



使用者指南

Amazon Elastic Compute Cloud



Amazon Elastic Compute Cloud: 使用者指南

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon 的商標和商業外觀不得用於任何非 Amazon 的產品或服務，也不能以任何可能造成客戶混淆、任何貶低或使 Amazon 名譽受損的方式使用 Amazon 的商標和商業外觀。所有其他非 Amazon 擁有的商標均為其各自擁有者的財產，這些擁有者可能隸屬於 Amazon，或與 Amazon 有合作關係，亦或受到 Amazon 贊助。

Table of Contents

什麼是 Amazon EC2 ?	1
功能	1
相關服務	2
存取 EC2	3
定價	4
估算、計費和成本最佳化	5
資源	6
開始教學課程	7
步驟 1：啟動執行個體	8
步驟 2：連線到您的執行個體	9
步驟 3：清理您的執行個體	12
後續步驟	12
最佳實務	14
Amazon Machine Image	16
使用 AMI	17
建立自己的 AMI	17
購買、共享和銷售 AMI	18
取消註冊您的 AMI	18
Amazon Linux 2023 和 Amazon Linux 2	18
視窗 AMI	19
AMI 類型	19
啟動許可	19
根設備儲存	20
虛擬化類型	23
開機模式	25
啟動執行個體	27
AMI 開機模式參數	33
執行個體類型開機模式	35
執行個體開機模式	37
作業系統開機模式	39
設定 AMI 開機模式	41
UEFI 變數	45
UEFI 安全開機	46
尋找 AMI	60

使用 Amazon EC2 主控台尋找 AMI	61
使用「」尋找 AMI AWS CLI	62
使用「」尋找 AMI AWS Tools for Windows PowerShell	63
使用 Systems Manager 參數尋找 AMI	63
使用 Systems Manager 尋找最新的 AMI	67
尋找 AMI 的更多資訊	68
共享 AMI	68
已驗證的供應商	68
尋找共享 AMI	69
使 AMI 公有化	74
與組織或 OU 共用 AMI	81
與指定的 AWS 帳戶共享 AMI	90
取消與您的帳戶共用 AMI	94
使用書籤	96
共享 Linux AMI 的準則	96
已支付 AMI	102
販售您的 AMI	103
尋找已支付 AMI	103
購買已支付 AMI	105
取得您執行個體的產品代碼	106
使用已支付支援	106
已支付及支援的 AMI 的帳單	107
管理您的 AWS Marketplace 訂閱	107
AMI 生命週期	108
建立 AMI	108
修改 AMI	173
複製 AMI	174
存放和還原 AMI	183
棄用 AMI	192
停用 AMI	199
封存 AMI 快照	205
取消註冊 (刪除) AMI	205
自動化 EBS 後端 AMI 生命週期	214
AMI 加密	214
執行個體啟動案例	215
映像複製案例	218

監控 AMI 事件	219
AMI 事件	221
創建 Amazon EventBridge 規則	224
了解 AMI 帳單	226
AMI 帳單欄位	227
尋找 AMI 帳單資訊	229
驗證帳單上的 AMI 費用	231
AMI 配額	232
請求增加 AMI 的配額	232
執行個體	234
執行個體與 AMI	234
執行個體	235
AMI	237
執行個體類型	238
可用的執行個體類型	238
硬體規格	239
AMI 虛擬化類型	241
尋找執行個體類型	242
取得建議	243
變更執行個體類型	250
爆量效能執行個體	259
GPU 執行個體	306
Mac 執行個體	316
考量事項	317
執行個體整備	318
EC2 macOS AMI	318
EC2 macOS Init	319
Amazon EC2 系統監視器	319
相關資源	319
啟動 Mac 執行個體	319
連接至您的 Mac 執行個體	322
更新 Mac 實例上的操作系統和軟件	325
增加 Mac 執行個體上 EBS 磁碟區的大小	332
停止並終止您的 Mac 執行個體	333
尋找專用主機支援的 macOS 版本	334
訂閱 macOS AMI 通知	335

EC2 macOS AMIs release notes	336
EBS 最佳化	339
支援的執行個體類型	339
獲得最大效能	407
檢視支援 EBS 最佳化的執行個體類型	407
在啟動時啟用 EBS 最佳化	409
啟用現有執行個體的 EBS 最佳化	410
執行個體購買選項	411
判斷執行個體生命週期	412
On-Demand Instances	413
Reserved Instances	415
Spot 執行個體	475
Dedicated Hosts	565
專用執行個體	618
Capacity Reservations	626
執行個體生命週期	700
執行個體啟動	702
執行個體停止和啟動	702
執行個體休眠	703
重新啟動執行個體	703
執行個體終止	704
重新啟動、停止、休眠和終止之間的差異	704
啟動	706
停止和啟動	775
休眠	783
重新開機	811
終止	813
淘汰	822
執行個體恢復	826
使用執行個體中繼資料	835
使用 IMDSv2	836
設定執行個體中繼資料選項	844
擷取執行個體中繼資料	866
使用執行個體使用者資料	888
擷取動態資料	892
執行個體中繼資料分類	893

Linux 實例：AMI 啟動指數值	905
執行個體身分文件	909
執行個體身分角色	973
啟動時執行命令	975
Amazon EC2 如何處理 Linux 執行個體的使用者資料	975
Amazon EC2 如何處理 Windows 執行個體的使用者資料	984
Connect 至您的 EC2 執行個體	998
連接至您的 Linux 執行個體	998
連線至 Windows 執行個體	1065
使用工作階段管理員進行連接	1076
使用 EC2 實例 Connect 端點進行 Connect	1077
將執行個體連線至資源	1101
識別執行個體	1139
檢查系統 UUID	1139
檢查系統虛擬機器產生識別符	1140
管理系統設定	1145
設定時間	1146
處理器狀態控制	1165
最佳化 CPU 選項	1167
AMD SEV-SNP	1305
新增視窗系統元件	1311
管理 Linux 系統使用者	1316
設定視窗管理員密碼	1320
管理裝置驅動	1321
安裝 NVIDIA 驅動程式	1321
安裝 AMD 驅動程式	1355
窗戶光伏驅動	1363
AWS 視窗 NVMe 驅動程式	1393
設定執行個體	1400
設定 Windows 啟動代理程式	1400
使用 EC2 快速啟動	1545
在 Windows 上使用彈性圖形加速器	1565
在視窗上安裝 WSL	1585
升級 Windows 執行個體	1587
執行就地升級	1587
執行自動升級	1591

移轉至目前一代的執行個體類型	1600
將 Microsoft SQL Server 從 Windows 遷移至 Linux	1609
故障診斷升級	1609
機群	1610
EC2 Fleet	1611
EC2 Fleet 限制	1613
爆量效能執行個體	1613
EC2 Fleet 請求類型	1613
EC2 Fleet 組態策略	1639
使用 EC2 Fleet	1672
Spot 叢集	1698
Spot Fleet 請求類型	1698
Spot Fleet 組態策略	1699
使用 Spot Fleet	1732
CloudWatch 競價型艦隊的指標	1762
Spot Fleet 的自動擴展	1766
監控機群事件	1773
EC2 Fleet 事件類型	1774
Spot Fleet 事件類型	1780
建立 EventBridge 規則	1786
教學課程	1796
教學：使用 EC2 Fleet 與執行個體權重	1796
教學：使用 EC2 Fleet 與隨需做為主要容量	1800
教學課程：使用目標容量預留啟動隨需執行個體	1801
教學課程：在容量區塊中啟動執行個體	1807
教學：使用 Spot Fleet 與執行個體權重	1809
範例組態	1812
EC2 Fleet 範例組態	1812
Spot Fleet 範例組態	1831
機群配額	1848
請求增加目標容量的配額	1849
監控	1851
自動和手動監控	1852
自動化監控工具	1852
手動監控工具	1853
監控最佳實務	1854

監控您的執行個體狀態	1855
執行個體狀態檢查	1855
狀態變更事件	1863
排程事件	1865
使用監控執行個體 CloudWatch	1896
實例警報	1896
啟用詳細監控	1897
列出可用的指標	1900
安裝和設定 CloudWatch 代理程式	1918
取得指標的統計資料	1922
圖形指標	1932
建立警示	1932
建立警示以停止、終止、重新啟動或復原執行個體	1933
使用自動化 EventBridge	1945
Amazon EC2 事件類型	1946
使用記錄 API 呼叫 CloudTrail	1946
Amazon EC2 API 信息 CloudTrail	1947
瞭解 Amazon EC2 API 日誌檔項目	694
透過 EC2 執行個體連線稽核連線	1949
監控您的 .NET 和 SQL Server 應用程式	1950
追蹤您的免費方案用量	1951
聯網	1954
區域 (Region) 和區域 (Zone)	1955
區域	1955
可用區域	1962
本機區域	1965
Wavelength 區域	1968
AWS Outposts	1971
執行個體 IP 定址	1972
私有 IPv4 地址	1973
公有 IPv4 地址	1974
公用 IPv4 位址最佳化	1975
彈性 IP 地址 (IPv4)	1976
IPv6 地址	1976
使用執行個體的 IPv4 地址	1977
使用執行個體的 IPv6 地址	1980

多個 IP 地址	1983
適用於視窗的多個私人 IPv4 位址	1991
EC2 執行個體主機名稱	1998
Link-local 地址	1998
執行個體主機名稱類型	1999
EC2 主機名稱的類型	1999
查看資源名稱與 IP 名稱的位置	2001
如何決定是選擇資源名稱還是 IP 名稱	2002
修改主機名稱類型和 DNS 主機名稱組態	2003
使用自有 IP 地址	2004
BYOIP 定義	2005
需求與配額	2006
加入先決條件	2006
加入您的 BYOIP	2014
使用您的位址範圍	2018
驗證您的 BYOIP	2019
區域可用性	2024
Local Zone 可用性	2024
進一步了解	2024
彈性 IP 地址	2024
彈性 IP 地址定價	2025
彈性 IP 地址基本概念	2025
使用彈性 IP 地址	2026
彈性 IP 地址配額	2040
網路介面	2041
網路介面基本概念	2042
網路卡	2044
每個執行個體類型每網路介面的 IP 地址	2045
使用網路介面	2046
設定網路介面的最佳實務	2057
網路介面案例	2059
申請者受管的網路介面	2061
指派字首	2063
網路頻寬	2079
可用執行個體頻寬	2079
監控執行個體頻寬	2081

增強型網路	2081
增強的網路支援	2082
彈性網路轉接器 (ENA)	2082
ENA Express	2107
Intel 82599 VF	2129
網路效能指標	2140
疑難排解 ENA	2149
排解 ENA 視窗驅動程式	2162
改善 Linux 執行個體的網路延遲	2179
硝基性能考量	2182
最佳化 Windows 執行個體的網路效能	2187
Elastic Fabric Adapter	2189
EFA 基本概念	2190
支援的介面和程式庫	2191
支援的執行個體類型	2191
支援的作業系統	2192
EFA 限制	2193
EFA 定價	2193
開始使用 P5 執行個體和 EFA	2193
開始使用 EFA 和 MPI	2197
開始使用 EFA 和 NCCL	2213
使用 EFA	2249
監控 EFA	2252
使用檢查總和驗證 EFA 安裝程式	2253
執行個體拓撲	2265
運作方式	2265
必要條件	2269
範例	2270
置放群組	2282
置放策略	2282
規則與限制	2285
使用置放群組	2287
共用置放群組	2300
放置群組 AWS Outposts	2305
網路 MTU	2306
巨型訊框 (9001 MTU)	2307

路徑 MTU 探索	2307
檢查兩個主機間的路徑 MTU	2308
檢查執行個體的 MTU	2310
為您的執行個體設定 MTU	2311
疑難排解	2313
Virtual Private Cloud (VPC)	2313
您的預設 VPC	2314
建立其他 VPC	2314
從您的執行個體中存取網際網路	2315
共用子網路	2316
僅限 IPv6 的子網路	2316
安全	2317
資料保護	2318
Amazon EBS 資料安全	2319
靜態加密	2319
傳輸中加密	2320
基礎設施安全	2321
網路隔離	2322
實體主機上的隔離	2322
控制網路流量	2322
恢復能力	2324
法規遵循驗證	2325
Identity and Access Management	2326
網路存取您的執行個體	2327
Amazon EC2 許可屬性	2327
IAM 和 Amazon EC2	2327
IAM 政策	2328
AWS 受管理政策	2394
IAM 角色	2397
AWS PrivateLink	2414
建立介面 VPC 端點	2414
建立端點政策	2414
更新管理	2416
Windows 執行個體的安全性最佳做法	2416
高階安全性最佳做法	2416
更新管理	2417

組態管理	2419
變更管理	2420
Amazon EC2 視窗執行個體的稽核和責任制	2420
金鑰對	2421
建立金鑰對	2422
標記金鑰對	2430
描述您的金鑰配對	2432
刪除您的金鑰對	2439
在 Linux 執行個體上新增或移除公開金鑰	2440
驗證指紋	2442
安全群組	2444
安全群組規則	2446
連線追蹤	2448
預設和自訂安全性群組	2452
使用安全群組	2454
不同使用案例的安全群組規則	2463
NitroTPM	2469
考量事項	2470
必要條件	2470
建立 Linux AMI 以取得 NitroTPM 支援	2472
驗證 AMI 是否已啟用 NitroTPM	2473
啟用或停用執行個體上的 NitroTPM	2474
擷取公開認可金鑰	2476
視窗執行個體的認證保護	2477
必要條件	2478
啟動支援的執行個體	2478
禁用內存完整性	2480
開啟認證保全	2480
確認認證保全正在執行	2482
儲存	2484
Amazon EBS	2485
執行個體存放區	2486
執行個體儲存體磁碟區和資料存留期	2487
執行個體儲存體磁碟區	2489
新增執行個體存放區磁碟區	2491
SSD 執行個體存放區磁碟區	2496

Linux 執行個體執行個體儲存交換磁碟	2499
最佳化 Linux 執行個體的磁碟效能	2503
檔案儲存	2504
Amazon S3	2505
Amazon EFS	2507
Amazon FSx	2510
Amazon 文件緩存	2516
執行個體磁碟區限制	2516
建置於 Nitro System 上的執行個體磁碟區限制	2516
Xen 型執行個體的磁碟區限制	2518
根設備磁碟區	2519
根磁碟區類型	2520
根磁碟區類型選擇一個 Linux AMI	2523
判斷 Linux 執行個體的根裝置類型	2523
將根磁碟區變更為持續存在	2524
變更根磁碟區的初始大小	2528
取代根磁碟區	2528
裝置名稱	2538
可用裝置名稱	2538
裝置名稱考量	2540
區塊型儲存設備對應	2541
區塊型儲存設備映射概念	2541
AMI 區塊型儲存設備映射	2545
執行個體區塊型儲存設備映射	2548
將磁碟映射至磁碟區	2555
列出 NVMe 磁碟區	2556
列出磁碟區	2561
視窗 VSS 快照	2570
什麼是 IVS ?	2571
必要條件	2572
建立 VSS 快照	2587
疑難排解以視窗 VSS 為基礎的 EBS 快照	2596
從 VSS 快照中還原磁碟區	2600
版本歷史記錄	2600
Linux 執行個體的寫入防止損毀	2603
定價	2604

支援的區塊大小和區塊邊界相符	2604
需求	2605
檢查預防未完成的寫入支援和設定	2605
設定軟體堆疊以預防未完成的寫入	2607
資源與標籤	2609
資源回收筒	2609
其運作方式？	2610
支援的資源	2611
考量事項	2611
配額	2614
相關服務	2614
定價	2614
所需的 IAM 許可	2615
使用保留規則	2619
使用資源回收筒中的資源	2632
監控資源回收筒	2641
資源位置	2660
資源 ID	2661
列出與篩選您的資源	2662
主控台步驟	2662
CLI 和 API 步驟	2668
全域檢視 (跨區域)	2670
全域檢視	2670
標記您的 資源	2673
標籤基本概念	2674
標記您的 資源	2675
標籤限制	2679
標籤與存取管理	2680
標記您的資源以便計費	2680
透過主控台使用標籤	2680
透過命令列使用標籤	2685
使用執行個體中繼資料中的執行個體標籤	2689
使用將標籤新增至資源 CloudFormation	2692
服務配額	2694
檢視您目前的配額	2694
請求增加	2695

使用通訊埠 25 傳送的電子郵件限制	2695
疑難排解	2696
Windows 執行個體的常見問題	2696
EBS 磁碟區不是在 Windows Server 2016 和 2019 和更新版本上初始化	2697
在目錄服務還原模式 (DSRM) 中開機 EC2 Windows 執行個體	2698
執行個體遺失網路連線能力或排程任務未如預期執行	2700
無法取得主控台輸出	2701
網路不提供 Windows Server 2012 R2	2701
磁碟簽章衝突	2701
Windows 執行個體的常見訊息	2703
「未提供密碼」	2703
「尚未提供密碼」	2704
「無法擷取 Windows 密碼」	2704
等待中繼資料服務	2705
無法啟動 Windows	2708
Windows 不是正版 (0x80070005)	2710
無終端機伺服器的授權伺服器可提供授權	2710
「部分設定是由您的組織所管理」	2711
啟動問題故障診斷	2711
無效裝置名稱	2712
超過執行個體限制	2712
執行個體容量不足	2713
目前並不支援要求的組態。請查看文件以了解支援的組態。	2713
立刻終止執行個體	2714
許可不足	2715
視窗啟動後不久的 CPU 使用率過高 (僅限 Windows 執行個體)	2716
連接至您的 Linux 執行個體	2717
連線問題的常見原因	2717
連線到執行個體時發生錯誤：連線時間逾時	2720
錯誤：無法載入金鑰 ... 預期為：任一私有金鑰	2722
錯誤：伺服器無法辨識使用者金鑰	2723
錯誤：許可遭拒或 [執行個體] 連接埠 22 已關閉連線	2725
錯誤：未受保護的私有金鑰檔案	2727
錯誤：私有金鑰的開頭必須為「-----BEGIN RSA PRIVATE KEY-----」，結尾為「-----END RSA PRIVATE KEY-----」	2729
錯誤：伺服器拒絕了我們的金鑰，或是沒有可用的支援驗證方法	2729

無法對執行個體執行 Ping 動作	2730
錯誤：伺服器意外關閉網路連線	2731
錯誤：EC2 Instance Connect 的主機金鑰驗證失敗	2731
無法使用 EC2 Instance Connect 連線至 Ubuntu 執行個體	2733
我的私有金鑰遺失。如何連線到我的 Linux 執行個體？	2733
連線至 Windows 執行個體	2740
遠端桌面無法連線到遠端電腦	2740
使用 macOS RDP 用戶端時發生錯誤	2743
RDP 顯示黑色螢幕而非桌面	2744
無法使用非管理員的使用者從遠端登入至執行個體	2744
使用的遠程桌面問題故障 AWS Systems Manager	2744
使用遠端登錄在 EC2 執行個體上啟用遠端桌面	2748
我的私有金鑰遺失。如何連接到我的 Windows 執行個體？	2749
重設遺失或過期的 Windows 管理員密碼	2750
使用 EC2Launch v2 重設	2751
使用 EC2Config 重設	2756
使用 EC2Launch 重設	2761
故障診斷無法連線執行個體的問題	2766
重新啟動執行個體	2766
執行個體主控台輸出	2767
擷取無法連線執行個體的螢幕擷取畫面	2768
Windows 執行個體的常見螢幕擷	2770
當主機電腦故障時的執行個體復原	2779
停止您的執行個體。	2779
強制停止執行個體	2779
建立替代執行個體	2780
終止您的執行個體	2783
立刻終止執行個體	2783
延遲的執行個體終止	2783
仍顯示已終止的執行個體	2783
錯誤：執行個體可能無法終止。修改其「禁用ApiTermination」實例屬性	2783
執行個體自動啟動或終止	2784
Linux 上的狀態檢查失敗	2784
檢閱狀態檢查資訊	2785
擷取系統日誌	2786
解決 Linux 執行個體的系統記錄錯誤	2786

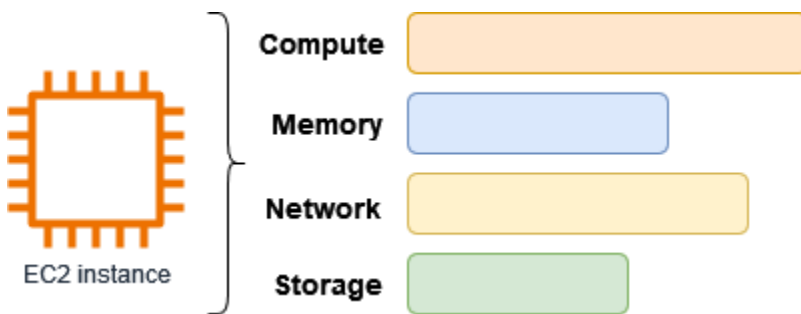
記憶體不足：終止程序	2787
錯誤：mmu_update 失敗 (記憶體管理更新作業失敗)	2788
I/O 錯誤 (區塊型儲存設備故障)	2789
I/O 錯誤：非本機也非遠端磁碟 (故障的分散式區塊型儲存設備)	2791
request_module：失控迴圈 modprobe (在舊版 Linux 上建立舊核心 modprobe 的迴圈)	2792
「嚴重：核心版本太舊」和「fsck：嘗試開啟 /dev 時，找不到此等檔案或目錄」(核心與 AMI 不符)	2793
「致命：無法載入 /lib/ 模組」或「BusyBox」(遺失核心模組)	2794
錯誤：無效的核心 (與 EC2 不相容的核心)	2796
fsck：嘗試開啟時找不到此等檔案或目錄... (找不到檔案系統)	2797
掛載檔案作業系統的一般錯誤 (掛載失敗)	2799
VFS：無法在未知的區塊上掛載根 fs (根檔案系統不符)	2801
錯誤：無法判定根設備的主/次編號... (根檔案系統/裝置不相符)	2802
XENBUS：裝置缺少驅動程式...	2804
...未進行檢查的天數，強制進行檢查 (需要進行檔案系統檢查)	2805
fsck 凍結於結束狀態... (缺少裝置)	2805
GRUB 提示 (grubdom>)	2807
帶出界面 eth0：裝置 eth0 的 MAC 地址和預期的不同，略過。(將 MAC 地址直接寫入程式)	2809
無法載入 SELinux 政策。機器正處於強制執行模式。現在停止中。(SELinux 組態設定不當)	2810
XENBUS：連線到裝置的作業逾時 (Xenbus 逾時)	2812
對 Linux 執行個體從錯誤的磁碟區開機	2813
疑難排解系統問題	2815
EC2Rescue for Linux	2816
安裝 適用於 Linux 的 EC2Rescue	2817
(選用) 確認 適用於 Linux 的 EC2Rescue 的簽章	2818
使用 適用於 Linux 的 EC2Rescue	2821
開發 EC2Rescue 模組	2823
EC2Rescue for Windows Server	2829
使用 GUI	2830
使用命令列	2836
使用 Systems Manager	2843
EC2 序列主控台	2847
必要條件	2847
設定對 EC2 序列主控台的存取	2855

連線至 EC2 序列主控台	2863
終止與 EC2 序列主控台的連線	2871
使用 EC2 序列主控台進行執行個體疑難排解	2872
傳送診斷中斷	2881
支援的執行個體類型	2881
必要條件	2882
傳送診斷中斷	2885
文件歷史紀錄	2887
2018 年及更早版本的歷史	2906
.....	mmcmxxv

什麼是 Amazon EC2 ？

Amazon Elastic Compute Cloud (Amazon EC2) 在 Amazon Web Services (AWS) Cloud 中提供可擴展的隨需運算容量。使用 Amazon EC2 可降低硬體成本，讓您以更快的速度開發和部署應用程式。您可以使用 Amazon EC2 按需要啟動任意數量的虛擬伺服器，設定安全性和聯網功能以及管理儲存。您可以新增容量 (擴充) 來處理需要大量運算的工作，例如每月或每年的例行程序，或是網站流量暴增。當使用量降低時，您可以再次減少容量 (縮減)。

EC2 執行個體是 AWS 雲端中的虛擬伺服器。啟動 EC2 執行個體時，您指定的執行個體類型會決定執行個體可用的硬體。每種執行個體類型都提供不同的運算、記憶體、網路和儲存資源平衡。如需詳細資訊，請參閱 [Amazon EC2 執行個體類型指南](#)。



Amazon EC2 的功能

Amazon EC2 提供以下進階功能：

執行個體

虛擬伺服器。

Amazon Machine Images (AMI)

供執行個體使用的預先設定範本，在其中封裝伺服器所需的元件 (包括作業系統和其他軟體)。

執行個體類型

適用執行個體的各種 CPU、記憶體、儲存、連網容量和圖形硬體組態。

Amazon EBS 磁碟區

資料的持久性儲存磁碟區請使用 Amazon Elastic Block Store (Amazon EBS)。

執行個體儲存體磁碟區

儲存磁碟區可供您停止、休眠或終止執行個體時所刪除的暫時性資料使用。

金鑰對

保護執行個體的登入資訊。AWS 存儲公鑰，並將私鑰存儲在安全的地方。

安全群組

虛擬防火牆可讓您指定存取執行個體的協定、連接埠和來源 IP 範圍，以及執行個體可連線的目的地 IP 範圍。

Amazon EC2 支援處理、儲存、傳輸商家或服務供應商的信用卡資料，並且已驗證符合支付卡產業 (PCI) 資料安全標準 (DSS)。如需 PCI DSS 的詳細資訊，包括如何要求 AWS PCI 相容性 Package 的複本，請參閱 [PCI DSS 等級 1](#)。

相關服務

與 Amazon EC2 搭配使用的服務

您可以在使用 Amazon EC2 部署 AWS 服務的執行個體搭配使用其他執行個體。

[Amazon EC2 Auto Scaling](#)

協助確保您有正確的 Amazon EC2 執行個體數量可應付應用程式的負載。

[AWS Backup](#)

自動備份 Amazon EC2 執行個體和連接至該執行個體的 Amazon EBS 磁碟區。

[Amazon CloudWatch](#)

監控執行個體和 Amazon EBS 磁碟區。

[Elastic Load Balancing](#)

自動將傳入應用程式流量分配至多個執行個體。

[Amazon GuardDuty](#)

偵測 EC2 執行個體可能未授權或惡意的使用情況。

[EC2 Image Builder](#)

自動建立、管理和部署自訂、安全和 up-to-date 伺服器影像。

[AWS Launch Wizard](#)

調整協力廠商應用程式的 AWS 資源大小、設定和部署資源，而不必手動識別和佈建個別 AWS 資源。

[AWS Systems Manager](#)

使用此安全 end-to-end 管理解決方案在 EC2 執行個體上大規模執行操作。

其他運算服務

您可以使用其他 AWS 運算服務而不是使用 Amazon EC2 來啟動執行個體。

[Amazon Lightsail](#)

使用 Amazon Lightsail 建置網站或 Web 應用程式，這是一個雲端平台，可提供快速部署專案所需的資源，而且每月價格低廉且可預測。若要比較 Amazon EC2 和 Lightsail，請參閱 [Amazon EC2](#)

[Amazon Elastic Container Service \(Amazon ECS\)](#)

在 EC2 執行個體的叢集上部署、管理和擴展容器化應用程式。如需詳細資訊，請參閱 [選擇 AWS 容器服務](#)。

[Amazon Elastic Kubernetes Service \(Amazon EKS\)](#)

在 AWS 上執行您的 Kubernetes 應用程式。如需詳細資訊，請參閱 [選擇 AWS 容器服務](#)。

存取 Amazon EC2

您可以使用下列界面來建立和管理 Amazon EC2 執行個體：

Amazon EC2 主控台

用於建立和管理 Amazon EC2 執行個體和資源的簡易 Web 界面。如果您已註冊 AWS 帳戶，則可以登入 AWS Management Console 並從主控台首頁選取 EC2 來存取 Amazon EC2 主控台。

AWS Command Line Interface

可讓您使用命令列殼層中的命令與 AWS 服務互動。Windows、Mac 和 Linux 系統皆提供支援。如需有關 AWS CLI 的詳細資訊，請參閱 [《AWS Command Line Interface 使用者指南》](#)。您可以在 [AWS CLI 命令參考](#) 中找到 Amazon EC2 命令。

AWS CloudFormation

Amazon EC2 支援使用 AWS CloudFormation 建立資源。您可以建立 JSON 或 YAML 格式的範本，以描述您的 AWS 資源，並為您 AWS CloudFormation 佈建和設定這些資源。您可以重複使用 CloudFormation 範本來多次佈建相同的資源，無論是在相同的區域和帳戶中，還是在多個區域和帳

戶中。如需 Amazon EC2 適用的支援資源類型和屬性的詳細資訊，請參閱《AWS CloudFormation 使用者指南》中的 [EC2 資源類型參考](#)。

AWS 開發套件

如果您偏好使用特定語言的 API 來建置應用程式，而不是透過 HTTP 或 HTTPS 提交要求，請為軟體開發人員 AWS 提供程式庫、範例程式碼、教學課程和其他資源。這些程式庫提供的基本功能可自動化任務，例如密碼編譯簽署要求、重試要求，以及處理錯誤回應，讓您更容易入門。如需詳細資訊，請參閱 [在 AWS 上建置的工具](#)。

AWS Tools for PowerShell

— PowerShell 組以公開的功能為基礎建置的模組 AWS SDK for .NET。「工具」可 PowerShell 讓您從命令列對資 AWS 源執行作業的 PowerShell 指令碼。若要開始使用，請參閱《[使用者指南 AWS Tools for Windows PowerShell](#)》。您可以在 [AWS Tools for PowerShell Cmdlet 參考](#) 中找到適用於 Amazon EC2 的 cmdlet。

查詢 API

Amazon EC2 提供查詢 API。這些 HTTP 或 HTTPS 請求，需要使用 HTTP 動詞 GET 或 POST，以及名為 Action 的查詢參數。如需適用於 Amazon EC2 的 API 動作詳細資訊，請參閱 Amazon EC2 API Reference 中的 [動作](#)。

Amazon EC2定價

Amazon EC2 提供下列定價選項：

免費方案

您可以免費開始使用 Amazon EC2。若要探索免費方案選項，請參閱 [AWS 免費方案](#)。

隨需執行個體

您只需要按秒支付使用執行個體的費用 (最少 60 秒)，無需長期購買或預付款。

Savings Plans

您可以透過對一致的使用量設定綁約來降低 Amazon EC2 成本，每小時以美金計價，為期一年或三年。

Reserved Instances

您可以透過設定綁約至特定的執行個體設定來降低您 Amazon EC2 的成本，包含執行個體類型和區域，為期一年或三年。

Spot Instances

請求未使用的 EC2 執行個體，可大幅降低您的 Amazon EC2 成本。

專用執行個體

使用完全供您使用的實體 EC2 伺服器 (隨需或 Savings Plan 的一部分) 來降低成本。您可以使用現有繫結伺服器的軟體授權，並取得符合合規要求的協助。

On-Demand Capacity Reservations

在特定可用區域中為 EC2 執行個體預留任何期限的運算容量。

以秒計費

從帳單中移除未使用分鐘和秒數的成本。

如需 Amazon EC2 的完整收費與定價清單，以及有關購買模型的詳細資訊，請參閱 [Amazon EC2 定價](#)。

估算、計費和成本最佳化

若要為您的 AWS 使用案例建立估算值，請使用 [AWS Pricing Calculator](#)。

若要估算將 Microsoft 工作負載轉換為使用部署開放原始碼和雲端原生服務的現代化架構的成本 AWS，請使用適用於 [Microsoft 工作負載的現 AWS 代化計算器](#)。

若要查看您的帳單，請前往 [AWS Billing and Cost Management 主控台](#) 中的帳單與成本管理儀表板。您的帳單內含用量報告的連結，可提供帳單的詳細資訊。若要深入瞭解 AWS 帳戶帳單，請參閱 [AWS Billing and Cost Management 使用者指南](#)。

如果您對帳 AWS 單、帳戶和事件有任何疑問，請聯絡 [Sup AWS port](#) 部門。

若要計算範本佈建環境的成本，請參閱 [雲端成本中心](#)。在計算佈建環境的成本時，請記住要包括附帶成本，如 EBS 磁碟區的快照儲存。

您可以使用最佳化 AWS 環境的成本、安全性和效能 [AWS Trusted Advisor](#)。

您可以使用 AWS Cost Explorer 來分析 EC2 執行個體的成本和用量。您可以檢視最近 13 個月的資料，並預測未來 12 個月的可能花費金額。有關更多信息，請參閱 [AWS Cost Management 用戶指南 AWS Cost Explorer 中的分析成本](#)。

資源

- [Amazon EC2 功能](#)
- [AWS Re: 文章](#)
- [AWS 技能生成器](#)
- [AWS Support](#)
- [動手教程](#)
- [網頁寄存](#)
- [視窗開啟 AWS](#)

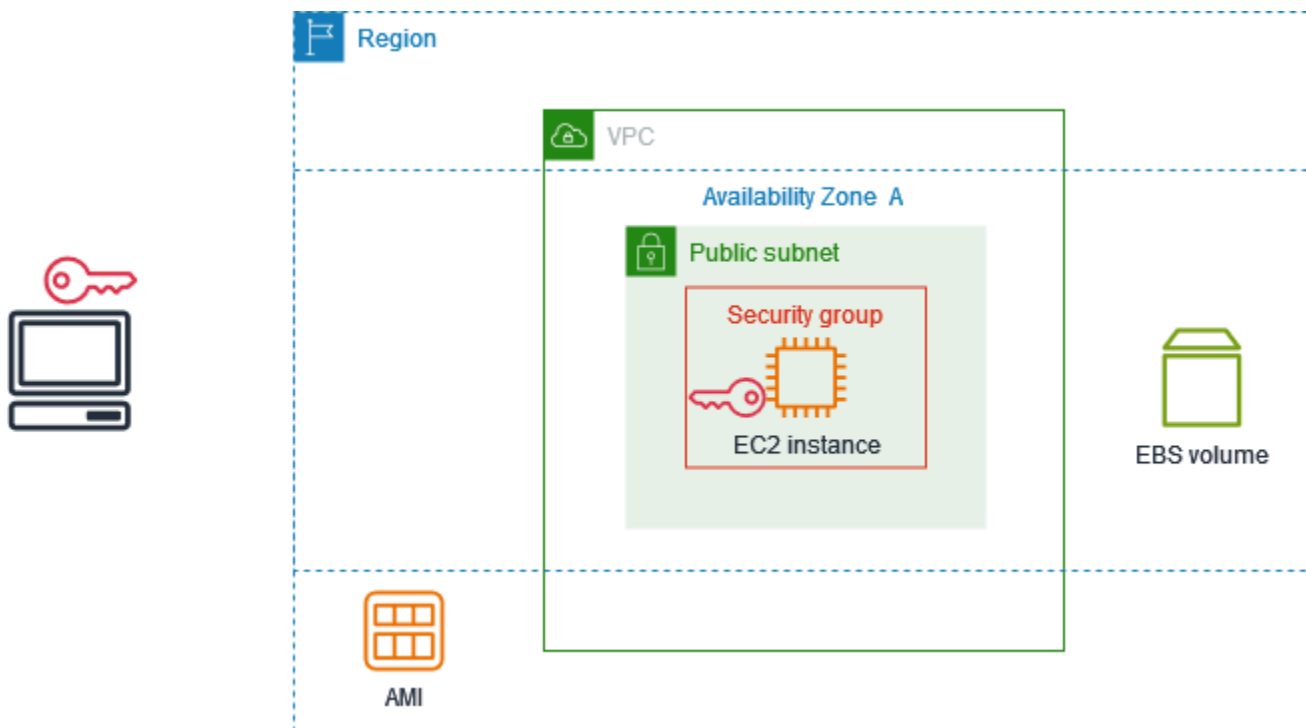
開始使用 Amazon EC2

使用此教學課程開始使用 Amazon Elastic Compute Cloud (Amazon EC2)。您將學習如何啟動並連接 EC2 執行個體。執行個體是 AWS 雲端中的虛擬伺服器。使用 Amazon EC2，依您的執行個體安裝和設定作業系統與應用程式。

概觀

下圖顯示您將在本教學課程中使用的關鍵元件：

- 映像檔 — 包含要在執行個體 (例如作業系統) 上執行之軟體的範本。
- key pair — 一組安全認證，用來在連線至執行個體時證明您的身分。公開金鑰位於您的執行個體上，私密金鑰位於您的電腦上。
- 網絡 — 虛擬私有雲 (VPC) 是一個虛擬網絡，專用於您 AWS 帳戶的。為了協助您快速開始使用，您的帳戶每個帳戶中都有一個預設 VPC AWS 區域，而每個預設 VPC 在每個可用區域中都有一個預設子網路。
- 安全群組 — 充當虛擬防火牆，以控制輸入和輸出流量。
- EBS 磁碟區 — 映像檔需要根磁碟區。您可以選擇性地新增資料磁碟區。



本教學課程的費用

當您註冊時 AWS，您可 Amazon EC2 用 [AWS 免費方案](#)。如果您 AWS 帳戶在 12 個月前建立的產品，但尚未超過 Amazon EC2 的免費方案權益，則完成本教學課程不會花費任何費用，因為我們會協助您選擇免費方案權益範圍內的選項。否則，從您啟動執行個體到終止執行個體 (這是本教學的最後步驟) 會產生基本的 Amazon EC2 使用費用，即使執行個體為閒置狀態。

如需判斷您是否符合免費方案資格的指示，請參閱 [the section called “追蹤您的免費方案用量”](#)。

任務

- [步驟 1：啟動執行個體](#)
- [步驟 2：連線到您的執行個體](#)
- [步驟 3：清理您的執行個體](#)
- [後續步驟](#)

步驟 1：啟動執行個體

您可以使用下列程序所 AWS Management Console 述啟動 EC2 執行個體。本教學課程旨在協助您在免費方案權益中快速啟動第一個執行個體，因此不涵蓋所有可能的選項。

啟動執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在屏幕頂部的導航欄中，我們顯示當前的 AWS 區域 -例如，俄亥俄州。您可以使用選取的「區域」，或選擇性地選取距離您較近的「區域」。
3. 從 EC2 主控台儀表板的 [啟動執行個體] 窗格中，選擇 [啟動執行個體]。
4. 在 Name and tags (名稱和標籤) 下的 Name (名稱) 欄位中，輸入執行個體的描述性名稱。
5. 在 Application and OS Images (Amazon Machine Image) (應用程式和作業系統映像 (Amazon Machine Image)) 下，執行下列動作：
 - a. 選擇 [快速入門]，然後為您的執行個體選擇作業系統 (OS)。對於您的第一個 Linux 執行個體，我們建議您選擇 Amazon Linux。
 - b. 從 Amazon 機器映像 (AMI) 中，選取標示為免費方案資格的 AMI。
6. 在 [執行個體類型] 下方，針對 [執行個體類型] 選擇 t2.micro 符合免費方案的資格。在無法使用 t2.micro 的區域中，符 t3.micro 合免費方案的資格。
7. 在 key pair (登入) 下，對於 key pair 名稱，選擇現有的 key pair，或選擇建立新的金鑰配對來建立您的第一個金鑰配對。

⚠ Warning

如果您選擇不使用 key pair 繼續 (不建議使用)，您將無法使用本教學課程所述的方法連線至執行個體。

8. 在 [網路設定] 下，請注意我們已選取您的預設 VPC、選取在我們為您選擇的可用區域中使用預設子網路的選項，並設定安全性群組，其中包含允許從任何地方連線至執行個體的規則。對於您的第一個執行個體，我們建議您使用預設設定。否則，您可以按照以下方式更新網路設置：
 - (選擇性) 若要使用特定的預設子網路，請選擇 [編輯]，然後選擇子網路。
 - (選擇性) 若要使用不同的 VPC，請選擇 [編輯]，然後選擇現有的 VPC。如果 VPC 未設定為公用網際網路存取，您將無法連線至執行個體。
 - (選擇性) 若要限制特定網路的輸入連線流量，請選擇 [自訂] 而非 [任何地方]，然後輸入網路的 CIDR 區塊。
 - (選擇性) 若要使用不同的安全性群組，請選擇選取現有的安全性群組，然後選擇現有的安全性群組。如果安全性群組沒有允許來自網路的連線流量的規則，您將無法連線至執行個體。對於 Linux 執行個體，您必須允許安全殼層流量。對於 Windows 執行個體，您必須允許 RDP 流量。
9. 在 [設定儲存空間] 下，請注意我們已設定根磁碟區，但沒有資料磁碟區。這足以用於測試目的。
10. 檢閱 Summary (摘要) 面板中執行個體組態的摘要，並在準備就緒時選擇 Launch instance (啟動執行個體)。
11. 如果啟動成功，請從「成功」通知中選擇執行個體的 ID，以開啟「執行個體」頁面並監控啟動狀態。
12. 選取該執行個體的核取方塊。初始執行個體狀態為 pending。在執行個體啟動後，它的生命週期狀態會變為 running。選擇 [狀態與警示] 索引標籤。執行個體通過其狀態檢查後，就可以接收連線要求。

步驟 2：連線到您的執行個體

您使用的程序視執行個體的作業系統而定。如果您無法連線至您的執行個體，請參閱[連線到 Linux 執行個體的疑難排解](#)取得協助。

Linux 執行個體

您可以使用任何安全殼層用戶端連線到 Linux 執行個體。如果您在電腦上執行 Windows，請開啟終端機並執行 ssh 指令以確認您已安裝 SSH 用戶端。如果找不到這個命令，請[安裝 OpenSSH 視窗版](#)。

使用 SSH 連線到您的執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體，然後選取 Connect (連線)。
4. 在 [Connect 至執行個體] 頁面上，選擇 [SSH 用戶端] 索引標籤。
5. (選擇性) 如果您在啟動執行個體時建立 key pair 組，並將私密金鑰 (.pem 檔案) 下載到執行 Linux 或 macOS 的電腦，請執行範例 `chmod` 命令來設定私密金鑰的權限。
6. 複製示例 SSH 命令。以下是範例，其中金####.pem 是私密金鑰檔案的名稱，*ec2-user* 是與映像檔相關聯的使用者名稱，而 @符號後面的字串是執行個體的公開 DNS 名稱。

```
ssh -i key-pair-name.pem ec2-user@ec2-198-51-100-1.us-east-2.compute.amazonaws.com
```

7. 在電腦的終端機視窗中，執行您在上一個步驟中儲存的 ssh 命令。如果私密金鑰檔案不在目前目錄中，您必須在此指令中指定金鑰檔案的完整路徑。

以下是回應範例：

```
The authenticity of host 'ec2-198-51-100-1.us-east-2.compute.amazonaws.com
(198-51-100-1)' can't be established.
ECDSA key fingerprint is 14UB/neBad9tvkgJf1QZWxheQmR59WgrgzEimCG6kZY.
Are you sure you want to continue connecting (yes/no)?
```

8. (選擇性) 在您第一次啟動執行個體時，確認安全性警示中的指紋是否符合主控台輸出中包含的執行個體指紋。若要取得主控台輸出，請選擇 [動作]、[監視和疑難排解]、[取得系統記錄 如果指紋不相符，可能是有人試圖 man-in-the-middle 攻擊。如果它們符合，請繼續下一個步驟。
9. 輸入 **yes**。

以下是回應範例：

```
Warning: Permanently added 'ec2-198-51-100-1.us-
east-2.compute.amazonaws.com' (ECDSA) to the list of known hosts.
```

Windows 執行個體

若要連線至 Windows 執行個體，您必須擷取初始管理員密碼，並在使用遠端桌面連線至執行個體時使用此密碼。在執行個體啟動後，需要幾分鐘的時間處理才能提供密碼。

系統管理員帳戶的預設使用者名稱取決於 AMI 中包含的作業系統 (OS) 的語言。若要確定正確的使用者名稱，請識別 AMI 作業系統的語言，然後選擇對應的使用者名稱。例如，對於英語操作系統，用戶名是 Administrator，對於法國操作系統來說 Administrateur，對於葡萄牙語操作系統而言，它是 Administrador。如果作業系統的語言版本沒有相同語言的使用者名稱，請選擇使用者名稱 Administrator (Other)。如需詳細資訊，請參閱 Microsoft TechNet Wiki [中的 Windows 系統管理員帳戶的本地化名稱](#)。

如果您已將執行個體加入網域，則可以使用已在 AWS Directory Service 中定義的網域憑證連線至您的執行個體。在遠端桌面登入畫面上，不要使用本機電腦名稱和產生的密碼，而是使用系統管理員的合格使用者名稱 (例如 `corp.example.com\Admin`) 和此帳戶的密碼。

如果您在嘗試連線至執行個體時遇到錯誤，請參閱 [the section called “遠端桌面無法連線到遠端電腦”](#)。

使用 RDP 用戶端連線至您的 Windows 執行個體。

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體，然後選取 Connect (連線)。
4. 在 [Connect 至執行個體] 頁面上，選擇 [RDP 用戶端] 索引標籤。
5. 在使用者名稱中，選擇管理員帳戶的預設使用者名稱。您選擇的使用者名稱必須與您用來啟動執行個體的 AMI 中所包含的作業系統 (OS) 語言相符。如果沒有與您的作業系統相同語言的使用者名稱，請選擇「管理員 (其他)」。
6. 選擇 [取得密碼]。
7. 在 [取得 Windows 密碼] 頁面上，執行下列動作：
 - a. 選擇「上傳私密金鑰檔案」，然後瀏覽至您在啟動執行個體時指定的私密金鑰 (.pem) 檔案。選取檔案並選取 Open (開啟)，將檔案的完整內容複製至此視窗。
 - b. 選擇「解密密碼」。[取得 Windows 密碼] 頁面會關閉，且執行個體的預設管理員密碼會顯示在 [密碼] 底下，取代先前顯示的 [取得密碼] 連結。
 - c. 複製密碼並將其保存在安全的地方。必須使用此密碼來連接至執行個體。
8. 選擇 Download Remote Desktop File (下載遠端桌面檔案)。當您完成下載檔案時，請選擇 Cancel (取消) 以返回 Instances (執行個體) 頁面。導覽至您的下載目錄，然後開啟 RDP 檔案。
9. 您可能會收到警告提示遠端連線的發佈者為未知。選擇 Connect (連接) 以繼續連接到您的執行個體。
10. 預設選擇系統管理員帳戶。貼上先前複製的密碼，然後選擇 [確定]。
11. 由於自我簽署憑證的性質，您可能會收到安全憑證無法驗證的警告。執行以下任意一項：

- 如果您信任憑證，請選擇 [是] 連線至您的執行個體。
- [Windows] 在繼續之前，請將證書的指紋與系統日誌中的值進行比較，以確認遠端電腦的身分。選擇檢視憑證，然後從詳細資料索引標籤中選擇指紋。將此值與 [動作]、[監視及疑難排解]、[取得系統記錄] RDPCERTIFICATE-THUMBPRINT 中的值進行比較。
- [Mac OS X] 繼續之前，請先將憑證的指紋與系統記錄檔中的值進行比較，以確認遠端電腦的身分識別。選擇顯示憑證，展開詳細資料，然後選擇 SHA1 指紋。將此值與 [動作]、[監視及疑難排解]、[取得系統記錄] RDPCERTIFICATE-THUMBPRINT 中的值進行比較。

步驟 3：清理您的執行個體

在您完成為本教學建立的執行個體後，您應該終止此執行個體完成清理。如果想在清理前使用此執行個體執行更多作業，請參閱 [後續步驟](#)。

Important

終止執行個體可有效的將其刪除，無法在終止執行個體後重新連接到執行個體。

如果您啟動不在 [AWS 免費方案](#) 內的執行個體，執行個體狀態一變更為 shutting down 或 terminated，就會立即停止產生費用。若要保留執行個體以待後用，但不想產生費用，您可立即停止執行個體，稍後再啟動它。如需詳細資訊，請參閱 [停止並啟動 Amazon EC2 執行個體](#)。

終止您的執行個體

1. 在導覽窗格中，選擇 Instances (執行個體)。在執行個體清單中，選取執行個體。
2. 選擇 Instance state (執行個體狀態)、Terminate instance (終止執行個體)。
3. 出現確認提示時，請選擇 Terminate (終止)。

Amazon EC2 關閉並終止您的執行個體。在您的執行個體終止後，它仍會短暫顯示於主控台，然後項目才會自動刪除。您無法自行從主控台顯示畫面中移除已終止的執行個體。

後續步驟

啟動執行個體之後，您可能會想要探索下列後續步驟：

- 瞭解如何追蹤您的免費方案使用情況，以避免帳單意外。如需詳細資訊，請參閱 [the section called “追蹤您的免費方案用量”](#)。

- 設定 CloudWatch 警示，以在您的使用量超過免費方案時通知您。如需詳細資訊，請參閱使用 AWS Billing 者指南中的 [追蹤您的 AWS 免費方案使用量](#)。
- 新增 EBS 磁碟區。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的建立 Amazon EBS 磁碟區](#)。
- 了解如何使用 Run 命令從遠端管理您的 EC2 執行個體。如需詳細資訊，請參閱 AWS Systems Manager 使用者指南中的 [AWS Systems Manager 執行命令](#)。
- 了解執行個體購買選項。如需詳細資訊，請參閱 [執行個體購買選項](#)。
- 取得有關執行個體類型的建議。如需更多詳細資訊，請參閱 [取得新工作負載的執行個體類型建議](#)。

Amazon EC2 的最佳實務

為了確保 Amazon EC2 獲得最大利益，我們建議您執行以下最佳實務。

安全

- 盡可能使用與身分提供者和 AWS IAM 角色聯合的身分識別來管理資源和 API 的存取。如需詳細資訊，請參閱《IAM 使用者指南》中的[建立 IAM 政策](#)。
- 為您的安全群組實作最嚴苛的規則。如需詳細資訊，請參閱[安全群組規則](#)。
- 定期修補、更新和保護您執行個體上的作業系統與應用程式。如需詳細資訊，請參閱[更新管理](#)。如需 Windows 作業系統的特定指導方針，請參閱[Windows 執行個體的安全性最佳做法](#)。
- 使用 Amazon Inspector 自動探索和掃描 Amazon EC2 執行個體的軟體漏洞和意外網路暴露。如需更多資訊，請參閱[Amazon Inspector 使用者指南](#)。
- 使用 AWS Security Hub 控制項根據安全最佳實務和安全標準監控 Amazon EC2 資源。如需有關使用 Security Hub 的詳細資訊，請參閱《AWS Security Hub 使用者指南》中的[Amazon Elastic Compute Cloud 控制項](#)。

儲存

- 了解根裝置類型對資料持久性、備份和復原的意涵。如需詳細資訊，請參閱[根設備儲存](#)。
- 為作業系統和資料使用不同的 Amazon EBS 磁碟區。確保執行個體終止後能持續儲存您資料的磁碟區。如需詳細資訊，請參閱[在執行個體終止時保留資料](#)。
- 使用您執行個體可用的執行個體存放區存放暫存資料。請記住，當您讓您的執行個體停止、休眠或終止時，會刪除存放在執行個體存放區的資料。如果您將執行個體存放區用為資料庫儲存體，請確保您的叢集有能確保容錯能力的複寫因素。
- 加密 EBS 磁碟區和快照。如需詳細資訊，請參閱[Amazon EBS 使用者指南中的 Amazon EBS 加密](#)。

資源管理

- 使用執行個體中繼資料和自訂資源標籤追蹤並找出您的 AWS 資源。如需詳細資訊，請參閱[使用執行個體中繼資料](#)和[標記您的 Amazon EC2 資源](#)。
- 檢視您目前的 Amazon EC2 限制。計劃在您需要的時候，先行請求提高限制。如需詳細資訊，請參閱[Amazon EC2 服務配額](#)。

- 用於檢 AWS Trusted Advisor 查您的 AWS 環境，然後在存在機會時提出建議，以節省成本、改善系統可用性和效能，或協助縮小安全性漏洞。如需詳細資訊，請參閱《AWS Support 使用者指南》中的 [AWS Trusted Advisor](#)。

備份與復原

- 使用 [Amazon EBS 快照](#) 定期備份您的 EBS 磁碟區，並從您的執行個體建立 [Amazon Machine Image \(AMI\)](#) 將組態儲存為範本，以啟動未來的執行個體。如需有助於實現此使用案例的 AWS 服務的詳細資訊，請參閱 [AWS Backup](#) 和 [Amazon Data Lifecycle Manager](#)。
- 跨多個可用區域部署您應用程式的關鍵元件，並適當複寫您的資料。
- 設計您的應用程式在執行個體重新啟動時處理動態 IP 定址。如需詳細資訊，請參閱 [Amazon EC2 執行個體 IP 定址](#)。
- 監控與回應事件。如需詳細資訊，請參閱 [監控 Amazon EC2](#)。
- 請確定您已準備好可處理容錯移轉。如需基本解決方案，您可將網路界面或彈性 IP 地址手動連接到替代執行個體。如需詳細資訊，請參閱 [彈性網路介面](#)。如需自動化解決方案，您可使用 Amazon EC2 Auto Scaling。如需詳細資訊，請參閱 [Amazon EC2 Auto Scaling 使用者指南](#)。
- 定期測試執行個體和 Amazon EBS 磁碟區的復原程序，以確保資料和各項服務成功還原。

聯網

- 針對 time-to-live IPv4 和 IPv6，將應用程式的值設定為 255。如果您使用較小的值，則存在 TTL 會在應用程式流量傳輸期間過期的風險，從而導致執行個體的可存取性問題。

Amazon Machine Image (AMI)

Amazon 機器映像 (AMI) 是支援且維護的映像檔 AWS ，提供啟動執行個體所需的資訊。在您啟動執行個體時，必須指定 AMI。當您需要多個具有相同組態的執行個體時，可以從單一 AMI 啟動多個執行個體。當您需要具有不同組態的執行個體時，可以使用不同的 AMI 啟動執行個體。

AMI 包括下列項目：

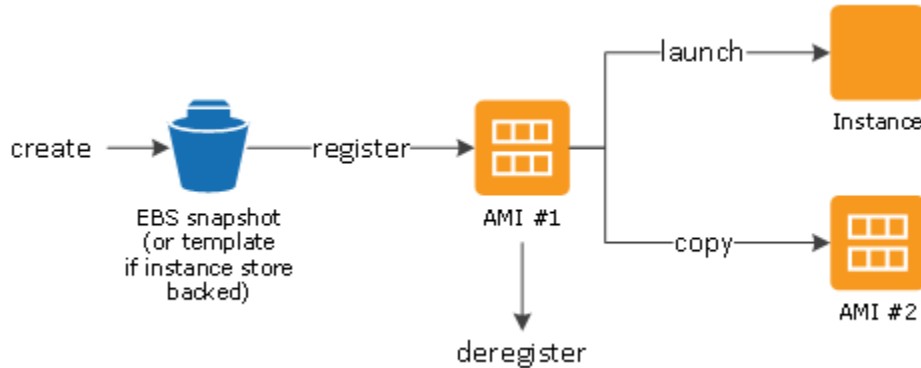
- 一或多個 Amazon Elastic Block Store (Amazon EBS) 快照，或對於 instance-store-backed AMI ，則為執行個體根磁碟區的範本 (例如，作業系統、應用程式伺服器 and 應用程式)。
- 控制哪些 AWS 帳戶可以使用 AMI 啟動執行個體的啟動權限。
- 區塊型設備映射，指定啟動執行個體時連接至該執行個體的磁碟區。

Amazon Machine Image (AMI) 主題

- [使用 AMI](#)
- [建立自己的 AMI](#)
- [購買、共享和銷售 AMI](#)
- [取消註冊您的 AMI](#)
- [Amazon Linux 2023 和 Amazon Linux 2](#)
- [視窗 AMI](#)
- [AMI 類型](#)
- [AMI 虛擬化類型](#)
- [Amazon EC2 啟動模式](#)
- [尋找 AMI](#)
- [共享 AMI](#)
- [已支付 AMI](#)
- [AMI 生命週期](#)
- [對 EBS 後端 AMI 使用加密](#)
- [使用 Amazon 監控 AMI 事件 EventBridge](#)
- [了解 AMI 帳單資訊](#)
- [AMI 配額](#)

使用 AMI

下圖摘要說明 AMI 生命週期。在您建立和註冊 AMI 之後，即可用以啟動新的執行個體。(如果 AMI 擁有者將啟動許可授予您，則您也可以從 AMI 啟動執行個體)。您可以將 AMI 複製到相同 AWS 區域內或不同的區 AWS 域。當您不再需要 AMI 時，可予以取消註冊。



您可以搜尋符合執行個體條件的 AMI。您可以搜尋由社群提供的 AMI AWS 或 AMI 提供。如需詳細資訊，請參閱 [AMI 類型](#) 和 [尋找 AMI](#)。

從 AMI 啟動執行個體之後，您便可以連線至其中。當您連線至執行個體時，使用方式如同其他伺服器。如需啟動、連線和使用執行個體的資訊，請參閱 [開始使用 Amazon EC2](#)。

建立自己的 AMI

您可以從現有 AMI 啟動執行個體、自訂執行個體 (例如，在執行個體上安裝軟體)，然後將此更新的設定儲存為自訂 AMI。從這個新自訂 AMI 啟動的執行個體包括您建立 AMI 時所設的自訂。

執行個體的根儲存裝置會決定您建立 AMI 時需遵循的程序。執行個體的根磁碟區為 Amazon Elastic Block Store (Amazon EBS) 磁碟區或執行個體存放區磁碟區。如需根設備磁碟區的詳細資訊，請參閱 [Amazon EC2 執行個體根磁碟區](#)。

- 若要建立 Amazon EBS 後端 AMI，請參閱 [創建一個 Amazon EBS-backed AMI](#)。
- 若要建立執行個體後端 AMI，請參閱 [建立執行個體存放區後端 Linux AMI](#)。

若要協助分類和管理 AMI，您可以指派自訂標籤到 AMI。如需詳細資訊，請參閱 [標記您的 Amazon EC2 資源](#)。

購買、共享和銷售 AMI

創建 AMI 後，您可以將其保密，以便只有您可以使用它，或者您可以與指定的 AWS 帳戶列表共享它。您也可以將自訂 AMI 設為公有，讓社群能夠使用。只要遵循一些簡單準則，您即可經由簡單明瞭的程序建立供公眾取用的安全可用 AMI。如需如何建立和使用共用 AMI 的資訊，請參閱 [共享 AMI](#)。

您可以從第三方購買 AMI (包括如 Red Hat 等組織之服務合約隨附的 AMI)。您也可以建立 AMI，然後銷售給其他 Amazon EC2 使用者。如需購買或銷售 AMI 的詳細資訊，請參閱 [已支付 AMI](#)。

取消註冊您的 AMI

您可以在 AMI 使用完畢之後予以取消註冊。在您取消註冊 AMI 之後，將無法用以啟動新的執行個體。從 AMI 啟動的現有執行個體不會受影響。如需詳細資訊，請參閱 [取消註冊 \(刪除\) AMI](#)。

Amazon Linux 2023 和 Amazon Linux 2

Amazon Linux 的最新版本 AL2023 已針對 Amazon EC2 最佳化，免費供 Amazon EC2 使用者使用。AL2023 的特色包括可預期的發佈節奏、頻繁更新和長期支援。

如需有關 AL2023 特色和啟動 AL2023 AMI 的詳細資訊，請參閱：

- [AL2023 特色](#)
- [開始使用 AL2023](#)

Amazon Linux 2 (AL2) 為在亞馬 Amazon EC2 上執行的應用程式提供穩定、安全且高效能的執行環境。有關 Amazon Linux 2 的更多信息，請參閱 [Amazon Linux 2 用戶指南中的 Amazon Amazon EC2 上的亞馬遜 Linux 2](#)。

Note

Amazon Linux AMI 於 2023 年 12 月 31 日 end-of-life 到期，自 2024 年 1 月 1 日起將不會收到任何安全更新或錯誤修復。如需有關 Amazon Linux AMI end-of-life 和維護支援的詳細資訊，請參閱部落格文章 [Amazon Linux AMI 上的更新 end-of-life](#)。建議您將應用程式升級至 AL2023，其中包括直到 2028 年的長期支援。

視窗 AMI

AWS 提供一組公開可用的 AMI，其中包含特定於 Windows 平台的軟體設定。您可以使用這些 AMI 在 Amazon EC2 上快速開始建置和部署應用程式。請先選擇符合特定要求的 AMI，然後使用該 AMI 來啟動執行個體。您可以擷取管理員帳戶的密碼，然後使用遠端桌面連線登入執行個體，如同使用其他 Windows 伺服器。如需有關 AWS 視窗 AMI 的詳細資訊，請參閱 [AWS 視窗 AMI 參考資料](#)。

當您從 Windows AMI 啟動執行個體時，Windows 執行個體的根裝置為 Amazon Elastic Block Store (Amazon EBS) 磁碟區。Windows AMI 不支援根裝置的執行個體存放區。

已設定為使用 EC2 快速啟動更快速啟動的 Windows AMI 會預先佈建，使用快照啟動執行個體的速度提高達 65%。若要進一步了解 EC2 快速啟動，請參閱 [為您的執行個體使用 EC2 快速啟動](#)。

Note

Microsoft 不再支持視窗服務器 2016 之前的版本視窗服務器。建議您使用支援的 Windows Server 版本來啟動新的 EC2 執行個體。如果您現有之 EC2 執行個體執行的是未受支援 Windows Server 版本，建議您將這些執行個體升級為受支援的 Windows Server 版本。如需詳細資訊，請參閱 [將 Amazon EC2 Windows 執行個體升級至較新版本的 Windows Server](#)。

AMI 類型

您可以根據下列特性來選取要使用的 AMI：

- 區域 (請參閱 [區域 \(Region\)](#) 和 [區域 \(Zone\)](#))
- 作業系統
- 架構 (32 位元或 64 位元)
- [啟動許可](#)
- [根設備儲存](#)

啟動許可

AMI 的擁有者可以指定啟動許可來判定其可用性。啟動許可分為以下類別。

啟動許可	描述
public	擁有者會授與所有 AWS 帳戶的啟動權限。
explicit	擁有者將啟動權限授與特定 AWS 帳戶、組織或組織單位 (OU)。
implicit	擁有者具有 AMI 的絕對啟動許可。

Amazon 和 Amazon EC2 社群可大量選取公用 AMI。如需詳細資訊，請參閱 [共享 AMI](#)。開發人員可以收取其 AMI 的費用。如需詳細資訊，請參閱 [已支付 AMI](#)。

根設備儲存

所有 AMI 都會分類為「Amazon EBS 後端」或「執行個體存放區後端」。

- Amazon EBS 後端 AMI – 從 AMI 啟動之執行個體的根設備是從 Amazon EBS 快照建立的 Amazon Elastic Block Store (Amazon EBS) 磁碟區。同時支援 Linux 和視窗 AMI。
- Amazon 執行個體後端 AMI – 從 AMI 所啟動之執行個體的根設備是從 Amazon S3 中存放的範本建立的執行個體存放區磁碟區。僅支援 Linux 使用者介面卡。Windows AMI 不支援根裝置的執行個體存放區。

如需詳細資訊，請參閱 [Amazon EC2 執行個體根磁碟區](#)。

下表摘要說明使用兩種類型之 AMI 時的重要差異。

特性	Amazon EBS 後端 AMI	Amazon 執行個體後端 AMI
執行個體的開機時間	通常少於 1 分鐘	通常少於 5 分鐘
根設備的大小限制	64 TiB**	10 GiB
根設備磁碟區	EBS 磁碟區	執行個體存放區磁碟區
資料持久性	根據預設，該執行個體終止時會刪除根磁碟區。* 任何其他 EBS 磁碟	

特性	Amazon EBS 後端 AMI	Amazon 執行個體後端 AMI
	區上的資料在執行個體終止之後，預設為會持續留存。	只有在執行個體的生命週期，才會持續保存執行個體存放區磁碟區上的資料。
修改	在執行個體停止時，可以變更執行個體類型、核心、RAM 磁碟和使用者資料。	在執行個體的生命週期，執行個體屬性是固定的。
費用	會向您收取執行個體使用費、EBS 磁碟區使用費，以及將 AMI 存放為 EBS 快照的費用。	會向您收取執行個體使用費，以及在 Amazon S3 中存放 AMI 的費用。
AMI 建立/綁定	使用單一命令/呼叫	需要安裝和使用 AMI 工具
停止狀態	可以處於停止狀態。即使執行個體已停止且未執行，根磁碟區仍會保留在 Amazon EBS	無法進入停止狀態；執行個體正在執行或已終止

* 根據預設，EBS 根磁碟區會將 `DeleteOnTermination` 標記設定為 `true`。如需如何變更此標記，以便在終止之後持續保存磁碟區的資訊，請參閱 [將根磁碟區變更為持續存在](#)。

** 僅支援 `io2` EBS Block Express。如需詳細資訊，請參閱 Amazon EBS 使用者指南中的 [佈建 IOPS 固態硬碟區塊快速磁碟區](#)。

判定 AMI 的根設備類型

使用主控台判定 AMI 的根設備類型

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選取 AMIs，然後選取 AMI。
3. 在 Details (詳細資訊) 索引標籤中，檢查 Root device type (根設備類型) 的值，如下所示：
 - `ebs` – 這是 EBS 後端 AMI。
 - `instance store` – 這是執行個體後端 AMI。

使用命令列判定 AMI 的根設備類型

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [describe-images](#) (AWS CLI)
- [Get-EC2Image](#) (AWS Tools for Windows PowerShell)

停止狀態

您可以停止具有其根設備之 EBS 磁碟區的執行個體，但無法停止具有其根設備之執行個體存放區磁碟區的執行個體。

停止會讓執行個體停止執行 (其狀態會從 `running` 變為 `stopping`，再變為 `stopped`)。停止的執行個體會持續保存在 Amazon EBS 中，讓其能夠重新啟動。停止與終止不同；您無法重新啟動已終止的執行個體。由於無法停止具有根設備之執行個體存放區磁碟區的執行個體，因此其狀態不是執行中就是已終止。如需停止執行個體時會發生的狀況與因應方式的詳細資訊，請參閱 [停止並啟動 Amazon EC2 執行個體](#)。

預設資料儲存體和持久性

具有根設備之執行個體存放區磁碟區的執行個體，會自動具備可用的執行個體存放區 (根磁碟區包含根分割區，且您可以存放額外的資料)。您可以連接一或多個 EBS 磁碟區，以將持久性儲存體新增至執行個體。執行個體失敗或終止時，會刪除執行個體存放區磁碟區上的任何資料。如需詳細資訊，請參閱 [執行個體儲存體磁碟區和資料存留期](#)。

具有根設備之 Amazon EBS 的執行個體會自動連接 EBS 磁碟區。磁碟區會出現在磁碟區清單中，如同任何磁碟區。對於大部分執行個體類型，具有根設備之 EBS 磁碟區的執行個體預設沒有執行個體存放區磁碟區。您可以使用區塊型設備映射來新增執行個體存放區磁碟區或額外 EBS 磁碟區。如需詳細資訊，請參閱 [區塊型儲存設備對應](#)。

開機時間

從 Amazon EBS 後端 AMI 啟動之執行個體的啟動速度，會比從執行個體後端 AMI 啟動的執行個體還要快。當您從執行個體後端 AMI 啟動執行個體時，必須先從 Amazon S3 擷取所有組件，才能使用執行個體。使用 Amazon EBS 後端 AMI，只需要從快照擷取執行個體開機時所需的組件，才能使用執行個體。不過，使用 EBS 磁碟區做為根設備之執行個體的效能會變慢一段時間，因為其餘的組件正同時從快照擷取並載入磁碟區。當您停止並重新啟動執行個體時，執行個體會快速啟動，因為其狀態存放在 EBS 磁碟區中。

建立 AMI

若要建立執行個體存放區後端 Linux AMI，您必須於執行個體本身使用 Amazon EC2 AMI 工具，從執行個體建立 AMI。請注意，Windows AMI 不支援根裝置的執行個體存放區。

Amazon EBS 後端 AMI 的 AMI 建立較為簡單。CreateImage API 動作會建立並註冊 Amazon EBS 後端 AMI。中還有一個按鈕，可 AWS Management Console 讓您從執行中的執行個體建立 AMI。如需詳細資訊，請參閱 [創建一個 Amazon EBS-backed AMI](#)。

收費方式

使用執行個體存放區後端 AMI，會向您收取執行個體使用費，以及在 Amazon S3 中存放 AMI 的費用。使用 Amazon EBS 後端 AMI，會向您收取執行個體使用費、EBS 磁碟區儲存和使用費，以及將 AMI 以 EBS 快照存放的費用。

使用 Amazon EC2 執行個體後端 AMI，每次您自訂 AMI 和建立新的 AMI 時，每個 AMI 的所有組件都會存放在 Amazon S3 中。因此，每個自訂 AMI 的儲存足跡會是 AMI 的完整大小。對於 Amazon EBS 後端 AMI，每次您自訂 AMI 和建立新的 AMI 時，只會存放變更。因此，您在第一個 AMI 後自訂之後續 AMI 的儲存足跡會比較小，而產生較低的 AMI 儲存費用。

具有其根設備之 EBS 磁碟區的執行個體停止時，不會向您收取執行個體使用費；不過，仍然會向您收取磁碟區儲存費用。一旦您開始執行個體，我們最少就會收取一分鐘的使用費用。在一分鐘之後，我們只會依使用秒數收取費用。例如，如果您執行執行個體 20 秒，然後予以停止，我們會收取完整一分鐘的費用。如果您執行執行個體 3 分 40 秒，我們只會收取 3 分 40 秒的費用。我們會在執行個體運作時持續以秒計費 (最少為一分鐘)，縱使執行個體處於閒置狀態且您未連線仍會向您收費。

AMI 虛擬化類型

Amazon Machine Image 會使用兩種虛擬化類型的其中之一：全虛擬化 (PV) 或硬體虛擬機器 (HVM)。PV 和 HVM AMI 之間的主要區別在於開機的方式以及是否可以利用特殊的硬體延伸 (CPU、網路和儲存) 來獲得更好的效能。Windows AMI 為 HVM AMI。

為了獲得最佳效能，我們建議您在啟動執行個體時使用最新一代執行個體類型和 HVM AMI。如需最新一代執行個體類型的詳細資訊，請參閱 [Amazon EC2 執行個體類型](#)。如果您正在使用上一代執行個體類型並想要升級，請參閱 [升級路徑](#) 和 [變更執行個體類型](#)。

下表比較 HVM 和 PV AMI。

	HVM	PV
描述	<p>透過執行映像之根區塊型儲存設備的主開機記錄，HVM AMI 會以完全虛擬化的一組硬體呈現並開機。這種虛擬化類型提供在沒有任何修改的情況下直接在虛擬機器上執行作業系統之能力，就像執行在裸機硬體上一樣。Amazon EC2 主機系統模擬提供給訪客部分或全部底層硬體。</p>	<p>PV AMIs 使用稱為 PV-GRUB 的特殊開機載入器開機，該開機載入器會啟動開機週期，然後將 menu.lst 檔案中指定的核心載入映像中。全虛擬化訪客可以在不明確支援虛擬化的主機硬體上執行。歷史上，在許多情況下，PV 訪客比 HVM 訪客具有更好的效能，但由於 HVM 虛擬化中的增強以及 HVM AMI 之 PV 驅動程式的可用性，這已不再成立。如需有關 PV-GRUB 及其在 Amazon EC2 中使用的詳細資訊，請參閱使用者提供的核心。</p>
硬體延伸支援	<p>是。與 PV 訪客不同，HVM 訪客可以利用硬體延伸來快速存取主機系統上的底層硬體。如需 Amazon EC2 中可用之 CPU 虛擬化擴充功能的詳細資訊，請參閱 Intel 網站上的 Intel 虛擬化技術。</p> <p>需要 HVM AMI 才能充分利用增強聯網與 GPU 處理。為了將指令傳遞給專用網路和 GPU 裝置，作業系統需要能夠存取本地硬體平台；HVM 虛擬化提供此存取。如需詳細資訊，請參閱 Amazon EC2 上的增強型聯網。</p>	<p>不支援，它們無法利用特殊的硬體延伸，例如增強聯網或 GPU 處理。</p>
支援的執行個體類型	<p>所有目前世代的執行個體類型都支援 HVM AMI。</p>	<p>下列的前代執行個體類型支援 PV AMI：C1、C3</p>

	HVM	PV
		、M1、M3、M2 和 T1。最新一代執行個體類型不支援 PV AMI。
支援的區域	所有區域皆可支援 HVM 執行個體。	亞太區域 (東京)、亞太區域 (新加坡)、亞太區域 (雪梨)、歐洲 (法蘭克福)、歐洲 (愛爾蘭)、南美洲 (聖保羅)、US East (N. Virginia)、美國西部 (加州北部) 和美國西部 (奧勒岡)
如何尋找	使用主控台或 describe-images 命令來確認 AMI 的虛擬化類型已設定為 hvm。如需詳細資訊，請參閱 尋找 AMI 。	使用主控台或 describe-images 命令來確認 AMI 的虛擬化類型已設定為 paravirtual。如需詳細資訊，請參閱 尋找 AMI 。

HVM 上的 PV

全虛擬化訪客傳統上在儲存和網路操作效能上較 HVM 訪客更佳，因為他們可以利用特殊的 I/O 驅動程式避免模擬網路和硬碟硬體的コスト，而 HVM 訪客必須將這些指令轉換為模擬硬體。現在，PV 驅動程式可用於 HVM 訪客，因此無法移植到全虛擬化環境中執行的作業系統，仍然可以藉此在儲存體和網路 I/O 中看到效能優勢。使用 HVM 驅動程式上的這些 PV，HVM 訪客可以取得與全虛擬化訪客相同或更佳的效能。

Amazon EC2 啟動模式

電腦開機時，第一個執行的軟體會負責初始化平台，並提供作業系統執行平台特定作業的介面。

在 Amazon EC2 中，支援兩種開機模式軟體：整合可延伸韌體介面 (UEFI) 和 Legacy BIOS。

AMI 上可能的開機模式參數

AMI 可以有下列其中一個開機模式參數值：uefi、legacy-bios 或 uefi-preferred。AMI 開機模式參數為選用。針對沒有開機模式參數的 AMI，從這些 AMI 啟動的執行個體會使用執行個體類型的預設開機模式值。

AMI 開機模式參數的用途

AMI 開機模式參數會向 Amazon EC2 發出訊號，這是在啟動執行個體時要使用的開機模式。當開機模式參數設定為 `uefi` 時，EC2 會嘗試在 UEFI 上啟動執行個體。如果作業系統未設定為支援 UEFI，則執行個體啟動將會失敗。

UEFI 偏好的開機模式參數

您可以使用 `uefi-preferred` 開機模式參數來建立同時支援 UEFI 和 Legacy BIOS 的 AMI。當開機模式參數設定為 `uefi-preferred`，並且如果執行個體類型支援 UEFI，則會在 UEFI 上啟動執行個體。如果執行個體類型不支援 UEFI，則會在 Legacy BIOS 上啟動執行個體。

Warning

某些功能 (例如 UEFI 安全開機) 僅適用於透過 UEFI 開機的執行個體。當您將 `uefi-preferred` AMI 開機模式參數與不支援 UEFI 的執行個體類型搭配使用時，執行個體會以 Legacy BIOS 的形式啟動，而且會停用與 UEFI 相關的功能。如果您依賴 UEFI 相關功能的可用性，則請將 AMI 開機模式參數設定為 `uefi`。

執行個體類型的預設開機模式

- Graviton 執行個體類型：UEFI
- Intel 和 AMD 執行個體類型：Legacy BIOS

在 UEFI 上執行 Intel 和 AMD 執行個體類型

[Most Intel and AMD instance types](#) 可以同時在 UEFI 和舊式 BIOS 上執行。若要使用 UEFI，您必須選取將開機模式參數設定為 `uefi` 或 `uefi-preferred` 的 AMI，且 AMI 中包含的作業系統必須設定為支援 UEFI。

開機模式主題

- [啟動執行個體](#)
- [確定 AMI 的開機模式參數](#)
- [確定執行個體類型的支援開機模式](#)
- [確定執行個體的開機模式](#)
- [確定作業系統的開機模式](#)

- [設定 AMI 的開機模式](#)
- [UEFI 變數](#)
- [UEFI 安全開機](#)

啟動執行個體

您可以在 UEFI 或 Legacy BIOS 開機模式下啟動執行個體。

主題

- [限制](#)
- [考量事項](#)
- [在 UEFI 上啟動執行個體的要求](#)

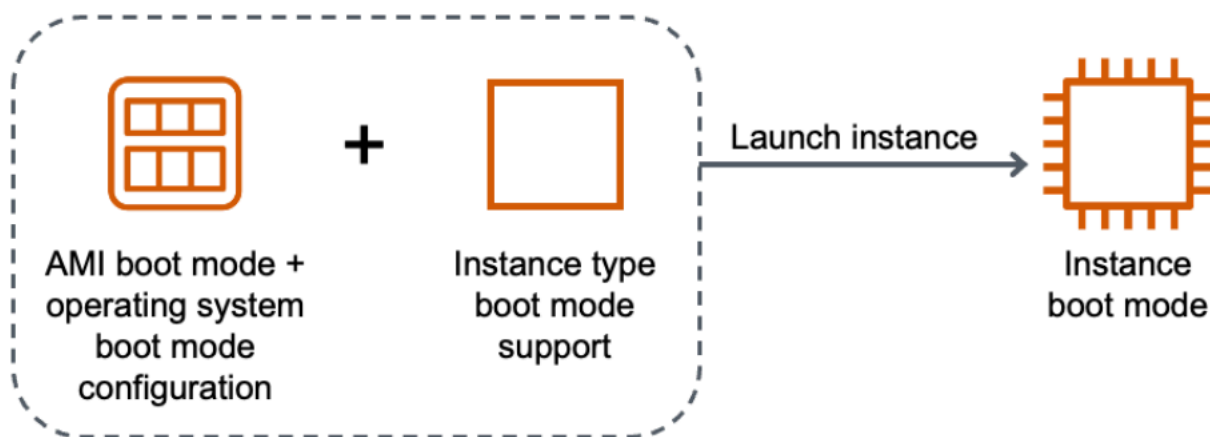
限制

在本機區域、Wavelength 區域中或使用 AWS Outposts 時不支援 UEFI 開機。

考量事項

啟動執行個體時請考慮下列情況：

- 執行個體的開機模式取決於 AMI 的組態、其中包含的作業系統以及執行個體類型，如下圖所示：



下表顯示執行個體的開機模式 (由產生的執行個體開機模式資料欄表示) 由 AMI 的開機模式參數 (資料欄 1)、AMI 所包含之作業系統的開機模式組態 (資料欄 2) 以及執行個體類型的開機模式支援 (資料欄 3) 的組合決定。

AMI 開機模式參數	作業系統開機模式組態	執行個體類型開機模式支援	產生的執行個體開機模式
UEFI	UEFI	UEFI	UEFI
Legacy BIOS	Legacy BIOS	Legacy BIOS	Legacy BIOS
UEFI Preferred	UEFI	UEFI	UEFI
UEFI Preferred	UEFI	UEFI 與 Legacy BIOS	UEFI
UEFI Preferred	Legacy BIOS	Legacy BIOS	Legacy BIOS
UEFI Preferred	Legacy BIOS	UEFI 與 Legacy BIOS	Legacy BIOS
未指定開機模式 - ARM	UEFI	UEFI	UEFI
未指定開機模式 - x86	Legacy BIOS	UEFI 與 Legacy BIOS	Legacy BIOS

- 預設開機模式：
 - Graviton 執行個體類型：UEFI
 - Intel 和 AMD 執行個體類型：Legacy BIOS
- 除 Legacy BIOS 之外，還支援 UEFI 的 Intel 和 AMD 執行個體類型：
 - 在 AWS 硝基系統上建置的所有執行個體，但裸機執行個體、DL1、G4ad、P4、u-3tb1、u-6tb1、u-9tb1、u-12tb1、u-18tb1、u-24 TB1 和 VT1 以外

查看特定區域中支援 UEFI 的可用執行個體類型

可用的執行個體類型因 AWS 區域而異。若要查看區域中支援 UEFI 的可用執行個體類型，請使用 [describe-instance-types](#) 命令，搭配 `--region` 參數。如果您省略 `--region` 參數，則會在請求中使用 [預設區域](#)。包含 `--filters` 參數以將結果範圍限定為支援 UEFI 的執行個體類型，以及包含 `--query` 參數以將輸出範圍限定為 InstanceType 的值。

對您的作業系統使用此指令。

Linux

AWS CLI

```
$ aws ec2 describe-instance-types --filters Name=supported-boot-mode,Values=uefi --query "InstanceTypes[*].[InstanceType]" --output text | sort
```

```
a1.2xlarge  
a1.4xlarge  
a1.large  
a1.medium  
a1.metal  
a1.xlarge  
c5.12xlarge  
...
```

PowerShell

```
PS C:\> Get-EC2InstanceType | `
  Where-Object {$_.SupportedBootModes -Contains "uefi"} | `
  Sort-Object InstanceType | `
  Format-Table InstanceType -GroupBy CurrentGeneration
```

```
CurrentGeneration: False
```

```
InstanceType
```

```
-----
```

```
a1.2xlarge  
a1.4xlarge  
a1.large  
a1.medium  
a1.metal  
a1.xlarge
```

```
CurrentGeneration: True
```

```
InstanceType
```

```
-----
```

```
c5.12xlarge  
c5.18xlarge  
c5.24xlarge
```



```
c5.2xlarge
c5.4xlarge
c5.9xlarge
...
```

Windows

AWS CLI

```
$ aws ec2 describe-instance-types --filters Name=supported-boot-mode,Values=uefi
Name=processor-info.supported-architecture,Values=x86_64 --query "InstanceTypes[*].
[InstanceType]" --output text | sort
```

```
c5.12xlarge
c5.18xlarge
c5.24xlarge
c5.2xlarge
c5.4xlarge
c5.9xlarge
c5.large
...
```

PowerShell

```
PS C:\> Get-EC2InstanceType | `
Where-Object {
    $_.SupportedBootModes -Contains "uefi" -and `
    $_.ProcessorInfo.SupportedArchitectures -eq "x86_64"
} | `
Sort-Object InstanceType | `
Format-Table InstanceType -GroupBy CurrentGeneration
```

```
CurrentGeneration: True
```

```
InstanceType
-----
c5.12xlarge
c5.18xlarge
c5.24xlarge
c5.2xlarge
c5.4xlarge
...
```

查看支援 UEFI 安全開機並在特定區域中保留非揮發性變數的可用執行個體類型

目前，裸機執行個體不支援 UEFI 安全開機和非揮發性變數。如上述範例中所述使用 [describe-instance-types](#) 命令，但是透過包含 `Name=bare-metal,Values=false` 篩選條件以篩選掉裸機執行個體。如需有關 UEFI 安全開機的資訊，請參閱 [UEFI 安全開機](#)。

對您的作業系統使用此指令。

Linux

AWS CLI

```
$ aws ec2 describe-instance-types --filters Name=supported-boot-mode,Values=uefi
Name=bare-metal,Values=false --query "InstanceTypes[*].[InstanceType]" --output
text | sort
```

```
a1.2xlarge
a1.4xlarge
a1.large
a1.medium
...
```

PowerShell

```
PS C:\> Get-EC2InstanceType | `
  Where-Object { `
    $_.SupportedBootModes -Contains "uefi" -and `
    $_.BareMetal -eq $False
  } | `
  Sort-Object InstanceType | `
  Format-Table InstanceType, SupportedBootModes, BareMetal,
  @{Name="SupportedArchitectures";
  Expression={$_.ProcessorInfo.SupportedArchitectures}}
```

InstanceType	SupportedBootModes	BareMetal	SupportedArchitectures
a1.2xlarge	{uefi}	False	arm64
a1.4xlarge	{uefi}	False	arm64
a1.large	{uefi}	False	arm64
a1.medium	{uefi}	False	arm64

a1.xlarge	{uefi}	False	arm64
c5.12xlarge	{legacy-bios, uefi}	False	x86_64
c5.18xlarge	{legacy-bios, uefi}	False	x86_64

Windows

AWS CLI

```
$ aws ec2 describe-instance-types --filters Name=supported-boot-
mode,Values=uefi Name=bare-metal,Values=false Name=processor-info.supported-
architecture,Values=x86_64 --query "InstanceTypes[*].[InstanceType]" --output text |
sort
```

```
c5.12xlarge
c5.18xlarge
c5.24xlarge
c5.2xlarge
...
```

PowerShell

```
PS C:\> Get-EC2InstanceType | `
  Where-Object { `
    $_.SupportedBootModes -Contains "uefi" -and `
    $_.BareMetal -eq $False -and `
    $_.ProcessorInfo.SupportedArchitectures -eq "x86_64" `
  } | `
  Sort-Object InstanceType | `
  Format-Table InstanceType, SupportedBootModes, BareMetal,
  @{Name="SupportedArchitectures";
  Expression={$_.ProcessorInfo.SupportedArchitectures}}
```

InstanceType	SupportedBootModes	BareMetal	SupportedArchitectures
c5.12xlarge	{legacy-bios, uefi}	False	x86_64
c5.18xlarge	{legacy-bios, uefi}	False	x86_64
c5.24xlarge	{legacy-bios, uefi}	False	x86_64
c5.2xlarge	{legacy-bios, uefi}	False	x86_64
c5.4xlarge	{legacy-bios, uefi}	False	x86_64
c5.9xlarge	{legacy-bios, uefi}	False	x86_64

在 UEFI 上啟動執行個體的要求

若要以 UEFI 開機模式啟動執行個體，您必須選取支援 UEFI 的執行個體類型，並針對 UEFI 設定 AMI 和作業系統，如下所示：

執行個體類型

啟動執行個體時，您必須選取支援 UEFI 的執行個體類型。如需詳細資訊，請參閱 [確定執行個體類型的支援開機模式](#)。

AMI

啟動執行個體時，您必須選取針對 UEFI 設定的 AMI。AMI 必須如下所示進行設定：

- 作業系統 - AMI 中包含的作業系統必須設定為使用 UEFI；否則，執行個體啟動將會失敗。如需詳細資訊，請參閱 [確定作業系統的開機模式](#)。
- AMI 開機模式參數 - AMI 的開機模式參數必須設定為 `uefi` 或 `uefi-preferred`。如需詳細資訊，請參閱 [確定 AMI 的開機模式參數](#)。

Linux — AWS 僅提供設定為支援以重力為基礎的執行個體類型 UEFI 的 Linux AMI。若要在其他 UEFI 執行個體類型上使用 Linux，您必須 [設定 AMI](#)、透過 [虛擬機器匯入/匯入 AMI 匯入 AMI](#)，或透過 [CloudEndure](#) 匯入 AMI。

視窗 — 下列視窗 AMI 支援 UEFI：

- TPM-Windows_Server-2022-English-Full-Base
- TPM-Windows_Server-2022-English-Core-Base
- TPM-Windows_Server-2019-English-Full-Base
- TPM-Windows_Server-2019-English-Core-Base
- TPM-Windows_Server-2016-English-Full-Base
- TPM-Windows_Server-2016-English-Core-Base

確定 AMI 的開機模式參數

AMI 開機模式參數為選用。AMI 可以有下列其中一個開機模式參數值：`uefi`、`legacy-bios` 或 `uefi-preferred`。

有些 AMI 沒有開機模式參數。當 AMI 沒有開機模式參數時，從 AMI 啟動的執行個體會使用執行個體類型的預設值，即 Graviton 上的 `uefi` 以及 Intel 和 AMD 執行個體類型上的 `legacy-bios`。

Console

確定 AMI (主控台) 的開機模式參數

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選取 AMI，然後選取該 AMI。
3. 檢查開機模式欄位。
 - 顯示 uefi 值表示 AMI 支援 UEFI。
 - 顯示 uefi-preferred 值表示 AMI 支援 UEFI 和 Legacy BIOS。
 - 如果未顯示任何值，表示透過 AMI 啟動的執行個體會使用其執行個體類型的預設值。

啟動執行個體時確定 AMI 的開機模式參數 (主控台)

使用啟動執行個體精靈啟動執行個體時，在選取 AMI 的步驟中檢查 Boot mode (開機模式) 欄位。如需詳細資訊，請參閱 [應用程式和作業系統映像 \(Amazon Machine Image\)](#)。

AWS CLI

確定 AMI 的開機模式參數 (AWS CLI)

使用 [describe-images](#) 操作來確定 AMI 的開機模式。

```
aws ec2 describe-images --region us-east-1 --image-id ami-0abcdef1234567890

{
  "Images": [
    {
      ...
    ],
    "EnaSupport": true,
    "Hypervisor": "xen",
    "ImageOwnerAlias": "amazon",
    "Name": "UEFI_Boot_Mode_Enabled-Windows_Server-2016-English-Full-Base-2020.09.30",
    "RootDeviceName": "/dev/sda1",
    "RootDeviceType": "ebs",
    "SriovNetSupport": "simple",
    "VirtualizationType": "hvm",
    "BootMode":
"uefi"
  }
}
```

```
]
}
```

在輸出中，`BootMode` 欄位會指示 AMI 的開機模式。`uefi` 的值表示 AMI 支援 UEFI。`uefi-preferred` 的值表示 AMI 支援 UEFI 和 Legacy BIOS。如果未顯示任何值，表示透過 AMI 啟動的執行個體會使用其執行個體類型的預設值。

PowerShell

要確定 AMI 的啟動模式參數 (工具 PowerShell)

請使用 [Get-EC2Image](#) Cmdlet 來確定 AMI 的開機模式。

```
PS C:\> Get-EC2Image -Region us-east-1 -ImageId ami-0abcdef1234567890 | Format-List
Name, BootMode, TpmSupport

Name      : TPM-Windows_Server-2016-English-Full-Base-2023.05.10
BootMode  : uefi
TpmSupport : v2.0
```

在輸出中，`BootMode` 欄位會指示 AMI 的開機模式。`uefi` 的值表示 AMI 支援 UEFI。`uefi-preferred` 的值表示 AMI 支援 UEFI 和 Legacy BIOS。如果未顯示任何值，表示透過 AMI 啟動的執行個體會使用其執行個體類型的預設值。

確定執行個體類型的支援開機模式

您可以使用 AWS CLI 或 [工具] PowerShell 來判斷執行個體類型支援的開機模式。

確定執行個體類型的支援開機模式

您可以使用下列方法來確定執行個體類型的支援開機模式。

AWS CLI

您可以使用 [describe-instance-types](#) 命令來確定執行個體類型的支援開機模式。透過包含 `--query` 參數，您可以篩選輸出。在此範例中，系統會篩選輸出，僅傳回支援的開機模式。

下列範例顯示 `m5.2xlarge` 同時支援 UEFI 和 Legacy BIOS 開機模式。

```
aws ec2 describe-instance-types --region us-east-1 --instance-types m5.2xlarge --
query "InstanceTypes[*].SupportedBootModes"
```

預期的輸出結果：

```
[
  [
    "legacy-bios",
    "uefi"
  ]
]
```

下列範例顯示 t2.xlarge 僅支援 Legacy BIOS。

```
aws ec2 describe-instance-types --region us-east-1 --instance-types t2.xlarge --
query "InstanceTypes[*].SupportedBootModes"
```

預期的輸出結果：

```
[
  [
    "legacy-bios"
  ]
]
```

PowerShell

您可以使用 [Get-EC2InstanceType](#)(Tools for PowerShell) 指令程式來判斷執行個體類型支援的開機模式。

下列範例顯示 m5.2xlarge 同時支援 UEFI 和 Legacy BIOS 開機模式。

```
Get-EC2InstanceType -Region us-east-1 -InstanceType m5.2xlarge | Format-List
InstanceType, SupportedBootModes
```

預期的輸出結果：

```
InstanceType      : m5.2xlarge
SupportedBootModes : {legacy-bios, uefi}
```

下列範例顯示 t2.xlarge 僅支援 Legacy BIOS。

```
Get-EC2InstanceType -Region us-east-1 -InstanceType t2.xlarge | Format-List
InstanceType, SupportedBootModes
```

預期的輸出結果：

```
InstanceType      : t2.xlarge
SupportedBootModes : {legacy-bios}
```

確定執行個體的開機模式

執行個體的開機模式會顯示在 Amazon EC2 主控台的開機模式欄位中，並透過 AWS CLI 中的 `currentInstanceBootMode` 參數顯示。

啟動執行個體時，其啟動模式參數的值取決於用於將其啟動的 AMI 開機模式參數的值，如下所示：

- 具有 `uefi` 開機模式參數的 AMI 會建立 `currentInstanceBootMode` 參數為 `uefi` 的執行個體。
- 具有 `legacy-bios` 開機模式參數的 AMI 會建立 `currentInstanceBootMode` 參數為 `legacy-bios` 的執行個體。
- 如果執行個體類型支援 UEFI，則開機模式參數為 `uefi-preferred` 的 AMI 會建立 `currentInstanceBootMode` 參數為 `uefi` 的執行個體；否則，其會建立 `currentInstanceBootMode` 參數為 `legacy-bios` 的執行個體。
- 沒有開機模式參數值的 AMI 會建立具有 `currentInstanceBootMode` 參數值的執行個體，該參數值取決於 AMI 架構是 ARM 還是 x86，以及支援的執行個體類型開機模式。Graviton 執行個體類型上的預設開機模式為 `uefi`，Intel 和 AMD 執行個體類型上的預設開機模式為 `legacy-bios`。

Console

確定執行個體的開機模式 (主控台)

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取您的執行個體。
3. 在 Details (詳細資訊) 標籤上，檢查 Boot mode (開機模式) 欄位。

AWS CLI

確定執行個體的開機模式 (AWS CLI)

使用 [describe-instances](#) 命令確定執行個體的開機模式。您也可以判斷用來建立執行個體之 AMI 的開機模式。


```
aws ec2 describe-instances --region us-east-1 --instance-ids i-1234567890abcdef0

{
  "Reservations": [
    {
      "Groups": [],
      "Instances": [
        {
          "AmiLaunchIndex": 0,
          "ImageId": "ami-0e2063e7f6dc3bee8",
          "InstanceId": "i-1234567890abcdef0",
          "InstanceType": "m5.2xlarge",
          ...
        },
        {
          "BootMode": "uefi",
          "CurrentInstanceBootMode": "uefi"
        }
      ],
      "OwnerId": "1234567890",
      "ReservationId": "r-1234567890abcdef0"
    }
  ]
}
```

PowerShell

判斷執行個體的開機模式 (適用於的工具 PowerShell)

請使用 [Get-EC2Image](#) Cmdlet 來確定執行個體的開機模式。您也可以判斷用來建立執行個體之 AMI 的開機模式。

[Get-EC2Image](#) (AWS Tools for Windows PowerShell)

```
(Get-EC2Instance -InstanceId i-1234567890abcdef0).Instances | Format-List BootMode,
CurrentInstanceBootMode, InstanceType, ImageId
```

```
BootMode           : uefi
CurrentInstanceBootMode : uefi
InstanceType       : c5a.large
ImageId            : ami-0265446f88eb4021b
```

在輸出中，以下參數描述開機模式：

- `BootMode` - 用於建立執行個體之 AMI 的開機模式。
- `CurrentInstanceBootMode` - 用於在啟動或開始時讓執行個體開機的開機模式。

確定作業系統的開機模式

AMI 的開機模式會引導 Amazon EC2 使用哪種開機模式來啟動執行個體。若要檢視執行個體的作業系統是否設定為 UEFI，您必須使用安全殼層 (Linux 執行個體) 或 RDP (Windows 執行個體) 連線至執行個體。

請遵循於執行個體作業系統的指示。

Linux

確定執行個體作業系統的開機模式

1. [使用 SSH 連線至您的 Linux 執行個體](#)。
2. 若要檢視作業系統的開機模式，請嘗試下列其中一個動作：
 - 執行下列命令。

```
[ec2-user ~]$ sudo /usr/sbin/efibootmgr
```

在 UEFI 開機模式下啟動的執行個體預期輸出

```
BootCurrent: 0001
Timeout: 0 seconds
BootOrder: 0000,0001
Boot0000* UiApp
Boot0001* UEFI Amazon Elastic Block Store vol-xyz
```

- 執行下列命令，以驗證 `/sys/firmware/efi` 目錄是否存在。只有在執行個體使用 UEFI 開機時，才會存在此目錄。如果此目錄不存在，命令會傳回 `Legacy BIOS Boot Detected`。

```
[ec2-user ~]$ [ -d /sys/firmware/efi ] && echo "UEFI Boot Detected" || echo "Legacy BIOS Boot Detected"
```

在 UEFI 開機模式下啟動的執行個體預期輸出

```
UEFI Boot Detected
```

在 Legacy BIOS 開機模式下啟動的執行個體預期輸出

```
Legacy BIOS Boot Detected
```

- 執行下列命令，以驗證 EFI 是否出現在 dmesg 輸出中。

```
[ec2-user ~]$ dmesg | grep -i "EFI"
```

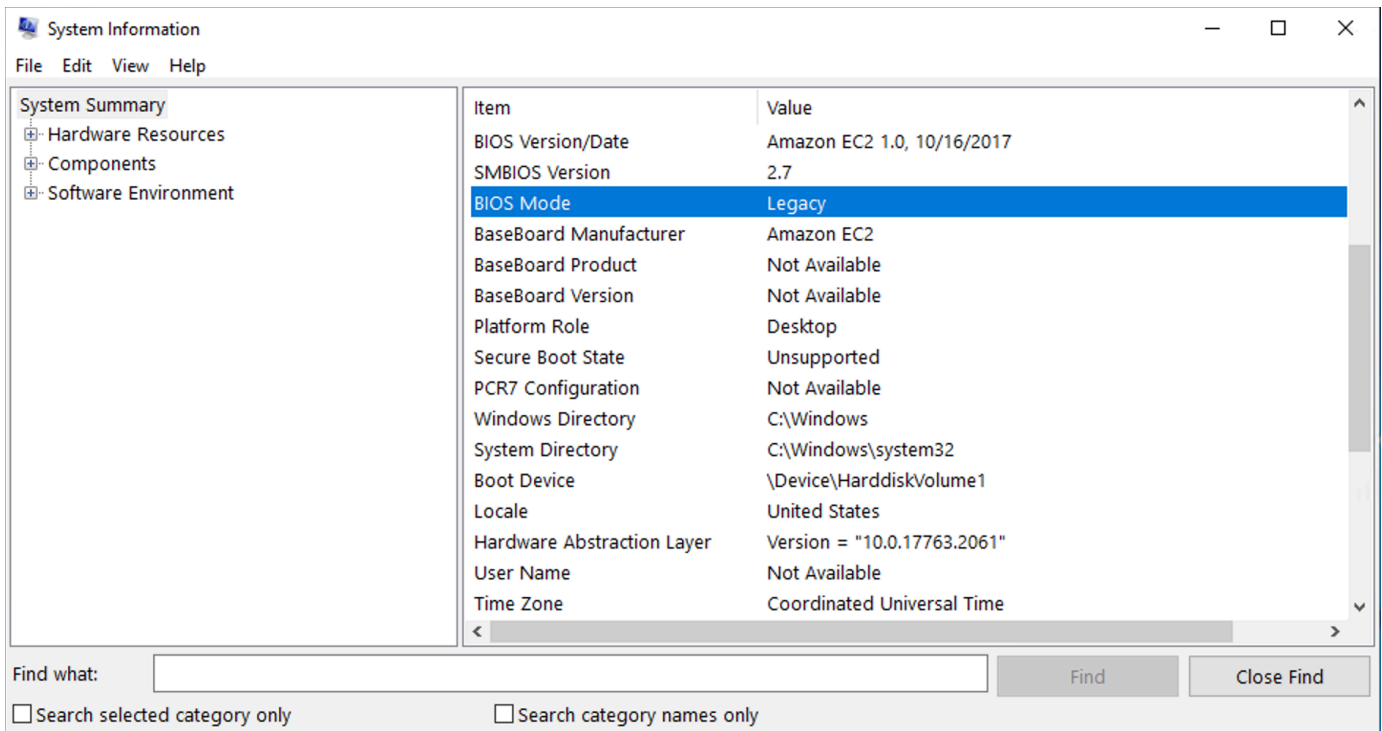
在 UEFI 開機模式下啟動的執行個體預期輸出

```
[ 0.000000] efi: Getting EFI parameters from FDT:  
[ 0.000000] efi: EFI v2.70 by EDK II
```

Windows

確定執行個體作業系統的開機模式

- [使用 RDP 連線至您的 Windows 執行個體。](#)
- 請移至 System Information (系統資訊)，然後檢查 BIOS Mode (BIOS 模式) 列。



設定 AMI 的開機模式

當您使用 [register-image](#) 命令建立 AMI 時，可以將 AMI 的開機模式設定為 `uefi`、`legacy-bios` 或 `uefi-preferred`。

當 AMI 開機模式設為 `uefi-preferred` 時，執行個體會以下列方式開機：

- 對於同時支援 UEFI 和 Legacy BIOS (例如 `m5.large`) 的執行個體類型，執行個體會使用 UEFI 開機。
- 對於僅支援 Legacy BIOS (例如 `m4.large`) 的執行個體類型，執行個體會使用 Legacy BIOS 開機。

Note

如果您將 AMI 開機模式設定為 `uefi-preferred`，則作業系統必須支援 UEFI 和 Legacy BIOS 開機功能。

目前，您無法使用 [register-image](#) 命令來建立同時支援 [NitroTPM](#) 和 UEFI Preferred 的 AMI。

Warning

某些功能 (例如 UEFI 安全開機) 僅適用於透過 UEFI 開機的執行個體。當您將 `uefi-preferred` AMI 開機模式參數與不支援 UEFI 的執行個體類型搭配使用時，執行個體會以 Legacy BIOS 的形式啟動，而且會停用與 UEFI 相關的功能。如果您依賴 UEFI 相關功能的可用性，則請將 AMI 開機模式參數設定為 `uefi`。

若要將現有的 Legacy BIOS 型執行個體轉換為 UEFI，或將現有的 UEFI 型執行個體轉換為 Legacy BIOS，您需要執行數個步驟：首先，修改執行個體的磁碟區和作業系統，以支援選取的開機模式。然後，建立磁碟區的快照。最後，使用 [register-image](#) 來透過快照建立 AMI。

您無法使用 [create-image](#) 命令來設定 AMI 的開機模式。使用 [create-image](#)，AMI 會繼承用於建立 AMI 的 EC2 執行個體的開機模式。例如，如果您從在 Legacy BIOS 上執行的 EC2 執行個體建立 AMI，AMI 開機模式將設定為 `legacy-bios`。如果您從使用 AMI 啟動且開機模式設定為 `uefi-preferred` 的 EC2 執行個體建立 AMI，則建立的 AMI 也會將其開機模式設定為 `uefi-preferred`。

⚠ Warning

設定 AMI 開機模式參數並不會自動針對指定的開機模式設定作業系統。在繼續執行這些步驟之前，您必須先對執行個體的磁碟區和作業系統進行適當的修改，以支援使用選取的開機模式開機；否則，產生的 AMI 將無法使用。例如，如果您要將舊版 BIOS 型 Windows 執行個體轉換為 UEFI，您可以使用 Microsoft 提供的 [MBR2GPT](#) 工具將系統磁碟從 MBR 轉換為 GPT。所需的修改視作業系統而異。如需詳細資訊，請參閱作業系統的手冊。

設定 AMI 的開機模式 (AWS CLI)

1. 對執行個體的磁碟區和作業系統進行適當的修改，以支援透過選取的開機模式開機。所需的修改視作業系統而異。如需詳細資訊，請參閱作業系統的手冊。

i Note

如果您不執行此步驟，則 AMI 將無法使用。

2. 若要尋找執行個體的磁碟區 ID，請使用 [describe-instances](#) 命令。您將在下一個步驟中建立此磁碟區的快照。

```
aws ec2 describe-instances --region us-east-1 --instance-ids i-1234567890abcdef0
```

預期的輸出結果

```
...
    "BlockDeviceMappings": [
      {
        "DeviceName": "/dev/sda1",
        "Ebs": {
          "AttachTime": "",
          "DeleteOnTermination": true,
          "Status": "attached",
          "VolumeId": "vol-1234567890abcdef0"
        }
      }
    ]
  ...
```

3. 若要建立磁碟區的快照，請使用 [create-snapshot](#) 命令。使用上一個步驟的磁碟區 ID。

```
aws ec2 create-snapshot --region us-east-1 --volume-id vol-1234567890abcdef0 --  
description "add text"
```

預期的輸出結果

```
{  
  "Description": "add text",  
  "Encrypted": false,  
  "OwnerId": "123",  
  "Progress": "",  
  "SnapshotId": "snap-01234567890abcdef",  
  "StartTime": "",  
  "State": "pending",  
  "VolumeId": "vol-1234567890abcdef0",  
  "VolumeSize": 30,  
  "Tags": []  
}
```

4. 請注意上一個步驟輸出中的快照 ID。
5. 等到快照建立 completed 後，再進行下一個步驟。若要查詢快照的狀態，請使用 [describe-snapshots](#) 命令。

```
aws ec2 describe-snapshots --region us-east-1 --snapshot-ids snap-01234567890abcdef
```

範例輸出

```
{  
  "Snapshots": [  
    {  
      "Description": "This is my snapshot",  
      "Encrypted": false,  
      "VolumeId": "vol-049df61146c4d7901",  
      "State": "completed",  
      "VolumeSize": 8,  
      "StartTime": "2019-02-28T21:28:32.000Z",  
      "Progress": "100%",  
      "OwnerId": "012345678910",  
      "SnapshotId": "snap-01234567890abcdef",  
      ...  
    }  
  ]  
}
```

6. 若要建立新 AMI，請使用 [register-image](#) 命令。使用您在之前步驟中記下的快照 ID。

- 若要將開機模式設定為 UEFI，請將 `--boot-mode` 參數新增至命令並將 `uefi` 指定為值。

```
aws ec2 register-image \  
  --region us-east-1 \  
  --description "add description" \  
  --name "add name" \  
  --block-device-mappings "DeviceName=/dev/  
sda1,Ebs={SnapshotId=snap-01234567890abcdef,DeleteOnTermination=true}" \  
  --architecture x86_64 \  
  --root-device-name /dev/sda1 \  
  --virtualization-type hvm \  
  --ena-support \  
  --boot-mode uefi
```

- 若要將開機模式設定為 `uefi-preferred`，請將 `--boot-mode` 參數新增至命令並將 `uefi-preferred` 指定為值。

```
aws ec2 register-image \  
  --region us-east-1 \  
  --description "add description" \  
  --name "add name" \  
  --block-device-mappings "DeviceName=/dev/  
sda1,Ebs={SnapshotId=snap-01234567890abcdef,DeleteOnTermination=true}" \  
  --architecture x86_64 \  
  --root-device-name /dev/sda1 \  
  --virtualization-type hvm \  
  --ena-support \  
  --boot-mode uefi-preferred
```

預期的輸出結果

```
{  
  "ImageId": "ami-new_ami_123"  
}
```

7. 若要驗證新建立的 AMI 具有您在上一個步驟中指定的開機模式，請使用 [describe-images](#) 命令。

```
aws ec2 describe-images --region us-east-1 --image-id ami-new_ami_123
```

預期的輸出結果

```
{
  "Images": [
    {
      "Architecture": "x86_64",
      "CreationDate": "2021-01-06T14:31:04.000Z",
      "ImageId": "ami-new_ami_123",
      "ImageLocation": "",
      ...
      "BootMode": "uefi"
    }
  ]
}
```

8. 使用新創建的 AMI 啟動新的執行個體。

如果 AMI 開機模式為 `uefi` 或 `legacy-bios`，則從此 AMI 建立的執行個體將具有與 AMI 相同的開機模式。如果 AMI 開機模式為 `uefi-preferred`，若執行個體類型支援 UEFI，執行個體將使用 UEFI 開機；否則，執行個體將使用 Legacy BIOS 開機。如需詳細資訊，請參閱 [考量事項](#)。

9. 若要驗證新執行個體是否具有預期的開機模式，請使用 [describe-instances](#) 命令。

UEFI 變數

啟動其中啟動模式設定為 UEFI 的執行個體時，就會建立變數的機碼值存放區。UEFI 和執行個體作業系統可以使用存放區來儲存 UEFI 變數。

開機載入器和作業系統會使用 UEFI 變數來設定早期系統啟動。它們允許作業系統管理開機程序的特定設定，例如開機順序，或管理 UEFI 安全開機的金鑰。

Warning

任何可以連線至執行個體的使用者 (也可能是執行個體上執行的任何軟體)，或任何擁有在執行個體上使用 [GetInstanceUefiData](#) API 權限的人都可以讀取變數。您決不應該將敏感資料 (例如密碼或個人識別資訊) 存放在 UEFI 變數存放區中。

UEFI 變數持久性

- 對於在 2022 年 5 月 10 日或之前啟動的執行個體，UEFI 變數將在重新開機或停止時被抹除。
- 對於在 2022 年 5 月 11 日或之後啟動的執行個體，標示為非揮發性的 UEFI 變數會在重新開機和停止/啟動時持續存在。
- 裸機執行個體不會在執行個體停止/啟動操作中保留 UEFI 非揮發性變數。

UEFI 安全開機

UEFI 安全開機建置於 Amazon EC2 的長期安全開機程序之上，並提供其他功能，協助客戶 defense-in-depth 保護軟體免於在重新開機期間持續存在的威脅。它確保執行個體僅對使用密碼編譯金鑰簽署的軟體進行開機。這些金鑰存放在 [UEFI 非揮發性變數存放區](#) 的金鑰資料庫中。UEFI 安全開機可避免對執行個體開機流程進行未經授權的修改。

主題

- [UEFI 安全開機的工作原理](#)
- [啟動具有 UEFI 安全開機支援的執行個體](#)
- [驗證執行個體是否已啟用 UEFI 安全開機](#)
- [建立 Linux AMI 以支援 UEFI 安全開機](#)
- [如何創建 AWS 二進制 blob](#)

UEFI 安全開機的工作原理

UEFI 安全開機是 UEFI 中指定的一項功能，它提供有關開機鏈狀態的驗證。該功能旨在確保韌體自我初始化後，只會執行以密碼編譯驗證的 UEFI 二進位檔案。這些二進位檔案包括 UEFI 驅動程式和主要開機載入器，以及鏈式載入元件。

UEFI 安全開機指定在信任鏈中使用的四個金鑰資料庫。資料庫存放在 UEFI 變數存放區中。

信任鏈如下：

平台金鑰 (PK) 資料庫

PK 資料庫是信任根。它包含在信任鏈中用於更新金鑰交換金鑰 (KEK) 資料庫的單一公有 PK 金鑰。

若要變更 PK 資料庫，您必須具有私有 PK 金鑰才能簽署更新要求。這包括透過寫入空 PK 金鑰來刪除 PK 資料庫。

金鑰交換金鑰 (KEK) 資料庫

KEK 資料庫是一份公有 KEK 金鑰清單，這些金鑰用於在信任鏈中更新簽章 (db) 和拒絕清單 (dbx) 資料庫。

若要變更公有 KEK 資料庫，您必須具有私有 PK 金鑰才能簽署更新要求。

簽章 (db) 資料庫

db 資料庫是一份公有金鑰和雜湊清單，這些金鑰和雜湊用於在信任鏈中驗證所有 UEFI 開機二進位檔案。

若要變更 db 資料庫，您必須具有私有 PK 金鑰或任何一項私有 KEK 金鑰才能簽署更新要求。

簽章拒絕清單 (dbx) 資料庫

dbx 資料庫是一份不受信任的公有金鑰和二進位檔案雜湊清單，在信任鏈中作為撤銷檔案使用。

dbx 資料庫始終優先於所有其他金鑰資料庫。

若要變更 dbx 資料庫，您必須具有私有 PK 金鑰或任何一項私有 KEK 金鑰才能簽署更新要求。

UEFI 論壇維護有適用於許多已知不良二進位檔案和憑證的公開可用的 dbx，網址為：<https://uefi.org/revocationlistfile>。

Important

UEFI 安全開機強制對任何 UEFI 二進位檔案執行簽章驗證。若要允許在 UEFI 安全開機中執行 UEFI 二進位檔案，請使用上述任何私有 db 金鑰予以簽署。

UEFI 安全開機預設為停用，且系統處於 SetupMode 模式。當系統處於 SetupMode 模式時，所有金鑰變數都可以在沒有密碼編譯簽章的情況下更新。設定 PK 時，會啟用 UEFI 安全開機並結束 SetupMode。

啟動具有 UEFI 安全開機支援的執行個體

當您[啟動執行個體](#) (具有下列先決條件) 時，執行個體將根據其 UEFI 安全開機資料庫自動驗證 UEFI 開機二進位檔案。您還可以在啟動後在執行個體上設定 UEFI 安全開機。

Note

UEFI 安全開機可保護您的執行個體及其作業系統免受開機流程修改的影響。通常，UEFI 安全開機設定為 AMI 的一部分。如果您使用與基本 AMI 不同的參數建立新 AMI，例如變更 AMI 中的 UefiData，您可以停用 UEFI 安全開機。

必要條件**AMI**

若要啟動 Linux 執行個體，Linux AMI 必須啟用 UEFI 安全開機。

Amazon Linux 從 AL2023 2023.1 版起開始支援 UEFI 安全開機。不過，AMI 在預設情況下並不會啟用 UEFI 安全開機。如需詳細資訊，請參閱《AL2023 使用者指南》中的 [UEFI 安全開機](#)。舊版 Amazon Linux AMI 並未啟用 UEFI 安全開機。若要使用支援的 AMI，必須在自有的 Linux AMI 上執行多個設定步驟。如需詳細資訊，請參閱 [建立 Linux AMI 以支援 UEFI 安全開機](#)。

視窗 AMI

若要啟動 Windows 執行個體，視窗 AMI 必須啟用 UEFI 安全開機。

下列 Windows AMI 已預先設定為使用 Microsoft 金鑰來啟用 UEFI 安全開機：

- TPM-Windows_Server-2022-English-Core-Base
- TPM-Windows_Server-2022-English-Full-Base
- TPM-Windows_Server-2022-English-Full-SQL_2022_Enterprise
- TPM-Windows_Server-2022-English-Full-SQL_2022_Standard
- TPM-Windows_Server-2019-English-Core-Base
- TPM-Windows_Server-2019-English-Full-Base
- TPM-Windows_Server-2019-English-Full-SQL_2019_Enterprise
- TPM-Windows_Server-2019-English-Full-SQL_2019_Standard
- TPM-Windows_Server-2016-English-Core-Base
- TPM-Windows_Server-2016-English-Full-Base

目前，我們不支援使用 [import-image](#) 命令來匯入具有 UEFI 安全開機的 Windows。

執行個體類型

- 支援：所有支援 UEFI 的虛擬化執行個體類型也支援 UEFI 安全開機。如需有關支援 UEFI 安全開機的執行個體類型詳細資訊，請參閱 [考量事項](#)。

- 不支援：裸機執行個體類型不支援 UEFI 安全開機。

驗證執行個體是否已啟用 UEFI 安全開機

Linux 執行個體

可使用 `mokutil` 公用程式來驗證 Linux 執行個體是否已啟用 UEFI 安全開機。如果執行個體上沒有安裝 `mokutil`，則必須安裝。如需 Amazon Linux 2 的安裝說明，請參閱<https://docs.aws.amazon.com/linux/al2/ug/find-install-software.html>。如需其他 Linux 發行版，請參閱其特定文件。

驗證 Linux 執行個體是否已啟用 UEFI 安全開機

在執行個體上將下列命令作為 `root` 執行。

```
mokutil --sb-state
```

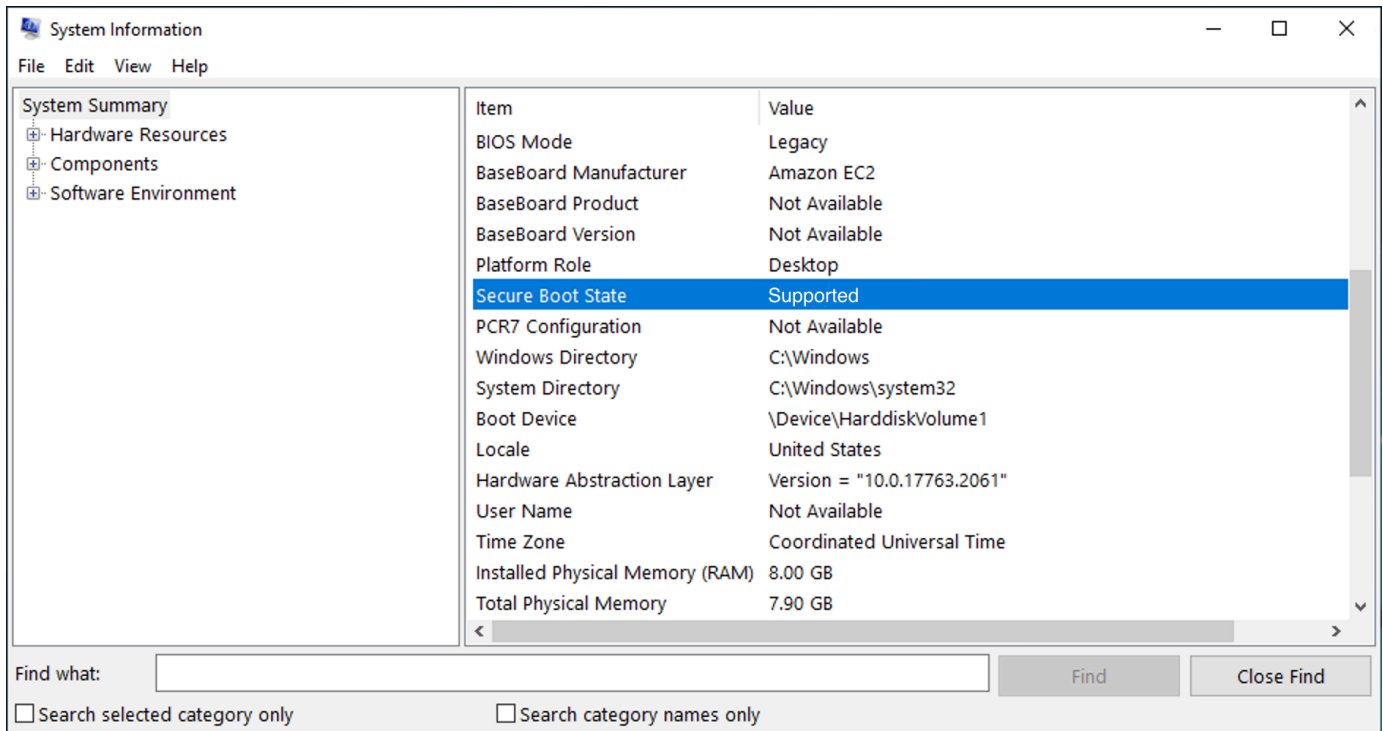
預期的輸出結果：

- 如果啟用了 UEFI 安全開機，則輸出會包含 `SecureBoot enabled`。
- 如果未啟用 UEFI 安全開機，則輸出會包含 `SecureBoot disabled` 或 `Failed to read SecureBoot`。

Windows 執行個體

驗證 Windows 執行個體是否已啟用 UEFI 安全開機

1. 開啟 `msinfo32` 工具。
2. 檢查 `Secure Boot State` (安全開機狀態) 欄位。Supported (已支援) 表示已啟用 UEFI 安全開機。



您也可以使用 Windows 指 PowerShell 令程式 `Confirm-SecureBootUEFI` 來檢查安全開機狀態。如需指令程式的相關資訊，請參閱 Microsoft 文件網站中的 [確認-SecureBoot UEFI](#)。

建立 Linux AMI 以支援 UEFI 安全開機

下列程序說明如何建立自己的 UEFI 變數存放區，以便使用自訂私有金鑰進行安全開機。Amazon Linux 從 AL2023 2023.1 版起開始支援 UEFI 安全開機。如需詳細資訊，請參閱《AL2023 使用者指南》中的 [UEFI 安全開機](#)。

⚠ Important

下列建立 AMI 以支援 UEFI 安全開機的程序僅供進階使用者使用。您必須具備充分的 SSL 和 Linux 發行版本開機流程的知識才能使用這些程序。

必要條件

- 將會使用下列工具：
 - OpenSSL - <https://www.openssl.org/>
 - efivar - <https://github.com/rhboot/efivar>
 - efitools - <https://git.kernel.org/pub/scm/linux/kernel/git/jejb/efitools.git/>

- [獲取實例 AWS CLI 例-數據命令](#)
- 您的 Linux 執行個體必須使用支援 UEFI 開機模式的 Linux AMI 啟動，並且有非揮發性資料存在。

沒有 UEFI 安全開機金鑰的新建立執行個體是在 SetupMode 中建立，允許您註冊自己的金鑰。某些 AMI 預先設定為具有 UEFI 安全開機功能，您無法變更現有金鑰。如果您想要變更金鑰，您必須根據原始 AMI 建立新 AMI。

您有兩種在變數存放區傳播金鑰的方法，這些方法在接下來的選項 A 和選項 B 中說明。選項 A 說明如何在執行個體中執行此操作，並模仿真實硬體的流程。選項 B 說明如何建立二進位檔案 Blob，其隨後會在建立 AMI 時作為 base64 編碼的檔案傳遞。對於這兩個選項，您必須先建立用於信任鏈的三對金鑰對。

若要建立支援 UEFI 安全開機的 Linux AMI，請先建立三個金鑰對，然後完成選項 A 或選項 B：

- [建立三個金鑰對](#)
- [選項 A：從執行個體中將金鑰新增至變數存放區](#)
- [選項 B：建立一個包含預先填寫的變數存放區的二進位 blob](#)

Note

這些指示只能用於建立 Linux AMI。如果您需要 Windows AMI，請使用其中一個支援的 Windows AMI。如需詳細資訊，請參閱 [啟動具有 UEFI 安全開機支援的執行個體](#)。

建立三個金鑰對

UEFI 安全開機基於下列用於信任鏈的三個金鑰資料庫：平台金鑰 (PK)、金鑰交換金鑰 (KEK) 和簽章資料庫 (db)。¹

您需在執行個體上建立每個金鑰。若要以對 UEFI 安全開機標準有效的格式準備公有金鑰，請為每個金鑰建立一個憑證。DER 定義 SSL 格式 (格式的二進位編碼)。然後，您可以將每個憑證轉換為 UEFI 簽章清單，該清單為 UEFI 安全開機中所理解的二進位格式。最後，您需使用相關金鑰簽署每個憑證。

主題

- [準備建立金鑰對](#)
- [金鑰對 1：建立平台金鑰 \(PK\)](#)
- [金鑰對 2：建立金鑰交換金鑰 \(KEK\)](#)

- [金鑰對 3：建立簽章資料庫 \(db\)](#)
- [使用私有金鑰簽署開機映像 \(核心\)](#)

準備建立金鑰對

建立金鑰對之前，請建立一個用於金鑰產生的全域唯一識別符 (GUID)。

1. [連線到執行個體。](#)
2. 在 shell 提示中執行下列命令。

```
uuidgen --random > GUID.txt
```

金鑰對 1：建立平台金鑰 (PK)

PK 是 UEFI 安全開機執行個體的信任根。私有 PK 用於更新 KEK，其又可接續用於將授權的金鑰新增至簽章資料庫 (db)。

X.509 標準用於建立金鑰對。如需有關標準的資訊，請參閱維基百科上的 [X.509](#)。

建立 PK

1. 建立金鑰。您必須將變數命名為 PK。

```
openssl req -newkey rsa:4096 -nodes -keyout PK.key -new -x509 -sha256 -days 3650 -  
subj "/CN=Platform key/" -out PK.crt
```

指定了下列參數：

- -keyout PK.key - 私有金鑰檔案。
 - -days 3650 - 憑證有效的天數。
 - -out PK.crt - 用於建立 UEFI 變數的憑證。
 - CN=*Platform key* - 金鑰的通用名稱 (CN)。您可以輸入自己的組織名稱，而不是####。
2. 建立憑證。

```
openssl x509 -outform DER -in PK.crt -out PK.cer
```

3. 將憑證轉換至 UEFI 簽章清單。

```
cert-to-efi-sig-list -g "$(< GUID.txt)" PK.crt PK.esl
```

4. 使用私有 PK (自我簽署) 簽署 UEFI 簽章清單。

```
sign-efi-sig-list -g "$(< GUID.txt)" -k PK.key -c PK.crt PK PK.esl PK.auth
```

金鑰對 2：建立金鑰交換金鑰 (KEK)

私有 KEK 用於將金鑰新增至 db，這是要在系統上開機使用的授權簽章清單。

建立 KEK

1. 建立金鑰。

```
openssl req -newkey rsa:4096 -nodes -keyout KEK.key -new -x509 -sha256 -days 3650 -  
subj "/CN=Key Exchange Key/" -out KEK.crt
```

2. 建立憑證。

```
openssl x509 -outform DER -in KEK.crt -out KEK.cer
```

3. 將憑證轉換至 UEFI 簽章清單。

```
cert-to-efi-sig-list -g "$(< GUID.txt)" KEK.crt KEK.esl
```

4. 使用私有 PK 簽署簽章清單。

```
sign-efi-sig-list -g "$(< GUID.txt)" -k PK.key -c PK.crt KEK KEK.esl KEK.auth
```

金鑰對 3：建立簽章資料庫 (db)

db 清單包含授權在系統上開機的授權金鑰。必須使用私有 KEK 才能修改清單。將使用在此步驟中建立的私有金鑰簽署開機映像。

建立 db

1. 建立金鑰。


```
openssl req -newkey rsa:4096 -nodes -keyout db.key -new -x509 -sha256 -days 3650 -  
subj "/CN=Signature Database key/" -out db.crt
```

2. 建立憑證。

```
openssl x509 -outform DER -in db.crt -out db.cer
```

3. 將憑證轉換至 UEFI 簽章清單。

```
cert-to-efi-sig-list -g "$(< GUID.txt)" db.crt db.esl
```

4. 使用私有 KEK 簽署簽章清單。

```
sign-efi-sig-list -g "$(< GUID.txt)" -k KEK.key -c KEK.crt db db.esl db.auth
```

使用私有金鑰簽署開機映像 (核心)

若為 Ubuntu 22.04，下列映像需要簽章。

```
/boot/efi/EFI/ubuntu/shimx64.efi  
/boot/efi/EFI/ubuntu/mmx64.efi  
/boot/efi/EFI/ubuntu/grubx64.efi  
/boot/vmlinuz
```

簽署映像

使用下列語法簽署映像。

```
sbsign --key db.key --cert db.crt --output /boot/vmlinuz /boot/vmlinuz
```

Note

您必須簽署所有新核心。*/boot/vmlinuz* 通常會以符號連結到上次安裝的核心。

請參閱您的發行版本說明文件，了解您的開機鏈和所需映像。

¹ 感謝 ArchWiki 社區對他們所做的所有工作。用於創建 PK，創建 KEK，創建數據庫和簽名映像的命令來自 [創建密鑰](#)，由 ArchWiki 維護團隊和/或貢獻者編寫。ArchWiki

選項 A：從執行個體中將金鑰新增至變數存放區

建立三個金鑰對後，您就可以連接到執行個體，並透過完成以下步驟從執行個體中將金鑰新增至變數存放區。

選項 A 步驟：

- [步驟 1：啟動將支援 UEFI 安全開機的執行個體](#)
- [步驟 2：設定執行個體以支援 UEFI 安全開機](#)
- [步驟 3：從執行個體建立 AMI](#)

步驟 1：啟動將支援 UEFI 安全開機的執行個體

當您[啟動執行個體](#) (具有下列先決條件) 時，執行個體將準備好設定為支援 UEFI 安全開機。您只能在啟動時在執行個體上啟用對 UEFI 安全開機的支援；以後無法啟用它。

必要條件

- AMI - Linux AMI 必須支援 UEFI 開機模式。若要驗證 AMI 是否支援 UEFI 開機模式，AMI 開機模式參數必須為 uefi。如需詳細資訊，請參閱 [確定 AMI 的開機模式參數](#)。

請注意，AWS 僅針對以重力為基礎的執行個體類型提供設定為支援 UEFI 的 Linux AMI。AWS 目前不提供支援 UEFI 開機模式的 x86_64 您可以設定自有的 AMI，以支援所有架構的 UEFI 開機模式。若要設定支援 UEFI 開機模式的 AMI，必須在自有的 AMI 上執行多個設定步驟。如需詳細資訊，請參閱 [設定 AMI 的開機模式](#)。

- 執行個體類型 - 所有支援 UEFI 的虛擬化執行個體類型也支援 UEFI 安全開機。裸機執行個體類型不支援 UEFI 安全開機。如需有關支援 UEFI 安全開機的執行個體類型詳細資訊，請參閱[考量事項](#)。
- 在發佈 UEFI 安全開機後啟動您的執行個體。只有在 2022 年 5 月 10 日 (發佈 UEFI 安全開機時) 後啟動的執行個體才能支援 UEFI 安全開機。

啟動執行個體後，您可以透過檢查 UEFI 資料是否存在，驗證執行個體是否已準備好設定為支援 UEFI 安全開機 (換句話說，您可以繼續執行[步驟 2](#))。UEFI 資料的存在表示非揮發性資料是持續的。

驗證執行個體是否準備好執行步驟 2

使用 [get-instance-uefi-data](#) 命令並指定執行個體 ID。

```
aws ec2 get-instance-uefi-data --instance-id i-0123456789example
```

如果輸出中存在 UEFI 資料，則執行個體已準備好執行步驟 2。如果輸出為空白，則無法將執行個體設定為支援 UEFI 安全開機。如果您的執行個體在 UEFI 安全開機支援可用之前啟動，則會發生這種情況。啟動新的執行個體，然後再試一次。

步驟 2：設定執行個體以支援 UEFI 安全開機

在執行個體上的 UEFI 變數存放區中註冊金鑰對

Warning

您必須在註冊金鑰之後簽署開機映像，否則您將無法啟動執行個體。

建立已簽署的 UEFI 簽章清單 (PK、KEK 以及 db) 後，必須將他們註冊至 UEFI 韌體。

只有在以下情況下才能寫入 PK 變數：

- 如果 SetupMode 變數為 1，表示尚未註冊 PK。透過以下命令來檢查這一點：輸出為 1 或 0。

```
efivar -d -n 8be4df61-93ca-11d2-aa0d-00e098032b8c-SetupMode
```

- 新 PK 由現有 PK 的私有金鑰簽署。

在 UEFI 變數存放區中註冊金鑰

必須在執行個體上執行以下命令。

如果 SetupMode 啟用 (值為1)，則可以在執行個體上執行下列命令來註冊金鑰：

```
[ec2-user ~]$ efi-updatevar -f db.auth db
```

```
[ec2-user ~]$ efi-updatevar -f KEK.auth KEK
```

```
[ec2-user ~]$ efi-updatevar -f PK.auth PK
```

驗證 UEFI 安全開機是否已啟用

若要驗證 UEFI 安全開機是否已啟用，請依照 [驗證執行個體是否已啟用 UEFI 安全開機](#) 中的步驟執行動作。

現在，您可以使用 [get-instance-uefi-data](#) CLI 命令匯出 UEFI 變數存放區，或者繼續執行下一個步驟。簽署開機映像以重新開機到已啟用 UEFI 安全開機的執行個體。

步驟 3：從執行個體建立 AMI

若要從執行個體建立 AMI，您可以使用主控台或 CreateImage API、CLI 或 SDK。如需主控台的說明，請參閱 [創建一個 Amazon EBS-backed AMI](#)。如需 API 指示，請參閱 [CreateImage](#)。

Note

CreateImage API 會自動將執行個體的 UEFI 變數存放區複製到 AMI。控制台使用 CreateImage API。使用此 AMI 啟動執行個體後，執行個體將具有相同的 UEFI 變數存放區。

選項 B：建立一個包含預先填寫的變數存放區的二進位 blob

建立 [三個金鑰對](#) 之後，您可以建立一個包含預先填寫的變數存放區 (包含 UEFI 安全開機金鑰) 的二進位 blob。

Warning

您必須在註冊金鑰之前簽署開機映像，否則您將無法啟動執行個體。

選項 B 步驟：

- [步驟 1：建立新的變數存放區或更新現有變數存放區](#)
- [步驟 2：在建立 AMI 時上傳二進位 blob](#)

步驟 1：建立新的變數存放區或更新現有變數存放區

您可以使用 python-uefivars 工具，在沒有執行個體執行的情況下離線建立變數存放區。該工具可以從您的金鑰建立一個新的變數存放區。指令碼目前支援 EDK2 格式、格式和 JSON 表示法，使用更高層級的工具更容易編輯。AWS

在沒有執行個體執行的情況下離線建立變數存放區

1. 在下列連結下載該工具。

```
https://github.com/aws-labs/python-uefivars
```

2. 通過執行下列命令從您的金鑰建立新的變數存放區。這將在 `your_binary_blob.bin` 中建立 base64 編碼二進位 blob。該工具還支援透過 `-I` 參數更新二進位 blob。

```
./uefivars.py -i none -o aws -O your_binary_blob.bin -P PK.esl -K KEK.esl --db db.esl --dbx dbx.esl
```

步驟 2：在建立 AMI 時上傳二進位 blob

使用 [register-image](#) 傳遞您的 UEFI 變數存放區資料。對於 `--uefi-data` 參數，請指定您的二進位 blob，而對於 `--boot-mode` 參數，則指定 `uefi`。

```
aws ec2 register-image \  
  --name uefi_sb_tpm_register_image_test \  
  --uefi-data $(cat your_binary_blob.bin) \  
  --block-device-mappings "DeviceName=/dev/sda1,Ebs={SnapshotId=snaph-0123456789example,DeleteOnTermination=true}" \  
  --architecture x86_64 \  
  --root-device-name /dev/sda1 \  
  --virtualization-type hvm \  
  --ena-support \  
  --boot-mode uefi
```

如何創建 AWS 二進制 blob

您可以使用以下步驟在 AMI 建立過程中自定義 UEFI 安全開機變數。在這些步驟中使用的 KEK 截至 2021 年 9 月都是最新的。如果 Microsoft 更新 KEK，則您必須使用最新的 KEK。

若要建立二 AWS 進位 blob

1. 建立一個空白的 PK 簽章清單。

```
touch empty_key.crt  
cert-to-efi-sig-list empty_key.crt PK.esl
```

2. 下載 KEK 憑證。

```
https://go.microsoft.com/fwlink/?LinkId=321185
```

3. 將 KEK 憑證包裝在 UEFI 簽章清單中 (siglist)。

```
sbsiglist --owner 77fa9abd-0359-4d32-bd60-28f4e78f784b --type x509 --output  
MS_Win_KEK.esl MicCorKEKCA2011_2011-06-24.crt
```

4. 下載 Microsoft 的 db 憑證。

```
https://www.microsoft.com/pkiops/certs/MicWinProPCA2011_2011-10-19.crt  
https://www.microsoft.com/pkiops/certs/MicCorUEFCA2011_2011-06-27.crt
```

5. 產生 db 簽章清單。

```
sbsiglist --owner 77fa9abd-0359-4d32-bd60-28f4e78f784b --type x509 --output  
MS_Win_db.esl MicWinProPCA2011_2011-10-19.crt  
sbsiglist --owner 77fa9abd-0359-4d32-bd60-28f4e78f784b --type x509 --output  
MS_UEFI_db.esl MicCorUEFCA2011_2011-06-27.crt  
cat MS_Win_db.esl MS_UEFI_db.esl > MS_db.esl
```

6. 從下列連結下載更新後的 dbx 變更要求。

```
https://uefi.org/revocationlistfile
```

7. 在上一步下載的 dbx 變更要求已與 Microsoft KEK 一起簽署，因此您需要將其剔除或解壓縮。您可以使用下列連結。

```
https://gist.github.com/out0xb2/f8e0bae94214889a89ac67fceb37f8c0
```

```
https://support.microsoft.com/en-us/topic/microsoft-guidance-for-applying-secure-  
boot-dbx-update-e3b9e4cb-a330-b3ba-a602-15083965d9ca
```

8. 使用 uefivars.py 指令碼建置 UEFI 變數存放區。

```
./uefivars.py -i none -o aws -0 uefiblob-microsoft-keys-empty-pk.bin -P ~/PK.esl -K  
~/MS_Win_KEK.esl --db ~/MS_db.esl --dbx ~/dbx-2021-April.bin
```

9. 檢查二進位 blob 和 UEFI 變數存放區。

```
./uefivars.py -i aws -I uefiblob-microsoft-keys-empty-pk.bin -o json | less
```

10. 您可以透過再次將 blob 傳遞給同一工具來更新 Blob。

```
./uefivars.py -i aws -I uefiblob-microsoft-keys-empty-pk.bin -o aws -O uefiblob-  
microsoft-keys-empty-pk.bin -P ~/PK.esl -K ~/MS_Win_KEK.esl --db ~/MS_db.esl --dbx  
~/dbx-2021-April.bin
```

預期的輸出結果

```
Replacing PK  
Replacing KEK  
Replacing db  
Replacing dbx
```

尋找 AMI

AMI 包含啟動執行個體所需的元件和應用程式，例如作業系統和根磁碟區類型。若要啟動符合您需求的執行個體，您必須找到符合您需求的 AMI。

選取 AMI 時，請考慮下列要啟動的執行個體需求：

- 地區 — AMI ID 對於每個 AWS 區域都是唯一的。
- 作業系統
- 架構：32 位元 (i386)、64 位元 (x86_64) 或 64 位元 ARM (arm64)
- 根裝置類型：Amazon EBS 或執行個體存放區
- 提供者 (例如，Amazon Web Services)
- 其他軟體 (例如，SQL server)

有多種方法可以找到滿足您需求的 AMI。本主題說明如何使用 Amazon EC2 主控台、AWS CLI AWS Tools for Windows PowerShell、和來尋找 AMI AWS Systems Manager。

主題

- [使用 Amazon EC2 主控台尋找 AMI](#)
- [使用「」尋找 AMI AWS CLI](#)
- [使用「」尋找 AMI AWS Tools for Windows PowerShell](#)
- [使用 Systems Manager 參數尋找 AMI](#)
- [使用 Systems Manager 尋找最新的 AMI](#)

- [尋找 AMI 的更多資訊](#)

使用 Amazon EC2 主控台尋找 AMI

您可以使用 Amazon EC2 主控台找到 AMI。當您使用啟動執行個體精靈來啟動執行個體時，可以從 AMI 清單中選取，也可以使用 Images (映像) 頁面搜尋所有可用的 AMI。

若要使用啟動執行個體精靈尋找 AMI

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 從導覽列選取要在其中啟動執行個體的區域。無論您的位置為何，皆可選取任何可用的區域。AMI ID 對於每個 AWS 地區都是唯一的。
3. 從主控台儀表板選擇 Launch Instance (啟動執行個體)。
4. (新主控台) 在 Application and OS Images (Amazon Machine Image) (應用程式和作業系統映像 (Amazon Machine Image)) 下，選取 Quick Start (快速入門)，為您的執行個體選取作業系統 (OS)，然後在 Amazon Machine Image (AMI) 中，從清單中選取其中一個常用的 AMI。若未看到您需要的 AMI，您可選擇 Browser more AMIs (瀏覽更多 AMI) 以瀏覽完整的 AMI 目錄。如需詳細資訊，請參閱 [應用程式和作業系統映像 \(Amazon Machine Image\)](#)。

(舊主控台) 在 Quick Start (快速入門) 索引標籤中，從清單中選取其中一個常用的 AMI。若未看到您想要使用的 AMI，請選擇 My AMIs (我的 AMI)、AWS Marketplace 或 Community AMIs (社群 AMI) 索引標籤，來尋找其他的 AMI。如需詳細資訊，請參閱 [步驟 1：選擇 Amazon Machine Image \(AMI\)](#)。

若要使用 AMI 頁面尋找 AMI

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 從導覽列選取要在其中啟動執行個體的區域。無論您的位置為何，皆可選取任何可用的區域。AMI ID 對於每個 AWS 地區都是唯一的。
3. 在導覽窗格中，選擇 AMIs (AMI)。
4. (選用) 利用 Filter (篩選) 和 Search (搜尋) 選項來限定所顯示 AMI 的清單範圍，只列出符合標準的 AMI。

例如，若要列出由提供的所有 AMI AWS，請選擇「公開映像檔」。然後使用 Search (搜尋) 選項進一步確定顯示 AMI 清單的範圍。選擇 Search (搜尋) 列，然後從選單中依序選擇 Owner alias(擁有者別名)、= 運算子和值 amazon。若要尋找符合特定平台 (例如 Linux 或 Windows) 的

AMI，請再次選擇 [搜尋] 列以選擇 [平台]，然後選擇 = 運算子，然後從提供的清單中選擇作業系統。

- (選用) 選擇偏好設定圖示，以選取要顯示的映像屬性，例如根裝置類型。或者，您可以從清單中選取 AMI，然後在 Details (詳細資訊) 索引標籤上檢視其屬性。
- 在選取 AMI 之前，重要的是檢查 AMI 是否為執行個體存放區後端執行個體或 Amazon EBS 後端執行個體，而且您知悉這種差異造成的影響。如需詳細資訊，請參閱 [根設備儲存](#)。
- 若要從這個 AMI 啟動執行個體，請先選取該 AMI，然後再選擇從映像啟動執行個體。如需有關使用主控台啟動執行個體的詳細資訊，請參閱[使用新的啟動執行個體精靈啟動執行個體](#)。如果您目前尚未準備好要啟動執行個體，請記下該 AMI ID，以待日後使用。

使用「」尋找 AMI AWS CLI

您可以使用[描述影像 AWS CLI 指令](#)，僅列出符合您需求的 AMI。找到符合需求的 AMI 之後，請記下其 ID，然後就可以使用此 AMI 來啟動執行個體。如需詳細資訊，請參閱《AWS Command Line Interface 使用者指南》中的[啟動您的執行個體](#)

[describe-images](#) 指令支援篩選參數。例如，使用 `--owners` 參數來顯示 Amazon 所擁有的公用 AMI。

```
aws ec2 describe-images --owners amazon
```

您可以在先前的指令中加入下列篩選條件，來只顯示 Windows AMI：

```
--filters "Name=platform,Values=windows"
```

您可以在先前的指令中加入下列篩選條件，來只顯示以 Amazon EBS 為後端的 AMI：

```
--filters "Name=root-device-type,Values=efs"
```

Important

如果您省略 `describe-images` 指令中的 `--owners` 參數，則會傳回您擁有啟動權限的所有影像，而不論擁有權為何。

使用「」尋找 AMI AWS Tools for Windows PowerShell

您可以使用 PowerShell 指令程式，僅列出符合您需求的 Windows AMI。如需相關資訊和範例，請參閱 [使用 AWS Tools for Windows PowerShell 者指南 PowerShell 中的使用 Windows 尋找 Amazon 電腦映像](#)。

找到符合需求的 AMI 之後，請記下其 ID，然後就可以使用此 AMI 來啟動執行個體。如需詳細資訊，請參閱 [使用 AWS Tools for Windows PowerShell 者指南 PowerShell 中的使用 Windows 啟動 Amazon EC2 執行個體](#)。

使用 Systems Manager 參數尋找 AMI

使用 Amazon EC2 主控台中的 EC2 啟動執行個體精靈啟動執行個體時，可以從清單中選取 AMI (如中所述 [使用 Amazon EC2 主控台尋找 AMI](#))，也可以選取指向 AMI ID 的 AWS Systems Manager 參數 (如本節所述)。如果您使用自動化程式碼來啟動執行個體，則可以指定 Systems Manager 參數而非 AMI ID。

Systems Manager 參數是您可以在 Systems Manager 參數存放區中建立的客戶定義索引鍵/值組。參數存放區提供一個中央存放區，可將您的應用程式組態值外部化。如需詳細資訊，請參閱 AWS Systems Manager 使用者指南中的 [AWS Systems Manager 參數存放區](#)。

當您建立指向 AMI ID 的參數時，請務必將資料類型指定為 `aws:ec2:image`。指定此資料類型可確保在建立或修改參數時，參數值會驗證為 AMI ID。如需詳細資訊，請參閱 AWS Systems Manager 使用者指南中的 [Amazon Machine Image ID 的原生參數支援](#)。

主題

- [使用案例](#)
- [許可](#)
- [限制](#)
- [使用 Systems Manager 參數啟動執行個體](#)

使用案例

使用 Systems Manager 參數指向 AMI ID 時，能讓使用者在啟動執行個體時更輕鬆選取正確的 AMI。Systems Manager 參數還可以簡化自動化程式碼的維護。

Easier for users (對使用者更簡單)

如果您要求使用特定 AMI 啟動執行個體，且該 AMI 定期更新，我們建議您要求使用者選取 Systems Manager 參數來尋找 AMI。透過要求使用者選取 Systems Manager 參數，您可以確保使用最新的 AMI 來啟動執行個體。

例如，您可以在組織中每月建立新版本 AMI，其中包含最新的作業系統和應用程式修補程式。您也要求使用者使用最新版本的 AMI 來啟動執行個體。為了確保您的使用者使用最新版本，您可以建立指向正確 AMI ID 的 Systems Manager 參數 (例如 `golden-ami`)。每次建立新版 AMI 時，您都會更新參數中的 AMI ID 值，使其永遠指向最新的 AMI。您的使用者不需要知道 AMI 的定期更新，因為他們每次都會繼續選取相同的 Systems Manager 參數。使用 AMI 的 Systems Manager 參數可在啟動執行個體時更輕鬆地選取正確的 AMI。

Simplify automation code maintenance (簡化自動化程式碼維護)

如果您使用自動化程式碼來啟動執行個體，則可以指定 Systems Manager 參數而非 AMI ID。如果新版本的 AMI 已建立，您可以變更參數中的 AMI ID 值，使其指向最新的 AMI。每次建立新版本的 AMI 時，都不需要修改參考參數的自動化程式碼。如此可簡化自動化維護作業，並有助於降低部署成本。

Note

當您變更 Systems Manager 參數指向的 AMI ID 時，執行中的執行個體不會受到影響。

許可

如果您在啟動執行個體精靈中使用指向 AMI ID 的 Systems Manager 參數，則必須將下列許可新增至 IAM 政策：

- `ssm:DescribeParameters`— 授予檢視和選取 Systems Manager 參數的權限。
- `ssm:GetParameters`— 授與擷取 Systems Manager 參數值的權限。

您也可以限制對特定 Systems Manager 參數的存取。如需詳細資訊和 IAM 政策範例，請參閱[範例：使用 EC2 啟動執行個體精靈](#)。

限制

AMI 和 Systems Manager 參數是區域特定的。若要跨區域使用相同的 Systems Manager 參數名稱，請在每個區域中建立具有相同名稱的 Systems Manager 參數 (例如 `golden-ami`)。在每個區域中，將 Systems Manager 參數指向該區域中的 AMI。

使用 Systems Manager 參數啟動執行個體

您可以使用主控台或 AWS CLI 來啟動執行個體。您可以指定指向 AMI 識別碼的 AWS Systems Manager 參數，而不是指定 AMI 識別碼。

New console

使用 Systems Manager 參數 (控制台) 查找 AMI

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 從導覽列選取要在其中啟動執行個體的區域。無論您的位置為何，皆可選取任何可用的區域。
3. 從主控台儀表板選擇 Launch Instance (啟動執行個體)。
4. 在 Application and OS Images (Amazon Machine Image) (應用程式和作業系統映像 (Amazon Machine Image)) 下，選擇 Browse more AMIs (瀏覽更多 AMI)。
5. 選擇搜尋列右側的箭頭按鈕，然後選擇 Search by Systems Manager parameter (依 Systems Manager 參數搜尋)。
6. 對於 Systems Manager parameter (Systems Manager 參數)，請選取一個參數。對應的 AMI ID 會出現在 Currently resolves to (目前解析為) 下方。
7. 選擇 Search (搜尋)。與 AMI ID 相符的 AMI 會顯示在清單中。
8. 從清單中選取 AMI，然後選取 Select (選取)。

如需有關使用啟動執行個體精靈啟動執行個體的詳細資訊，請參閱 [使用新的啟動執行個體精靈啟動執行個體](#)。

Old console

使用 Systems Manager 參數 (控制台) 查找 AMI

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 從導覽列選取要在其中啟動執行個體的區域。無論您的位置為何，皆可選取任何可用的區域。
3. 從主控台儀表板選擇 Launch Instance (啟動執行個體)。
4. 選擇 Search by Systems Manager parameter (依 Systems Manager 參數搜尋) (位於右上角)。
5. 對於 Systems Manager parameter (Systems Manager 參數)，請選取一個參數。對應的 AMI ID 會出現在 Currently resolves to (目前解析為) 旁邊。
6. 選擇 Search (搜尋)。與 AMI ID 相符的 AMI 會顯示在清單中。

7. 從清單中選取 AMI，然後選取 Select (選取)。

如需有關使用啟動執行個體精靈從 AMI 啟動執行個體的詳細資訊，請參閱 [步驟 1：選擇 Amazon Machine Image \(AMI\)](#)。

若要使用 AWS Systems Manager 參數而非 AMI ID (AWS CLI) 啟動執行個體

下列範例會使用 Systems Manager 參數 `golden-ami` 來啟動 `m5.xlarge` 執行個體。該參數指向 AMI ID。

若要在命令中指定參數，請使用下列語法：`resolve:ssm:/parameter-name`，其中 `resolve:ssm` 是標準字首，`parameter-name` 是唯一的參數名稱。請注意，參數名稱區分大小寫。參數名稱的反斜線只有在參數是階層的一部分時才需要，例如 `/amis/production/golden-ami`。如果參數不是階層架構的一部分，則可以省略反斜線。

在此範例中，`--count` 和 `--security-group` 參數不包含在內。對於 `--count`，預設值為 1。如果您有預設 VPC 和預設安全群組，則會使用這些預設項目。

```
aws ec2 run-instances
  --image-id resolve:ssm:/golden-ami
  --instance-type m5.xlarge
  ...
```

若要使用特定版本的 AWS Systems Manager 參數啟動執行個體 (AWS CLI)

Systems Manager 參數擁有版本支援。參數的每個反覆項目都會指定一個唯一的版本號碼。您可以參考參數的版本，如下所示 `resolve:ssm:parameter-name:version`，其中 `version` 是唯一的版本號碼。根據預設，在未指定任何版本時，會使用最新版本的參數。

下列範例會使用參數的版本 2。

在此範例中，`--count` 和 `--security-group` 參數不包含在內。`--count` 的預設值為 1。如果您有預設 VPC 和預設安全群組，則會使用這些預設值。

```
aws ec2 run-instances
  --image-id resolve:ssm:/golden-ami:2
  --instance-type m5.xlarge
  ...
```

若要使用提供的 `public` 參數啟動執行個體 AWS

Systems Manager 為公共 AMI 提供了公共參數。AWS 您可以在啟動執行個體時使用公用參數，以確保您使用的是最新的 AMI。

如需詳細資訊，請參閱 [使用 Systems Manager 尋找最新的 AMI](#)。

使用 Systems Manager 尋找最新的 AMI

AWS Systems Manager 為由 AWS 維護的公用 AMI 提供公用參數。您可以在啟動執行個體時使用公用參數，以確保您使用的是最新的 AMI。例如，公共參數 `/aws/service/ami-amazon-linux-latest/al2023-ami-kernel-default-arm64` 在所有區域都可以使用，而且一律會指向指定區域中適用於 arm64 架構的最新版本的 Amazon Linux 2023 AMI。

公用參數可從下列路徑取得：

- Linux – `/aws/service/ami-amazon-linux-latest`
- Windows – `/aws/service/ami-windows-latest`

若要檢視目前 AWS 區域中所有 Linux 或視窗 AMI 的清單

使用下列 [逐一路徑取得參數](#) AWS CLI 指令，檢視目前「區域」中所有 Linux 或 Windows AMI 的清單。AWS 對於 Linux 和視窗而言，`--path` 參數的值不同。

對於 Linux：

```
aws ssm get-parameters-by-path \  
  --path /aws/service/ami-amazon-linux-latest \  
  --query "Parameters[].Name"
```

針對 Windows：

```
aws ssm get-parameters-by-path \  
  --path /aws/service/ami-windows-latest \  
  --query "Parameters[].Name"
```

使用公有參數啟動執行個體

下列範例會為映像 ID 指定一個 Systems Manager 公用參數，以便使用最新的 Amazon Linux 2023 AMI 啟動執行個體。

若要在命令中指定參數，請使用下列語法：`resolve:ssm:public-parameter`，其中 `resolve:ssm` 是標準前綴，而 `public-parameter` 是公有參數的路徑和名稱。

在此範例中，`--count` 和 `--security-group` 參數不包含在內。對於 `--count`，預設值為 1。如果您有預設 VPC 和預設安全群組，則會使用這些預設項目。

```
aws ec2 run-instances \  
  --image-id resolve:ssm:/aws/service/ami-amazon-linux-latest/al2023-ami-kernel-  
  default-x86_64 \  
  --instance-type m5.xlarge \  
  --key-name MyKeyPair
```

如需詳細資訊，請參閱 AWS Systems Manager 使用者指南中的 [使用公有參數](#)。

如需使用 Systems Manager 參數的範例，請參閱 [使用參數存放區查詢最新的 Amazon Linux AMI ID](#) 和 [使用 AWS Systems Manager 參數存放區查詢最新的 Windows AMI](#)。AWS Systems Manager

尋找 AMI 的更多資訊

要查找 Amazon 2023 AMI，請參閱 [Amazon 2023 用戶指南中的亞馬遜 EC2 上的 AL2023](#)。

要查找 Ubuntu AMI，請參閱規範 Ubuntu 網站上的 [亞 Amazon EC2 AMI 定位器](#)。

若要尋找 RHEL AMI，請參閱 [紅帽網站上的 Amazon Web Services \(\) 上的紅帽企業 Linux 映像檔 \(AMI AWS\)](#)。

共享 AMI

共享 AMI 是由開發人員建置並可讓其他人員使用的 AMI。開始使用 Amazon EC2 最簡單的方式之一，就是使用具備您所需元件的共享 AMI，然後再新增自訂的內容。您也可以建立自己的 AMI 並與其他人共享。

但使用共享 AMI 所面臨的風險由您自行承擔。對於其他 Amazon EC2 使用者共享的 AMI，Amazon 不保證其完整性或安全性。因此，您對待共享 AMI 的方式應與您在考慮將任何外部程式碼部署到自己的資料中心時相同，且應善盡相關責任。建議您從信任的來源取得 AMI，例如經過驗證的供應商。

已驗證的供應商

在 Amazon EC2 主控台中，Amazon 或已驗證的 Amazon 合作夥伴擁有的公用 AMI 標記為 Verified provider (已驗證的供應商)。

您也可以使用[描述影像](#) AWS CLI 指令來識別來自已驗證提供者的公用 AMI。Amazon 或已驗證的合作夥伴擁有的公有映像有一個採用別名的擁有者，即 `amazon` 或 `aws-marketplace`。在 CLI 輸出中，這些值顯示為 `ImageOwnerAlias`。其他使用者無法為其 AMI 設定別名。這可讓您輕鬆地尋找來自 Amazon 或已驗證的合作夥伴的 AMI。

若要成為已驗證的供應商，您必須在 AWS Marketplace 上註冊為賣方。註冊後，您可以將您的 AMI 在 AWS Marketplace 上列出。如需詳細資訊，請參閱《AWS Marketplace 賣方指南》中的[賣方入門](#)和[以 AMI 為基礎的產品](#)。

共享 AMI 主題

- [尋找共享 AMI](#)
- [使 AMI 公有化](#)
- [與特定組織或組織單位共用 AMI](#)
- [與指定的 AWS 帳戶共享 AMI](#)
- [取消將 AMI 與您的共享 AWS 帳戶](#)
- [使用書籤](#)
- [共享 Linux AMI 的準則](#)

如果您正在尋找有關其他主題的資訊

- 如需有關建立 AMI 的資訊，請參閱[the section called “建立執行個體存放區後端 Linux AMI”](#)或[the section called “創建一個 Amazon EBS-backed AMI”](#)。
- 如需有關在 AWS Marketplace 上建置、交付和維護應用程式的資訊，請參閱 [AWS Marketplace 文件](#)。

尋找共享 AMI

您可用 Amazon EC2 主控台或命令列來尋找共享 AMI。

AMI 是一種區域性的資源。因此，您必須在 AMI 進行共享的區域中搜尋，才能搜尋到共享 AMI (公有或私有)。若要讓 AMI 在其他區域可用，請將 AMI 複製到該區域，然後共享。如需詳細資訊，請參閱[複製 AMI](#)。

任務

- [尋找共享 AMI \(主控台\)](#)

- [尋找共享 AMI \(AWS CLI\)](#)
- [尋找共用 AMI \(適用於視窗的工具 PowerShell\)](#)
- [使用共享 AMI](#)

尋找共享 AMI (主控台)

使用主控台尋找共享私有 AMI

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 AMIs (AMI)。
3. 在第一個篩選條件中，選擇 Private images (私有映像)。將列出與您共享的所有 AMI。若要縮小搜尋範圍，請選擇 Search (搜尋) 列並使用選單中提供的篩選條件選項。

使用主控台尋找共享公用 AMI

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 AMIs (AMI)。
3. 在第一個篩選條件中，選擇 Public images (公有映像)。若要縮小搜尋範圍，請選擇 Search (搜尋) 欄位並使用功能表中提供的篩選條件選項。

使用主控台尋找 Amazon 的共享公用 AMI

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 AMIs (AMI)。
3. 在第一個篩選條件中，選擇 Public images (公有映像)。
4. 選擇 Search (搜尋) 欄位，然後從顯示的功能表選項中依序選擇 Owner alias (擁有者別名)、= 和 amazon，以僅顯示 Amazon 的公有映像。

使用主控台尋找來自己驗證的供應商的共享公用 AMI

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 AMI Catalog (AMI 型錄)。
3. 選擇 Community AMI (社群 AMI)。
4. Verified provider (已驗證的供應商) 標籤指示來自 Amazon 或已驗證的合作夥伴的 AMI。

尋找共享 AMI (AWS CLI)

使用 [describe-images](#) 命令 (AWS CLI) 來列出 AMI。您可縮小範圍，僅列出感興趣的 AMI 類型，如下列範例所示。

範例：列出所有公用 AMI

下列命令列出所有公用 AMI，包括您所擁有的任何公用 AMI。

```
aws ec2 describe-images --executable-users all
```

範例：列出具有明確啟動許可的 AMI

下列命令列出您擁有明確啟動許可的 AMI。此清單不含您所擁有的任何 AMI。

```
aws ec2 describe-images --executable-users self
```

範例：列出已驗證的供應商擁有的 AMI

下列命令會列出已驗證的供應商擁有的 AMI。已驗證的供應商 (Amazon 或已驗證的合作夥伴) 擁有的公用 AMI 有一個採用別名的擁有者，其在帳戶欄位顯示為 `amazon` 或 `aws-marketplace`。這可協助您輕鬆尋找來自己驗證的供應商的 AMI。其他使用者無法為其 AMI 設定別名。

```
aws ec2 describe-images \  
  --owners amazon aws-marketplace \  
  --query 'Images[*].[ImageId]' \  
  --output text
```

範例：列出某一帳戶擁有的 AMI

下列命令會列出特定 AWS 帳戶擁有的 AMI。

```
aws ec2 describe-images --owners 123456789012
```

範例：使用篩選條件縮小 AMI 範圍

若要減少顯示的 AMI 數量，請使用篩選條件，只列出感興趣的 AMI 類型。例如，下列篩選條件只會顯示 EBS 後端 AMI。

```
--filters "Name=root-device-type,Values=ebs"
```

尋找共用 AMI (適用於視窗的工具 PowerShell)

使用指 [Get-EC2Image](#) 令 (視窗工具 PowerShell) 列示 AMI。您可縮小範圍，僅列出感興趣的 AMI 類型，如下列範例所示。

範例：列出所有公用 AMI

下列命令列出所有公用 AMI，包括您所擁有的任何公用 AMI。

```
PS C:\> Get-EC2Image -ExecutableUser all
```

範例：列出具有明確啟動許可的 AMI

下列命令列出您擁有明確啟動許可的 AMI。此清單不含您所擁有的任何 AMI。

```
PS C:\> Get-EC2Image -ExecutableUser self
```

範例：列出已驗證的供應商擁有的 AMI

下列命令會列出已驗證的供應商擁有的 AMI。已驗證的供應商 (Amazon 或已驗證的合作夥伴) 擁有的公用 AMI 有一個採用別名的擁有者，其在帳戶欄位顯示為 `amazon` 或 `aws-marketplace`。這可協助您輕鬆尋找來自己驗證的供應商的 AMI。其他使用者無法為其 AMI 設定別名。

```
PS C:\> Get-EC2Image -Owner amazon aws-marketplace
```

範例：列出某一帳戶擁有的 AMI

下列命令會列出特定 AWS 帳戶擁有的 AMI。

```
PS C:\> Get-EC2Image -Owner 123456789012
```

範例：使用篩選條件縮小 AMI 範圍

若要減少顯示的 AMI 數量，請使用篩選條件，只列出感興趣的 AMI 類型。例如，下列篩選條件只會顯示 EBS 後端 AMI。

```
-Filter @{ Name="root-device-type"; Values="ebs" }
```

使用共享 AMI

開始使用共享 AMI 之前，請先採取下列步驟，確定沒有預先安裝的登入資料可能允許第三方對您的執行個體執行不必要的存取，並確定沒有預先設定的遠端登入可能將機密資料傳送至第三方。請參閱 AMI 所使用 Linux 發行版本的說明文件，以取得改善系統安全性的詳細資訊。

為確保您不會意外失去執行個體的存取權，建立您初始化兩個 SSH 工作階段，並保持第二個工作階段為開啟狀態，直到您移除無法識別的登入資料並確認您仍可用 SSH 登入您的執行個體為止。

1. 識別並停用任何未經授權的公有 SSH 金鑰。檔案中唯一能存在的金鑰應該是您用來啟動 AMI 的金鑰。以下命令可找出 `authorized_keys` 檔案：

```
[ec2-user ~]$ sudo find / -name "authorized_keys" -print -exec cat {} \;
```

2. 停用根使用者的密碼型態身分驗證。開啟 `sshd_config` 檔案並編輯 `PermitRootLogin` 行，如下所示：

```
PermitRootLogin without-password
```

或者，您可停用公用程式，以根使用者身分登入執行個體：

```
PermitRootLogin No
```

重新啟動 `sshd` 服務。

3. 檢查是否有任何其他使用者能夠登入您的執行個體。具有超級使用者權限的使用者特別危險。移除或鎖定任何未知帳戶的密碼。
4. 檢查未使用且正在執行網路服務偵聽是否有傳入連線的開放中連接埠。
5. 若要避免預先設定的遠端登入，您應刪除現有的組態檔案，然後重新啟動 `rsyslog` 服務。例如：

```
[ec2-user ~]$ sudo rm /etc/rsyslog.conf  
[ec2-user ~]$ sudo service rsyslog restart
```

6. 確認所有的 `cron` 任務均為合法。

如果發現可能存有安全風險的公用 AMI，請聯絡 AWS 安全團隊。如需詳細資訊，請參閱 [AWS 安全中心](#)。

使 AMI 公有化

您可以通過與所有人共享來使您的 AMI 公開可用 AWS 帳戶。

如果您希望阻止公開共用您的 AMI，則可以啟用封鎖 AMI 的公有存取權。這樣可以阻止任何公開 AMI 的嘗試，有助於防止未經授權的存取和 AMI 資料的潛在濫用。請注意，啟用封鎖公開存取並不會影響您已公開提供的 AMI；它們仍然可以公開使用。

若要只允許特定帳戶使用 AMI 來啟動執行個體，請參閱 [與指定的 AWS 帳戶共享 AMI](#)。

目錄

- [考量事項](#)
- [與所有 AWS 帳戶共享 AMI \(公開共享 \)](#)
- [封鎖 AMI 的公有存取權](#)

考量事項

在將 AMI 設定為公有之前，請考慮以下項目。

- 所有權 — 要公開 AMI，您 AWS 帳戶 必須擁有 AMI。
- 區域 - AMI 是一種區域性的資源。共享 AMI 後，只能從共享 AMI 的區域中使用 AMI。若要讓 AMI 在其他區域可用，請將 AMI 複製到該區域，然後共用。如需詳細資訊，請參閱 [複製 AMI](#)。
- 封鎖公開存取 – 若要公開共用 AMI，必須在將公開共用 AMI 的每個區域中停用 [封鎖 AMI 的公有存取權](#)。公開共用 AMI 後，您可以重新啟用「封鎖 AMI 的公有存取權」，以防止進一步公開共用您的 AMI。
- 部分 AMI 無法設定為公有 - 如果您的 AMI 有下列任一情況，則無法將其設定為公有 (但您可以 [與指定的 AWS 帳戶共用 AMI](#))：
 - 加密磁碟區
 - 加密磁碟區的快照
 - 產品代碼
- 避免敏感資料外洩 - 若要避免共享 AMI 時洩漏敏感資料，請詳讀 [共享 Linux AMI 的準則](#) 中的安全性考量，並遵照推薦的動作進行。
- 使用 - 共享 AMI 後，使用者只能從 AMI 啟動執行個體。他們無法刪除、共用或修改它。不過，他們在使用您的 AMI 啟動了執行個體之後，就可以從其啟動的執行個體建立 AMI。

- 自動棄用 - 在預設情況下，所有公用 AMI 的棄用日期都設定為自 AMI 建立日期起兩年。您可將棄用日期設定為早於兩年。要取消棄用日期或將棄用日期移至以後的日期，您必須通過僅與特定的方式[共享 AMI 將其設為私有](#)。AWS 帳戶
- 移除過時的 AMI — 在公用 AMI 達到淘汰日期之後，如果六個月以上沒有從 AMI 啟動新執行個體，AWS 最終會移除公用共用屬性，以便過時的 AMI 不會出現在公用 AMI 清單中。
- 帳單 — 當其 AWS 帳戶 他人使用您的 AMI 啟動執行個體時，不會向您收取費用。使用 AMI 啟動執行個體的帳戶會根據它們啟動的執行個體計費。

與所有 AWS 帳戶共享 AMI (公開共享)

將 AMI 公有化之後，可以在主控台的社群 AMI 中使用，您可以從 EC2 主控台左側導覽器中的 AMI 目錄存取，也可在使用主控台啟動執行個體時進行存取。請注意，AMI 公有化後要經過一段時間才會出現在 Community AMI (社群 AMI) 內。

Console

使 AMI 公有化

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 AMIs (AMI)。
3. 從清單中選取您的 AMI，然後選取 Actions (動作)、Edit AMI permissions (編輯 AMI 許可)。
4. 在 AMI 可用性下，選擇公有。
5. 選擇儲存變更。

AWS CLI

每個 AMI 都有一個 `launchPermission` 屬性 AWS 帳戶，用於控制除了所有者之外，允許哪些屬性使用該 AMI 來啟動實例。藉由修改 AMI 的 `launchPermission` 屬性，您可以將 AMI 設為公開 (這會將啟動權限授予所有人 AWS 帳戶)，或僅與您指定的 AWS 帳戶 屬性共用。

您可在具有 AMI 啟動許可的帳戶清單中新增或移除帳戶 ID。若要將 AMI 設定為公有，請指定 `all` 群組。您可同時指定公有和明確啟動許可。

使 AMI 公有化

1. 如下所示，使用 [modify-image-attribute](#) 命令，新增 `all` 群組至指定 AMI 的 `launchPermission` 清單。

```
aws ec2 modify-image-attribute \  
  --image-id ami-0abcdef1234567890 \  
  --launch-permission "Add=[{Group=all}]"
```

- 若要確認 AMI 的啟動許可，請使用 [describe-image-attribute](#) 命令。

```
aws ec2 describe-image-attribute \  
  --image-id ami-0abcdef1234567890 \  
  --attribute launchPermission
```

- (選用) 若要重新將 AMI 設定為私有，請從啟動許可中移除 all 群組。請注意，AMI 擁有者始終具備啟動許可，因此不受此命令影響。

```
aws ec2 modify-image-attribute \  
  --image-id ami-0abcdef1234567890 \  
  --launch-permission "Remove=[{Group=all}]"
```

PowerShell

每個 AMI 都有一個 launchPermission 屬性 AWS 帳戶，用於控制除了所有者之外，允許哪些屬性使用該 AMI 來啟動實例。藉由修改 AMI 的 launchPermission 屬性，您可以將 AMI 設定為公開 (這會將啟動權限授予所有人 AWS 帳戶)，或僅與您指定的 AWS 帳戶 屬性共用。

您可在具有 AMI 啟動許可的帳戶清單中新增或移除帳戶 ID。若要將 AMI 設定為公有，請指定 all 群組。您可同時指定公有和明確啟動許可。

使 AMI 公有化

- 如下所示，使用 [Edit-EC2ImageAttribute](#) 命令，新增 all 群組至指定 AMI 的 launchPermission 清單。

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute  
  launchPermission -OperationType add -UserGroup all
```

- 若要確認 AMI 的啟動許可，請使用下列 [Get-EC2ImageAttribute](#) 命令。

```
PS C:\> Get-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute  
  launchPermission
```

3. (選用) 若要重新將 AMI 設定為私有，請從啟動許可中移除 all 群組。請注意，AMI 擁有者始終具備啟動許可，因此不受此命令影響。

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute
launchPermission -OperationType remove -UserGroup all
```

封鎖 AMI 的公有存取權

若要阻止公開共用您的 AMI，您可以啟用封鎖 AMI 的公有存取權。此設定在帳戶層級啟用，但您必須在每一個要防止 AMI 公開共用的設定 AWS 區域中啟用此設定。

啟用「封鎖公有存取權」時，將會自動封鎖任何將 AMI 公開的嘗試。但是，如果您已經擁有公開 AMI，它們仍然可以公開使用。

若要公開共用 AMI，您必須停用封鎖公用存取。完成共用後，最佳實務是重新啟用「封鎖公有存取權」，以防止任何非預期性公開共用您的 AMI。

您可以將 IAM 許可限制為管理員使用者，這樣便只有他們才能啟用或停用「封鎖 AMI 的公有存取權」。

目錄

- [預設設定](#)
- [所需的 IAM 許可](#)
- [啟用「封鎖 AMI 的公有存取權」](#)
- [停用「封鎖 AMI 的公有存取權」](#)
- [檢視封鎖 AMI 的公有存取權的狀態](#)

預設設定

根據您的帳戶是新帳戶還是現有帳戶，以及您是否擁有公有 AMI，封鎖 AMI 的公有存取權設定預設為啟用或停用。下表列出了預設的設定：

AWS 帳戶	封鎖 AMI 預設設定的公有存取權
新帳戶	已啟用
沒有公有 AMI 的現有帳戶 ¹	已啟用

AWS 帳戶

封鎖 AMI 預設設定的公有存取權

具有一個或多個公有 AMI 的現有帳戶

已停用

¹ 如果您的帳戶在 2023 年 7 月 15 日或之後擁有一個或多個公有 AMI，即使您隨後將所有 AMI 設為私有，依預設會停用您帳戶的封鎖 AMI 的公有存取權。

所需的 IAM 許可

若要使用「封鎖 AMI 的公有存取權」，您必須具備下列 IAM 許可：

- EnableImageBlockPublicAccess
- DisableImageBlockPublicAccess
- GetImageBlockPublicAccessState

啟用「封鎖 AMI 的公有存取權」

若要阻止公開共用您的 AMI，請在帳戶層級啟用「封鎖 AMI 的公有存取權」。您必須在您希望阻止公開共用您的 AMI 的每個 AWS 區域中啟用「封鎖 AMI 的公有存取權」。若您已有公有 AMI，則它們將保持公開可用。

Console

在指定區域中啟用「封鎖 AMI 的公有存取權」

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽列 (位於畫面頂端) 中，選取要啟用「封鎖 AMI 的公有存取權」的區域。
3. 如果儀表板未顯示，請在導覽窗格中，選擇 EC2 儀表板。
4. 在帳戶屬性下，選擇資料保護和安全性。
5. 在封鎖 AMI 的公有存取權下，選擇管理。
6. 選取封鎖新的公開共用核取方塊，然後選擇更新。

 Note

API 可能最多需要 10 分鐘的時間來設定此設定。在此期間，該值將是已允許新的公開共用。當 API 完成設定時，該值會自動變更為已封鎖新的公開共用。

AWS CLI

在指定區域中啟用「封鎖 AMI 的公有存取權」

使用 [enable-image-block-public-access](#) 命令，並指定要在其中啟用「封鎖 AMI 的公有存取權」的區域。針對 `--image-block-public-access-state` 參數，請指定 `block-new-sharing`。

```
aws ec2 enable-image-block-public-access \  
  --region us-east-1 \  
  --image-block-public-access-state block-new-sharing
```

預期的輸出結果

```
{  
  "ImageBlockPublicAccessState": "block-new-sharing"  
}
```

Note

API 可能最多需要 10 分鐘的時間來設定此設定。在此期間，如果您執行 [get-image-block-public-access-state](#) 命令，則回應將為 `unblocked`。當 API 完成設定時，回應將為 `block-new-sharing`。

停用「封鎖 AMI 的公有存取權」

若要允許您帳戶中的使用者公開共用您的 AMI，請在帳戶層級停用「封鎖 AMI 的公有存取權」。您必須在每個 AWS 區域 要允許 AMI 公開共用的情況下停用 AMI 的封鎖公用存取。

Console

在指定區域中停用「封鎖 AMI 的公有存取權」

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽列 (位於畫面頂端) 中，選取要停用「封鎖 AMI 的公有存取權」的區域。
3. 如果儀表板未顯示，請在導覽窗格中，選擇 EC2 儀表板。
4. 在帳戶屬性下，選擇資料保護和安全性。
5. 在封鎖 AMI 的公有存取權下，選擇管理。
6. 清除封鎖新的公開共用核取方塊，然後選擇更新。

7. 出現確認提示時，請輸入 **confirm**，然後選擇允許公開共用。

Note

API 可能最多需要 10 分鐘的時間來設定此設定。在此期間，該值將是已封鎖新的公開共用。當 API 完成設定時，該值會自動變更為已允許新的公開共用。

AWS CLI

在指定區域中停用「封鎖 AMI 的公有存取權」

使用 [disable-image-block-public-access](#) 命令，並指定要在其中停用「封鎖 AMI 的公有存取權」的區域。

```
aws ec2 disable-image-block-public-access --region us-east-1
```

預期的輸出結果

```
{
  "ImageBlockPublicAccessState": "unblocked"
}
```

Note

API 可能最多需要 10 分鐘的時間來設定此設定。在此期間，如果您執行 [get-image-block-public-access-state](#) 命令，則回應將為 `block-new-sharing`。當 API 完成設定時，回應將為 `unblocked`。

檢視封鎖 AMI 的公有存取權的狀態

若要查看您帳戶中 AMI 的公開共用是否已封鎖，您可以檢視封鎖 AMI 之公有存取權的狀態。您必須檢視您希望查看您的 AMI 的公開共用是否已封鎖之每個 AWS 區域的狀態。

Console

在指定區域中檢視封鎖 AMI 的公有存取權的狀態

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。

2. 在導覽列 (位於畫面頂端) 中，選取要檢視封鎖 AMI 的公有存取權的狀態的區域。
3. 如果儀表板未顯示，請在導覽窗格中，選擇 EC2 儀表板。
4. 在帳戶屬性下，選擇資料保護和安全性。
5. 在封鎖 AMI 的公有存取權下，請勾選公有存取權欄位。值可能是已封鎖新的公開共用，也可能是已允許新的公開共用。

AWS CLI

在指定區域中獲取封鎖 AMI 的公有存取權的狀態

使用 [get-image-block-public-access-state](#) 命令，並指定要在其中獲取封鎖 AMI 的公有存取權的狀態的區域。

```
aws ec2 get-image-block-public-access-state --region us-east-1
```

預期的輸出結果 – 值為 `block-new-sharing` 或 `unblocked`。

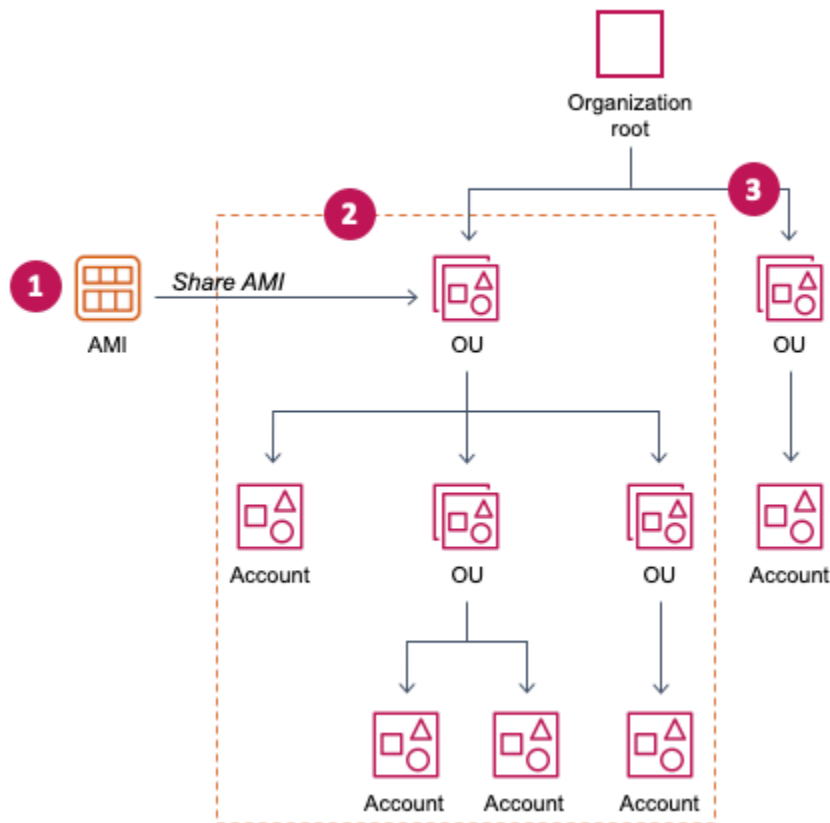
```
{
  "ImageBlockPublicAccessState": "block-new-sharing"
}
```

與特定組織或組織單位共用 AMI

[AWS Organizations](#) 是一種帳戶管理服務，可讓您 AWS 帳戶 將多個組織合併到您建立並集中管理的組織中。除了[與指定的帳戶共享 AMI](#) 之外，您還可以與組織或您建立的組織單位 (OU) 共享 AMI。

組織是您建立來合併並集中管理 AWS 帳戶的實體。您可以將帳戶組織為階層式、類似樹狀目錄的結構，其中的[根](#)位於頂層，而[組織單位](#)在組織根下形成巢狀。每個帳戶都可以直接新增至根中，或放在階層中的其中一個 OU 中。如需帳戶類型的詳細資訊，請參閱 AWS Organizations 使用者指南中的 [AWS Organizations 術語與概念](#)。

當您與組織或 OU 共用 AMI 時，所有子帳戶都可以存取 AMI。例如，在下圖中，與頂層 OU 共用 AMI (由編號 1 的箭頭表示)。在該頂層 OU 之下形成巢狀的所有 OU 和帳戶 (由編號 2 的虛線表示) 也可以存取 AMI。組織中的帳戶和虛線外的 OU (由編號 3 表示) 無法存取 AMI，因為它們不是與其共用 AMI 之 OU 的子項。



考量事項

與特定組織或組織單位共享 AMI 時，請考慮以下項目。

- 擁有權 – 若要共用 AMI，您的 AWS 帳戶 必須擁有 AMI。
- 共用限制 – AMI 擁有者可以與任何組織或 OU 共用 AMI，包括他們不是其成員的組織和 OU。

如需可在區域內共用 AMI 的最大實體數量，請參閱 [Amazon EC2 服務配額](#)。

- 標籤 – 您無法共用使用者定義的標籤 (連接至 AMI 的標籤)。當您共用 AMI 時，您的使用者定義標籤不適用 AWS 帳戶 於與 AMI 共用的組織或 OU 中的任何標籤。
- ARN 格式 - 在命令中指定組織或 OU 時，請務必使用正確的 ARN 格式。如果只指定 ID，則您會收到錯誤，例如，如果只指定 o-123example 或 ou-1234-5example。

正確的 ARN 格式：

- 組織 ARN : `arn:aws:organizations::account-id:organization/organization-id`
- OU ARN : `arn:aws:organizations::account-id:ou/organization-id/ou-id`

其中：

- *account-id* 是 12 位數的管理帳號，例如，123456789012。如果不知道管理帳號，您可以描述組織或組織單位以取得 ARN，其中包括管理帳號。如需詳細資訊，請參閱 [取得 ARN](#)。
- *organization-id* 是組織 ID，例如，o-123example。
- *ou-id* 是組織單位 ID，例如，ou-1234-5example。

如需 ARN 格式的詳細資訊，請參閱 IAM 使用者指南中的 [Amazon 資源名稱 \(ARN\)](#)。

- 加密和金鑰 – 您可以共用未加密和加密快照所支援的 AMI。
 - 加密快照必須透過客戶受管金鑰進行加密。您無法共用由使用預設 AWS 受管理金鑰加密的快照支援的 AMI。
 - 如果您共用由加密快照支援的 AMI，您必須允許組織或 OU 使用用於加密快照的客戶管理金鑰。如需詳細資訊，請參閱 [允許組織和 OU 使用 KMS 金鑰](#)。
- 區域 - AMI 是一種區域性的資源。共享 AMI 後，只能從共享 AMI 的區域中使用 AMI。若要讓 AMI 在其他區域可用，請將 AMI 複製到該區域，然後共用。如需詳細資訊，請參閱 [複製 AMI](#)。
- 使用 - 共享 AMI 後，使用者只能從 AMI 啟動執行個體。他們無法刪除、共用或修改它。不過，他們在使用您的 AMI 啟動了執行個體之後，就可以從其啟動的執行個體建立 AMI。
- 帳單 — 當其 AWS 帳戶 他人使用您的 AMI 啟動執行個體時，不會向您收取費用。使用 AMI 啟動執行個體的帳戶會根據它們啟動的執行個體計費。

允許組織和 OU 使用 KMS 金鑰

如果您共用由加密快照支援的 AMI，您也必須允許組織或 OU 使用 AWS KMS keys 用來加密快照的 AMI。

使用 `aws:PrincipalOrgID` 和 `aws:PrincipalOrgPaths` 引鍵來比較要求原則中 AWS Organizations 路徑的主參與者路徑的路徑。該主體可以是使用者、IAM 角色、聯合身分使用者或 AWS 帳戶 根使用者。在政策中，此條件金鑰可確保申請者是 AWS Organizations 中所指定組織根或中 OU 內的帳戶成員。如需更多條件陳述式範例，請參閱《IAM 使用者指南》中的 [aws:PrincipalOrgID](#) 和 [aws:PrincipalOrgPaths](#)。

如需有關編輯金鑰原則的詳細資訊，請參閱 AWS Key Management Service 開發人員指南 [中的允許其他帳戶中的使用者使用 KMS 金鑰](#)。

若要許可組織或 OU 使用 KMS 金鑰，請將以下陳述式新增至金鑰政策。

```
{
  "Sid": "Allow access for organization root",
  "Effect": "Allow",
```

```

"Principal": "*",
"Action": [
  "kms:Describe*",
  "kms:List*",
  "kms:Get*",
  "kms:Encrypt",
  "kms:Decrypt",
  "kms:ReEncrypt*",
  "kms:GenerateDataKey*"
],
"Resource": "*",
"Condition": {
  "StringEquals": {
    "aws:PrincipalOrgID": "o-123example"
  }
}
}

```

若要與多個 OU 共用 KMS 金鑰，您可以使用類似下列範例的政策。

```

{
  "Sid": "Allow access for specific OUs and their descendants",
  "Effect": "Allow",
  "Principal": "*",
  "Action": [
    "kms:Describe*",
    "kms:List*",
    "kms:Get*",
    "kms:Encrypt",
    "kms:Decrypt",
    "kms:ReEncrypt*",
    "kms:GenerateDataKey*"
  ],
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "aws:PrincipalOrgID": "o-123example"
    },
    "ForAnyValue:StringLike": {
      "aws:PrincipalOrgPaths": [
        "o-123example/r-ab12/ou-ab12-33333333/*",
        "o-123example/r-ab12/ou-ab12-22222222/*"
      ]
    }
  }
}

```

```
    }  
  }  
}
```

共用 AMI

您可以使用 Amazon EC2 主控台或與組織或 OU 共用 AMI。AWS CLI

共用 AMI (主控台)

使用主控台與組織或 OU 共用 AMI

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 AMIs (AMI)。
3. 在清單中選取您的 AMI，然後選取 Actions (動作)、Edit AMI permissions (編輯 AMI 許可)。
4. 在 AMI availability (AMI 可用性) 下，選擇 Private (私有)。
5. 在 Shared organizations/OUs (共用組織/OU) 旁邊，選擇 Add organization/OU ARN (新增組織/OU ARN)。
6. 對於 Organization/OU ARN (組織/OU ARN)，輸入要與之共用 AMI 的組織 ARN 或 OU ARN，然後選擇 Share AMI (共用 AMI)。請注意，您必須指定完整的 ARN，而不只是 ID。

若要與多個組織或 OU 共用此 AMI，請重覆此步驟，直到新增所有需要的組織或 OU。

Note

共享 AMI 時，不需要共享 AMI 所參考的 Amazon EBS 快照。需要共用的只有 AMI 本身，而且系統可自動讓執行個體存取參考的 Amazon EBS 快照，以進行啟動。不過，您確實需要共用用來加密 AMI 參照之快照的 KMS 金鑰。如需詳細資訊，請參閱[允許組織和 OU 使用 KMS 金鑰](#)。

7. 完成後，請選擇 Save changes (儲存變更)。
8. (選用) 若要查看與其共用 AMI 的組織或 OU，請從清單中選取 AMI，然後選取 Permissions (許可) 索引標籤，向下滾動至 Shared organizations/OUs (共用的組織/OU)。若要尋找與您共用的 AMI，請參閱[尋找共享 AMI](#)。

共用 AMI (視窗工具 PowerShell)

使用 [Edit-EC2ImageAttribute](#) 命令 (Windows 工具 PowerShell) 來共用 AMI，如下列範例所示。

與組織或 OU 共用 AMI

下列命令可將所指定 AMI 的啟動許可授予指定的組織。

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -  
Attribute launchPermission -OperationType add -OrganizationArn  
"arn:aws:organizations::123456789012:organization/o-123example"
```

Note

共享 AMI 時，不需要共享 AMI 所參考的 Amazon EBS 快照。需要共用的只有 AMI 本身，而且系統可自動讓執行個體存取參考的 Amazon EBS 快照，以進行啟動。不過，您的確需要共用 KMS 金鑰，而這些金鑰用來加密 AMI 參考的快照。如需詳細資訊，請參閱[允許組織和 OU 使用 KMS 金鑰](#)。

停止與組織或 OU 共用 AMI

下列命令可從指定的組織移除所指定 AMI 的啟動許可。

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -  
Attribute launchPermission -OperationType remove -OrganizationArn  
"arn:aws:organizations::123456789012:organization/o-123example"
```

停止與所有組織、OU 和共用 AMI AWS 帳戶

下列命令可從指定的 AMI 移除所有的公有和明確啟動許可。請注意，AMI 擁有者一律具備啟動許可，因此不受此命令影響。

```
PS C:\> Reset-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute  
launchPermission
```

共用 AMI (AWS CLI)

使用 [modify-image-attribute](#) 命令 (AWS CLI) 來共用 AMI。

若要與組織共用 AMI，請使用 AWS CLI

[modify-image-attribute](#) 命令可將所指定 AMI 的啟動許可授予給指定組織。請注意，您必須指定完整的 ARN，而不只是 ID。

```
aws ec2 modify-image-attribute \  
  --image-id ami-0abcdef1234567890 \  
  --launch-permission  
  "Add=[{OrganizationArn=arn:aws:organizations::123456789012:organization/  
o-123example}]]"
```

若要使用與 OU 共用 AMI AWS CLI

[modify-image-attribute](#) 命令可將所指定 AMI 的啟動許可授予指定的 OU。請注意，您必須指定完整的 ARN，而不只是 ID。

```
aws ec2 modify-image-attribute \  
  --image-id ami-0abcdef1234567890 \  
  --launch-permission  
  "Add=[{OrganizationalUnitArn=arn:aws:organizations::123456789012:ou/o-123example/  
ou-1234-5example}]]"
```

Note

共享 AMI 時，不需要共享 AMI 所參考的 Amazon EBS 快照。需要共用的只有 AMI 本身，而且系統可自動讓執行個體存取參考的 Amazon EBS 快照，以進行啟動。不過，您的確需要共用 KMS 金鑰，而這些金鑰用來加密 AMI 參考的快照。如需詳細資訊，請參閱 [允許組織和 OU 使用 KMS 金鑰](#)。

停止共用 AMI

您可以使用 Amazon EC2 主控台或停 AWS CLI 止與組織或 OU 共用 AMI。

停止共用 AMI (主控台)

使用主控台停止與組織或 OU 共用 AMI

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 AMIs (AMI)。
3. 在清單中選取您的 AMI，然後選取 Actions (動作)、Edit AMI permissions (編輯 AMI 許可)。
4. 在 Shared organizations/OUs (共用的組織/OU)，選取要停止與其共用 AMI 的組織或 OU，然後選取 Remove Selected (移除所選項)。

5. 完成後，請選擇 Save changes (儲存變更)。
6. (選用) 若要確認您已停止與組織或 OU 共用 AMI，請從清單中選取此 AMI，然後選取 Permissions (許可) 索引標籤，向下滾動至 Shared organizations/OUs (共用的組織/OU)。

停止共用 AMI (AWS CLI)

使用 [modify-image-attribute](#) 或 [reset-image-attribute](#) 命令 (AWS CLI) 來停止共用 AMI。

若要停止與組織或 OU 共用 AMI AWS CLI

[modify-image-attribute](#) 命令可從指定的組織中移除所指定 AMI 的啟動許可。請注意，您必須指定 ARN。

```
aws ec2 modify-image-attribute \  
  --image-id ami-0abcdef1234567890 \  
  --launch-permission  
  "Remove=[{OrganizationArn=arn:aws:organizations::123456789012:organization/  
o-123example}]"
```

若要停止與所有組織、OU 共用 AMI，並 AWS 帳戶 使用 AWS CLI

[reset-image-attribute](#) 命令可從指定的 AMI 移除所有的公有和明確啟動許可。請注意，AMI 擁有者始終具備啟動許可，因此不受此命令影響。

```
aws ec2 reset-image-attribute \  
  --image-id ami-0abcdef1234567890 \  
  --attribute launchPermission
```

Note

如果特定帳戶位於與其共用 AMI 的組織或 OU 中，則無法停止與該帳戶共用 AMI。如果您嘗試透過移除帳戶的啟動許可來停止共用 AMI，Amazon EC2 會傳回成功訊息。不過，會繼續與該帳戶共用 AMI。

查看與之共用 AMI 的組織和 OU

您可以使用 Amazon EC2 主控台或檢 AWS CLI 查您已與哪些組織和 OU 共用您的 AMI。

查看與之共用 AMI 的組織和 OU (主控台)

使用主控台檢查已與哪些組織和 OU 共用 AMI

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 AMIs (AMI)。
3. 從清單中選取 AMI，選取 Permissions (許可) 索引標籤，然後向下滾動至 Shared organizations/ OUs (共用的組織/OU)。

若要尋找與您共用的 AMI，請參閱[尋找共享 AMI](#)。

查看與之共用 AMI 的組織和 OU (AWS CLI)

您可以使用 [describe-image-attribute](#) 命令 (AWS CLI) 和 `launchPermission` 屬性，檢查已與哪些組織和 OU 共用您的 AMI。

若要檢查您使用的 AMI 與哪些組織和 OU 共用 AWS CLI

[describe-image-attribute](#) 命令會描述所指定 AMI 的 `launchPermission` 屬性，並傳回您與其共用 AMI 的組織和 OU。

```
aws ec2 describe-image-attribute \  
  --image-id ami-0abcdef1234567890 \  
  --attribute launchPermission
```

回應範例

```
{  
  "ImageId": "ami-0abcdef1234567890",  
  "LaunchPermissions": [  
    {  
      "OrganizationalUnitArn": "arn:aws:organizations::111122223333:ou/  
o-123example/ou-1234-5example"  
    }  
  ]  
}
```

取得 ARN

組織和組織單位 ARN 包含 12 位數的管理帳號。如果不知道管理帳號，您可以描述組織或組織單位以取得其中每一個的 ARN。在下列範例中，123456789012 是管理帳號。

在取得 ARN 之前，您必須具有描述組織和組織單位的許可。下列政策提供必要的許可。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "organizations:Describe*"
      ],
      "Resource": "*"
    }
  ]
}
```

取得組織的 ARN

使用 [describe-organization](#) 命令和設定為 'Organization.Arn' 的 --query 參數，以只傳回組織 ARN。

```
aws organizations describe-organization --query 'Organization.Arn'
```

回應範例

```
"arn:aws:organizations::123456789012:organization/o-123example"
```

取得組織單位的 ARN

使用 [describe-organizational-unit](#) 命令，指定 OU ID，並將 --query 參數設定為 'OrganizationalUnit.Arn'，以只傳回組織單位 ARN。

```
aws organizations describe-organizational-unit --organizational-unit-id ou-1234-5example --query 'OrganizationalUnit.Arn'
```

回應範例

```
"arn:aws:organizations::123456789012:ou/o-123example/ou-1234-5example"
```

與指定的 AWS 帳戶共享 AMI

您可以與特定的 AMI 共享，AWS 帳戶而無需公開 AMI。所有你需要的是 AWS 帳戶 ID。

AWS 帳戶 ID 是一個 12 位數的數字，例如 012345678901，可唯一 AWS 帳戶識別。如需詳細資訊，請參閱《AWS Account Management 參考指南》中的[檢視 AWS 帳戶 識別符](#)。

考量事項

與特 AWS 帳戶定的共用 AMI 時，請考慮下列事項。

- 擁有權 – 若要共用 AMI，您的 AWS 帳戶 必須擁有 AMI。
- 共用限制 - 如需可在區域內共用 AMI 的最大實體數量，請參閱 [Amazon EC2 服務配額](#)。
- 標籤 – 您無法共用使用者定義的標籤 (連接至 AMI 的標籤)。當您共享 AMI 時，您的用戶定義標籤不適用於與 AMI 共享的任何 AWS 帳戶 標籤。
- 加密和金鑰 – 您可以共用未加密和加密快照所支援的 AMI。
 - 加密快照必須透過 KMS 金鑰進行加密。您無法共用透過預設 AWS 受管金鑰加密之快照支援的 AMI。
 - 如果您共用由加密快照支援的 AMI，則必須允許使用用 AWS 帳戶 來加密快照的 KMS 金鑰。如需詳細資訊，請參閱[允許組織和 OU 使用 KMS 金鑰](#)。若要設定在使用客戶受管金鑰進行加密時啟動 Auto Scaling 執行個體所需的金鑰 [AWS KMS key 政策](#)，請參閱 [Amazon EC2 Auto Scaling 使用者指南中的加密磁碟區](#)使用的必要政策。
- 區域 - AMI 是一種區域性的資源。當您共用 AMI 時，其只能在該區域中使用。若要讓 AMI 在其他區域可用，請將 AMI 複製到該區域，然後共用。如需詳細資訊，請參閱 [複製 AMI](#)。
- 使用 - 共享 AMI 後，使用者只能從 AMI 啟動執行個體。他們無法刪除、共用或修改它。不過，在使用您的 AMI 啟動了執行個體之後，他們可以從其執行個體建立 AMI。
- 複製共享 AMI – 如果其他帳戶中的使用者想要複製共享 AMI，則必須向他們授予用於支援 AMI 的儲存的讀取許可。如需詳細資訊，請參閱 [跨帳戶複製](#)。
- 帳單 — 當其 AWS 帳戶 他人使用您的 AMI 啟動執行個體時，不會向您收取費用。使用 AMI 啟動執行個體的帳戶會根據它們啟動的執行個體計費。

共享 AMI (主控台)

使用主控台授予明確啟動許可

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 AMIs (AMI)。
3. 在清單中選取您的 AMI，然後選取 Actions (動作)、Edit AMI permissions (編輯 AMI 許可)。

4. 選擇 Private (私有)。
5. 在 Shared accounts (共享帳戶) 中，選擇 Add account ID (新增帳戶 ID)。
6. 針對 AWS 帳戶 ID，輸入您要與其共用 AMI 的 AWS 帳戶 ID，然後選擇 [共用 AMI]。

若要與多個帳戶共用此 AMI，請重覆步驟 5 和 6，直到您新增了所有必要的帳戶 ID。

Note

共享 AMI 時，不需要共享 AMI 所參考的 Amazon EBS 快照。要共享的只有 AMI 本身；系統會自動為執行個體提供所參考 Amazon EBS 快照的存取，供其啟動。不過，您的確需要共享任何用來加密快照的 KMS 金鑰，而 AMI 會參考這些快照。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的共用 Amazon EBS 快照](#)。

7. 完成後，請選擇 Save changes (儲存變更)。
8. (選擇性) 若要檢視與您共用 AMI 的 AWS 帳戶 ID，請在清單中選取 AMI，然後選擇權限索引標籤。若要尋找與您共用的 AMI，請參閱 [尋找共享 AMI](#)。

共用 AMI (視窗工具 PowerShell)

使用 [Edit-EC2ImageAttribute](#) 命令 (Windows 工具 PowerShell) 來共用 AMI，如下列範例所示。

授予明確啟動許可

下列命令可向指定的 AWS 帳戶授予指定 AMI 的啟動許可。在下列範例中，以有效的 AMI 識別碼取代範例 AMI ID，並以 12 位數 AWS 帳戶 識別碼取代 *account-id*。

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute  
launchPermission -OperationType add -UserId "account-id"
```

Note

共享 AMI 時，不需要共享 AMI 所參考的 Amazon EBS 快照。要共享的只有 AMI 本身；系統會自動為執行個體提供所參考 Amazon EBS 快照的存取，供其啟動。不過，您的確需要共享任何用來加密快照的 KMS 金鑰，而 AMI 會參考這些快照。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的共用 Amazon EBS 快照](#)。

移除帳戶的啟動許可

下列命令可從指定的 AWS 帳戶移除指定 AMI 的啟動許可。在下列範例中，以有效的 AMI 識別碼取代範例 AMI ID，並以 12 位數 AWS 帳戶 識別碼取代 *account-id*。

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute  
launchPermission -OperationType remove -UserId "account-id"
```

移除所有的啟動許可

下列命令可從指定的 AMI 移除所有的公有和明確啟動許可。請注意，AMI 擁有者始終具備啟動許可，因此不受此命令影響。在下列範例中，將範例 AMI ID 取代為有效的 AMI ID。

```
PS C:\> Reset-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute  
launchPermission
```

共用 AMI (AWS CLI)

使用 [modify-image-attribute](#) 命令 (AWS CLI) 來共用 AMI，如下列範例所示。

授予明確啟動許可

下列命令可向指定的 AWS 帳戶授予指定 AMI 的啟動許可。在下列範例中，以有效的 AMI 識別碼取代範例 AMI ID，並以 12 位數 AWS 帳戶 識別碼取代 *account-id*。

```
aws ec2 modify-image-attribute \  
  --image-id ami-0abcdef1234567890 \  
  --launch-permission "Add=[{UserId=account-id}]"
```

Note

共享 AMI 時，不需要共享 AMI 所參考的 Amazon EBS 快照。要共享的只有 AMI 本身；系統會自動為執行個體提供所參考 Amazon EBS 快照的存取，供其啟動。不過，您的確需要共享任何用來加密快照的 KMS 金鑰，而 AMI 會參考這些快照。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的共用 Amazon EBS 快照](#)。

移除帳戶的啟動許可

下列命令可從指定的 AWS 帳戶移除指定 AMI 的啟動許可。在下列範例中，以有效的 AMI 識別碼取代範例 AMI ID，並以 12 位數 AWS 帳戶 識別碼取代 *account-id*。


```
aws ec2 modify-image-attribute \  
  --image-id ami-0abcdef1234567890 \  
  --launch-permission "Remove=[{UserId=account-id}]"
```

移除所有的啟動許可

下列命令可從指定的 AMI 移除所有的公有和明確啟動許可。請注意，AMI 擁有者始終具備啟動許可，因此不受此命令影響。在下列範例中，將範例 AMI ID 取代為有效的 AMI ID。

```
aws ec2 reset-image-attribute \  
  --image-id ami-0abcdef1234567890 \  
  --attribute launchPermission
```

取消將 AMI 與您的共享 AWS 帳戶

透過將帳戶新增到 AMI 的啟動許可，可以[與指定的 AWS 帳戶共用](#) Amazon Machine Image (AMI)。如果 AMI 已與您共享，AWS 帳戶而您不再希望該 AMI 與您的帳戶共享，則可以從 AMI 的啟動權限中刪除您的帳戶。您可以透過執行 `cancel-image-launch-permission` AWS CLI 命令來執行此操作。執行此命令時，會 AWS 帳戶 從指定 AMI 的啟動權限中移除。

例如，您可能會取消與您的帳戶共用 AMI，以降低啟動與您共用的未使用或已棄用 AMI 的執行個體的可能性。當您取消與您的帳戶共用 AMI 時，它將不再出現在 EC2 主控台的任何 AMI 清單或 [describe-images](#) 的輸出中。

主題

- [限制](#)
- [取消與您的帳戶共用 AMI](#)
- [尋找與您帳戶共用的 AMI](#)

限制

- 您可以從僅與您共享的 AMI 的啟動權限中刪除您的帳戶。您無法使用 `cancel-image-launch-permission` 從[與組織或組織單位 \(OU\) 共用的 AMI](#) 的啟動許可中移除帳戶，也無法移除公用 AMI 的存取權。
- 您無法從 AMI 的啟動許可中永久移除您的帳戶。AMI 擁有者可再次與您的帳戶共用 AMI。
- AMI 是一種區域性的資源。執行 `cancel-image-launch-permission` 時，必須指定 AMI 所在的區域。在命令中指定區域，或使用 `AWS_DEFAULT_REGION` [環境變數](#)。

- 只有 AWS CLI 和 SDK 支援從 AMI 的啟動權限中移除您的帳戶。EC2 主控台目前不支援此動作。

取消與您的帳戶共用 AMI

Note

取消與您的帳戶共用 AMI 後，將無法復原。若要重新獲得對 AMI 的存取權，AMI 擁有者必須與您的帳戶共用它。

AWS CLI

若要取消與您的共用 AMI AWS 帳戶

使用 [cancel-image-launch-permission](#) 命令並指定 AMI ID。

```
aws ec2 cancel-image-launch-permission \  
  --image-id ami-0123456789example \  
  --region us-east-1
```

預期的輸出結果

```
{  
  "Return": true  
}
```

PowerShell

若要取消 AMI 與您的 AWS 帳戶 使用 AWS Tools for PowerShell

使用 [Stop-EC2ImageLaunchPermission](#) 命令並指定 AMI ID。

```
Stop-EC2ImageLaunchPermission \  
  -ImageId ami-0123456789example \  
  -Region us-east-1
```

預期的輸出結果

```
True
```

尋找與您帳戶共用的 AMI

若要尋找與您共用的 AMI AWS 帳戶，請參閱[尋找共享 AMI](#)。

使用書籤

如果您已建立公用 AMI，或與其他人共用 AMI AWS 帳戶，您可以建立書籤，讓使用者存取您的 AMI，並立即在自己的帳戶中啟動執行個體。這是共享 AMI 參考的簡單方法，如此使用者便不需要為了使用您的 AMI 而花時間尋找。

請注意，您的 AMI 必須為公有，或您必須與想要傳送書籤的使用者共享。

建立 AMI 的書籤

1. 輸入 URL 和下列資訊，其中 region 為您 AMI 所在的區域：

```
https://console.aws.amazon.com/ec2/v2/home?
region=region#LaunchInstanceWizard:ami=ami_id
```

例如，此 URL 可從美國東部 (維吉尼亞北部) us-east-1 區域的 ami-0abcdef1234567890 AMI 中啟動執行個體：

```
https://console.aws.amazon.com/ec2/v2/home?region=us-
east-1#LaunchInstanceWizard:ami=ami-0abcdef1234567890
```

2. 將連結發佈給想要使用您 AMI 的使用者。
3. 若要使用書籤，請選擇連結或複製連結並貼到瀏覽器中。Launch Wizard 會隨即開啟，並選取 AMI。

共享 Linux AMI 的準則

請使用下列準則來減少可能遭受攻擊的範圍，以及改善所建立 AMI 的可靠性。

Important

沒有任何一份清單可詳盡列出所有的安全性準則。請小心建置您的共享 AMI，並花些時間思考那些位置可能暴露機密資料。

目錄

- [更新 AMI 工具之後再使用](#)
- [停用根使用者的密碼型態遠端登入](#)
- [停用本機根存取](#)
- [移除 SSH 主機金鑰對](#)
- [安裝公有金鑰登入資料](#)
- [停用 sshd DNS 檢查 \(選擇性\)](#)
- [自我保護](#)

如果您要為其建立 AMI AWS Marketplace，請參閱《AWS Marketplace 賣家指南》中[建立 AMI 的最佳做法](#)，以瞭解準則、政策和最佳做法。

如需有關安全共享 AMI 的詳細資訊，請參閱以下文章：

- [如何安全地共享及使用公用 AMI](#)
- [Public AMI Publishing: Hardening and Clean-up Requirements](#)

更新 AMI 工具之後再使用

如為執行個體存放區後端 AMI，建議您的 AMI 先下載並升級 Amazon EC2 AMI 建立工具之後再使用。如此可確保以共享 AMI 為基礎的新 AMI 取得最新的 AMI 工具。

若為 [Amazon Linux 2](#)，請使用下列命令安裝 aws-amitools-ec2 套件，並將 AMI 工具新增至 PATH。若為 [Amazon Linux AMI](#)，則預設已安裝 aws-amitools-ec2 套件。

```
[ec2-user ~]$ sudo yum install -y aws-amitools-ec2 && export PATH=$PATH:/opt/aws/bin  
> /etc/profile.d/aws-amitools-ec2.sh && . /etc/profile.d/aws-amitools-ec2.sh
```

使用以下命令更新 AMI 工具：

```
[ec2-user ~]$ sudo yum upgrade -y aws-amitools-ec2
```

如為其他的發行版本，請確定您取得最新的 AMI 工具。

停用根使用者的密碼型態遠端登入

公用 AMI 使用固定的根密碼會帶來安全風險，且資訊會迅速向外散佈。即便仰賴使用者於第一次登入時變更密碼，都會留下一些可能遭到濫用的空窗。

若要解決此問題，請停用根使用者的密碼型態遠端登入。

停用根使用者的密碼型態遠端登入

1. 使用文字編輯器開啟 `/etc/ssh/sshd_config` 檔案，並找出下行：

```
#PermitRootLogin yes
```

2. 將行變為：

```
PermitRootLogin without-password
```

此組態檔案的位置可能與您的發行版本不同，或您不是執行 OpenSSH。若是這種情況，請參閱相關文件。

停用本機根存取

使用共享 AMI 時，最佳實務之一就是停用直接根登入。若要停用，請登入運作中的執行個體，然後發出下列命令：

```
[ec2-user ~]$ sudo passwd -l root
```

Note

此命令不會影響 `sudo` 的使用。

移除 SSH 主機金鑰對

如果您打算共享來自公用 AMI 的 AMI，請移除 `/etc/ssh` 中現有的 SSH 主機金鑰對。有其他人使用您的 AMI 啟動執行個體時，會強迫 SSH 產生新的唯一 SSH 金鑰對，藉此以改善安全性和減少發生「中間人」攻擊的可能性。

移除位在您系統中所有下列的金鑰檔案。

- `ssh_host_dsa_key`
- `ssh_host_dsa_key.pub`
- `ssh_host_key`
- `ssh_host_key.pub`

- ssh_host_rsa_key
- ssh_host_rsa_key.pub
- ssh_host_ecdsa_key
- ssh_host_ecdsa_key.pub
- ssh_host_ed25519_key
- ssh_host_ed25519_key.pub

您可以使用下列命令安全地移除所有這些檔案：

```
[ec2-user ~]$ sudo shred -u /etc/ssh/*_key /etc/ssh/*_key.pub
```

Warning

像是 **shred** 等安全移除公用程式可能不會從儲存媒體中移除檔案的所有複本。檔案的隱藏複本可能由日誌檔案系統 (包含 Amazon Linux 預設 ext4)、快照、備份、RAID 和暫時性快取建立。如需詳細資訊，請參閱 [shred 文件](#)。

Important

如果您忘記從公用 AMI 中移除現有的 SSH 主機金鑰對，我們的定期稽核流程會通知您和執行您可能有安全風險之 AMI 執行個體的所有客戶。過一段短時間的寬限期後，我們會將 AMI 標示為私有。

安裝公有金鑰登入資料

設定 AMI 避免使用密碼登入後，您必須確定使用者可用其他機制登入。

Amazon EC2 允許使用者在啟動執行個體時指定公私金鑰對名稱。為 RunInstances API 呼叫 (或透過命令列 API 工具) 提供有效的金鑰對名稱時，公有金鑰 (呼叫 CreateKeyPair 或 ImportKeyPair 後金鑰對中 Amazon EC2 保留在伺服器上的部分) 會透過執行個體中繼資料的 HTTP 查詢提供給執行個體。

若要透過 SSH 登入，您的 AMI 必須在開機時擷取金鑰值，並將其附加至 /root/.ssh/authorized_keys (或 AMI 上任何其他使用者帳戶的對等檔案)。使用者可用金鑰對啟動您 AMI 的執行個體並登入，無需根密碼。

包括 Amazon Linux 和 Ubuntu 等許多發行版本皆使用 `cloud-init` 套件來注入所設定使用者的公有金鑰登入資料。如果您的發行版本不支援 `cloud-init`，您可新增下列程式碼至系統啟動指令碼 (例如 `/etc/rc.local`)，以納入您在啟動時為根使用者指定的公有金鑰。

Note

在下列範例中，IP 地址 `http://169.254.169.254/` 是 link-local 地址且僅在執行個體中有效。

IMDSv2

```
if [ ! -d /root/.ssh ] ; then
    mkdir -p /root/.ssh
    chmod 700 /root/.ssh
fi
# Fetch public key using HTTP
TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-
data/public-keys/0/openssh-key > /tmp/my-key
if [ $? -eq 0 ] ; then
    cat /tmp/my-key >> /root/.ssh/authorized_keys
    chmod 700 /root/.ssh/authorized_keys
    rm /tmp/my-key
fi
```

IMDSv1

```
if [ ! -d /root/.ssh ] ; then
    mkdir -p /root/.ssh
    chmod 700 /root/.ssh
fi
# Fetch public key using HTTP
curl http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key > /tmp/my-key
if [ $? -eq 0 ] ; then
    cat /tmp/my-key >> /root/.ssh/authorized_keys
    chmod 700 /root/.ssh/authorized_keys
    rm /tmp/my-key
fi
```

這會套用到任何使用者；無需將其限制到 `root` 使用者。

Note

重新綁定以此 AMI 為基礎的執行個體，會加入其啟動金鑰。若要避免加入金鑰，您必須清除 (或刪除) `authorized_keys` 檔案或將此檔案排除於重新綁定之外。

停用 sshd DNS 檢查 (選擇性)

停用 sshd DNS 檢查會稍微降低 sshd 的安全性。但假如 DNS 解析失敗，SSH 登入仍能如預期運作。如果您不停用 sshd 檢查，DNS 解析失敗將封鎖所有登入。

停用 sshd DNS 檢查

1. 使用文字編輯器開啟 `/etc/ssh/sshd_config` 檔案，並找出下行：

```
#UseDNS yes
```

2. 將行變為：

```
UseDNS no
```

Note

此組態檔案的位置可能與您的發行版本不同，或您不是執行 OpenSSH。若是這種情況，請參閱相關文件。

自我保護

我們建議您不要將機密資料或軟體存放在共享的任何 AMI 上。啟動共享 AMI 的使用者可能會重新綁定，並將其註冊為自己的 AMI。請按照以下準則幫助您避免一些容易遭到忽視的安全風險：

- 建議使用 `--exclude directory` 的 `ec2-bundle-vol` 選項跳過含有您不想加入到套件內之機密資訊的任何目錄和子目錄。尤其請在綁定映像時排除所有使用者擁有的 SSH 公有/私有金鑰對和 SSH `authorized_keys` 檔案。Amazon 公有 AMI 會將這些資訊存放在根使用者的 `/root/.ssh`，和一般使用者的 `/home/user_name/.ssh/` 內。如需詳細資訊，請參閱 [ec2-bundle-vol](#)。

- 綁定前務必刪除 shell 歷程記錄。如果您在同一 AMI 嘗試一次以上的套件上傳，shell 歷程記錄會包含您的存取金鑰。以下範例應該是您在從執行個體內綁定之前執行的最後一項命令。

```
[ec2-user ~]$ shred -u ~/.*history
```

Warning

上面警告中提及的 **shred** 限制同樣適用於此處。

請注意，bash 會在退出時將目前工作階段的歷程記錄寫入到磁碟。如果您在刪除 `~/.bash_history` 後登出執行個體，然後再重新登入，您會發現 `~/.bash_history` 已重新建立，並包含在先前工作階段中執行的所有命令。

bash 以外的其他程式也會將歷程記錄寫入到磁碟。請小心使用，並移除或排除不必要以點開頭的檔案或以點開頭的目錄。

- 綁定執行中的執行個體需要您的私有金鑰和 X.509 憑證。因此請將這些憑證和其他登入資料放在不會加入綁定的位置 (例如執行個體存放區)。

已支付 AMI

付費 AMI 是在中列出出售的 AMI AWS Marketplace。這 AWS Marketplace 是一個在線商店，您可以在其中購買可運行的軟件 AWS，包括可用於啟動 EC2 實例的 AMI。AWS Marketplace AMI 分為類別，例如「開發人員工具」，以便您找到符合您需求的產品。有關更多信息 AWS Marketplace，請[AWS Marketplace](#)訪問網站。

您可以 AWS Marketplace 從第三方購買 AMI，包括來自 Red Hat 等組織的服務合約的 AMI。您還可以創建 AMI 並將其出售 AWS Marketplace 給其他 Amazon EC2 用戶。只要遵循一些簡單準則，您即可經由簡單明瞭的程序建立供公眾取用的安全可用 AMI。如需如何建立和使用共用 AMI 的資訊，請參閱[共享 AMI](#)。

從已支付 AMI 啟動執行個體與從任何其他 AMI 啟動執行個體的方式相同。不需額外的參數。執行個體會根據 AMI 擁有者設定的費率，以及相關 Web 服務的標準使用費 (例如，在 Amazon EC2 中執行 m5.small 執行個體類型的小時費率) 收費。可能也需支付額外稅金。已支付 AMI 的擁有者可確認特定執行個體是否使用該已支付 AMI 啟動。

⚠ Important

Amazon DevPay 不再接受新的賣家或產品。AWS Marketplace 現在是通過銷售軟件和服務的單一統一電子商務平台 AWS。如需如何部署及銷售軟體的詳細資訊 AWS Marketplace，請參閱在 [AWS Marketplace 中銷售](#)。AWS Marketplace 支援由 Amazon EBS 支援的 AMI。

目錄

- [販售您的 AMI](#)
- [尋找已支付 AMI](#)
- [購買已支付 AMI](#)
- [取得您執行個體的產品代碼](#)
- [使用已支付支援](#)
- [已支付及支援的 AMI 的帳單](#)
- [管理您的 AWS Marketplace 訂閱](#)

販售您的 AMI

您可以使用出售 AMI AWS Marketplace。AWS Marketplace 提供有組織的購物體驗。此外，AWS Marketplace AWS 還支援 Amazon EBS 支援的 AMI、預留執行個體和競價型執行個體等功能。

有關如何在「」上銷售 AMI 的詳細資訊 AWS Marketplace，請參閱在 [AWS Marketplace 中銷售](#)。

尋找已支付 AMI

有數種方法能讓您尋找可購買的 AMI。例如，您可以使用 [AWS Marketplace](#)、Amazon EC2 主控台或命令列。或者，開發人員也可自行讓您了解已支付 AMI。

使用主控台尋找已支付 AMI

使用主控台尋找已支付 AMI

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 AMIs (AMI)。
3. 中，選擇 Public images (公有映像) 作為第一個篩選條件。
4. 在搜尋列中，依序選擇 Owner alias (擁有者別名)、= 和 aws-marketplace。

5. 如果您知道產品代碼，請選擇 Product code (產品代碼) 和 =，然後輸入產品代碼。

使用查找付費 AMI AWS Marketplace

要使用查找付費 AMI AWS Marketplace

1. 打開 [AWS Marketplace](#).
2. 在搜尋欄位中輸入作業系統的名稱，然後選擇搜尋按鈕 (放大鏡)。
3. 若要進一步限制搜尋結果的範圍，請使用其中一種分類或篩選條件。
4. 每一種產品都帶有其產品類型的標籤：AMI 或 Software as a Service

使用查找付費 AMI AWS CLI

您可以使用下列 [describe-images](#) 命令 (AWS CLI) 尋找已支付 AMI。

```
aws ec2 describe-images
  --owners aws-marketplace
```

此命令會傳回描述每一個 AMI 的數種詳細資訊，包含已支付 AMI 的產品代碼。describe-images 的輸出包含產品代碼的項目，如下所示：

```
"ProductCodes": [
  {
    "ProductCodeId": "product_code",
    "ProductCodeType": "marketplace"
  }
],
```

如果您知道產品代碼，您可以依產品代碼篩選結果。此範例會傳回最新 AMI 與指定的產品代碼。

```
aws ec2 describe-images
  --owners aws-marketplace \
  --filters "Name=product-code,Values=product_code" \
  --query "sort_by(Images, &CreationDate)[-1].[ImageId]"
```

使用視窗工具尋找付費 AMI PowerShell

您可以使用以下 [Get-EC2Image](#) 命令找到付費 AMI。

```
PS C:\> Get-EC2Image -Owner aws-marketplace
```

已支付 AMI 的輸出包含產品代碼。

ProductCodeId	ProductCodeType
-----	-----
<i>product_code</i>	marketplace

如果您知道產品代碼，您可以依產品代碼篩選結果。此範例會傳回最新 AMI 與指定的產品代碼。

```
PS C:\> (Get-EC2Image -Owner aws-marketplace -Filter @{"Name"="product-  
code";"Value"="product_code"} | sort CreationDate -Descending | Select-Object -First  
1).ImageId
```

購買已支付 AMI

您必須先為 (購買) 已支付 AMI 進行註冊，才能使用 AMI 啟動執行個體。

通常已支付 AMI 的賣方會提供您 AMI 的相關資訊，包含價格和可購買該 AMI 的連結。當您按一下連結時，系統會先要求您登入 AWS，然後您可以購買 AMI。

使用主控台購買已支付 AMI

您可以使用 Amazon EC2 Launch Wizard 購買已支付 AMI。如需詳細資訊，請參閱 [啟動 AWS Marketplace 執行個體](#)。

使用訂閱產品 AWS Marketplace

若要使用 AWS Marketplace，您必須擁有一個 AWS 帳戶。若要從產 AWS Marketplace 品啟動執行個體，您必須註冊才能使用 Amazon EC2 服務，而且您必須訂閱要從中啟動執行個體的產品。有兩種方法可訂閱 AWS Marketplace 中的產品：

- AWS Marketplace 網站：您可以使用一鍵式部署功能快速啟動預先配置的軟件。
- Amazon EC2 啟動精靈：您可以搜尋 AMI，然後直接從精靈啟動執行個體。如需詳細資訊，請參閱 [啟動 AWS Marketplace 執行個體](#)。

取得您執行個體的產品代碼

您可以使用執行個體中繼資料擷取執行個體的 AWS Marketplace 產品代碼。若執行個體具有產品代碼，Amazon EC2 便會傳回它。如需擷取中繼資料的詳細資訊，請參閱 [擷取執行個體中繼資料](#)。

若要擷取產品代碼，請使用執行個體作業系統的指令。

Linux

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/product-codes
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/product-codes
```

Windows

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/product-codes
```

使用已支付支援

Amazon EC2 同時也可讓開發人員提供軟體 (或衍生 AMI) 的支援。開發人員能建立可讓您註冊使用的支援產品。在註冊支援產品時，開發人員會給予您產品代碼，您必須將該產品代碼與您擁有的 AMI 建立關聯。這可讓開發人員確認您的執行個體符合取得支援的資格。它也能確認在您執行產品執行個體的時候，您確實有根據開發人員指定的產品規定支付費用。

Important

您無法搭配預留執行個體使用支援產品。您一律必須支付支援產品賣方指定的價格。

若要將產品代碼與您的 AMI 建立關聯，請使用下列其中一個命令。其中，ami_id 為 AMI 的 ID，product_code 則是產品代碼：

- [modify-image-attribute](#) (AWS CLI)

```
aws ec2 modify-image-attribute --image-id ami_id --product-codes "product_code"
```

- [Edit-EC2ImageAttribute](#) (AWS Tools for Windows PowerShell)

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami_id -ProductCode product_code
```

在您設定產品代碼屬性後，您便無法變更或移除它。

已支付及支援的 AMI 的帳單

在每個月結束的時候，您會收到一封電子郵件，其中包含在該月份中使用任何已支付或支援的 AMI，而向您信用卡收取的金額。此帳單與您的一般 Amazon EC2 帳單各自獨立。如需詳細資訊，請參閱 AWS Marketplace 買方指南中的 [產品付款](#)。

管理您的 AWS Marketplace 訂閱

在 AWS Marketplace 網站上，您可以檢查訂閱詳細資料、檢視廠商的使用說明、管理您的訂閱等。

檢查您的訂閱詳細資訊

1. 登入 [AWS Marketplace](#)。
2. 選擇 Your Marketplace Account (您的 Marketplace 帳戶)。
3. 選擇 Manage your software subscriptions (管理您的軟體訂閱)。
4. 將會列出所有您目前的訂閱。選擇 Usage Instructions (使用說明) 檢視使用產品的特定說明，例如：用來連線到您執行中執行個體的使用者名稱。

若要取消訂 AWS Marketplace 閱

1. 請確認您已終止任何從該訂閱執行的執行個體。
 - a. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
 - b. 在導覽窗格中，選擇執行個體。
 - c. 選取執行個體，然後選取 Instance state (執行個體狀態)、Terminate instance (終止執行個體)。
 - d. 出現確認提示時，請選擇 Terminate (終止)。

2. 登入 [AWS Marketplace](#)，然後依序選擇 Your Marketplace Account (您的 Marketplace 帳戶) 及 Manage your software subscriptions (管理您的軟體訂閱)。
3. 選擇 Cancel subscription (取消訂閱)。系統會提示您確認取消。

Note

在您取消您的訂閱之後，您便無法從該 AMI 啟動任何執行個體。若要再次使用該 AMI，您需要在 AWS Marketplace 網站上或透過 Amazon EC2 主控台內的啟動精靈重新訂閱該 AMI。

AMI 生命週期

您可建立自己的 AMI、加以複製、備份並進行維護，直至準備好取代或取消註冊。

目錄

- [建立 AMI](#)
- [修改 AMI](#)
- [複製 AMI](#)
- [使用 S3 存放和還原 AMI](#)
- [棄用 AMI](#)
- [停用 AMI](#)
- [封存 AMI 快照](#)
- [取消註冊 \(刪除\) AMI](#)
- [自動化 EBS 後端 AMI 生命週期](#)

建立 AMI

您可以建立由 Amazon EBS 磁碟區支援的 Linux 或視窗 AMI。您也可以建立由執行個體儲存磁碟區支援的 Linux AMI (Windows AMI 不支援根裝置的執行個體存放區)。您也可以使用視窗系統來建立視窗 AMI。

主題

- [創建一個 Amazon EBS-backed AMI](#)
- [建立執行個體存放區後端 Linux AMI](#)

- [使用視窗系統建立 AMI](#)

創建一個 Amazon EBS-backed AMI

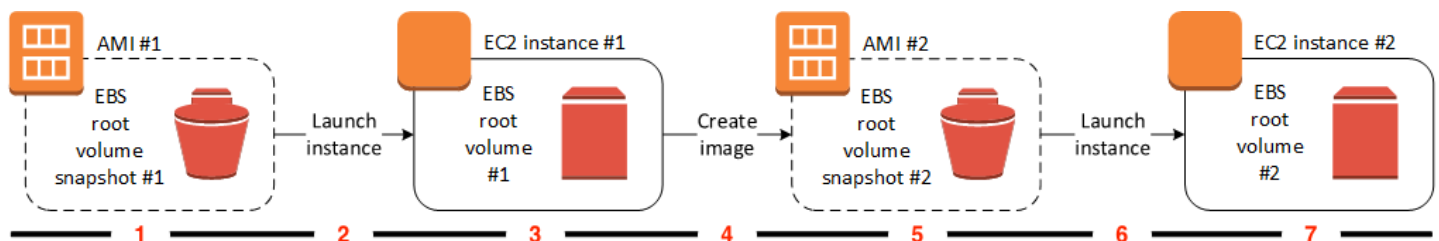
若要建立 Amazon EBS-backed AMI，請從您從現有的 Amazon EBS-backed AMI 啟動的執行個體開始。這可以是您從中取得的 AMI AWS Marketplace、您使用 [AWS Server Migration Service](#) 或 [VM 匯入/匯出](#) 建立的 AMI，或是您可以存取的任何其他 AMI。在您將執行個體自訂到符合您的需求後，建立及註冊新的 AMI。您可以搭配這些自訂項目，使用這個新的 AMI 啟動新的執行個體。

下列說明的程序同時適用於加密 Amazon Elastic Block Store (Amazon EBS) 磁碟區 (包含根磁碟區) 及未加密磁碟區後端 Amazon EC2 執行個體。

執行個體存放區後端 AMI 有不同的 AMIs 建立程序。如需 Amazon EBS 後端及執行個體後端執行個體之間差異的資訊，以及如何判斷執行個體的根裝置類型，請參閱 [根設備儲存](#)。如需建立 instance store-backed AMI 的相關資訊，請參閱 [建立執行個體存放區後端 Linux AMI](#)

建立 Amazon EBS 後端的 AMIs 概觀

下列圖表總結了從正在執行的 EC2 執行個體建立 Amazon EBS 後端 AMI 的過程：從現有 AMI 開始，啟動執行個體，自訂執行個體，根據其建立新的 AMI，以及最終啟動您的新 AMI 執行個體。該圖表中的數字與隨後描述中的數字相符。



1 – AMI #1：從現有 AMI 開始

查找與要建立的 AMI 相似的現有 AMI。這可以是您從中取得的 AMI AWS Marketplace、您使用 [AWS Server Migration Service](#) 或 [VM 匯入/匯出](#) 建立的 AMI，或是您可以存取的任何其他 AMI。您將根據需要自訂此 AMI。

在圖表中，EBS 根磁碟區快照 #1 指示 AMI 是 Amazon EBS 後端 AMI，並且有關根磁碟區的資訊存放在此快照中。

2 – 從現有 AMI 啟動執行個體

設定 AMI 的方法是從您希望作為新 AMI 基礎的 AMI 啟動執行個體，然後自訂執行個體 (由圖表中的 3 指示)。然後，您將建立新的 AMI，其中包含自訂執行個體 (由圖表中的 4 指示)。

3 – EC2 執行個體 #1：自訂執行個體

連線到您的執行個體並根據需要自訂。您的新 AMI 將包含這些自訂執行個體。

您可以在執行個體上執行下列任一個動作以自訂：

- 安裝軟體和應用程式
- 複製資料
- 刪除暫存檔案並重組您的硬碟，來減少啟動時間。
- 連接額外的 EBS 磁碟區

4 – 建立映像

從執行個體建立 AMI 時，Amazon EC2 會在建立 AMI 前關閉執行個體，確認執行個體上的一切在建立程序中都維持在停止且一致的狀態。若您相信您的執行個體處於一致且可用來建立 AMI 的狀態，您可以告知 Amazon EC2 不要關閉及重新開機執行個體。有些檔案系統 (例如 XFS) 可能會凍結及取消凍結活動，使其足夠安全，可用來建立映像，而無須重新開機執行個體。

在 AMI 建立程序期間，Amazon EC2 會建立您執行個體根磁碟區及任何連接到您執行個體之其他 EBS 磁碟區的快照。您必須為快照支付費用，直到您[取消註冊 AMI](#) 及刪除快照為止。若任何連接到執行個體的磁碟區經過加密，則新的 AMI 只能在支援 Amazon EBS 加密的執行個體上成功啟動。

AMI 建立程序可能需要數分鐘的時間完成 (有些可能會花上 24 小時)，具體取決於磁碟區的大小。您可能會發現在建立您的 AMI 前先建立您磁碟區的快照會更有效率。透過此方式，建立 AMI 時便只需要建立小型的增量快照，且程序將能更快完成 (建立快照的總時間維持不變)。

5 – AMI #2：新 AMI

在程序完成後，您便會擁有新的 AMI 及從執行個體根磁碟區建立的快照 (快照 #2)。若除了根設備磁碟區之外，您還為您的執行個體新增執行個體存放磁碟區或 EBS 磁碟區，則新 AMI 的區塊型設備映射便會包含這些磁碟區的資訊。

Amazon EC2 會自動為您註冊 AMI。

6 – 從新的 AMI 啟動執行個體

您可以使用新 AMI 啟動執行個體。

7 – EC2 執行個體 #2：新執行個體

當您使用新的 AMI 啟動執行個體時，Amazon EC2 會使用快照，建立此執行個體的根磁碟區的新 EBS 磁碟區。如果您在自訂執行個體時為新增執行個體存放磁碟區或 EBS 磁碟區，則新 AMI 的區

塊型設備映射便會包含這些磁碟區的資訊，並且您從新的 AMI 啟動之執行個體的區塊型設備映射也會自動包含這些磁碟區的資訊。新執行個體的區塊型設備映射中指定的執行個體儲存體磁碟區都是全新的，不會包含您用來建立 AMI 的執行個體之執行個體儲存體磁碟區中的任何資料。EBS 磁碟區上的資料會持續。如需詳細資訊，請參閱 [區塊型儲存設備對應](#)。

當您從 EBS 後端 AMI 建立新的執行個體時，建議您在將其用於生產環境前初始化其根磁碟區和任何額外的 EBS 儲存體。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的初始化 Amazon EBS 磁碟區](#)。

從執行個體建立 AMI

您可以使用 AWS Management Console 或命令列建立 AMI。

Console

若要建立 AMI

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取要從中建立 AMI 的執行個體，然後依序選擇 Actions (動作)、Image and templates (映像和範本)，Create image (建立映像)。

Tip

若此選項已停用，表示您的執行個體並非 Amazon EBS 後端執行個體。

4. 在 Create image (建立映像) 頁面上，指定下列資訊：
 - a. 對於 Image name (映像名稱)，輸入映像的唯一名稱，最多 127 個字元。
 - b. 對於 Image description (映像描述)，輸入映像的選擇性描述，最多 255 個字元。
 - c. 對於 No reboot (不重新啟動)，請將 Enable (啟用) 核取方塊 (預設值) 保留為未核取，或選取它。
 - 如果清除了不重新開機的啟用核取方塊，則 Amazon EC2 建立新 AMI 時，會重新啟動執行個體，以便在資料處於靜態時可以拍攝連接磁碟區的快照，以確保狀態一致。
 - 如果選取了不重新開機的啟用核取方塊，則 Amazon EC2 建立新 AMI 時，不會關閉及重新啟動執行個體。

⚠ Warning

若您選擇啟用 No reboot (不重新開機)，我們無法保證建立映像的檔案系統完整性。

- d. Instance volumes (執行個體磁碟區) – 您可以修改根磁碟區，以及新增額外的 Amazon EBS 及執行個體儲存體磁碟區，如下所示：
 - i. 根磁碟區在第一列中定義。
 - 若要變更根磁碟區的大小，請在 Size (大小) 中輸入所需的值。
 - 若您選取 Delete on termination (終止時刪除)，則當您終止從此 AMI 建立的執行個體時，便會刪除 EBS 磁碟區。若您清除 Delete on termination (終止時刪除)，則當您終止執行個體時，便不會刪除 EBS 磁碟區。如需詳細資訊，請參閱 [在執行個體終止時保留資料](#)。
 - ii. 若要新增 EBS 磁碟區，請選擇 Add volume (新增磁碟區) (這會新增一個新的資料列)。針對儲存體類型，選擇 EBS，然後填入資料列中的欄位。當您從您新的 AMI 啟動執行個體時，額外的磁碟區會自動連接到執行個體。空白磁碟區必須進行格式化及掛載。以快照為基礎的磁碟區必須掛載。
 - iii. 若要新增執行個體存放區磁碟區，請參閱 [將執行個體存放區的磁碟區新增至 AMI](#)。當您從您新的 AMI 啟動執行個體時，額外的磁碟區會自動初始化並掛載。這些磁碟區不會包含做為您 AMI 基礎之執行中執行個體的執行個體存放區磁碟區中的任何資料。
 - e. Tags (標籤) - 當您建立 AMI 時，可以使用相同的標籤來標記 AMI 和快照，或者可以使用不同的標籤來進行標記。
 - 若要使用相同的 標籤來標記 AMI 和快照，請選擇 Tag image and snapshots together (共同標記映像和快照)。相同的標籤即套用至 AMI 和每個建立的快照。
 - 若要使用不同的 標籤來標記 AMI 和快照，請選擇 Tag image and snapshots separately (個別標記映像和快照)。不同的標籤即套用至 AMI 和建立的快照。但是所有快照都會取得相同的標籤；您無法使用不同的標籤標記每個快照。

若要新增標籤，請選擇 Add tag (新增標籤)，然後輸入標籤的鍵和值。針對每個標籤，重複執行上述動作。
 - f. 當您準備好建立 AMI 時，請選擇 Create image (建立映像)。
5. 若要在建立您 AMI 的過程中檢視狀態：

- a. 在導覽窗格中，選擇 AMIs (AMI)。
- b. 將篩選條件設為 Owned by me (由我擁有)，然後在清單中尋找您的 AMI。

一開始，狀態為 pending，但在幾分鐘後應該會變更為 available。

6. (選用) 若要檢視為新 AMI 建立的快照：

- a. 記下您在先前步驟中找到的 AMI ID。
- b. 在導覽窗格中，選擇快照。
- c. 將篩選條件設定為 Owned by me (由我擁有)，然後在 Description (說明) 欄中尋找含有新 AMI ID 的快照。

當您從此 AMI 啟動執行個體時，Amazon EC2 會使用此快照建立其根設備磁碟區。

AWS CLI

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱[存取 Amazon EC2](#)。

- [create-image](#) (AWS CLI)
- [New-EC2Image](#) (AWS Tools for Windows PowerShell)

從快照建立 Linux AMI

如果您有執行個體根裝置磁碟區的快照，您可以使用 AWS Management Console 或指令列從此快照建立 Linux AMI。此功能目前不適用於 Windows 執行個體。

Console

若要從快照建立 AMI

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇快照。
3. 選取要從中建立 AMI 的快照，然後選取 Actions (動作)、Create image from snapshot (從快照建立映像)。
4. 在 [從快照建立映像] 頁面上，指定下列資訊：
 - a. 對於 Image name (映像名稱)，輸入映像的描述性名稱。
 - b. 對於 Description (描述)，輸入映像的簡短描述。

- c. 對於 Architecture (架構), 選擇映像架構。如果是 32 位元, 請選擇 i386, 64 位元選擇 64 位元, 64 位元 ARM 則選擇 64 位元, 或 64 位元 macOS 本則選擇 64 位元。
- d. 對於 Root device name (根設備名稱), 輸入要用於根設備磁碟區的設備名稱。如需詳細資訊, 請參閱 [Amazon EC2 實例上的設備名稱](#)。
- e. 對於 Virtualization type (虛擬化類型), 選擇從這個 AMI 啟動之執行個體要使用的虛擬化類型。如需詳細資訊, 請參閱 [AMI 虛擬化類型](#)。
- f. (僅適用於半虛擬化) 對於 Kernel ID (核心 ID), 選取映像的作業系統核心。如果您是使用執行個體的根設備磁碟區快照, 請選取與原始執行個體相同的核心 ID。如果不確定, 請使用預設核心。
- g. (僅適用於半虛擬化) 對於 RAM disk ID (RAM 磁碟 ID), 選取映像的 RAM 磁碟。如果選取特定核心, 則可能需要選取具有可支援之驅動程式的特定 RAM 磁碟。
- h. 針對開機模式, 請選擇映像檔的開機模式, 或選擇 [使用預設值], 如此當執行個體使用此 AMI 啟動時, 它會以執行個體類型支援的開機模式啟動。如需詳細資訊, 請參閱 [設定 AMI 的開機模式](#)。
- i. (選擇性) 在封鎖裝置對應下, 自訂根磁碟區並新增其他資料磁碟區。

針對每個磁碟區, 您可以指定大小、類型、效能特性、終止時的刪除行為, 以及加密狀態。若為根磁碟區, 大小不可小於快照的大小。磁碟區類型的預設選項是「一般用途 SSD gp3」。
- j. (選用) 在標籤下, 您可以新增一或多個標籤至新 AMI。若要新增標籤, 請選擇 Add tag (新增標籤), 然後輸入標籤的鍵和值。針對每個標籤, 重複執行上述動作。
- k. 當您準備好建立 AMI 時, 請選擇 Create image (建立映像)。

AWS CLI

使用命令列從快照建立 AMI

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊, 請參閱[存取 Amazon EC2](#)。

- [寄存器影像 \(CLI\)](#)AWS
- [Register-EC2Image](#) (AWS Tools for Windows PowerShell)

從您建立的 AMI 啟動執行個體

您可以從已從執行個體或快照建立的 AMI 啟動執行個體。

如何從 AMI 啟動執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的映像下，選擇 AMIs。
3. 將篩選條件設為 Owned by me (由我擁有)，然後選取您的 AMI。
4. 選擇從 AMI 啟動執行個體。
5. 接受啟動執行個體精靈中的預設值或指定自訂值。如需詳細資訊，請參閱 [使用新的啟動執行個體精靈啟動執行個體](#)。

建立執行個體存放區後端 Linux AMI

您啟動執行個體時指定的 AMI，會判定根設備磁碟區的類型。

若要建立執行個體後端 Linux AMI，請從您已從現有執行個體後端 Linux AMI 啟動的執行個體開始。在您將執行個體自訂到符合您的需求後，建立磁碟區並註冊新的 AMI。您可以搭配這些自訂項目，使用這個新的 AMI 啟動新的執行個體。

您無法建立執行個體存放區支援的 Windows AMI，因為 Windows AMI 不支援根裝置的執行個體存放區。

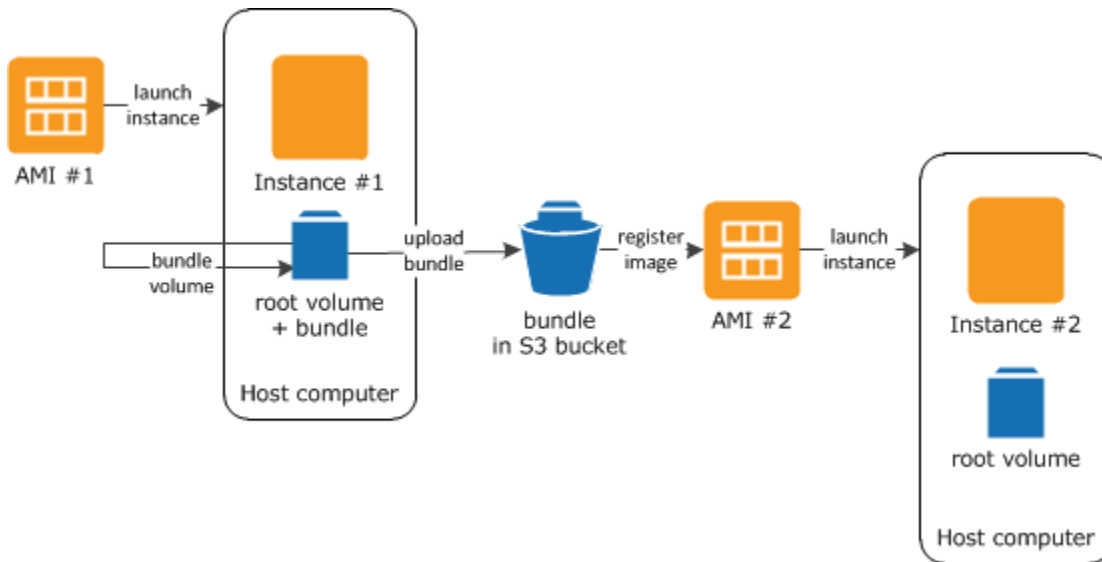
Important

只有下列執行個體類型支援執行個體儲存區磁碟區做為根裝置：
C1、C3、D2、I2、M2、M2、M3、R3 和 X1。

AMI 建立程序與 Amazon EBS 後端 AMI 不同。如需 Amazon EBS 後端及執行個體後端執行個體之間差異的詳細資訊，以及如何判斷執行個體的根裝置類型，請參閱 [根設備儲存](#)。如果您需要建立 Amazon EBS-backed AMI，請參閱 [創建一個 Amazon EBS-backed AMI](#)。

執行個體後端 AMI 建立程序概觀

以下圖表顯示從執行個體後端執行個體建立 AMI 之程序的摘要。



首先，請先從與您要建立的 AMI 相似的 AMI 啟動執行個體。您可以連線到您的執行個體並自訂。設定好所要的執行個體後，您可以組合執行個體。完成綁定程序需要幾分鐘。程序完成後，您將擁有 bundle，其中包含映像工作資訊清單檔案 (image.manifest.xml) 和含有根磁碟區範本的檔案 (image.part.xx)。接著您上傳 bundle 至 Amazon S3 儲存貯體，然後登錄您的 AMI。

Note

若要為您的執行個體存放區後端 Linux AMI 將物件上傳到 S3 儲存貯體，必須為儲存貯體啟用 ACL。否則，Amazon EC2 將無法在要上傳的物件上設定 ACL。如果您的目的地儲存貯體將儲存貯體擁有者強制設定用於 S3 物件擁有權，這將無法運作，因為 ACL 已停用。如需詳細資訊，請參閱 [使用 S3 物件擁有權控制上傳物件的擁有權](#)。

當您使用新的 AMI 啟動執行個體時，我們會使用上傳至 Amazon S3 的 bundle 建立執行個體的根磁碟區。Amazon S3 中 bundle 所用的儲存空間會使您的帳戶產生費用，直到刪除為止。如需詳細資訊，請參閱 [取消註冊 \(刪除\) AMI](#)。

若除了根設備磁碟區之外，您還為您的執行個體新增執行個體存放磁碟區，則新 AMI 的區塊型設備映射便會包含這些磁碟區的資訊，並且您從新的 AMI 啟動之執行個體的區塊型設備映射也會自動包含這些磁碟區的資訊。如需詳細資訊，請參閱 [區塊型儲存設備對應](#)。

必要條件

在您可以開始建立 AMI 之前，務必完成以下工作：

- 安裝 AMI 工具。如需詳細資訊，請參閱 [設定 AMI 工具](#)。

- 安裝 AWS CLI。如需詳細資訊，請參閱[開始設定 AWS Command Line Interface](#)。
- 請確定您具有套件組合的 S3 儲存貯體，並確定您的儲存貯體已啟用 ACL。如需有關設定 ACL 的詳細資訊，請參閱[設定 ACL](#)。
 - 若要使用建立 S3 儲存貯體 AWS Management Console，請在 <https://console.aws.amazon.com/s3/> 開啟 Amazon S3 主控台，然後選擇建立儲存貯體。
 - 若要使用建立 S3 儲存貯體 AWS CLI，您可以使用 `mb` 命令。如果安裝的 AMI 工具版本為 1.5.18 或更新版本，也可以使用 `ec2-upload-bundle` 命令建立 S3 儲存貯體。如需詳細資訊，請參閱[ec2-upload-bundle](#)。
- 確保您有您的 AWS 帳戶 ID。如需詳細資訊，請參閱[AWS 帳戶管理參考指南中的檢視 AWS 帳戶 識別碼](#)。
- 請確定您擁有可使用 AWS CLI 的憑證。如需詳細資訊，請參閱《AWS Account Management 參考指南》中的[AWS 帳戶最佳做法](#)。
- 請確定您擁有 X.509 憑證和對應的私有金鑰。
 - 如果需要建立 X.509 憑證，請參閱[管理簽署憑證](#)。X.509 憑證和私有金鑰用於加密和解密 AMI。
 - [中國 (北京)] 使用 `$EC2_AMITOOL_HOME/etc/ec2/amitools/cert-ec2-cn-north-1.pem` 憑證。
 - [AWS GovCloud (美國西部)] 使用 `$EC2_AMITOOL_HOME/etc/ec2/amitools/cert-ec2-gov.pem` 憑證。
- 連線到您的執行個體並加以自訂。例如，您可以安裝軟體和應用程式、複製資料、刪除暫存檔，以及修改 Linux 組態。

任務

- [設定 AMI 工具](#)
- [從執行個體後端 Amazon Linux 執行個體建立 AMI](#)
- [從執行個體後端 Ubuntu 執行個體建立 AMI](#)
- [將執行個體後端 AMI 轉換為 Amazon EBS 後端的 AMI](#)

設定 AMI 工具

您可使用 AMI 工具來建立及管理執行個體存放區後端 Linux AMIs。若要使用工具，您必須在 Linux 執行個體上安裝工具。AMI 工具在不支援 RPM 的 Linux 發行版本上會同時以 RPM 和 .zip 檔案的方式提供。

使用 RPM 設定 AMI 工具

1. 使用 Linux 發行版本的套件管理工具 (例如 yum) 來安裝 Ruby。例如：

```
[ec2-user ~]$ sudo yum install -y ruby
```

2. 使用 wget 或 curl 等工具下載 RPM 檔案。例如：

```
[ec2-user ~]$ wget https://s3.amazonaws.com/ec2-downloads/ec2-ami-tools.noarch.rpm
```

3. 使用下列命令驗證 RPM 檔案的簽章：

```
[ec2-user ~]$ rpm -K ec2-ami-tools.noarch.rpm
```

上述命令應該會顯示檔案的 SHA1 和 MD5 雜湊為 OK。如果命令顯示雜湊為 NOT OK，請用下列命令檢視檔案的 Header SHA1 和 MD5 雜湊：

```
[ec2-user ~]$ rpm -Kv ec2-ami-tools.noarch.rpm
```

接著比較檔案的 Header SHA1 和 MD5 雜湊與下列已驗證的 AMI 工具雜湊，以確認檔案的真偽：

- Header SHA1 : a1f662d6f25f69871104e6a62187fa4df508f880
- MD5 : 9faff05258064e2f7909b66142de6782

如果檔案的 Header SHA1 和 MD5 雜湊符合已驗證的 AMI 工具雜湊，請繼續下一步。

4. 使用以下命令安裝 RPM：

```
[ec2-user ~]$ sudo yum install ec2-ami-tools.noarch.rpm
```

5. 使用 [ec2-ami-tools-version](#) 命令確認 AMI 工具的安裝。

```
[ec2-user ~]$ ec2-ami-tools-version
```

Note

如果您收到載入錯誤，例如「無法載入此類檔案-ec2/amitools/版本 (LoadError)」，請完成下一步，將 AMI 工具安裝的位置新增至您的路徑。RUBYLIB

6. (選用) 如果您在上一步驟中發生錯誤，請將 AMI 工具的安裝位置加入到 RUBYLIB 路徑。
 - a. 執行下列命令來判斷要加入的路徑。

```
[ec2-user ~]$ rpm -qil ec2-ami-tools | grep ec2/amitools/version
/usr/lib/ruby/site_ruby/ec2/amitools/version.rb
/usr/lib64/ruby/site_ruby/ec2/amitools/version.rb
```

在上例中，先前載入錯誤所缺少的檔案位在 `/usr/lib/ruby/site_ruby` 和 `/usr/lib64/ruby/site_ruby`。

- b. 在先前的步驟中將位置加入到 RUBYLIB 路徑。

```
[ec2-user ~]$ export RUBYLIB=$RUBYLIB:/usr/lib/ruby/site_ruby:/usr/lib64/ruby/site_ruby
```

- c. 使用 [ec2-ami-tools-version](#) 命令確認 AMI 工具的安裝。

```
[ec2-user ~]$ ec2-ami-tools-version
```

使用 .zip 檔案設定 AMI 工具

1. 使用 Linux 發行版本的套件管理工具 (例如 apt-get) 來安裝 Ruby 及解壓縮。例如：

```
[ec2-user ~]$ sudo apt-get update -y && sudo apt-get install -y ruby unzip
```

2. 使用 wget 或 curl 等工具下載 .zip 檔案。例如：

```
[ec2-user ~]$ wget https://s3.amazonaws.com/ec2-downloads/ec2-ami-tools.zip
```

3. 將檔案解壓縮到適當的安裝目錄，例如 `/usr/local/ec2`。

```
[ec2-user ~]$ sudo mkdir -p /usr/local/ec2
$ sudo unzip ec2-ami-tools.zip -d /usr/local/ec2
```

請注意，.zip 檔案包含資料夾 `ec2-ami-tools-x.x.x`，其中 `x.x.x` 是工具的版本號碼 (例如，`ec2-ami-tools-1.5.7`)。

4. 將 `EC2_AMITOOL_HOME` 環境變數設定為工具的安裝目錄。例如：

```
[ec2-user ~]$ export EC2_AMIT00L_HOME=/usr/local/ec2/ec2-ami-tools-x.x.x
```

- 將工具加入到 PATH 環境變數。例如：

```
[ec2-user ~]$ export PATH=$EC2_AMIT00L_HOME/bin:$PATH
```

- 使用 [ec2-ami-tools-version](#) 命令確認 AMI 工具的安裝。

```
[ec2-user ~]$ ec2-ami-tools-version
```

管理簽署憑證

AMI 工具中的部分命令需要簽署憑證 (亦稱為 X.509 憑證)。您必須建立憑證，然後將其上傳至 AWS。例如，您可用 OpenSSL 等第三方工具來建立憑證。

建立簽署憑證

- 安裝及設定 OpenSSL。
- 使用 `openssl genrsa` 命令建立私有金鑰，並將輸出儲存至 `.pem` 檔案。建議新建一個 2048 或 4096 位元 RSA 金鑰。

```
openssl genrsa 2048 > private-key.pem
```

- 使用 `openssl req` 命令產生憑證。

```
openssl req -new -x509 -nodes -sha256 -days 365 -key private-key.pem -outform PEM -out certificate.pem
```

若要將憑證上傳至 AWS，請使用 [上傳簽署憑證指令](#)。

```
aws iam upload-signing-certificate --user-name user-name --certificate-body  
file://path/to/certificate.pem
```

若要列出使用者的憑證，請用 [list-signing-certificates](#) 命令：

```
aws iam list-signing-certificates --user-name user-name
```

若要停用或重新啟用簽署憑證，請用 [update-signing-certificate](#) 命令。下列命令可停用憑證：

```
aws iam update-signing-certificate --certificate-id OFHPLP4ZULTHYPMSYEX704BEXAMPLE --status Inactive --user-name user-name
```

若要刪除憑證，請用 [delete-signing-certificate](#) 命令：

```
aws iam delete-signing-certificate --user-name user-name --certificate-id OFHPLP4ZULTHYPMSYEX704BEXAMPLE
```

從執行個體存放區後端執行個體建立 AMI

以下程序可從執行個體後端執行個體建立執行個體後端 AMI。開始之前，請務必先詳閱所列各項[先決條件](#)。

主題

- [從執行個體後端 Amazon Linux 執行個體建立 AMI](#)
- [從執行個體後端 Ubuntu 執行個體建立 AMI](#)

從執行個體後端 Amazon Linux 執行個體建立 AMI

本節說明如何從 Amazon Linux 執行個體建立 AMI。以下程序不一定適用於執行其他 Linux 發行版本的執行個體。如需 Ubuntu 專屬程序，請參閱[從執行個體後端 Ubuntu 執行個體建立 AMI](#)。

準備使用 AMI 工具 (僅適用於 HVM 執行個體)

1. AMI 工具需要 GRUB 舊版才能正常開機。請使用下列命令安裝 GRUB：

```
[ec2-user ~]$ sudo yum install -y grub
```

2. 使用下列命令安裝分割區管理套件：

```
[ec2-user ~]$ sudo yum install -y gdisk kpartx parted
```

從執行個體後端 Amazon Linux 執行個體建立 AMI

此程序假設您已滿足 [必要條件](#) 中的先決條件。

在下列命令中，將每個#####替換為您自己的資訊。

1. 將您的登入資料上傳到執行個體。這些登入資料是用來確定只有您與 Amazon EC2 可存取您的 AMI。
 - a. 在執行個體上為登入資料建立如下的暫時目錄：

```
[ec2-user ~]$ mkdir /tmp/cert
```

如此您就能將登入資料從建立的映像中排除。

- b. 使用 [scp](#) 等安全複製工具從電腦上將 X.509 憑證和對應的私有金鑰複製到執行個體上的 /tmp/cert 目錄內。以下 `-i my-private-key.pem` 命令中的 `scp` 選項為用來透過 SSH 連接執行個體的私有金鑰，而非 X.509 私有金鑰。例如：

```
you@your_computer:~ $ scp -i my-private-key.pem /  
path/to/pk-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem /  
path/to/cert-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem ec2-  
user@ec2-203-0-113-25.compute-1.amazonaws.com:/tmp/cert/  
pk-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem 100% 717 0.7KB/s 00:00  
cert-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem 100% 685 0.7KB/s 00:00
```

或者，由於這些檔案為純文字檔案，因此您可用文字編輯器開啟憑證和金鑰，然後將內容複製到 /tmp/cert 中的新檔案內。

2. 從執行個體內執行 [ec2-bundle-vol](#) 命令，準備要上傳至 Amazon S3 的 bundle。務必指定 `-e` 選項，以排除憑證存放位置的目錄。根據預設，bundle 程序不包含可能帶有敏感資訊的檔案。這些檔案包含 `*.sw`、`*.swo`、`*.swp`、`*.pem`、`*.priv`、`*id_rsa*`、`*id_dsa*`、`*.gpg`、`*.jks`、`*/.ssh/authorized_keys` 和 `*/.bash_history`。若要包含所有這些檔案，請使用 `--no-filter` 選項。若要包含這些檔案中的一部分，請使用 `--include` 選項。

Important

根據預設，AMI 綁定程序會在代表根磁碟區的 /tmp 目錄內建立經過壓縮加密的檔案集合。如果 /tmp 內沒有足夠的可用磁碟空間可存放 bundle，您需要用 `-d /path/to/bundle/storage` 選項指定其他的 bundle 存放位置。某些執行個體已掛載於 /mnt 或您可以使用 /media/ephemeral0 的暫時儲存，或者您也可以建立、連接和掛接新的 Amazon EBS) 磁碟區來存放服務包。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的建立 Amazon EBS 磁碟區](#)。

- a. 您必須以 root 執行 `ec2-bundle-vol` 命令。針對多數命令，您皆能用 `sudo` 來取得更高級的許可，但在此情況下您應執行 `sudo -E su` 以保持環境變數。


```
[ec2-user ~]$ sudo -E su
```

請注意，bash 提示現在將您識別為根使用者，且貨幣符號已換成雜湊標籤，表示您正在根 shell 下：

```
[root ec2-user]#
```

- b. 若要建立 AMI bundle，請照下列所示執行 `ec2-bundle-vol` 命令：

```
[root ec2-user]# ec2-bundle-vol -k /tmp/cert/pk-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem -c /tmp/cert/cert-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem -u 123456789012 -r x86_64 -e /tmp/cert --partition gpt
```

 Note

對於中國 (北京) 和 AWS GovCloud (美國西部) 區域，請使用 `--ec2cert` 參數並根據 [必要條件](#) 指定憑證。

建立映像可能需要幾分鐘。完成此命令後，您的 `/tmp` (或非預設的) 目錄將包含 bundle (image.manifest.xml，加上多個 image.part.xx 檔案)。

- c. 退出根 shell。

```
[root ec2-user]# exit
```

3. (選用) 若要新增更多執行個體儲存體磁碟區，請在 AMI 的 image.manifest.xml 檔案中編輯區塊型裝置映射。如需詳細資訊，請參閱 [區塊型儲存設備對應](#)。


- a. 為您的 image.manifest.xml 檔案建立備份。

```
[ec2-user ~]$ sudo cp /tmp/image.manifest.xml /tmp/image.manifest.xml.bak
```

- b. 重新格式化 image.manifest.xml 檔案，以方便讀取和編輯。

```
[ec2-user ~]$ sudo xmllint --format /tmp/image.manifest.xml.bak > /tmp/  
image.manifest.xml
```

- c. 使用文字編輯器編輯 `image.manifest.xml` 中的區塊型裝置映射。下列範例顯示 `ephemeral1` 執行個體存放磁碟區的新項目。

 Note

如需排除檔案清單，請參閱 [ec2-bundle-vol](#)。

```
<block_device_mapping>  
  <mapping>  
    <virtual>ami</virtual>  
    <device>sda</device>  
  </mapping>  
  <mapping>  
    <virtual>ephemeral0</virtual>  
    <device>sdb</device>  
  </mapping>  
  <mapping>  
    <virtual>ephemeral1</virtual>  
    <device>sdc</device>  
  </mapping>  
  <mapping>  
    <virtual>root</virtual>  
    <device>/dev/sda1</device>  
  </mapping>  
</block_device_mapping>
```

- d. 儲存 `image.manifest.xml` 檔案並結束文字編輯器。
4. 若要上傳 bundle 至 Amazon S3，請照下列所示執行 [ec2-upload-bundle](#) 命令。

```
[ec2-user ~]$ ec2-upload-bundle -b my-s3-bucket/bundle_folder/bundle_name -m /tmp/  
image.manifest.xml -a your_access_key_id -s your_secret_access_key
```

⚠ Important

若要在 US East (N. Virginia) 以外的區域登錄 AMI，您必須使用 `--region` 選項指定兩個目標區域，和目標區域中已存在的儲存貯體路徑，或可在目標區域中建立的唯一儲存貯體路徑。

5. (選用) bundle 上傳至 Amazon S3 之後，您可使用下列 `/tmp` 命令將 bundle 從執行個體的 `rm` 目錄中移除：

```
[ec2-user ~]$ sudo rm /tmp/image.manifest.xml /tmp/image.part.* /tmp/image
```

⚠ Important

如果用 `-d /path/to/bundle/storage` 選項指定 [Step 2](#) 中的路徑，請使用該路徑，而非 `/tmp`。

6. 若要登錄 AMI，請照下列所示執行 [register-image](#) 命令。

```
[ec2-user ~]$ aws ec2 register-image --image-location my-s3-bucket/bundle_folder/bundle_name/image.manifest.xml --name AMI_name --virtualization-type hvm
```

⚠ Important

如果您先前已指定 [ec2-upload-bundle](#) 命令的區域，請在此命令中再次指定該區域。

從執行個體後端 Ubuntu 執行個體建立 AMI

本節說明如何從 Ubuntu Linux 執行個體建立 AMI，並將執行個體儲存區磁碟區做為根磁碟區。以下程序不一定適用於執行其他 Linux 發行版本的執行個體。如需特定的程序 Amazon Linux，請參閱[從執行個體後端 Amazon Linux 執行個體建立 AMI](#)。

準備使用 AMI 工具 (僅適用於 HVM 執行個體)

AMI 工具需要 GRUB 舊版才能正常開機。但 Ubuntu 已設定為使用 GRUB 2。您必須檢查執行個體是否使用 GRUB 舊版，假如不是，您需要進行安裝及設定。

HVM 執行個體也要求必須安裝分割工具，AMI 工具才能正常運作。

1. 您的執行個體上必須安裝 GRUB 舊版 (0.9x 版或更低)。請檢查是否有 GRUB 舊版，並請視需要安裝。
 - a. 檢查 GRUB 安裝的版本。

```
ubuntu:~$ grub-install --version
grub-install (GRUB) 1.99-21ubuntu3.10
```

在此範例中，GRUB 版本為 0.9x 以上，因此您必須安裝 GRUB 舊版。繼續執行「[Step 1.b](#)」。如果您已有 GRUB 舊版，則可以跳至[Step 2](#)步驟。

- b. 使用以下命令安裝 grub 套件。

```
ubuntu:~$ sudo apt-get install -y grub
```

2. 使用發行版本的套件管理工具安裝下列分割區管理套件。

- gdisk (某些發行版本可能稱此套件為 gptfdisk)
- kpartx
- parted

使用下列命令。

```
ubuntu:~$ sudo apt-get install -y gdisk kpartx parted
```

3. 檢查執行個體的核心參數。

```
ubuntu:~$ cat /proc/cmdline
BOOT_IMAGE=/boot/vmlinuz-3.2.0-54-virtual root=UUID=4f392932-ed93-4f8f-
aee7-72bc5bb6ca9d ro console=ttyS0 xen_emul_unplug=unnecessary
```

請注意核心及根設備參數後方的選項：ro、console=ttyS0 和 xen_emul_unplug=unnecessary。您的選項可能不盡相同。

4. 檢查 /boot/grub/menu.lst 中的核心項目。

```
ubuntu:~$ grep ^kernel /boot/grub/menu.lst
kernel /boot/vmlinuz-3.2.0-54-virtual root=LABEL=cloudimg-rootfs ro console=hvc0
```

```
kernel /boot/vmlinuz-3.2.0-54-virtual root=LABEL=cloudimg-rootfs ro single
kernel /boot/memtest86+.bin
```

請注意，`console` 參數指向 `hvc0`，而非 `ttyS0`，且 `xen_emul_unplug=unnecessary` 參數遺失。再次強調，您的選項可能不盡相同。

5. 用您偏好的文字編輯器 (例如 `/boot/grub/menu.lst` 或 `vim`) 編輯 `nano` 檔案，以變更主控台並將先前找出的參數加入到開機項目。

```
title          Ubuntu 12.04.3 LTS, kernel 3.2.0-54-virtual
root           (hd0)
kernel         /boot/vmlinuz-3.2.0-54-virtual root=LABEL=cloudimg-rootfs
               ro console=ttyS0 xen_emul_unplug=unnecessary
initrd         /boot/initrd.img-3.2.0-54-virtual

title          Ubuntu 12.04.3 LTS, kernel 3.2.0-54-virtual (recovery mode)
root           (hd0)
kernel         /boot/vmlinuz-3.2.0-54-virtual root=LABEL=cloudimg-rootfs ro
               single console=ttyS0 xen_emul_unplug=unnecessary
initrd         /boot/initrd.img-3.2.0-54-virtual

title          Ubuntu 12.04.3 LTS, memtest86+
root           (hd0)
kernel         /boot/memtest86+.bin
```

6. 請確認您的核心項目現在包含正確參數。

```
ubuntu:~$ grep ^kernel /boot/grub/menu.lst
kernel /boot/vmlinuz-3.2.0-54-virtual root=LABEL=cloudimg-rootfs ro console=ttyS0
       xen_emul_unplug=unnecessary
kernel /boot/vmlinuz-3.2.0-54-virtual root=LABEL=cloudimg-rootfs ro single
       console=ttyS0 xen_emul_unplug=unnecessary
kernel /boot/memtest86+.bin
```

7. (僅適用於 Ubuntu 14.04 和更新版本) 自 Ubuntu 14.04 起，執行個體後端 Ubuntu AMI 使用 GPT 分割表格和掛載於 `/boot/efi` 的獨立 EFI 分割區。 `ec2-bundle-vol` 命令不會組合此開機分割區，因此您需要將 EFI 分割區的 `/etc/fstab` 項目變更為註解，如下範例所示。

```
LABEL=cloudimg-rootfs /          ext4  defaults          0 0
#LABEL=UEFI           /boot/efi                       vfat  defaults          0 0
/dev/xvdb             /mnt    auto  defaults,nobootwait,comment=cloudconfig 0 2
```

從執行個體後端 Ubuntu 執行個體建立 AMI

此程序假設您已滿足 [必要條件](#) 中的先決條件。

在下列命令中，將每個#####替換為您自己的資訊。

1. 將您的登入資料上傳到執行個體。這些登入資料是用來確定只有您與 Amazon EC2 可存取您的 AMI。
 - a. 在執行個體上為登入資料建立如下的暫時目錄：

```
ubuntu:~$ mkdir /tmp/cert
```

如此您就能將登入資料從建立的映像中排除。

- b. 使用 [scp](#) 等安全複製工具從電腦上將 X.509 憑證和私有金鑰複製到執行個體上的 /tmp/cert 目錄內。以下 `-i my-private-key.pem` 命令中的 scp 選項為用來透過 SSH 連接執行個體的私有金鑰，而非 X.509 私有金鑰。例如：

```
you@your_computer:~ $ scp -i my-private-key.pem /  
path/to/pk-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem /  
path/to/cert-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem ec2-  
user@ec2-203-0-113-25.compute-1.amazonaws.com:/tmp/cert/  
pk-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem 100% 717 0.7KB/s 00:00  
cert-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem 100% 685 0.7KB/s 00:00
```

或者，由於這些檔案為純文字檔案，因此您可用文字編輯器開啟憑證和金鑰，然後將內容複製到 /tmp/cert 中的新檔案內。

2. 從執行個體執行 [ec2-bundle-vol](#) 命令，準備要上傳至 Amazon S3 的 bundle。務必指定 `-e` 選項，以排除憑證存放位置的目錄。根據預設，bundle 程序不包含可能帶有敏感資訊的檔案。這些檔案包含 *.sw、*.swo、*.swp、*.pem、*.priv、*id_rsa*、*id_dsa*、*.gpg、*.jks、*/.ssh/authorized_keys 和 */.bash_history。若要包含所有這些檔案，請使用 `--no-filter` 選項。若要包含這些檔案中的一部分，請使用 `--include` 選項。

Important

根據預設，AMI 綁定程序會在代表根磁碟區的 /tmp 目錄內建立經過壓縮加密的檔案集合。如果 /tmp 內沒有足夠的可用磁碟空間可存放 bundle，您需要用 `-d /path/to/bundle/storage` 選項指定其他的 bundle 存放位置。某些執行個體已掛載於 /mnt 或

您可以使用 `/media/ephemeral0` 的暫時儲存，或者您也可以建立、連接和掛接新的 Amazon EBS) 磁碟區來存放服務包。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的建立 Amazon EBS 磁碟區](#)。

- a. 您必須以根使用者身分執行 `ec2-bundle-vol` 命令。針對多數命令，您皆能用 `sudo` 來取得更高級的許可，但在此情況下您應執行 `sudo -E su` 以保持環境變數。

```
ubuntu:~$ sudo -E su
```

請注意，`bash` 提示現在將您識別為根使用者，且貨幣符號已換成雜湊標籤，表示您正在根 shell 下：

```
root@ubuntu:~#
```

- b. 若要建立 AMI bundle，請照下列所示執行 [ec2-bundle-vol](#) 命令。

```
root@ubuntu:~# ec2-bundle-vol -k /tmp/cert/pk-HKZYKTAIG2ECMYIBH3HXV4ZBEXAMPLE.pem -c /tmp/cert/cert-HKZYKTAIG2ECMYIBH3HXV4ZBEXAMPLE.pem -u your_aws_account_id -r x86_64 -e /tmp/cert --partition gpt
```

Important

針對 Ubuntu 14.04 和更新版本的 HVM 執行個體，請新增 `--partition mbr` 旗標，以正確組合開機指令；否則新建立的 AMI 將無法開機。

建立映像可能需要幾分鐘。完成此命令後，您的 `tmp` 目錄將包含 `bundle` (`image.manifest.xml`，加上多個 `image.part.xx` 檔案)。

- c. 退出根 shell。

```
root@ubuntu:~# exit
```

3. (選用) 若要新增更多執行個體儲存體磁碟區，請在 AMI 的 `image.manifest.xml` 檔案中編輯區塊型裝置映射。如需詳細資訊，請參閱 [區塊型儲存設備對應](#)。

- a. 為您的 `image.manifest.xml` 檔案建立備份。

```
ubuntu:~$ sudo cp /tmp/image.manifest.xml /tmp/image.manifest.xml.bak
```

- b. 重新格式化 `image.manifest.xml` 檔案，以方便讀取和編輯。

```
ubuntu:~$ sudo xmllint --format /tmp/image.manifest.xml.bak > /tmp/  
image.manifest.xml
```

- c. 使用文字編輯器編輯 `image.manifest.xml` 中的區塊型裝置映射。下列範例顯示 `ephemeral1` 執行個體存放磁碟區的新項目。

```
<block_device_mapping>  
  <mapping>  
    <virtual>ami</virtual>  
    <device>sda</device>  
  </mapping>  
  <mapping>  
    <virtual>ephemeral0</virtual>  
    <device>sdb</device>  
  </mapping>  
  <mapping>  
    <virtual>ephemeral1</virtual>  
    <device>sdc</device>  
  </mapping>  
  <mapping>  
    <virtual>root</virtual>  
    <device>/dev/sda1</device>  
  </mapping>  
</block_device_mapping>
```

- d. 儲存 `image.manifest.xml` 檔案並結束文字編輯器。
4. 若要上傳 bundle 至 Amazon S3，請照下列所示執行 [ec2-upload-bundle](#) 命令。

```
ubuntu:~$ ec2-upload-bundle -b my-s3-bucket/bundle_folder/bundle_name -m /tmp/  
image.manifest.xml -a your_access_key_id -s your_secret_access_key
```

⚠ Important

如果您想要在US East (N. Virginia) 以外的區域登錄 AMI，您必須使用 `--region` 選項指定兩個目標區域和目標區域中已存在的儲存貯體路徑，或可在目標區域中建立的唯一儲存貯體路徑。

5. (選用) bundle 上傳至 Amazon S3 之後，您可使用下列 `/tmp` 命令將 bundle 從執行個體的 `rm` 目錄中移除：

```
ubuntu:~$ sudo rm /tmp/image.manifest.xml /tmp/image.part.* /tmp/image
```

⚠ Important

如果用 `-d /path/to/bundle/storage` 選項指定 [Step 2](#) 中的路徑，請使用底下的相同路徑，而非 `/tmp`。

6. 若要登錄 AMI，請照下列所示執行 [register-image](#) AWS CLI 命令。

```
ubuntu:~$ aws ec2 register-image --image-location my-s3-bucket/bundle_folder/bundle_name/image.manifest.xml --name AMI_name --virtualization-type hvm
```

⚠ Important

如果您先前已指定 [ec2-upload-bundle](#) 命令的區域，請在此命令中再次指定該區域。

7. (Ubuntu 14.04 和更新版本) 取消 `/etc/fstab` 中 EFI 項目的註解；否則，已在執行中的執行個體將無法重新啟動。

將執行個體後端 AMI 轉換為 Amazon EBS 後端的 AMI

您可將擁有的執行個體後端 Linux AMI 轉換為 Amazon EBS 後端 Linux AMI。

⚠ Important

您無法轉換您不擁有的 AMI。

將執行個體後端 AMI 轉換為 Amazon EBS 後端 AMI

1. 從 Amazon EBS 後端 AMI 啟動 Amazon Linux 執行個體。如需詳細資訊，請參閱 [使用新的啟動執行個體精靈啟動執行個體](#)。Amazon Linux 執行個體已預先安裝 AWS CLI 和 AMI 工具。
2. 請將您用來組合執行個體後端 AMI 的 X.509 私有金鑰上傳至您的執行個體。此金鑰是用來確定只有您與 Amazon EC2 可存取您的 AMI。

- a. 在執行個體上為 X.509 私有金鑰建立如下的暫時目錄：

```
[ec2-user ~]$ mkdir /tmp/cert
```

- b. 使用 [scp](#) 等安全複製工具從電腦上將 X.509 私有金鑰複製到執行個體上的 /tmp/cert 目錄內。以下命令中的 *my-private-key* 參數為用來透過 SSH 連接執行個體的私有金鑰。例如：

```
you@your_computer:~ $ scp -i my-private-key.pem /  
path/to/pk-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem ec2-  
user@ec2-203-0-113-25.compute-1.amazonaws.com:/tmp/cert/  
pk-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem 100% 717 0.7KB/s 00:00
```

3. 設定環境變數以使用 AWS CLI。如需詳細資訊，請參閱 [建立金鑰對](#)。

- a. (建議使用) 設定 AWS 存取金鑰、私密金鑰和工作階段權杖的環境變數。

```
[ec2-user ~]$ export AWS_ACCESS_KEY_ID=your_access_key_id  
[ec2-user ~]$ export AWS_SECRET_ACCESS_KEY=your_secret_access_key  
[ec2-user ~]$ export AWS_SESSION_TOKEN=your_session_token
```

- b. 為您的 AWS 訪問密鑰和密鑰設置環境變量。

```
[ec2-user ~]$ export AWS_ACCESS_KEY_ID=your_access_key_id  
[ec2-user ~]$ export AWS_SECRET_ACCESS_KEY=your_secret_access_key
```

4. 為新的 AMI 準備 Amazon Elastic Block Store (Amazon EBS) 磁碟區。

- a. 使用 [create-volume](#) 命令在與您執行個體相同的可用區域中建立空的 EBS 磁碟區。記下命令輸出中的磁碟區 ID。

⚠ Important

此 EBS 磁碟區的大小必須等於或大於原本的執行個體存放區根磁碟區。

```
[ec2-user ~]$ aws ec2 create-volume --size 10 --region us-west-2 --  
availability-zone us-west-2b
```

- b. 使用 [attach-volume](#) 命令將磁碟區連接到您的 Amazon EBS 後端執行個體。

```
[ec2-user ~]$ aws ec2 attach-volume --volume-id volume_id --instance-  
id instance_id --device /dev/sdb --region us-west-2
```

5. 建立您的 bundle 資料夾。

```
[ec2-user ~]$ mkdir /tmp/bundle
```

6. 使用 `/tmp/bundle` 命令將執行個體後端 AMI 的 bundle 下載至 [ec2-download-bundle](#)。

```
[ec2-user ~]$ ec2-download-bundle -b my-s3-bucket/bundle_folder/bundle_name -m  
image.manifest.xml -a $AWS_ACCESS_KEY_ID -s $AWS_SECRET_ACCESS_KEY --privatekey /  
path/to/pk-HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem -d /tmp/bundle
```

7. 使用 [ec2-unbundle](#) 命令從 bundle 重新建構映像檔。

- a. 將目錄變更為 bundle 資料夾。

```
[ec2-user ~]$ cd /tmp/bundle/
```

- b. 執行 [ec2-unbundle](#) 命令。

```
[ec2-user bundle]$ ec2-unbundle -m image.manifest.xml --privatekey /path/to/pk-  
HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem
```

8. 從解開映像將檔案複製到新的 EBS 磁碟區。

```
[ec2-user bundle]$ sudo dd if=/tmp/bundle/image of=/dev/sdb bs=1M
```

9. 探測磁碟區中是否有任何解開的新分割區。


```
[ec2-user bundle]$ sudo partprobe /dev/sdb1
```

- 列出區塊型儲存設備，以找出要掛載的設備名稱。

```
[ec2-user bundle]$ lsblk
NAME          MAJ:MIN RM  SIZE RO  TYPE MOUNTPOINT
/dev/sda      202:0    0   8G  0  disk
##/dev/sda1  202:1    0   8G  0  part /
/dev/sdb      202:80   0  10G  0  disk
##/dev/sdb1  202:81   0  10G  0  part
```

在此範例中，要掛載的分割區為 `/dev/sdb1`，但您的設備名稱可能不同。如果磁碟區尚未分割，要掛載的設備會類似 `/dev/sdb` (沒有設備分割區尾碼)。

- 為新的 EBS 磁碟區建立掛載點，然後掛載磁碟區。

```
[ec2-user bundle]$ sudo mkdir /mnt/ebs
[ec2-user bundle]$ sudo mount /dev/sdb1 /mnt/ebs
```

- 用您偏好的文字編輯器 (例如 `/etc/fstab` 或 `vim`) 在 EBS 磁碟區上開啟 `nano` 檔案，並移除執行個體 (暫時性) 存放磁碟區的任何項目。EBS 磁碟區掛載於 `/mnt/ebs`，因此 `fstab` 檔案位於 `/mnt/ebs/etc/fstab`。

```
[ec2-user bundle]$ sudo nano /mnt/ebs/etc/fstab
#
LABEL=/      /          ext4      defaults,noatime 1 1
tmpfs        /dev/shm   tmpfs     defaults         0 0
devpts       /dev/pts   devpts    gid=5,mode=620  0 0
sysfs        /sys       sysfs     defaults         0 0
proc         /proc      proc       defaults         0 0
/dev/sdb     /media/ephemeral0 auto      defaults,comment=cloudconfig 0
2
```

在此範例中，應該移除最後一行。

- 卸載磁碟區，並從執行個體分離。

```
[ec2-user bundle]$ sudo umount /mnt/ebs
[ec2-user bundle]$ aws ec2 detach-volume --volume-id volume_id --region us-west-2
```

- 從新的 EBS 磁碟區建立 AMI，如下所示。

- a. 建立新 EBS 磁碟區的快照。

```
[ec2-user bundle]$ aws ec2 create-snapshot --region us-west-2 --description  
"your_snapshot_description" --volume-id volume_id
```

- b. 檢查快照是否完整。

```
[ec2-user bundle]$ aws ec2 describe-snapshots --region us-west-2 --snapshot-  
id snapshot_id
```

- c. 使用 describe-images 命令找出原始 AMI 的處理器架構、虛擬化類型與核心映像 (aki)。在此步驟中，您需要擁有原始執行個體後端 AMI 的 AMI ID。

```
[ec2-user bundle]$ aws ec2 describe-images --region us-west-2 --image-id ami-id  
--output text  
IMAGES x86_64 amazon/amzn-ami-pv-2013.09.2.x86_64-s3 ami-8ef297be amazon  
available public machine aki-fc8f11cc instance-store paravirtual xen
```

在此範例中，其架構為 x86_64，核心映像 ID 為 aki-fc8f11cc。請將這些數值使用在下列步驟。如果上述命令的輸出也列出 ari ID，請將其記下。

- d. 使用新 EBS 磁碟區的快照 ID 和先前步驟中的數值登錄您的新 AMI。如果先前的命令輸出列出 ari ID，請用 --ramdisk-id *ari_id* 將其加入到下列命令。

```
[ec2-user bundle]$ aws ec2 register-image --region us-west-2 --  
name your_new_ami_name --block-device-mappings DeviceName=device-  
name,Ebs={SnapshotId=snapshot_id} --virtualization-type paravirtual --  
architecture x86_64 --kernel-id aki-fc8f11cc --root-device-name device-name
```

15. (選用) 確認可以從新 AMI 啟動執行個體後，您便能刪除為此程序建立的 EBS 磁碟區。

```
aws ec2 delete-volume --volume-id volume_id
```

AMI 工具參考

您可使用 AMI 工具命令來建立及管理執行個體存放區後端 Linux AMI。欲設定這些工具，請參閱[設定 AMI 工具](#)。

如需有關存取金鑰的詳細資訊，請參閱《AWS Account Management 參考指南》中的[AWS 帳戶的最佳實務](#)。

命令

- [ec2-ami-tools-version](#)
- [ec2-bundle-image](#)
- [ec2-bundle-vol](#)
- [ec2-delete-bundle](#)
- [ec2-download-bundle](#)
- [ec2-migrate-manifest](#)
- [ec2-unbundle](#)
- [ec2-upload-bundle](#)
- [AMI 工具常用選項](#)

ec2-ami-tools-version

描述

描述 AMI 工具的版本。

語法

ec2-ami-tools-version

輸出

版本資訊。

範例

此範例命令會顯示您正使用之 AMI 工具的版本資訊。

```
[ec2-user ~]$ ec2-ami-tools-version  
1.5.2 20071010
```

ec2-bundle-image

描述

從迴路檔案中建立之作業系統映像建立執行個體存放區後端 Linux AMI。

語法

```
ec2-bundle-image -c path -k path -u account -i path [-d path] [--ec2cert path] [-r architecture] [--productcodes code1,code2,...] [-B mapping] [-p prefix]
```

選項

-c, --cert *path*

使用者以 PEM 編碼的 RSA 公有金鑰憑證檔案。

必要：是

-k, --privatekey *path*

指向以 PEM 編碼的 RSA 金鑰檔案路徑。您必須指定此金鑰來解開此 bundle，因此請將其置於安全位置。請注意，密鑰不必註冊到您的 AWS 帳戶。

必要：是

-u, --user *account*

使用者的 AWS 帳號 ID，不含破折號。

必要：是

-i, --image *path*

欲綁定的映像路徑。

必要：是

-d, --destination *path*

欲建立 bundle 的目錄。

預設：/tmp

必要：否

--ec2cert *path*

用於加密映像資訊清單的 Amazon EC2 X.509 公有金鑰憑證路徑。

us-gov-west-1 和 cn-north-1 區域使用非預設的公有金鑰憑證，而該憑證的路徑必須以此選項加以指定。憑證的路徑視 AMI 工具的安裝方法而異。以 Amazon Linux 而言，此憑證位於 /opt/aws/amitools/ec2/etc/ec2/amitools/。若您從 RPM 或 ZIP 檔案將 AMI 工具安裝進 [設定 AMI 工具](#)，則憑證位於 \$EC2_AMITOOL_HOME/etc/ec2/amitools/。

必要：僅適用於 us-gov-west-1 和 cn-north-1 區域。

-r, --arch architecture

映像架構。若您未於命令列提供架構，則會在綁定開始時提示您輸入架構。

有效值：i386 | x86_64

必要：否

--productcodes code1,code2,...

欲在註冊時連接至映像的產品代碼，以逗號分隔。

必要：否

-B, --block-device-mapping mapping

定義區塊型儲存設備向此 AMI 的執行個體公開的方式 (若其執行個體類型支援所指定的設備)。

指定以逗號分隔的索引鍵/值組清單，其中索引鍵為虛擬名稱，而值為相對應的設備名稱。虛擬名稱包括下列：

- ami - 執行個體所看見的根檔案系統設備
- root - 核心所看見的根檔案系統設備
- swap - 執行個體所看見的置換裝置
- ephemeralN - 第 N 個執行個體存放磁碟區

必要：否

-p, --prefix prefix

綁定的 AMI 檔案之檔案名稱字首。

預設：映像檔案名稱。舉例來說，若映像路徑為 /var/spool/my-image/version-2/debian.img，則預設字首為 debian.img。

必要：否

`--kernel kernel_id`

已廢除。使用 [register-image](#) 來設定核心。

必要：否

`--ramdisk ramdisk_id`

已廢除。使用 [register-image](#) 來視需要設定 RAM 磁碟。

必要：否

輸出

描述綁定程序階段與狀態的狀態訊息。

範例

此範例會從迴路檔案中建立的作業系統映像，建立綁定 AMI。

```
[ec2-user ~]$ ec2-bundle-image -k pk-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem -c cert-
HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem -u 111122223333 -i image.img -d bundled/ -r x86_64
Please specify a value for arch [i386]:
Bundling image file...
Splitting bundled/image.gz.crypt...
Created image.part.00
Created image.part.01
Created image.part.02
Created image.part.03
Created image.part.04
Created image.part.05
Created image.part.06
Created image.part.07
Created image.part.08
Created image.part.09
Created image.part.10
Created image.part.11
Created image.part.12
Created image.part.13
Created image.part.14
Generating digests for each part...
Digests generated.
Creating bundle manifest...
```

```
ec2-bundle-image complete.
```

ec2-bundle-vol

描述

針對執行個體的根設備磁碟區，透過壓縮、加密並簽署其複本，建立執行個體存放區後端 Linux AMI。

Amazon EC2 會嘗試從執行個體繼承產品代碼、核心設定、RAM 磁碟設定和區塊型設備映射。

根據預設，bundle 程序不包含可能帶有敏感資訊的檔案。這些檔案包含

.sw、.swo、*.swp、*.pem、*.priv、*id_rsa*、*id_dsa* *.gpg、*.jks、*/.ssh/authorized_keys 和 */.bash_history。若要包含所有這些檔案，請使用 --no-filter 選項。若要包含這些檔案中的一部分，請使用 --include 選項。

如需詳細資訊，請參閱 [建立執行個體存放區後端 Linux AMI](#)。

語法

```
ec2-bundle-vol -c path -k path -u account [-d path] [--ec2cert path] [-r architecture] [--productcodes code1,code2,...] [-B mapping] [--all] [-e directory1,directory2,...] [-i file1,file2,...] [--no-filter] [-p prefix] [-s size] [--[no-]inherit] [-v volume] [-P type] [-S script] [--fstab path] [--generate-fstab] [--grub-config path]
```

選項

-c, --cert *path*

使用者以 PEM 編碼的 RSA 公有金鑰憑證檔案。

必要：是

-k, --privatekey *path*

使用者以 PEM 編碼的 RSA 金鑰檔案路徑。

必要：是

-u, --user *account*

使用者的 AWS 帳號 ID，不含破折號。

必要：是

`-d, --destination destination`

欲建立 bundle 的目錄。

預設：/tmp

必要：否

`--ec2cert path`

用於加密映像資訊清單的 Amazon EC2 X.509 公有金鑰憑證路徑。

us-gov-west-1 和 cn-north-1 區域使用非預設的公有金鑰憑證，而該憑證的路徑必須以此選項加以指定。憑證的路徑視 AMI 工具的安裝方法而異。以 Amazon Linux 而言，此憑證位於 /opt/aws/amitools/ec2/etc/ec2/amitools/。若您從 RPM 或 ZIP 檔案將 AMI 工具安裝進 [設定 AMI 工具](#)，則憑證位於 \$EC2_AMITOOL_HOME/etc/ec2/amitools/。

必要：僅適用於 us-gov-west-1 和 cn-north-1 區域。

`-r, --arch architecture`

映像架構。若您未於命令列提供架構，則會在綁定開始時提示您提供架構。

有效值：i386 | x86_64

必要：否

`--productcodes code1,code2,...`

欲在註冊時連接至映像的產品代碼，以逗號分隔。

必要：否

`-B, --block-device-mapping mapping`

定義區塊型儲存設備向此 AMI 的執行個體公開的方式 (若其執行個體類型支援所指定的設備)。

指定以逗號分隔的索引鍵/值組清單，其中索引鍵為虛擬名稱，而值為相對應的設備名稱。虛擬名稱包括下列：

- ami - 執行個體所看見的根檔案系統設備
- root - 核心所看見的根檔案系統設備
- swap - 執行個體所看見的置換裝置
- ephemeralN - 第 N 個執行個體存放磁碟區

必要：否

`-a, --all`

綁定所有目錄，包括遠端掛載之檔案系統上的目錄。

必要：否

`-e, --exclude directory1,directory2,...`

欲從 bundle 操作排除的絕對目錄路徑和檔案清單。此參數會覆寫 `--all` 選項。指定排除項目時，隨參數列出的目錄和子目錄將不會與磁碟區一同綁定。

必要：否

`-i, --include file1,file2,...`

欲納入 bundle 操作的檔案清單。因為指定的檔案可能具備敏感資訊，若未指定則會從 AMI 排除。

必要：否

`--no-filter`

若已指定，我們不會因其可能具備敏感資訊而將其從 AMI 排除。

必要：否

`-p, --prefix prefix`

綁定的 AMI 檔案之檔案名稱字首。

預設：image

必要：否

`-s, --size size`

欲建立的映像檔案大小，以 MB (1024 * 1024 位元) 表示。大小上限為 10240 MB。

預設：10240

必要：否

`--[no-]inherit`

指示映像是否繼承執行個體的中繼資料 (預設為繼承)。若啟用 `--inherit` 但無法存取執行個體中繼資料，則綁定會失敗。

必要：否

`-v, --volume volume`

欲從中建立 bundle 的掛載磁碟區絕對路徑。

預設：根目錄 (/)

必要：否

`-P, --partition type`

指示磁碟映像是否使用分割區表格。若未指定分割區表格類型，預設為用於磁碟區父區塊型儲存設備的類型 (若適用)，否則預設為 gpt。

有效值：mbr | gpt | none

必要：否

`-S, --script script`

欲在綁定前執行的自訂指令碼。此指令碼必須具備單一引數，亦即磁碟區的掛載點。

必要：否

`--fstab path`

欲綁定至映像的 fstab 路徑。若未指定，Amazon EC2 會綁定 /etc/fstab。

必要：否

`--generate-fstab`

使用 Amazon EC2 提供的 fstab 綁定磁碟區。

必要：否

`--grub-config`

欲綁定至映像的替代 grub 組態檔案路徑。根據預設，ec2-bundle-vol 預期所複製映像上應存在 /boot/grub/menu.lst 或 /boot/grub/grub.conf。此選項可讓您指定替代 grub 組態檔案路徑，並將其複製以覆蓋預設值 (如有)。

必要：否

`--kernel kernel_id`

已廢除。使用 [register-image](#) 來設定核心。

必要：否

--ramdiskramdisk_id

已廢除。使用 [register-image](#) 來視需要設定 RAM 磁碟。

必要：否

輸出

描述綁定階段與狀態的狀態訊息。

範例

此範例會壓縮、加密並簽署本機機器根檔案系統的快照，藉此建立綁定的 AMI。

```
[ec2-user ~]$ ec2-bundle-vol -d /mnt -k pk-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem -c
cert-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem -u 111122223333 -r x86_64
Copying / into the image file /mnt/image...
Excluding:
  sys
  dev/shm
  proc
  dev/pts
  proc/sys/fs/binfmt_misc
  dev
  media
  mnt
  proc
  sys
  tmp/image
  mnt/img-mnt
1+0 records in
1+0 records out
mke2fs 1.38 (30-Jun-2005)
warning: 256 blocks unused.

Splitting /mnt/image.gz.crypt...
Created image.part.00
Created image.part.01
Created image.part.02
Created image.part.03
...
Created image.part.22
```

```
Created image.part.23
Generating digests for each part...
Digests generated.
Creating bundle manifest...
Bundle Volume complete.
```

ec2-delete-bundle

描述

從 Amazon S3 儲存空間刪除指定 bundle。刪除 bundle 之後，即無法從相對應 AMI 啟動執行個體。

語法

```
ec2-delete-bundle -b bucket -a access_key_id -s secret_access_key [-t token] [--url url] [--region region] [--sigv version] [-m path] [-p prefix] [--clear] [--retry] [-y]
```

選項

-b, --bucket *bucket*

內含綁定 AMI 的 Amazon S3 儲存貯體名稱，其後為選用「/」以分隔路徑字首

必要：是

-a, --access-key *access_key_id*

AWS 存取金鑰識別碼。

必要：是

-s, --secret-key *secret_access_key*

AWS 秘密存取金鑰。

必要：是

-t, --delegation-token *token*

要傳遞給 AWS 要求的委派權杖。如需詳細資訊，請參閱[使用臨時安全登入資料](#)。

必要：僅適用您使用暫時安全登入資料時。

預設：AWS_DELEGATION_TOKEN 環境變數的值 (如已設定)。

--region

於請求簽章內使用的區域。

預設：us-east-1

必要：若使用簽章第 4 版則為必要

--sigvversion

簽署請求時使用的簽章版本。

有效值：2 | 4

預設：4

必要：否

-m, --manifestpath

指向資訊清單檔案的路徑。

必要：您必須指定 **--prefix** 或 **--manifest**。

-p, --prefix prefix

綁定的 AMI 檔案名稱字首。請提供完整字首。舉例來說，若字首為 image.img，則使用 **-p image.img** 而非 **-p image**。

必要：您必須指定 **--prefix** 或 **--manifest**。

--clear

若 Amazon S3 儲存貯體在刪除指定 bundle 後為空白，則加以刪除。

必要：否

--retry

在所有 Amazon S3 錯誤後自動重試，每次操作至多五次。

必要：否

-y, --yes

自動假設所有提示的答案為 Yes (是)。

必要：否

輸出

Amazon EC2 顯示的狀態訊息，會指出刪除程序的階段與狀態。

範例

此範例會自 Amazon S3 刪除一個 bundle。

```
[ec2-user ~]$ ec2-delete-bundle -b DOC-EXAMPLE-BUCKET1 -a your_access_key_id -s your_secret_access_key
Deleting files:
DOC-EXAMPLE-BUCKET1/image.manifest.xml
DOC-EXAMPLE-BUCKET1/image.part.00
DOC-EXAMPLE-BUCKET1/image.part.01
DOC-EXAMPLE-BUCKET1/image.part.02
DOC-EXAMPLE-BUCKET1/image.part.03
DOC-EXAMPLE-BUCKET1/image.part.04
DOC-EXAMPLE-BUCKET1/image.part.05
DOC-EXAMPLE-BUCKET1/image.part.06
Continue? [y/n]
y
Deleted DOC-EXAMPLE-BUCKET1/image.manifest.xml
Deleted DOC-EXAMPLE-BUCKET1/image.part.00
Deleted DOC-EXAMPLE-BUCKET1/image.part.01
Deleted DOC-EXAMPLE-BUCKET1/image.part.02
Deleted DOC-EXAMPLE-BUCKET1/image.part.03
Deleted DOC-EXAMPLE-BUCKET1/image.part.04
Deleted DOC-EXAMPLE-BUCKET1/image.part.05
Deleted DOC-EXAMPLE-BUCKET1/image.part.06
ec2-delete-bundle complete.
```

ec2-download-bundle

描述

從 Amazon S3 儲存體下載指定的執行個體存放區後端 Linux AMIs。

語法

```
ec2-download-bundle -b bucket -a access_key_id -s secret_access_key -k path [--url url] [--region region] [--sigv version] [-m file] [-p prefix] [-d directory] [--retry]
```

選項

`-b, --bucket bucket`

bundle 所在的 Amazon S3 儲存貯體名稱，其後為選用「/」以分隔路徑字首。

必要：是

`-a, --access-key access_key_id`

AWS 存取金鑰識別碼。

必要：是

`-s, --secret-key secret_access_key`

AWS 秘密存取金鑰。

必要：是

`-k, --privatekey path`

用於解密資訊清單的私有金鑰。

必要：是

`--url url`

Amazon S3 服務的 URL。

預設：`https://s3.amazonaws.com/`

必要：否

`--region region`

於請求簽章內使用的區域。

預設：`us-east-1`

必要：若使用簽章第 4 版則為必要

`--sigv version`

簽署請求時使用的簽章版本。

有效值：`2 | 4`

預設：4

必要：否

`-m, --manifest file`

不含路徑的資訊清單檔案名稱。建議您指定資訊清單 (-m) 或字首 (-p)。

必要：否

`-p, --prefix prefix`

綁定的 AMI 檔案之檔案名稱字首。

預設：image

必要：否

`-d, --directory directory`

所下載 bundle 的儲存目錄。此目錄必須存在。

預設：目前使用中的目錄。

必要：否

`--retry`

在所有 Amazon S3 錯誤後自動重試，每次操作至多五次。

必要：否

輸出

會顯示指出下載程序各個階段的狀態訊息。

範例

此範例會建立 `bundled` 目錄 (使用 Linux `mkdir` 命令) 並從 `DOC-EXAMPLE-BUCKET1` Amazon S3 儲存貯體下載 bundle。

```
[ec2-user ~]$ mkdir bundled
[ec2-user ~]$ ec2-download-bundle -b DOC-EXAMPLE-BUCKET1/bundles/bundle_name
-m image.manifest.xml -a your_access_key_id -s your_secret_access_key -k pk-
HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem -d mybundle
```



```

Downloading manifest image.manifest.xml from DOC-EXAMPLE-BUCKET1 to mybundle/
image.manifest.xml ...
Downloading part image.part.00 from DOC-EXAMPLE-BUCKET1/bundles/bundle_name to
mybundle/image.part.00 ...
Downloaded image.part.00 from DOC-EXAMPLE-BUCKET1
Downloading part image.part.01 from DOC-EXAMPLE-BUCKET1/bundles/bundle_name to
mybundle/image.part.01 ...
Downloaded image.part.01 from DOC-EXAMPLE-BUCKET1
Downloading part image.part.02 from DOC-EXAMPLE-BUCKET1/bundles/bundle_name to
mybundle/image.part.02 ...
Downloaded image.part.02 from DOC-EXAMPLE-BUCKET1
Downloading part image.part.03 from DOC-EXAMPLE-BUCKET1/bundles/bundle_name to
mybundle/image.part.03 ...
Downloaded image.part.03 from DOC-EXAMPLE-BUCKET1
Downloading part image.part.04 from DOC-EXAMPLE-BUCKET1/bundles/bundle_name to
mybundle/image.part.04 ...
Downloaded image.part.04 from DOC-EXAMPLE-BUCKET1
Downloading part image.part.05 from DOC-EXAMPLE-BUCKET1/bundles/bundle_name to
mybundle/image.part.05 ...
Downloaded image.part.05 from DOC-EXAMPLE-BUCKET1
Downloading part image.part.06 from DOC-EXAMPLE-BUCKET1/bundles/bundle_name to
mybundle/image.part.06 ...
Downloaded image.part.06 from DOC-EXAMPLE-BUCKET1

```

ec2-migrate-manifest

描述

修改執行個體存放區後端 Linux AMI (如其憑證、核心和 RAM 磁碟) , 以支援不同區域。

語法

```
ec2-migrate-manifest -c path -k path -m path {(-a access_key_id -s secret_access_key --region region) | (--no-mapping)} [--ec2cert ec2_cert_path] [--kernel kernel-id] [--ramdisk ramdisk_id]
```

選項

-c, --cert path

使用者以 PEM 編碼的 RSA 公有金鑰憑證檔案。

必要：是

`-k, --privatekey path`

使用者以 PEM 編碼的 RSA 金鑰檔案路徑。

必要：是

`--manifest path`

指向資訊清單檔案的路徑。

必要：是

`-a, --access-key access_key_id`

AWS 存取金鑰識別碼。

必要：若使用自動映射則為必要。

`-s, --secret-key secret_access_key`

AWS 秘密存取金鑰。

必要：若使用自動映射則為必要。

`--region region`

欲在映射檔案內查詢的區域。

必要：若使用自動映射則為必要。

`--no-mapping`

停用核心和 RAM 磁碟的自動映射。

在遷移過程中，Amazon EC2 會將資訊清單檔案內的核心和 RAM 磁碟，取代為專為目的地區域設計的核心與 RAM 磁碟。除非提供 `--no-mapping` 參數，否則 `ec2-migrate-bundle` 可能會使用 `DescribeRegions` 和 `DescribeImages` 操作來執行自動映射。

必要：若您未提供用於自動映射的 `-a`、`-s` 和 `--region` 選項則為必要。

`--ec2cert path`

用於加密映像資訊清單的 Amazon EC2 X.509 公有金鑰憑證路徑。

`us-gov-west-1` 和 `cn-north-1` 區域使用非預設的公有金鑰憑證，而該憑證的路徑必須以此選項加以指定。憑證的路徑視 AMI 工具的安裝方法而異。以 Amazon Linux 而言，此憑證位於 `/opt/`

aws/amitools/ec2/etc/ec2/amitools/。若您從 ZIP 檔案將 AMI 工具安裝進 [設定 AMI 工具](#)，則憑證位於 \$EC2_AMITOOL_HOME/etc/ec2/amitools/。

必要：僅適用於 us-gov-west-1 和 cn-north-1 區域。

--kernel kernel_id

欲選取的核心 ID。

 Important

建議您使用 PV-GRUB，而不要使用核心與 RAM 磁碟。如需詳細資訊，請參閱 Amazon Linux 2 使用者指南中的使用者[提供的核心](#)。

必要：否

--ramdisk ramdisk_id

欲選取的 RAM 磁碟 ID。

 Important

建議您使用 PV-GRUB，而不要使用核心與 RAM 磁碟。如需詳細資訊，請參閱 Amazon Linux 2 使用者指南中的使用者[提供的核心](#)。

必要：否

輸出

描述綁定程序階段與狀態的狀態訊息。

範例

此範例會將 my-ami.manifest.xml 資訊清單中指定的 AMI，從美國複製至歐洲。

```
[ec2-user ~]$ ec2-migrate-manifest --manifest my-ami.manifest.xml
--cert cert-HKZYKTAIG2ECMXIYIBH3HXV4ZBZQ55CL0.pem --privatekey pk-
HKZYKTAIG2ECMXIYIBH3HXV4ZBZQ55CL0.pem --region eu-west-1
```

```
Backing up manifest...
```

```
Successfully migrated my-ami.manifest.xml It is now suitable for use in eu-west-1.
```

ec2-unbundle

描述

從執行個體存放區後端 Linux AMI 重新建立 bundle。

語法

```
ec2-unbundle -k path -m path [-s source_directory] [-d destination_directory]
```

選項

-k, --privatekey *path*

指向您以 PEM 編碼的 RSA 金鑰檔案路徑。

必要：是

-m, --manifest *path*

指向資訊清單檔案的路徑。

必要：是

-s, --source *source_directory*

內含 bundle 的目錄。

預設：目前的目錄。

必要：否

-d, --destination *destination_directory*

欲從中解開 AMI 的目錄。此目的地目錄必須存在。

預設：目前的目錄。

必要：否

範例

此 Linux 和 UNIX 範例會解開 `image.manifest.xml` 檔案內指定的 AMI。

```
[ec2-user ~]$ mkdir unbundled
$ ec2-unbundle -m mybundle/image.manifest.xml -k pk-
HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem -s mybundle -d unbundled
$ ls -l unbundled
total 1025008
-rw-r--r-- 1 root root 1048578048 Aug 25 23:46 image.img
```

輸出

會顯示指出解開程序各個階段的狀態訊息。

ec2-upload-bundle

描述

將執行個體存放區後端 Linux AMI 的套件組合上傳至 Amazon S3，並在所上傳的物件上設定適當的存取控制清單 (ACL)。如需詳細資訊，請參閱 [建立執行個體存放區後端 Linux AMI](#)。

Note

若要為您的執行個體存放區後端 Linux AMI 將物件上傳到 S3 儲存貯體，必須為儲存貯體啟用 ACL。否則，Amazon EC2 將無法在要上傳的物件上設定 ACL。如果您的目的地儲存貯體將儲存貯體擁有者強制設定用於 S3 物件擁有權，這將無法運作，因為 ACL 已停用。如需詳細資訊，請參閱 [使用 S3 物件擁有權控制上傳物件的擁有權](#)。

語法

```
ec2-upload-bundle -b bucket -a access_key_id -s secret_access_key [-t
token] -m path [--url url] [--region region] [--sigv version] [--acl acl]
[-d directory] [--part part] [--retry] [--skipmanifest]
```

選項

-b, --bucket *bucket*

存放 bundle 的 Amazon S3 儲存貯體名稱，其後為選用「/」以分隔路徑字首。若該儲存貯體不存在，則會加以建立 (如該儲存貯體名稱為可用)。此外，如果儲存貯體不存在且 AMI 工具版本為 1.5.18 或更新版本，則此命令會設定儲存貯體的 ACL。

必要：是

`-a, --access-key access_key_id`

您的 AWS 存取金鑰 ID。

必要：是

`-s, --secret-key secret_access_key`

您的 AWS 秘密訪問密鑰。

必要：是

`-t, --delegation-token token`

要傳遞給 AWS 要求的委派權杖。如需詳細資訊，請參閱[使用臨時安全登入資料](#)。

必要：僅適用您使用暫時安全登入資料時。

預設：AWS_DELEGATION_TOKEN 環境變數的值 (如已設定)。

`-m, --manifest path`

指向資訊清單檔案的路徑。資訊清單檔案會於綁定程序期間建立，位於內含 bundle 的目錄。

必要：是

`--url url`

已廢除。改用 `--region` 選項，除非您的儲存貯體被限制在 EU 位置 (而非 `eu-west-1`)。 `--location` 旗標是確定該特定位置限制的唯一方法。

Amazon S3 端點服務 URL。

預設：`https://s3.amazonaws.com/`

必要：否

`--region region`

要在請求簽章內針對目的地 S3 儲存貯體使用的區域。

- 若該儲存貯體不存在且您未指定區域，此工具會建立不具位置限制的儲存貯體 (位於 `us-east-1`)。
- 若該儲存貯體不存在且您已指定區域，此工具會在該區域建立儲存貯體。
- 若該儲存貯體存在但您未指定區域，此工具會使用該儲存貯體的位置。
- 若該儲存貯體存在且您指定 `us-east-1` 為區域，此工具會使用該儲存貯體的實際位置，不會出現錯誤訊息，而且會覆寫所有現有的相符檔案。

- 若該儲存貯體存在且您指定的區域 (us-east-1 之外) 與儲存貯體的實際位置不相符，此工具會退出並顯示錯誤。

若您的儲存貯體被限制在 EU 位置 (而非 eu-west-1)，請改用 `--location` 旗標。 `--location` 旗標是確定該特定位置限制的唯一方法。

預設：us-east-1

必要：若使用簽章第 4 版則為必要

`--sigv version`

簽署請求時使用的簽章版本。

有效值：2 | 4

預設：4

必要：否

`--acl acl`

綁定映像的存取控制清單政策。

有效值：public-read | aws-exec-read

預設：aws-exec-read

必要：否

`-d, --directory directory`

內含綁定 AMI 組件的目錄。

預設：內含資訊清單檔案的目錄 (請參閱 `-m` 選項)。

必要：否

`--part part`

開始上傳指定組件及所有後續組件。例如 `--part 04`。

必要：否

`--retry`

在所有 Amazon S3 錯誤後自動重試，每次操作至多五次。

必要：否

--skipmanifest

不會上傳資訊清單。

必要：否

--location location

已廢除。改用 --region 選項，除非您的儲存貯體被限制在 EU 位置 (而非 eu-west-1)。--location 旗標是確定該特定位置限制的唯一方法。

目的地 Amazon S3 儲存貯體的位置限制。若該儲存貯體存在且您指定的位置與儲存貯體的實際位置不相符，此工具會退出並顯示錯誤。若該儲存貯體存在但您未指定位置，此工具會使用該儲存貯體的位置。若該儲存貯體不存在且您已指定位置，此工具會在該位置建立儲存貯體。若該儲存貯體不存在且您未指定位置，此工具會建立不具位置限制的儲存貯體 (位於 us-east-1)。

預設：若已指定 --region，則位置會設定為該指定區域。若未指定 --region，則位置預設為 us-east-1。

必要：否

輸出

Amazon EC2 顯示的狀態訊息，會指出上傳程序的階段與狀態。

範例

此範例會上傳 image.manifest.xml 資訊清單指定的 bundle。

```
[ec2-user ~]$ ec2-upload-bundle -b DOC-EXAMPLE-BUCKET1/bundles/bundle_name -m
image.manifest.xml -a your_access_key_id -s your_secret_access_key
Creating bucket...
Uploading bundled image parts to the S3 bucket DOC-EXAMPLE-BUCKET1 ...
Uploaded image.part.00
Uploaded image.part.01
Uploaded image.part.02
Uploaded image.part.03
Uploaded image.part.04
Uploaded image.part.05
Uploaded image.part.06
```



```
Uploaded image.part.07
Uploaded image.part.08
Uploaded image.part.09
Uploaded image.part.10
Uploaded image.part.11
Uploaded image.part.12
Uploaded image.part.13
Uploaded image.part.14
Uploading manifest ...
Uploaded manifest.
Bundle upload completed.
```

AMI 工具常用選項

多數 AMI 工具接受下列選用參數。

`--help, -h`

顯示協助訊息。

`--version`

顯示版本及著作權聲明。

`--manual`

顯示手動輸入。

`--batch`

以批次模式執行，不使用互動式提示。

`--debug`

顯示可用於故障排除問題的資訊。

使用視窗系統建立 AMI

Microsoft System Preparation (Sysprep) 工具可簡化複製 Windows 自訂安裝的程序。您可以使用 Sysprep 來建立標準化的 Amazon Machine Image (AMI)。您接著可以從此標準化映像建立 Windows 的新 Amazon EC2 執行個體。

我們建議您使用 [EC2 Image Builder](#) 來自動建立、管理和部署自訂、安全和 up-to-date 「黃金」伺服器映像，這些影像已預先安裝並預先設定軟體和設定。

如果您使用視窗系統表示建立標準化 AMI，我們建議您使用執行 Sysprep 與 [EC2Launch v2](#) 如果您仍在使用 EC2Config (Windows Server 2012 R2 及更早版本)或 EC2Launch (Windows Server 2016 及 2019) 代理程式，請參閱下面關於與 EC2Config 和 EC2Launch 一起搭配使用 Sysprep 的說明文件。

Important

請不要使用 Sysprep 來建立執行個體備份。Sysprep 會移除系統特定資訊；移除此資訊可能會在執行個體備份時發生意料之外的結果。

若要疑難排解 Sysprep，請參閱 [疑難排解 Windows 執行個體的系統問題](#)。

內容

- [開始之前](#)
- [使用 Sysprep 搭配 EC2Launch v2](#)
- [使用 Sysprep 搭配 EC2Launch](#)
- [使用 Sysprep 搭配 EC2Config](#)

開始之前

- 執行 Sysprep 之前，用於執行 Sysprep 的單一管理員帳戶除外，建議您移除其他所有本機使用者帳戶和所有帳戶描述檔。如果您以其他帳戶和描述檔執行 Sysprep，可能會發生非預期的行為，包括遺失描述檔資料或無法完成 Sysprep。
- 了解更多關於 [Microsoft 的系統](#)。TechNet
- 了解 [Sysprep 可支援的伺服器角色](#)。

使用 Sysprep 搭配 EC2Launch v2

本節包含不同 Sysprep 執行階段，以及映像就緒時 EC2Launch v2 服務所執行之任務的詳細資訊。它也包括使用 Sysprep 搭配 EC2Launch v2 服務來建立標準化 AMI 的步驟。

Sysprep 搭配 EC2Launch v2 主題

- [Sysprep 階段](#)
- [Sysprep 動作](#)
- [Sysprep 後續階段](#)
- [使用 EC2Launch v2 執行 Sysprep](#)

Sysprep 階段

Sysprep 會執行下列階段：

- 一般化：此工具會移除映像特定資訊和組態。例如，Sysprep 會移除安全識別符 (SID)、電腦名稱、事件日誌和特定驅動程式等。完成本階段之後，作業系統 (OS) 即準備好建立 AMI。

Note

當您使用 EC2Launch v2 服務執行 Sysprep 時，系統會防止驅動程式被移除，因為 `PersistAllDeviceInstalls` 設定預設為 `true`。

- 特製化：隨插即用會掃描電腦，並為任何偵測到的裝置安裝驅動程式。此工具會產生電腦名稱和 SID 這類 OS 要求。或者，您可以在此階段執行命令。
- 立即可用體驗 (OOBE)：系統執行縮簡版的 Windows 安裝程式，並要求您輸入系統語言、時區和已註冊的組織這類資訊。當您於 EC2Launch v2 執行 Sysprep 時，回應檔案會自動執行此階段。

Sysprep 動作

Sysprep 和 EC2Launch v2 會在準備映像時執行下列動作。

1. 當您在 EC2Launch settings (EC2Launch 設定) 對話方塊中選擇 Shutdown with Sysprep (使用 Sysprep 關機) 時，系統會執行 `ec2launch sysprep` 命令。
2. EC2Launch v2 透過讀取位於 `unattend.xml` 的登錄值來編輯 `HKEY_USERS\DEFAULT\Control Panel\International\LocaleName` 檔案的內容。檔案位於以下目錄：`C:\ProgramData\Amazon\EC2Launch\sysprep`。
3. 系統會執行 `BeforeSysprep.cmd`。此命令會建立一個登錄機碼，如下所示：

```
reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /v fDenyTSConnections /t REG_DWORD /d 1 /f
```

除非重新啟用 RDP 連接，否則此登錄機碼會停用連線。停用 RDP 連線是必要的安全措施，因為在執行 Sysprep 之後的第一個開機工作階段期間，RDP 會有一小段時間允許連接，而且管理員密碼會空白。

4. EC2Launch v2 服務會執行下列命令來呼叫 Sysprep：

```
sysprep.exe /oobe /generalize /shutdown /unattend: "C:\ProgramData\Amazon\EC2Launch\sysprep\unattend.xml"
```

一般化階段

- EC2Launch v2 會移除映像特定資訊和組態 (例如電腦名稱和 SID)。如果執行個體是網域成員，即會從網域中移除。unattend.xml 回應檔案包括下列可影響此階段的設定：
 - PersistAllDeviceInstalls：此設定可防止 Windows 安裝程式移除和重新設定裝置，進而加快映像準備程序的速度，因為 Amazon AMI 需要執行特定驅動程式，並且重新偵測這些驅動程式需要一些時間。
 - DoNotCleanUpNonPresent裝置：此設定會保留目前不存在的裝置的隨插即用資訊。
- Sysprep 會在準備建立 AMI 時關閉 OS。系統會啟動新的執行個體，或啟動原始執行個體。

特製化階段

系統會產生電腦名稱和 SID 這類 OS 特定要求。系統也會根據您在 unattend.xml 回應檔案中指定的組態，來執行下列動作。

- CopyProfile：Sysprep 可設定為刪除所有使用者設定檔，包括內建的管理員設定檔。此設定會保留內建管理員帳戶，因此您對該帳戶進行的任何自訂都會帶到新的映像。預設值為 True。

CopyProfile以現有的本機管理員設定檔取代預設設定檔。執行 Sysprep 之後登入的所有帳戶，將於第一次登入時收到該描述檔的複本及其內容。

如果您沒有特定的使用者描述檔自訂要帶到新映像，請將此設定變更為 False。Sysprep 將移除所有使用者描述檔 (這樣可節省時間和磁碟空間)。

- TimeZone：依預設，時區設定為「座標世界時間 (UTC)」。
- 順序為 1 的同步命令：系統會執行下列命令，以啟用管理員帳戶並指定密碼要求。

```
net user Administrator /ACTIVE:YES /LOGONPASSWORDCHG:NO /EXPIRES:NEVER /  
PASSWORDREQ:YES
```

- 順序為 2 的同步命令：系統會將管理員密碼變碼。此安全性措施的設計目的在於防止在 Sysprep 完成後，如果您未設定工作，則無法存取執行個體setAdminAccount。

系統會從您的本機啟動代理程式目錄 (C:\Program Files\Amazon\EC2Launch\) 執行下列命令。

```
EC2Launch.exe internal randomize-password --username Administrator
```

- 若要啟用遠端桌面連線，系統會將終端機伺服器fDenyTSConnections登錄機碼設定為 false。

OOBE 階段

1. 系統使用 EC2Launch v2 回應檔案指定下列組態：

- `<InputLocale>en-US</InputLocale>`
- `<SystemLocale>en-US</SystemLocale>`
- `<UILanguage>en-US</UILanguage>`
- `<UserLocale>en-US</UserLocale>`
- `<HideEULAPage>true</HideEULAPage>`
- `<HideWirelessSetupInOOBE>true</HideWirelessSetupInOOBE>`
- `<ProtectYourPC>3</ProtectYourPC>`
- `<BluetoothTaskbarIconEnabled>>false</BluetoothTaskbarIconEnabled>`
- `<TimeZone>UTC</TimeZone>`
- `<RegisteredOrganization>Amazon.com</RegisteredOrganization>`
- `<RegisteredOwner>EC2</RegisteredOwner>`

Note

在一般化和特製化階段期間，EC2Launch v2 會監控 OS 的狀態。如果 EC2Launch v2 偵測到 OS 處於 Sysprep 階段，即會將下列訊息發佈至系統日誌：
Windows is being configured. SysprepState= 不可部署的影像狀態

2. 系統會執行 EC2Launch v2。

Sysprep 後續階段

Sysprep 完成之後，EC2Launch v2 會將下列訊息傳送至主控台輸出：

```
Windows sysprep configuration complete.
```

EC2Launch v2 接著會執行下列動作：

1. 讀取 `agent-config.yml` 檔案的內容並執行已設定的任務。
2. 執行 `preReady` 階段中的所有任務。
3. 完成後，會將 `Windows is ready` 訊息傳送至執行個體系統日誌。
4. 執行 `PostReady` 階段中的所有任務。

如需 EC2Launch v2 的詳細資訊，請參閱[使用 EC2Launch v2 設定 Windows 執行個體](#)。

使用 EC2Launch v2 執行 Sysprep

請使用下列程序，使用 Sysprep 搭配 EC2Launch v2 來建立標準化 AMI。

1. 在 Amazon EC2 主控台中，找到您要複製的 AMI。
2. 啟動和連線至 Windows 執行個體。
3. 自訂執行個體。
4. 在 Windows 的開始功能表中，搜尋並選擇 Amazon EC2Launch 設定。如需有關 Amazon EC2Launch 設定對話方塊中選項和設定的詳細資訊，請參閱[EC2Launch v2 設定](#)。
5. 選取 Shutdown with Sysprep (使用 Sysprep 關機) 或 Shutdown without Sysprep (不使用 Sysprep 關機)。

系統會要求您確認要執行 Sysprep 並將執行個體關機，請按一下 Yes (是)。EC2Launch v2 執行 Sysprep。接著，系統會將您登出執行個體，然後執行個體會關機。如果您檢查 Amazon EC2 主控台內的 Instances (執行個體) 頁面，執行個體狀態會從 Running 變更為 Stopping，再變更為 Stopped。這時便能放心地從此執行個體建立 AMI。

您可以從命令列使用下列命令手動呼叫 Sysprep 工具：

```
"%programfiles%\amazon\ec2launch\ec2launch.exe" sysprep --shutdown=true
```

使用 Sysprep 搭配 EC2Launch

EC2Launch 會為 Sysprep 提供預設回應檔案和批次檔案，這可自動化和保護您 AMI 上的映像準備程序。您可選擇性修改這些檔案。這些檔案預設位於下列目錄：C:\ProgramData\Amazon\EC2-Windows\Launch\Sysprep。

Important

請不要使用 Sysprep 來建立執行個體備份。Sysprep 會移除系統專屬資訊。如果您移除此資訊，執行個體備份可能產生意外結果。

Sysprep 搭配 EC2Launch 主題

- [Sysprep 的 EC2Launch 回應和批次檔案](#)

- [以 EC2Launch 執行 Sysprep](#)
- [啟動自訂 AMI 時更新 Server 2016 和更新版本的中繼資料/KMS 路由](#)

Sysprep 的 EC2Launch 回應和批次檔案

Sysprep 的 EC2Launch 回應檔案和批次檔案包含下列內容：

`Unattend.xml`

這是預設回應檔案。如果您在使用者介面中執行 `SysprepInstance.ps1` 或選擇 `ShutdownWithSysprep`，系統會從此檔案讀取設定。

`BeforeSysprep.cmd`

自訂此批次檔案可在 EC2Launch 執行 Sysprep 前執行命令。

`SysprepSpecialize.cmd`

自訂此批次檔案可在 Sysprep 特製化階段期間執行命令。

以 EC2Launch 執行 Sysprep

完整安裝 Windows Server 2016 和更新版本 (包含桌面體驗) 後，您可以使用 EC2Launch 手動執行 Sysprep，或使用 EC2 Launch Settings (EC2 Launch 設定) 應用程式執行 Sysprep。

使用 EC2Launch Settings (EC2Launch 設定) 應用程式執行 Sysprep

1. 在 Amazon EC2 主控台中，尋找或建立 Windows Server 2016 或更新版本 AMI。
2. 從 AMI 啟動 Windows 執行個體。
3. 連線到您的 Windows 執行個體並加以自訂。
4. 搜尋並執行 EC2 應LaunchSettings用程式。根據預設，其位於下列目錄：`C:\ProgramData\Amazon\EC2-Windows\Launch\Settings`。

Ec2 Launch Settings

General

Set Computer Name

Set the computer name of the instance ip- <hex internal IP>. Disable this feature to persist your own computer name setting.

Set Wallpaper

Overlay instance information on the current wallpaper.

Extend Boot Volume

Extend OS partition to consume free space for boot volume.

Add DNS Suffix List

Add DNS suffix list to allow DNS resolution of servers running in EC2 without providing the fully qualified domain name.

Handle User Data

Execute user data provided at instance launch.
Note: This will be re-enabled when running shutdown with sysprep below.

Administrator Password

Random (Retrieve from console)

Specify (Temporarily store in config file)

Do Nothing (Customize Unattend.xml for sysprep)

These changes will take effect on next boot if Ec2Launch script is scheduled. By default, it is scheduled by shutdown options below.

Sysprep

Sysprep is a Microsoft tool that prepares an image for multiple launches.

Ec2Launch Script Location: **Found**

Run EC2Launch on every boot (instead of just the next boot).

- 視需要選取或清除選項。這些設定存放在 LaunchConfig.json 檔案中。
- 針對 Administrator Password (管理員密碼)，請執行下列其中一項作業：

- 選擇 Random (隨機)。EC2Launch 會產生密碼並以使用者金鑰將其加密。系統會在執行個體啟動後停用此設定，以便執行個體重新開機或停止再啟動時，能持續使用這個密碼。
 - 選擇 Specify (指定) 並輸入符合系統要求的密碼。系統會將該密碼以純文字存放於 LaunchConfig.json，並在 Sysprep 設定管理員密碼後予以刪除。如果您立即關機，即會立即設定密碼。EC2Launch 會以使用者金鑰加密該密碼。
 - 在檔案中選擇 DoNothing 並指定 unattend.xml 密碼。如果您未在 unattend.xml 中指定密碼，會停用管理員帳戶。
7. 選擇 Shutdown with Sysprep (使用 Sysprep 關機)。

使用 EC2Launch 手動執行 Sysprep

1. 在 Amazon EC2 主控台中尋找或建立您想要複製的 Windows Server 2016 或更新版本 Datacenter 版 AMI。
2. 啟動和連線至 Windows 執行個體。
3. 自訂執行個體。
4. 在 LaunchConfig.json 檔案中指定設定。根據預設，此檔案位於 C:\ProgramData\Amazon\EC2-Windows\Launch\Config 目錄。

針對 adminPasswordType，請指定下列其中一個值：

Random

EC2Launch 會產生密碼並以使用者金鑰將其加密。系統會在執行個體啟動後停用此設定，以便執行個體重新開機或停止再啟動時，能持續使用這個密碼。

Specify

EC2Launch 會使用您在 adminPassword 中指定的密碼。如果該密碼不符合系統要求，EC2Launch 會改為產生隨機密碼。系統會將該密碼以純文字存放於 LaunchConfig.json，並在 Sysprep 設定管理員密碼後予以刪除。EC2Launch 會以使用者金鑰加密該密碼。

DoNothing

EC2Launch 會使用您在 unattend.xml 檔案中指定的密碼。如果您未在 unattend.xml 中指定密碼，會停用管理員帳戶。

5. (選用) 請在 `unattend.xml` 和其他組態檔案中指定設定。如果您計畫安裝，就不需要變更這些檔案。根據預設，檔案位於下列目錄：`C:\ProgramData\Amazon\EC2-Windows\Launch\Sysprep`。
6. 在視窗中 PowerShell，執行 `./InitializeInstance.ps1 -Schedule`。根據預設，指令碼位於下列目錄：`C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts`。此指令碼會排程執行個體在下次開機期間初始化。您必須在執行下一步的 `SysprepInstance.ps1` 指令碼前，執行此指令碼。
7. 在視窗中 PowerShell，執行 `./SysprepInstance.ps1`。根據預設，指令碼位於下列目錄：`C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts`。

系統會將您登出執行個體，然後執行個體會關機。如果您檢查 Amazon EC2 主控台中的 Instances (執行個體) 頁面，執行個體狀態會從 `Running` 變更為 `Stopping`，然後變更為 `Stopped`。這時便能放心地從這個執行個體建立 AMI。

啟動自訂 AMI 時更新 Server 2016 和更新版本的中繼資料/KMS 路由

若要在啟動自訂 AMI 時更新 Server 2016 和更新版本的中繼資料/KMS 路由，請執行下列其中一項動作：

- 運行 EC2 LaunchSettings 圖形用戶界面 (`C:\ProgramData\Amazon\EC2-視窗\啟動\設置\EC2 LaunchSettings.exe` 文件)，然後選擇關閉與系統的選項。
- 在建立 AMI 之前，執行 EC2 LaunchSettings 並在沒有 Sysprep 的情況下關閉。這樣會將 EC2 啟動初始化任務設為在下次開機時執行，如此將會根據執行個體的子網來設定路由。
- 在從 [PowerShell](#) 中建立 AMI 之前，手動重新排程 EC2 啟動初始化工作。

Important

在重新排定任務之前，請記下預設密碼重設行為。

- 若要在發生 Windows 啟用失敗或與執行個體中繼資料通訊失敗的執行中的執行個體上更新路由，請參閱 [「無法啟用 Windows」](#)。

使用 Sysprep 搭配 EC2Config

本節包含不同 Sysprep 執行階段，以及映像就緒時 EC2Config 服務所執行之任務的詳細資訊。它也包括使用 Sysprep 搭配 EC2Config 服務來建立標準化 AMI 的步驟。

Sysprep 搭配 EC2Config 主題

- [Sysprep 階段](#)
- [Sysprep 動作](#)
- [Sysprep 後續階段](#)
- [搭配 EC2Config 服務執行 Sysprep](#)

Sysprep 階段

Sysprep 會執行下列階段：

- 一般化：此工具會移除映像特定資訊和組態。例如，Sysprep 會移除安全識別符 (SID)、電腦名稱、事件日誌和特定驅動程式等。完成本階段之後，作業系統 (OS) 即準備好建立 AMI。

Note

當您使用 EC2Config 服務執行 Sysprep 時，系統會防止驅動程式遭到移除，因為依預設 `PersistAllDeviceInstalls` 設定為 `true`，因此系統會防止驅動程式遭到移除。

- 特製化：隨插即用會掃描電腦，並為任何偵測到的裝置安裝驅動程式。此工具會產生電腦名稱和 SID 這類 OS 要求。或者，您可以在此階段執行命令。
- 立即可用體驗 (OOBE)：系統執行縮簡版的 Windows 安裝程式，並要求使用者輸入系統語言、時區和已註冊的組織這類資訊。當您於 EC2Config 執行 Sysprep 時，回應檔案會自動執行此階段。

Sysprep 動作

Sysprep 和 EC2Config 服務會在準備映像時執行下列動作。

1. 當您在 EC2 服務屬性對話方塊中選擇使用 Sysprep 關機時，系統會執行 `ec2config.exe -sysprep` 命令。
2. EC2Config 服務會讀取 `BundleConfig.xml` 檔案的內容。此檔案預設位在下列目錄：`C:\Program Files\Amazon\Ec2ConfigService\Settings`。

`BundleConfig.xml` 檔案包括下列設定。您可以變更這些設定：

- `AutoSysprep`：指出是否要自動使用系統表示。如果您要從 EC2 Service Properties (EC2 服務屬性) 對話方塊中執行 Sysprep，則不需要變更此值。預設值為 `No`。

- `SetRDPCertificate`：設定遠端桌面伺服器的自簽憑證。這可讓您安全地使用遠端桌面通訊協定 (RDP) 連線至執行個體。如果新的執行個體應該使用憑證，請將值變更為 `Yes`。這個設定不適用於 Windows 伺服器 2012 執行個體，因為這些作業系統可以產生自己的憑證。預設值為 `No`。
 - `SetPasswordAfterSysprep`：在新啟動的執行個體上設定隨機密碼，使用使用者啟動金鑰對其進行加密，然後將加密的密碼輸出至主控台。如果新執行個體不應該設定為隨機加密密碼，請將值變更為 `No`。預設值為 `Yes`。
 - `PreSysprepRunCmd`：要執行的命令的位置。根據預設，命令位於下列目錄：`C:\Program Files\Amazon\Ec2ConfigService\Scripts\BeforeSysprep.cmd`。
3. 系統會執行 `BeforeSysprep.cmd`。此命令會建立一個登錄機碼，如下所示：

```
reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /v  
fDenyTSConnections /t REG_DWORD /d 1 /f
```

除非重新啟用 RDP 連接，否則此登錄機碼會停用連線。停用 RDP 連線是必要的安全措施，因為在執行 Sysprep 之後的第一個開機工作階段期間，RDP 會有一小段時間允許連接，而且管理員密碼會空白。

4. EC2Config 服務會執行下列命令來呼叫 Sysprep：

```
sysprep.exe /unattend: "C:\Program Files\Amazon\Ec2ConfigService\sysprep2008.xml" /  
oobe /generalize /shutdown
```

一般化階段

- 此工具會移除映像特定資訊和組態 (例如電腦名稱和 SID)。如果執行個體是網域成員，即會從網域中移除。`sysprep2008.xml` 回應檔案包括下列可影響此階段的設定：
 - `PersistAllDeviceInstalls`：此設定可防止 Windows 安裝程式移除和重新設定裝置，進而加快映像準備程序的速度，因為 Amazon AMI 需要執行特定驅動程式，並且重新偵測這些驅動程式需要一些時間。
 - `DoNotCleanUpNonPresent` 裝置：此設定會保留目前不存在的裝置的隨插即用資訊。
- Sysprep 會在準備建立 AMI 時關閉 OS。系統會啟動新的執行個體，或啟動原始執行個體。

特製化階段

系統會產生電腦名稱和 SID 這類 OS 特定要求。系統也會根據您在 `sysprep2008.xml` 回應檔案中指定的組態，來執行下列動作。

- CopyProfile : Sysprep 可設定為刪除所有使用者設定檔，包括內建的管理員設定檔。此設定會保留內建管理員帳戶，因此您對該帳戶進行的任何自訂都會帶到新的映像。預設值為 True。

CopyProfile以現有的本機管理員設定檔取代預設設定檔。執行 Sysprep 之後登入的所有帳戶，將於第一次登入時收到該描述檔的複本及其內容。

如果您沒有特定的使用者描述檔自訂要帶到新映像，請將此設定變更為 False。Sysprep 將移除所有使用者描述檔；這樣可節省時間和磁碟空間。

- TimeZone : 依預設，時區設定為「座標世界時間 (UTC)」。
- 順序為 1 的同步命令：系統會執行下列命令，以啟用管理員帳戶並指定密碼要求。

```
net user Administrator /ACTIVE:YES /LOGONPASSWORDCHG:NO /EXPIRES:NEVER /
PASSWORDREQ:YES
```

- 順序為 2 的同步命令：系統會將管理員密碼變碼。此安全措施的用途在於您未啟用 ec2setpassword 設定時，可防止在 Sysprep 完成之後存取執行個體。

```
C:\Program 文件\ Amazon\ EC2ConfigService\ ScramblePassword .exe」 -u 管理員
```

- 順序為 3 的同步命令：系統會執行下列命令：

```
C:\Program 文件\ Amazon\ EC2\ 腳本ConfigService\ SysprepSpecialize 相位. CMD
```

此命令會新增下列登錄機碼，以重新啟用 RDP：

```
註冊添加「HKEY_本地_機器\系統\CurrentControl 設置\控制\終端服務器」/v
fDenyTSConnections 接/T REG_DWORD /d 0 /F
```

OOBE 階段

1. 使用 EC2Config 服務回應檔案，系統可以指定下列組態：

- < InputLocale > 一般美國 InputLocale </ >
- < SystemLocale > 一般美國 SystemLocale </ >
- < UILanguage > en-US </ UILanguage >
- < UserLocale > 一般美國 UserLocale </ >
- < HideEULAPage > true </ HideEULAPage >
- < HideWireless SetupIn 屋頂 > 真實 </ OBE HideWireless SetupIn >
- < NetworkLocation > 其他 </ > NetworkLocation

- `< ProtectYour 個人電腦 >`
- `BluetoothTaskbarIconEnabled`
- `< TimeZone >UTC</ > TimeZone`
- `< RegisteredOrganization > 亞馬遜網站 </ > RegisteredOrganization`
- `< RegisteredOwner > Amazon </ > RegisteredOwner`

Note

在一般化和特製化階段期間，EC2Config 服務會監控 OS 狀態。如果 EC2Config 偵測到 OS 處於 Sysprep 階段，即會將下列訊息發佈至系統日誌：

EC2 ConfigMonitor 狀態：0 視窗正在設定中。SysprepState= 不可部署的影像狀態

2. OOBE 階段完成後，系統會從下列位置執行 `SetupComplete.cmd`：C:\Windows\Setup\Scripts\SetupComplete.cmd。在 2015 年 4 月之前的 Amazon 公有 AMI 中，此檔案為空白，而且不會對映像執行任何操作。在 2015 年 4 月之後的公有 AMI 中，此檔案包含以下數值：`call "C:\Program Files\Amazon\Ec2ConfigService\Scripts\PostSysprep.cmd"`。
3. 系統會執行 `PostSysprep.cmd`，執行下列操作：
 - 將本機管理員密碼設定為不過期。如果密碼過期，管理員可能會無法登入。
 - 設定 MSSQLServer 機器名稱 (如果已安裝)，讓名稱與 AMI 同步。

Sysprep 後續階段

Sysprep 完成之後，EC2Config 服務會將下列訊息傳送至主控台輸出：

```
Windows sysprep configuration complete.
Message: Sysprep Start
Message: Sysprep End
```

EC2Config 接著會執行下列動作：

1. 讀取 `config.xml` 檔案的內容，並列出所有已啟用的外掛程式。
2. 同時執行所有「Windows 就緒之前」(“Before Windows is ready”) 外掛程式。
 - EC2 SetPassword
 - EC2 SetComputer 名稱
 - EC2 InitializeDrives

- EC2 EventLog
 - Ec2ConfigureRDP
 - Ec2OutputRDP Cert
 - EC2 SetDrive 信
 - EC2 WindowsActivate
 - EC2 DynamicBoot VolumeSize
3. 完成之後，會將「Windows 就緒」(“Windows is ready”) 訊息傳送至執行個體系統日誌。
 4. 同時執行所有「Windows 就緒之後」(“After Windows is ready”) 外掛程式。
 - Amazon CloudWatch 日誌
 - UserData
 - AWS Systems Manager (Systems Manager)

如需 Windows 外掛程式的詳細資訊，請參閱[使用 EC2Config 服務 \(舊版\) 設定 Windows 執行個體](#)。

搭配 EC2Config 服務執行 Sysprep

使用下列程序，使用 Sysprep 和 EC2Config 服務來建立標準化 AMI。

1. 在 Amazon EC2 主控台中，找出或[建立](#)您要複製的 AMI。
2. 啟動和連線至 Windows 執行個體。
3. 自訂執行個體。
4. 在 EC2Config 服務回應檔案中指定組態設定：

```
C:\Program Files\Amazon\Ec2ConfigService\sysprep2008.xml
```

5. 從 Windows 的 [開始] 功能表中，選擇 [所有程式]，然後選擇 ConfigService[EC2 設定]。
6. 選擇 Ec2 Service Properties (Ec2 服務屬性) 對話方塊中的 Image (映像) 標籤。如需 Ec2 Service Properties (Ec2 服務屬性) 對話方塊中之選項和設定的詳細資訊，請參閱[Ec2 服務屬性](#)。
7. 選取管理員密碼的選項，然後選取 Shutdown with Sysprep (使用 Sysprep 關機) 或 Shutdown without Sysprep (不使用 Sysprep 關機)。EC2Config 會根據您選取的密碼選項來編輯設定檔案。
 - Random (隨機)：EC2Config 會產生密碼，並以使用者金鑰加密，然後將已加密的密碼輸出到主控台。我們會在第一次啟動後停用此設定，以便執行個體重新開機或是停止後再啟動時，能持續使用這個密碼。

- **Specify (指定)**：密碼會以未加密形式 (純文字) 存放於 Sysprep 回應檔案。Sysprep 接著執行時，會設定管理員密碼。如果您立即關機，即會立即設定密碼。在服務重新啟動時，會移除管理員密碼。請務必記住此密碼，之後即無法再擷取此密碼。
- **Keep Existing (保留現有項目)**：執行 Sysprep 或重新啟動 EC2Config 時，管理員帳戶的現有密碼不會變更。請務必記住此密碼，之後即無法再擷取此密碼。

8. 選擇 OK (確定)。

系統會要求您確認要執行 Sysprep 並將執行個體關機，請按一下 Yes (是)。您會注意到 EC2Config 執行 Sysprep。接著，系統會將您登出執行個體，然後執行個體會關機。如果您查看 Amazon EC2 主控台的 Instances (執行個體) 頁面，執行個體狀態會從 Running 變更為 Stopping，最後再變更為 Stopped。這時便能放心地從此執行個體建立 AMI。

您可以從命令列使用下列命令手動呼叫 Sysprep 工具：

```
"%programfiles%\amazon\ec2configservice\"ec2config.exe -sysprep"
```

Note

如果您的 CMD 殼層已經在 C:\Program 檔案\Amazon\EC2ConfigService\ 目錄中，則不需要使用命令中的雙引號。

不過，您必須審慎確認 Ec2ConfigService\Settings 資料夾中指定的 XML 檔案選項正確；否則，您可能無法連線至該執行個體。如需設定檔案的詳細資訊，請參閱 [EC2Config 設定檔](#)。如需從命令列設定 Sysprep 並執行的範例，請參閱 Ec2ConfigService\Scripts\InstallUpdates.ps1。

修改 AMI

您可以修改一組有限的 Amazon Machine Image (AMI) 屬性，例如 AMI 的描述和共用屬性。但是，AMI 內容 (磁碟區二進位資料) 無法修改。若要修改 AMI 內容，您必須 [建立新的 AMI](#)。

Important

您無法修改 EBS 支援的 AMI 內容 (磁碟區二進位資料)，因為備份它們的快照是不可變的。您也無法修改執行個體存放區支援 (S3 支援) Linux AMI 的內容 (磁碟區二進位資料)，因為內容已簽署，如果簽章不相符，執行個體啟動也會失敗。

如需可修改的 AMI 屬性，請參閱 Amazon EC2 API 參考中的 [ModifyImage](#) 屬性。

下列主題提供使用 Amazon EC2 主控台和修 AWS CLI 改 AMI 屬性的指示：

- [使 AMI 公有化](#)
- [與特定組織或組織單位共用 AMI](#)
- [與指定的 AWS 帳戶共享 AMI](#)
- [使用已支付支援](#)
- [設定 AMI](#)

複製 AMI

您可以在區域內或跨 AWS 區域複製 Amazon 機器映像 (AMI)。您可以同時複製 Amazon EBS 支援的 AMI 和執行個體商店支援的 AMI。您可以複製具有加密快照的 EBS 支援 AMI，也可以在複製過程中變更加密狀態。您可複製共享給您的 AMI。

複製來源 AMI 會產生相同但不同的新 AMI，我們也稱之為目標 AMI。目標 AMI 有其自己唯一的 AMI 識別碼。您可變更來源 AMI 或取消註冊，不會影響目標 AMI。反之亦然。

使用 EBS 支援的 AMI，其每個支援快照都會複製到相同但不同的目標快照。如果您複製 AMI 到新區域，則快照為完整 (非增量) 複本。如果您加密未加密的備份快照，或將其加密為新的 KMS 金鑰，則快照為完整 (非增量) 複本。AMI 的後續複製作業會產生備份快照的增量複本。

目錄

- [考量事項](#)
- [成本](#)
- [IAM 許可](#)
- [複製 AMI](#)
- [停用待定的 AMI 複製作業](#)
- [跨區域複製](#)
- [跨帳戶複製](#)
- [加密和複製](#)

考量事項

- 複製 AMI 的權限 — 您可以使用 IAM 政策授予或拒絕使用者複製 AMI 的權限。為 CopyImage 動作指定的資源層級許可僅適用於新的 AMI。您無法指定來源 AMI 的資源層級許可。
- 啟動許可和 Amazon S3 儲存貯體許可 — AWS 不會將啟動許可或 Amazon S3 儲存貯體許可從來源 AMI 複製到新的 AMI。複製操作完成後，您可將啟動許可和 Amazon S3 儲存貯體許可套用至新的 AMI。
- 標籤 — 您只能複製附加至來源 AMI 的使用者定義 AMI 標籤。將不會複製連接至其他 AWS 帳戶的系統標籤 (字首為 aws:) 和使用使用者定義的標籤。複製 AMI 時，您可以將新標籤附加到目標 AMI 及其後備快照。

成本

複製 AMI 不須支付費用。然而，將收取標準儲存和資料傳輸費用。如果您複製 EBS 後端 AMI，任何其他 EBS 快照的儲存則都會產生費用。

IAM 許可

若要複製 EBS 支援或 instance store-backed AMI，您需要下列 IAM 許可：

- ec2:CopyImage— 要複製 AMI。對於 EBS 支援的 AMI，它也會授予複製 AMI 支援快照的權限。
- ec2:CreateTags— 標記目標 AMI。對於 EBS 支援的 AMI，它也會授予標記目標 AMI 支援快照的權限。

如果您要複製儲存支援 AMI 的執行個體，則需要下列其他 IAM 許可：

- s3:CreateBucket— 在目標區域中為新 AMI 建立 S3 儲存貯體
- s3:GetBucketAcl— 讀取來源值區的 ACL 權限
- s3:ListAllMyBuckets— 尋找目標區域中 AMI 的現有 S3 儲存貯體
- s3:GetObject— 讀取來源值區中的物件
- s3:PutObject— 在目標值區中寫入物件
- s3:PutObjectAcl— 寫入目標值區中新物件的權限

複製 EBS 支援 AMI 並標記目標 AMI 和快照的 IAM 政策範例

下列範例原則授予您複製任何 EBS 支援 AMI 並標記目標 AMI 及其支援快照的權限。

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "PermissionToCopyAllImages",
    "Effect": "Allow",
    "Action": [
      "ec2:CopyImage",
      "ec2:CreateTags"
    ],
    "Resource": "arn:aws:ec2:*::image/*"
  }]
}
```

複製 EBS 支援 AMI 但拒絕標記新快照的 IAM 政策範例

當您取得`ec2:CopySnapshot`權限時，系統會自動授與`ec2:CopyImage`權限。這包括標記目標 AMI 的新支援快照的權限。可明確拒絕標記新支援快照的權限。

下列範例原則授予您複製 EBS 支援 AMI 的權限，但拒絕您標記目標 AMI 的新支援快照。

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "ec2:CopyImage",
      "ec2:CreateTags"
    ],
    "Resource": "arn:aws:ec2:*::image/*"
  },
  {
    "Effect": "Deny",
    "Action": "ec2:CreateTags",
    "Resource": "arn:aws:ec2::*:snapshot/*"
  }
]
```

複製執行個體支援存放區的 AMI 和標記目標 AMI 的 IAM 政策範例

下列範例政策授予您權限，將指定來源儲存貯體中的任何 instance store-backed AMI 複製到指定的區域，並標記目標 AMI。

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "PermissionToCopyAllImages",
    "Effect": "Allow",
    "Action": [
      "ec2:CopyImage",
      "ec2:CreateTags"
    ],
    "Resource": "arn:aws:ec2:*::image/*"
  },
  {
    "Effect": "Allow",
    "Action": "s3:ListAllMyBuckets",
    "Resource": [
      "arn:aws:s3::*"
    ]
  },
  {
    "Effect": "Allow",
    "Action": "s3:GetObject",
    "Resource": [
      "arn:aws:s3:::ami-source-bucket/*"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "s3:CreateBucket",
      "s3:GetBucketAcl",
      "s3:PutObjectAcl",
      "s3:PutObject"
    ],
    "Resource": [
      "arn:aws:s3:::amis-for-account-in-region-hash"
    ]
  }
  ]
}
```

若要尋找 AMI 來源儲存貯體的 Amazon Resource Name (ARN)，請透過 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台 ()，在導覽窗格中選擇 AMI，然後在 Source (來源) 欄中找到該儲存貯體名稱。

Note

只有在您第一次將 instance store-backed AMI 複製到個別區域時，才需要此 `s3:CreateBucket` 權限。之後，已在區域中建立的 Amazon S3 儲存貯體用來儲存您複製到該區域的所有未來 AMIs。

複製 AMI

您可以使用 AWS Management Console、AWS Command Line Interface 或開發套件或 Amazon EC2 API 複製 AMI，所有這些 API 都支援該 CopyImage 動作。

Console

若要複製 AMI

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 從主控台導覽列選取內含該 AMI 的區域。
3. 在導覽窗格中，選擇 AMI 以顯示「地區」中可用的 AMI 清單。
4. 如果您沒有看到要複製的 AMI，請選擇其他篩選器。您可以依據我擁有的 AMI、私人映像、公開映像和已停用的映像進行篩選。
5. 選取要複製的 AMI，然後選擇 [動作] > [複製 AMI]。
6. 在 Copy AMI (複製 AMI) 頁面上，指定下列資訊：
 - a. AMI copy name (AMI 複本名稱)：新 AMI 的名稱。您可以在名稱中包含作業系統資訊，因為 Amazon EC2 在顯示 AMI 的詳細資料時並未提供此資訊。
 - b. AMI copy description (AMI 複本描述)：根據預設，描述包括來源 AMI 的資訊，讓您能夠辨識原始和複本內容。您可視需要變更此描述。
 - c. Destination region (目的地區域)：AMI 複製到的區域。如需詳細資訊，請參閱 [跨區域複製](#)。
 - d. Copy tags (複製標籤)：選取此核取方塊可在複製 AMI 時包含使用者定義的 AMI 標籤。將不會複製連接至其他 AWS 帳戶的系統標籤 (字首為 `aws:`) 和使用者定義的標籤。
 - e. (僅 EBS 支援 AMI) 加密 AMI 副本的 EBS 快照：選取此核取方塊可加密目標快照，或使用不同的金鑰重新加密它們。如果預設為啟用加密，則會選取 [加密 AMI 副本的 EBS 快照] 核取方塊，且無法清除。如需詳細資訊，請參閱 [加密和複製](#)。
 - f. (僅限 EBS 支援的 AMI) KMS 金鑰：用來加密目標快照的 KMS 金鑰。

- g. 標籤：您可以使用相同的標籤標記新的 AMI 和新快照，也可以使用不同的標籤來標記它們。
- 若要使用相同標籤標記新的 AMI 和新快照，請選擇「將映像檔和快照標記在一起」。相同的標籤會套用至新的 AMI 和所建立的每個快照。
 - 若要使用不同標籤標記新 AMI 和新快照，請分別選擇「標記映像檔和快照」。不同的標籤會套用至新的 AMI 和建立的快照。不過請注意，建立的所有新快照都會取得相同的標籤；您無法使用不同的標籤來標記每個新快照。

若要新增標籤，請選擇 Add tag (新增標籤)，然後輸入標籤的鍵和值。針對每個標籤，重複執行上述動作。

- h. 當您準備好複製 AMI 時，請選擇「複製 AMI」。

新 AMI 的初始狀態為 Pending。當狀態為 Available 時，表示 AMI 複製作業完成。

AWS CLI

若要使用複製 AMI AWS CLI

您可使用 [copy-image](#) 命令來複製 AMI。您必須同時指定來源和目的地區域。請使用 `--source-region` 參數指定來源區域。請使用 `--region` 參數或環境變數指定目的地區域。若要取得更多資訊，請參閱 [〈規劃 AWS 指令行介面〉](#)。

(僅 EBS 支援 AMI) 當您在複製期間加密目標快照時，必須指定下列其他參數：和 `--encrypted`
`--kms-key-id`

如需範例命令，請參閱 AWS CLI 命令參考 中的 [copy-image](#) 下的 [Examples](#) (範例)。

PowerShell

若要使用視窗適用的工具複製 AMI PowerShell

您可以使用 [Copy-EC2Image](#) 指令複製 AMI。您必須同時指定來源和目的地區域。請使用 `-SourceRegion` 參數指定來源區域。請使用 `-Region` 參數或 `Set-AWSDefaultRegion` 命令指定目的地區域。如需詳細資訊，請參閱 [指定 AWS 區域](#)。

(僅 EBS 支援 AMI) 當您在複製期間加密目標快照時，必須指定下列其他參數：和 `-Encrypted -KmsKeyId`

停用待定的 AMI 複製作業

您可以使用 AWS Management Console 或命令列停止擱置的 AMI 副本。

Console

使用主控台停止 AMI 複製作業

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 從導覽列中的區域選擇器，選取目的地區域。
3. 在導覽窗格中，選擇 AMIs (AMI)。
4. 選取要停止複製的 AMI，然後選擇「動作」、「取消註冊 AMI」。
5. 要求確認時，請選擇 Deregister AMI (取消註冊 AMI)。

Command line

使用命令列停止 AMI 複製作業

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱[存取 Amazon EC2](#)。

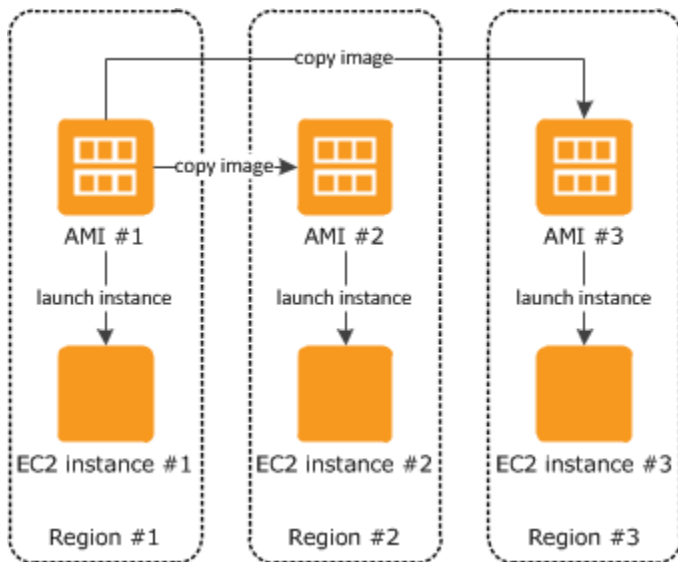
- [deregister-image](#) (AWS CLI)
- [Unregister-EC2Image](#) (AWS Tools for Windows PowerShell)

跨區域複製

跨越地理區域複製 AMI，具有下列優點：

- 一致的全球部署：將 AMI 從一個區域複製到另一個區域，可讓您以相同 AMI 在不同區域啟動一致的執行個體。
- 擴展能力：無論使用者的位置，設計並建置滿足全球使用者需求的應用程式將變得更輕鬆。
- 效能：您可分散應用程式，並將應用程式的重要元件放置於靠近使用者的位置，藉此增加效能。您也可以利用特定區域的功能，例如執行個體類型或其他 AWS 服務。
- 高可用性：您可跨 AWS 區域設計並部署應用程式，增加可用性。

下圖顯示來源 AMI 與不同區域中兩個複製 AMI 之間的關係，以及從每個執行個體啟動的 EC2 執行個體。從 AMI 啟動執行個體時，其所處位置與該 AMI 的區域相同。若您變更來源 AMI 並希望在目標區域的 AMIs 反映這些變更，您必須將來源 AMI 重新複製到目標區域。



首次將執行個體後端 AMI 複製到某個區域時，我們會為複製到該區域的 Amazon S3 建立 AMIs 儲存貯體。所有複製到該區域的執行個體後端 AMIs，都會存放於此儲存貯體。儲存貯體名稱為下列格式：`amis-for-account-in-region-hash`。例如：`amis-for-123456789012-in-us-east-2-yhjmvp6`。

先決條件

複製 AMI 前，您必須確認來源 AMI 的內容已更新為可支援在不同區域運作。例如，您應更新資料庫連線字串或類似的應用程式組態資料，以指向適當的資源。否則，從目的地區域中的新 AMI 啟動的執行個體仍可能會使用來源區域的資源，這可能會影響效能和成本。

限制

- 目的地區域限制為 100 個並行 AMI 複本。
- 您無法將半虛擬 (PV) AMI 複製到不支援 PV AMI 的區域。如需詳細資訊，請參閱 [AMI 虛擬化類型](#)。

跨帳戶複製

您可以與其他 AWS 帳戶共享 AMI。共享 AMI 不會影響該 AMI 的所有權。擁有帳戶須支付該區域的儲存費用。如需詳細資訊，請參閱 [與指定的 AWS 帳戶共享 AMI](#)。

若您複製的 AMI 已經與您的帳戶共享，則您是您帳戶內目標 AMI 的擁有者。來源 AMI 的擁有者須支付 Amazon EBS 或 Amazon S3 標準傳輸費用，而您則須支付目的地區域目標 AMI 的儲存費用。

資源許可

若要從另一個帳戶複製已與您共用的 AMI，來源 AMI 的擁有者必須授予您支援 AMI 的儲存體的讀取許可。該儲存體為相關聯的 EBS 快照 (Amazon EBS 支援的 AMI) 或相關聯的 S3 儲存貯體 (執行個體儲存體支援的 AMI)。如果共享 AMI 具有加密的快照，則擁有者也必須與您共享金鑰。如需授予資源許可的詳細資訊，對於 EBS 快照，請參閱 [Amazon EBS 使用者指南中的共用 Amazon EBS 快照](#)，如需 S3 儲存貯體，請參閱 Amazon 簡單儲存服務使用者指南中的 Amazon S3 中的身分和存取管理。

Note

若要複製 AMI 及其標籤，您必須擁有來源 AMI 的啟動許可。

加密和複製

下表說明各種 AMI 複製案例支援的加密。雖然您可將未加密快照複製為加密快照，但是無法將加密快照複製為未加密快照。

案例	描述	支援
1	未加密到未加密	是
2	加密到加密	是
3	未加密到加密	是
4	加密到未加密	否

Note

在 CopyImage 動作期間的加密僅適用於 Amazon EBS 後端 AMIs。因為執行個體後端 AMI 不仰賴快照，所以您無法透過複製來變更其加密狀態。

根據預設 (也就是，未指定加密參數)，AMI 的備份快照會以原始的加密狀態進行複製。複製以未加密快照為後端的 AMI，將產生相同未加密的目標快照。如果來源 AMI 是由加密快照所支援，則複製它會產生相同的目標快照，並以相同的 AWS KMS 金鑰加密。若 AMI 的後端為多個快照，則進行複製時，每個目標快照預設都會保留來源加密狀態。

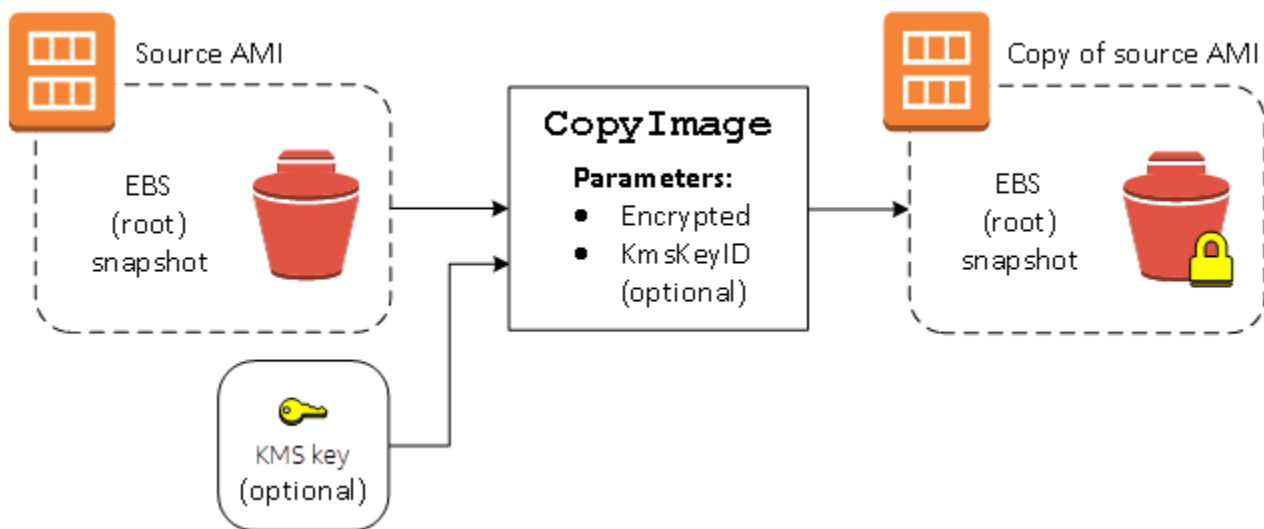
如果您在複製 AMI 時指定加密參數，則可加密或重新加密其支援快照。以下範例顯示將加密參數提供給 CopyImage 動作的非預設案例，以變更目標 AMI 的加密狀態。

將未加密來源 AMI 複製到加密目標 AMI

在此案例中，未加密根快照所支援的 AMI 將複製到具有加密根快照的 AMI。透過兩個加密參數叫用 CopyImage 動作，包括客戶受管金鑰。因此，根快照的加密狀態會變更，目標 AMI 會受到包含與來源快照相同資料的根快照支援，但使用指定的金鑰進行加密。您需要為兩個 AMI 中的快照支付儲存成本，以及從任一 AMI 啟動的任何執行個體的費用。

Note

依預設啟用加密，與將 Encrypted=true 參數設定為 AMI 中所有快照的效果相同。



設定 Encrypted 參數會加密此執行個體的單一快照。若您未指定 KmsKeyId 參數，則會使用預設的客戶受管金鑰來加密快照複本。

如需複製具有加密快照的 AMIs 的詳細資訊，請參閱[對 EBS 後端 AMI 使用加密](#)。

使用 S3 存放和還原 AMI

您可以將 Amazon Machine Image (AMI) 存放在 Amazon S3 儲存貯體中，將 AMI 複製到另一個 S3 儲存貯體，然後從 S3 儲存貯體中還原。透過使用 S3 儲存貯體儲存和還原 AMI，您可以將 AMI 從一個 AWS 分割區複製到另一個分割區，例如，從主要商業分割區複製到分 AWS GovCloud (US) 分割區。您也可以透過將其存放在 S3 儲存貯體中，來建立 AMI 的封存複本。

支援使用 S3 存放和還原 AMI 的 API 包括

CreateStoreImageTask、DescribeStoreImageTasks 和 CreateRestoreImageTask。

CopyImage 是建議用於在 AWS 分區內複製 AMI 的 API。但是，CopyImage 無法將 AMI 複製到另一個分割區。

如需 # AWS 割區的相關資訊，請參閱 IAM 使用者指南中的 [Amazon 資源名稱 \(ARN\)](#) 頁面上的分割區。

Warning

在分 AWS 區或 AWS 區域之間移動資料時，請確保您遵守所有適用的法律和業務要求，包括但不限於任何適用的政府法規和資料落地要求。

主題

- [使用案例](#)
- [AMI 存放和還原 API 如何運作](#)
- [限制](#)
- [成本](#)
- [保護您的 AMI](#)
- [使用 S3 存放和還原 AMI 的許可](#)
- [使用 AMI 存放和還原 API](#)
- [在 S3 中使用檔案路徑](#)

使用案例

使用存放和還原 API 來執行下列動作：

- [將 AMI 從一個 AWS 分區複製到另一個 AWS 分區](#)
- [製作 AMI 的封存複本](#)

將 AMI 從一個 AWS 分區複製到另一個 AWS 分區

透過使用 S3 儲存貯體儲存和還原 AMI，您可以將 AMI 從一個 AWS 分割區複製到另一個分割區，或從一個 AWS 區域複製到另一個區域。在下列範例中，您將 AMI 從主要商業磁碟分割複製到磁碟分 AWS GovCloud (US) 割，特別是從「us-east-2 區域」複製到「us-gov-east-1 區域」。

如果要將 AMI 從一個分割區複製到另一個分割區，請依照下列步驟執行：

- 透過使用 `CreateStoreImageTask` 將 AMI 存放在當前區域的 S3 儲存貯體中。在此範例中，S3 儲存貯體位於 `us-east-2`。如需命令範例，請參閱[將 AMI 存放在 S3 儲存貯體中](#)。
- 使用 `DescribeStoreImageTasks` 監控存放任務的進度。當任務完成時，該物件在 S3 儲存貯體中變得可見。如需命令範例，請參閱[描述 AMI 存放任務的進度](#)。
- 使用您選擇的程序，將存放的 AMI 物件複製到目標分割區中的 S3 儲存貯體。在此範例中，S3 儲存貯體位於 `us-gov-east-1`。

Note

因為每個分割區需要不同的 AWS 登入資料，因此無法將 S3 物件直接從一個分割區複製到另一個分割區。跨分割區複製 S3 物件的程序超出本文件的範圍。我們提供下列複製程序作為範例，但您必須使用符合安全性需求的複製程序。

- 若要跨分割區複製一個 AMI，複製程序非常簡單明瞭，具體如下：從來源儲存貯體[下載物件](#)至中繼主機 (例如 EC2 執行個體或筆記型電腦)，然後從中繼主機[上傳物件](#)至目標儲存貯體。針對程序的每個階段，請使用磁碟分割的 AWS 認證。
 - 為了更能夠持續使用，請考慮開發一個能管理複製的應用程式，最好可使用 S3 [分段下載和上傳](#)。
- 使用 `CreateRestoreImageTask` 從目標分割區中的 S3 儲存貯體還原 AMI。在此範例中，S3 儲存貯體位於 `us-gov-east-1`。如需命令範例，請參閱[從 S3 儲存貯體還原 AMI](#)。
 - 透過描述 AMI 來監控還原任務的進度，以查看其狀態何時變為可用。您也可以透過描述快照來監控組成還原 AMI 的快照進度百分比。

製作 AMI 的封存複本

您可以透過將其存放在 S3 儲存貯體中，來建立 AMI 的封存複本。如需命令範例，請參閱[將 AMI 存放在 S3 儲存貯體中](#)。

AMI 被打包為 S3 中的單一物件，並且所有 AMI 中繼資料 (不包括共享資訊) 作為存放之 AMI 的一部分保留。AMI 資料被壓縮為儲存程序的一部分。包含可輕鬆壓縮之資料的 AMI 會在 S3 中產生較小的物件。為了降低成本，您可以使用較低成本的 S3 儲存層。如需詳細資訊，請參閱 [Amazon S3 儲存類別](#) 和 [Amazon S3 定價](#)

AMI 存放和還原 API 如何運作

要使用 S3 存放和還原 AMI，請使用以下 API：

- [CreateStoreImageTask](#) – 將 AMI 存放在 S3 儲存貯體中
- [DescribeStoreImageTasks](#) – 提供 AMI 存放任務的進度
- [CreateRestoreImageTask](#) – 從 S3 儲存貯體還原 AMI

API 的運作方式

- [CreateStoreImageTask](#)
- [DescribeStoreImageTasks](#)
- [CreateRestoreImageTask](#)

CreateStoreImageTask

該 [CreateStoreImageTask](#) API 將 AMI 作為單個對象存儲在 S3 存儲桶中。

API 建立一個從 AMI 及其快照中讀取所有資料的任務，然後使用 [S3 分段上傳](#) 將資料存放在 S3 物件中。API 採用 AMI 的所有元件，包括大多數非特定區域的 AMI 中繼資料以及 AMI 中包含的所有 EBS 快照，並將其打包為 S3 中的單一物件。資料會在上傳過程中進行壓縮，以減少 S3 中使用的空間量，因此 S3 中的物件可能會小於 AMI 中快照大小的總和。

如果呼叫此 API 的帳戶可見 AMI 和快照標籤，則會將其保留。

S3 中的物件與 AMI 具有相同的 ID，但有 .bin 副檔名。下列資料也會作為 S3 中繼資料標籤存放於 S3 物件：AMI 名稱、AMI 描述、AMI 註冊日期、AMI 擁有者帳戶，以及存放操作的時間戳記。

完成任務所需的時間取決於 AMI 的大小。這也取決於有多少其他任務正在進行中，因為任務已排入佇列。您可以透過呼叫 [DescribeStoreImageTasks](#) API 來追蹤任務的進度。

進行中的所有 AMI 的大小總和限制為每個帳戶 600 GB EBS 快照資料。系統會拒絕更多的任務建立，除非進行中的任務數量小於限制數量。例如，如果目前存放了一個具有 100 GB 快照資料的 AMI，以及另一個具有 200 GB 快照資料的 AMI，則會接受另一個請求，因為進行中的總量為小於限制的 300 GB。但是，如果目前存放了具有 800 GB 快照資料的單一 AMI，則在任務完成之前，更多的任務將被拒絕。

DescribeStoreImageTasks

該 [DescribeStoreImageTasks](#) API 描述了 AMI 存儲任務的進度。您可以描述指定 AMI 的任務。如果未指定 AMI，您會取得過去 31 天內處理的所有存放映像任務的分頁清單。

對於每個 AMI 任務，回應會指示任務為 InProgress、Completed 或 Failed。對於任務 InProgress，回應會以百分比顯示預估進度。

任務會以相反的時間順序列出。

目前，只能檢視上個月的任務。

CreateRestoreImageTask

[CreateRestoreImageTask](#) API 會啟動從先前使用 [CreateStoreImageTask](#) 要求建立的 S3 物件還原 AMI 的工作。

還原任務可以在執行存放任務的相同或不同區域中執行。

將從中還原 AMI 物件的 S3 儲存貯體必須位於請求還原任務的相同區域。AMI 將在此區域還原。

AMI 會還原其中繼資料，例如名稱、描述和對應於存放 AMI 之值的區塊型儲存設備映射。此帳戶的區域中，AMI 的名稱必須是唯一的。如果您沒有提供名稱，新的 AMI 將取得與原始 AMI 相同的名稱。AMI 取得在還原過程中產生的新 AMI ID。

完成 AMI 還原任務所需的時間取決於 AMI 的大小。這也取決於有多少其他任務正在進行中，因為任務已排入佇列。您可以透過描述 AMI ([describe-images](#)) 或其 EBS 快照 ([describe-snapshots](#)) 來查看任務的進度。如果任務失敗，AMI 和快照集會轉為失敗狀態。

進行中的所有 AMI 的大小總和限制為每個帳戶 300 GB EBS 快照資料 (根據還原後的大小)。在進行中的任務小於限制之前，系統會拒絕更多的任務建立。

限制

- 要存儲 AMI，您 AWS 帳戶 必須擁有 AMI 及其快照，或者 AMI 及其快照必須 [直接與您的帳戶共享](#)。如果 AMI 僅 [公開共用](#)，則無法儲存 AMI。
- 使用這些 API 僅能存放 EBS 後端 AMI。
- 不支援半虛擬 (PV) AMI。
- 可存放的 AMI (壓縮之前) 大小限制為 5,000 GB。
- [存放映像](#) 請求的配額：600 GB 的進行中儲存工作 (快照資料)。
- [還原映像](#) 請求的配額：300 GB 的進行中還原工作 (快照資料)。
- 在存放任務期間，不得刪除快照，且執行存放的 IAM 主體必須具有快照的存取權，否則存放程序將會失敗。
- 您無法在同一個 S3 儲存貯體中建立 AMI 的多個複本。
- 存放在 S3 儲存貯體中的 AMI 無法使用其原始 AMI ID 還原。您可以透過使用 [AMI 別名](#) 來減輕這種情況。

- 目前只有使用 AWS Command Line Interface、AWS 開發套件和 Amazon EC2 API 才能支援存放區和還原 API。您無法使用 Amazon EC2 主控台存放和還原 AMI。

成本

當您使用 S3 存放和還原 AMI 時，需支付存放和還原 API 所使用之服務的費用，以及資料傳輸的費用。這些 API 使用 S3 和 EBS 直接 API (由這些 API 內部使用以存取快照資料)。如需詳細資訊，請參閱 [Amazon S3 定價](#) 和 [Amazon EBS 定價](#)。

保護您的 AMI

若要使用存放和還原 API，S3 儲存貯體和 AMI 必須位於相同的區域。請務必確保 S3 儲存貯體具有足夠的安全性來保護 AMI 的內容，而且只要 AMI 物件保留在儲存貯體中，安全性就必須要維護。如果無法完成，則不建議使用這些 API。確保不允許公開存取 S3 儲存貯體。我們建議針對存放 AMI 的 S3 儲存貯體啟用 [伺服器端加密](#)，儘管這不是必要的。

如需如何為 S3 儲存貯體設置適當安全性設定的相關資訊，請檢閱下列安全性主題：

- [封鎖對 Amazon S3 儲存體的公開存取權](#)
- [對 Amazon S3 儲存貯體設定預設伺服器端加密行為](#)
- [我可以使用的 S3 存儲桶政策來遵守僅限 S3 桶-ssl 請求的 AWS Config 規則？](#)
- [啟用 Amazon S3 伺服器存取記錄日誌](#)

將 AMI 快照複製到 S3 物件時，資料會透過 TLS 連線複製。您可以將 AMI 與加密的快照一起存放，但快照會在存放程序中解密。

使用 S3 存放和還原 AMI 的許可

若您的 IAM 主體將使用 Amazon S3 存放或還原 AMI，則您需要授予其所需的許可。

下列範例政策包含允許 IAM 主體執行存放和還原任務所需的所有動作。

您還可建立僅授予委託人存取特定資源的 IAM 政策。如需更多範例政策，請參閱 [IAM 使用者指南中的 AWS 資源存取管理](#)。

Note

若組成 AMI 的快照已進行加密，或者您的帳戶依預設啟用了加密功能，則您的 IAM 主體必須具有使用 KMS 金鑰的許可權限。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:DeleteObject",
        "s3:GetObject",
        "s3:ListBucket",
        "s3:PutObject",
        "s3:PutObjectTagging",
        "s3:AbortMultipartUpload",
        "ebs:CompleteSnapshot",
        "ebs:GetSnapshotBlock",
        "ebs:ListChangedBlocks",
        "ebs:ListSnapshotBlocks",
        "ebs:PutSnapshotBlock",
        "ebs:StartSnapshot",
        "ec2:CreateStoreImageTask",
        "ec2:DescribeStoreImageTasks",
        "ec2:CreateRestoreImageTask",
        "ec2:GetEbsEncryptionByDefault",
        "ec2:DescribeTags",
        "ec2:CreateTags"
      ],
      "Resource": "*"
    }
  ]
}
```

使用 AMI 存放和還原 API

主題

- [將 AMI 存放在 S3 儲存貯體中](#)
- [描述 AMI 存放任務的進度](#)
- [從 S3 儲存貯體還原 AMI](#)

將 AMI 存放在 S3 儲存貯體中

若要存放 AMI (AWS CLI)

使用 [create-store-image-task](#) 命令。指定 AMI 的 ID 以及要存放 AMI 的 S3 儲存貯體名稱。

```
aws ec2 create-store-image-task \  
  --image-id ami-1234567890abcdef0 \  
  --bucket myamibucket
```

預期的輸出結果

```
{  
  "ObjectKey": "ami-1234567890abcdef0.bin"  
}
```

描述 AMI 存放任務的進度

若要描述 AMI 存放任務的進度 (AWS CLI)

使用 [describe-store-image-tasks](#) 命令。

```
aws ec2 describe-store-image-tasks
```

預期的輸出結果

```
{  
  "AmiId": "ami-1234567890abcdef0",  
  "Bucket": "myamibucket",  
  "ProgressPercentage": 17,  
  "S3ObjectKey": "ami-1234567890abcdef0.bin",  
  "StoreTaskState": "InProgress",  
  "StoreTaskFailureReason": null,  
  "TaskStartTime": "2021-01-01T01:01:01.001Z"  
}
```

從 S3 儲存貯體還原 AMI

若要還原 AMI (AWS CLI)

使用 [create-restore-image-task](#) 命令。使用來自 `describe-store-image-tasks` 輸出的 `S3ObjectKey` 和 `Bucket` 值，指定 AMI 的物件索引鍵，以及將 AMI 複製到其中之 S3 儲存貯體的名稱。同時指定已還原 AMI 的名稱。此帳戶的區域中，AMI 的名稱必須是唯一的。

Note

還原的 AMI 會取得一個新的 AMI ID。

```
aws ec2 create-restore-image-task \  
  --object-key ami-1234567890abcdef0.bin \  
  --bucket myamibucket \  
  --name "New AMI Name"
```

預期的輸出結果

```
{  
  "ImageId": "ami-0eab20fe36f83e1a8"  
}
```

在 S3 中使用檔案路徑

您可以透過下列方式在存放和還原 AMI 時使用檔案路徑：

- 在 S3 中存放 AMI 時，可以將檔案路徑新增到儲存貯體名稱。在內部，系統會將路徑與儲存貯體名稱分隔開來，然後將路徑新增至產生用來存放 AMI 的物件金鑰。完整的物件路徑會顯示在 API 呼叫的回應中。
- 還原 AMI 時，由於物件索引鍵參數可用，因此可以將路徑新增至物件索引鍵值的開頭。

您可以在使用 AWS CLI 和 SDK 時使用檔案路徑。

範例：存放和還原 AMI (AWS CLI) 時使用檔案路徑

下列範例會先將 AMI 存放在 S3 中，檔案路徑會附加至儲存貯體名稱。然後，範例會從 S3 中還原 AMI，並在物件索引鍵參數前面加上檔案路徑。

1. 存放 AMI。對於 `--bucket`，指定儲存貯體名稱後面的檔案路徑，如下所示：

```
aws ec2 create-store-image-task \  
  --image-id ami-1234567890abcdef0 \  
  --bucket myamibucket/path1/path2
```

預期的輸出結果

```
{  
  "ObjectKey": "path1/path2/ami-1234567890abcdef0.bin"  
}
```

2. 還原 AMI。對於 `--object-key`，指定上一步驟中輸出的值，包括檔案路徑。

```
aws ec2 create-restore-image-task \  
  --object-key path1/path2/ami-1234567890abcdef0.bin \  
  --bucket myamibucket \  
  --name "New AMI Name"
```

棄用 AMI

您可以取代 AMI 來指出它已過期且不應該使用。您也可以為 AMI 指定未來的取代日期，指出 AMI 何時過期。例如，您可能會取代不再主動維護的 AMI，或者您可能會取代已被較新版本替代的 AMI。根據預設，已取代的 AMI 不會出現在 AMI 清單中，以防止新使用者使用 out-of-date AMI。但是，現有使用者和啟動服務 (例如啟動範本和 Auto Scaling 群組) 可以透過指定其 ID 來繼續使用已取代的 AMI。若要刪除 AMI，以便使用者和服務無法使用它，您必須[取消註冊](#)它。

AMI 被取代後：

- 對於 AMI 使用者，除非您指定其 ID 或指定必須出現已取代的 AMI，否則已取代的 AMI 不會出現在 [DescribeImages](#) API 呼叫中。AMI 擁有者繼續在 [DescribeImages](#) API 呼叫中看到已淘汰的 AMI。
- 對於 AMI 使用者，不能透過 EC2 主控台選取已取代的 AMI。例如，已取代的 AMI 不會出現在啟動執行個體精靈的 AMI 目錄中。AMI 擁有者繼續在 EC2 主控台中查看已取代的 AMI。
- 對於 AMI 使用者，如果您知道已取代的 AMI 的 ID，則可以使用 API、CLI 或 SDK 透過已取代的 AMI 來啟動執行個體。
- 啟動服務 (例如啟動範本和 Auto Scaling 群組) 可以繼續參考已取代的 AMI。
- 使用隨後被取代的 AMI 來啟動的 EC2 執行個體不會受到影響，而且可以停止、啟動和重新啟動。

您可以同時取代私有和公有 AMI。

您也可建立 Amazon Data Lifecycle Manager EBS 後端 AMI 政策，以自動化 EBS 後端 AMI 的取代。如需詳細資訊，請參閱[自動化 AMI 生命週期](#)。

Note

在預設情況下，所有公用 AMI 的棄用日期都設定為自 AMI 建立日期起兩年。您可將棄用日期設定為早於兩年。若要取消棄用日期或將棄用日期延後，您必須將[與指定的 AWS 帳戶共享 AMI](#)，使該 AMI 成為私有。

主題

- [成本](#)
- [限制](#)
- [棄用 AMI](#)
- [描述已取代的 AMI](#)
- [取消取代 AMI](#)

成本

當您取代 AMI 時，不會刪除 AMI。AMI 擁有者繼續為 AMI 的快照付費。要停止支付快照的費用，AMI 擁有者必須透過[取消註冊](#) AMI 來刪除它。

限制

- 若要取代 AMI，您必須是 AMI 的擁有者。

棄用 AMI

您可以在特定的日期和時間取代 AMI。您必須是 AMI 擁有者才能執行此程序。

Console**在特定日期取代 AMI**

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽窗格中，選擇 AMI。
3. 從篩選條件列中，選擇 Owned by me (由我擁有)。
4. 選取 AMI，然後依序選取 Actions (動作)、Manage AMI Deprecation (管理 AMI 取代)。您可以選取多個 AMI，一次設定多個 AMI 的相同取代日期。

5. 選取 Enable (啟用) 核取方塊，然後輸入取代日期和時間。

棄用日期的上限是從現在起 10 年，但公用 AMI 除外，其上限是自建立日期起 2 年。您無法指定過去的日期。

6. 選擇儲存。

AWS CLI

在特定日期取代 AMI

使用 [enable-image-deprecation](#) 命令。指定 AMI 的 ID 以及取代 AMI 的日期和時間。如果您指定秒數值，Amazon EC2 會將秒數四捨五入到最接近的分鐘。

deprecate-at 的上限是從現在起 10 年，但公用 AMI 除外，其上限是自建立日期起 2 年。您無法指定過去的日期。

```
aws ec2 enable-image-deprecation \  
  --image-id ami-1234567890abcdef0 \  
  --deprecate-at "2021-10-15T13:17:12.000Z"
```

預期的輸出結果

```
{  
  "Return": "true"  
}
```

檢查上次使用 AMI 的時間

LastLaunchedTime 是一個時間戳記，表示您的 AMI 上次用於啟動執行個體的時間。最近未用於啟動執行個體的 AMI 可能就是待移除或[取消註冊](#)的項目。

Note

- 使用 AMI 啟動執行個體時，在報告該使用情況之前會有 24 小時延遲。
- lastLaunchedTime 資料從 2017 年 4 月開始提供。

Console

檢視 AMI 的上次啟動時間

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽窗格中，選擇 AMI。
3. 從篩選條件列中，選擇 Owned by me (由我擁有)。
4. 選取 AMI，然後檢查 Last launched time (上次啟動時間) 欄位 (若您選取 AMI 旁的核取方塊，其位於 Details (詳細資料) 索引標籤)。此欄位顯示上次使用 AMI 啟動執行個體的日期和時間。

AWS CLI

檢視 AMI 的上次啟動時間

執行 [describe-image-attribute](#) 命令並指定 `--attribute lastLaunchedTime`。您必須是 AMI 擁有者才能執行此命令。

```
aws ec2 describe-image-attribute \  
  --image-id ami-1234567890example \  
  --attribute lastLaunchedTime
```

範例輸出

```
{  
  "LastLaunchedTime": {  
    "Value": "2022-02-10T02:03:18Z"  
  },  
  "ImageId": "ami-1234567890example",  
}
```

描述已取代的 AMI

您可以檢視 AMI 的取代日期和時間，並依取代日期篩選所有 AMI。您也可以使用 AWS CLI 來描述所有已被棄用的 AMI，其中的取代日期是過去的。

Console

檢視 AMI 的取代日期

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽器中，選取 AMI，然後選取該 AMI。
3. 檢查 Deprecation time (取代時間) 欄位 (若您選取 AMI 旁的核取方塊，其位於 Details (詳細資料) 索引標籤)。該欄位顯示 AMI 的取代日期和時間。若欄位為空的，則 AMI 不會遭到取代。

依取代日期篩選 AMI

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽窗格中，選擇 AMI。
3. 從篩選條件列中，選擇 Owned by me (由我擁有) 或 Private images (私有映像) (私有映像包括與您共享以及您擁有的 AMI)。
4. 在搜尋列中，輸入 **Deprecation time** (當您輸入字母時，Deprecation time (取代時間) 篩選條件隨即顯示)，隨後選擇一個運算子及一個日期和時間。

AWS CLI

當您使用 [describe-images](#) 命令描述所有 AMI 時，結果會有所不同，這取決於您是 AMI 使用者還是 AMI 擁有者。

- 如果您是 AMI 使用者：

根據預設，當您使用 [describe-images](#) 命令時，不屬於您但是與您共用的已取代的 AMI 不會出現在結果中。這是因為預設值為 `--no-include-deprecated`。若要在結果中包含已取代的 AMI，您必須指定 `--include-deprecated` 參數。

- 如果您是 AMI 擁有者：

當您使用 [describe-images](#) 命令描述所有 AMI 時，您擁有的所有 AMI (包括已取代的 AMI) 都會出現在結果中。您不需要指定 `--include-deprecated` 參數。此外，您無法使用 `--no-include-deprecated` 從結果中排除您擁有的已取代的 AMI。

如果已取代 AMI，則 `DeprecationTime` 欄位會出現在結果中。

Note

已取代的 AMI 是一種其取代日期在過去的 AMI。如果您已將取代日期設定為未來的日期，則尚未取代 AMI。

在描述所有 AMI 時包含所有已取代的 AMI

使用 [describe-images](#) 命令並指定 `--include-deprecated` 參數，即可在結果中包含不為您擁有的所有已棄用 AMI。

```
aws ec2 describe-images \  
  --region us-east-1 \  
  --owners 123456example \  
  --include-deprecated
```

描述 AMI 的取代日期

使用 [describe-images](#) 命令並指定 AMI 的 ID。

請注意，如果您同時指定 `--no-include-deprecated` 和 AMI ID，則會在結果中傳回已取代的 AMI。

```
aws ec2 describe-images \  
  --region us-east-1 \  
  --image-ids ami-1234567890EXAMPLE
```

預期的輸出結果

`DeprecationTime` 欄位會顯示 AMI 被設定為已取代的日期。如果 AMI 未設定為已取代，`DeprecationTime` 欄位不會出現在輸出中。

```
{  
  "Images": [  
    {  
      "VirtualizationType": "hvm",  
      "Description": "Provided by Red Hat, Inc.",  
      "PlatformDetails": "Red Hat Enterprise Linux",  
      "EnaSupport": true,  
      "Hypervisor": "xen",
```



```

    "State": "available",
    "SriovNetSupport": "simple",
    "ImageId": "ami-1234567890EXAMPLE",
    "DeprecationTime": "2021-05-10T13:17:12.000Z"
    "UsageOperation": "RunInstances:0010",
    "BlockDeviceMappings": [
      {
        "DeviceName": "/dev/sda1",
        "Ebs": {
          "SnapshotId": "snap-111222333444aaabb",
          "DeleteOnTermination": true,
          "VolumeType": "gp2",
          "VolumeSize": 10,
          "Encrypted": false
        }
      }
    ],
    "Architecture": "x86_64",
    "ImageLocation": "123456789012/RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-
GP2",
    "RootDeviceType": "ebs",
    "OwnerId": "123456789012",
    "RootDeviceName": "/dev/sda1",
    "CreationDate": "2019-05-10T13:17:12.000Z",
    "Public": true,
    "ImageType": "machine",
    "Name": "RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-GP2"
  }
]
}

```

取消取代 AMI

您可以取消取代 AMI，此會從 Deprecation time (取代時間) 欄位 (主控台) 或從 [describe-images](#) 輸出 (DeprecationTime) 的 AWS CLI 欄位移除日期和時間。您必須是 AMI 擁有者才能執行此程序。

Console

取消取代 AMI

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽窗格中，選擇 AMI。

3. 從篩選條件列中，選擇 Owned by me (由我擁有)。
4. 選取 AMI，然後依序選取 Actions (動作)、Manage AMI Deprecation (管理 AMI 取代)。您可以選取多個 AMI，一次取消多個 AMI 的取代。
5. 清除 Enable (啟用) 核取方塊，然後選擇 Save (儲存)。

AWS CLI

取消取代 AMI

使用 [disable-image-deprecation](#) 命令並指定 AMI 的 ID。

```
aws ec2 disable-image-deprecation \  
  --image-id ami-1234567890abcdef0
```

預期的輸出結果

```
{  
  "Return": "true"  
}
```

停用 AMI

可以停用 AMI 以防止其用於執行個體啟動。無法從已停用的 AMI 中啟動新執行個體。可以重新啟用已停用的 AMI，以便在執行個體啟動時再次使用它。

Warning

停用 AMI 會移除其所有啟動權限。

停用 AMI 時：

- AMI 的狀態會變更為 disabled。
- 無法共用已停用的 AMI。如果 AMI 為公有或先前被共用，則會將其設為私有。如果 AMI 與 AWS 帳戶、組織或組織單位共用，則他們將無法存取已停用 AMI。
- 依預設，已停用的 AMI 不會出現在 [DescribeImages](#) API 呼叫中。

- 已停用的 AMI 不會出現在我所擁有主控台篩選條件下。若要尋找已停用的 AMI，請使用已停用的映像主控台篩選條件。
- 對於 EC2 主控台中的執行個體啟動，不可選擇已停用的 AMI。例如，在啟動執行個體精靈中或建立啟動範本時，已停用的 AMI 不會出現在 AMI 目錄中。
- 啟動服務 (例如啟動範本和 Auto Scaling 群組) 可以繼續參考已停用的 AMI。從已停用的 AMI 中進行後續執行個體啟動將會失敗，因此建議您更新啟動範本和 Auto Scaling 群組以僅參考可用的 AMI。
- 先前使用隨後被停用的 AMI 來啟動的 EC2 執行個體不會受到影響，而且可以停止、啟動和重新啟動。
- 您無法刪除與已停用的 AMI 相關聯的快照。嘗試刪除關聯的快照會導致 `snapshot is currently in use` 錯誤。

重新啟用 AMI 時：

- AMI 的狀態會變更為 `available`，而且可用來啟動執行個體。
- 可以共用 AMI。
- 由於停用 AMI 而無法對其進行存取的 AWS 帳戶、組織和組織單位不會自動重新獲得存取權，但可以再次與他們共用 AMI。

可以同時停用私有和公有 AMI。

主題

- [成本](#)
- [必要條件](#)
- [所需的 IAM 許可](#)
- [停用 AMI](#)
- [描述已停用的 AMI](#)
- [重新啟用已停用的 AMI](#)

成本

當您停用 AMI 時，不會刪除 AMI。如果 AMI 是 EBS 後端 AMI，您會繼續支付 AMI 的 EBS 快照。如果想要保留 AMI，可以透過封存快照來降低儲存成本。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的封存 Amazon EBS 快照](#)。如果不想保留 AMI 及其快照，必須取消註冊 AMI 並刪除快照。如需詳細資訊，請參閱 [刪除與您的 Amazon EBS-backed AMI 聯的資源](#)。

必要條件

若要停用或重新啟用 AMI，您必須是 AMI 的擁有者。

所需的 IAM 許可

若要停用和重新啟用 AMI，您必須具有下列 IAM 許可：

- `ec2:DisableImage`
- `ec2:EnableImage`

停用 AMI

您可以使用 EC2 主控台或 AWS Command Line Interface (AWS CLI) 來停用 AMI。您必須是 AMI 擁有者才能執行此程序。

Console

停用 AMI

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽窗格中選擇 AMI。
3. 從篩選條件列中，選擇 Owned by me (由我擁有)。
4. 選取 AMI，然後選擇動作、停用 AMI。可以選取多個 AMI，以便一次停用。
5. 在停用 AMI 視窗中，選擇停用 AMI。

AWS CLI

停用 AMI

使用 [disable-image](#) 命令並指定 AMI 的 ID。

```
aws ec2 disable-image --image-id ami-1234567890abcdef0
```

預期的輸出結果

```
{  
  "Return": "true"
```

```
}
```

描述已停用的 AMI

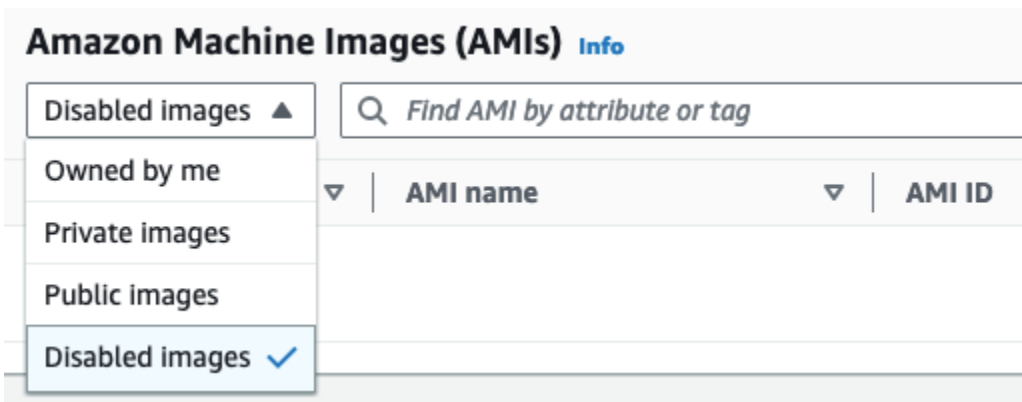
可以使用 EC2 主控台或 AWS CLI 來檢視已停用的 AMI。

您必須是 AMI 擁有者，才能檢視已停用的 AMI。因為已停用的 AMI 會設為私有，如果您不是擁有者，就無法檢視已停用的 AMI。

Console

若要檢視已停用的 AMI

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽窗格中選擇 AMI。
3. 在篩選列中，選擇 [停用的影像]。



AWS CLI

依預設，當您使用 [describe-images](#) 命令來描述所有 AMI 時，已停用的 AMI 不會顯示在結果中。這是因為預設值為 `--no-include-disabled`。若要在結果中包含已停用的 AMI，必須指定 `--include-disabled` 參數。

在描述所有 AMI 時包含所有已停用的 AMI

使用 [describe-images](#) 命令並指定 `--include-disabled` 參數來擷取已停用的 AMI 以及所有其他 AMI。或者，指定 `--owners self` 僅擷取您擁有的 AMI。

```
aws ec2 describe-images \
```

```
--region us-east-1 \  
--owners self  
--include-disabled
```

如果指定已停用的 AMI 的 ID，但未指定 `--include-disabled`，則會在結果中傳回已停用的 AMI。

```
aws ec2 describe-images \  
  --region us-east-1 \  
  --image-ids ami-1234567890EXAMPLE
```

僅擷取已停用的 AMI

指定 `--filters Name=state,Values=disabled`。還必須指定 `--include-disabled`，否則會收到錯誤。

```
aws ec2 describe-images \  
  --include-disabled \  
  --filters Name=state,Values=disabled
```

範例輸出

State 欄位會顯示 AMI 的狀態。disabled 表示 AMI 已停用。

```
{  
  "Images": [  
    {  
      "VirtualizationType": "hvm",  
      "Description": "Provided by Red Hat, Inc.",  
      "PlatformDetails": "Red Hat Enterprise Linux",  
      "EnaSupport": true,  
      "Hypervisor": "xen",  
      "State": "disabled",  
      "SriovNetSupport": "simple",  
      "ImageId": "ami-1234567890EXAMPLE",  
      "DeprecationTime": "2023-05-10T13:17:12.000Z",  
      "UsageOperation": "RunInstances:0010",  
      "BlockDeviceMappings": [  
        {  
          "DeviceName": "/dev/sda1",  
          "Ebs": {  
            "SnapshotId": "snap-111222333444aaabb",
```

```
        "DeleteOnTermination": true,  
        "VolumeType": "gp2",  
        "VolumeSize": 10,  
        "Encrypted": false  
      }  
    }  
  ],  
  "Architecture": "x86_64",  
  "ImageLocation": "123456789012/RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-  
GP2",  
  "RootDeviceType": "ebs",  
  "OwnerId": "123456789012",  
  "RootDeviceName": "/dev/sda1",  
  "CreationDate": "2019-05-10T13:17:12.000Z",  
  "Public": false,  
  "ImageType": "machine",  
  "Name": "RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-GP2"  
}  
]  
}
```

重新啟用已停用的 AMI

可以重新啟用已停用的 AMI。您必須是 AMI 擁有者才能執行此程序。

Console

重新啟用已停用的 AMI

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽窗格中選擇 AMI。
3. 在篩選列中，選擇 [停用的影像]。
4. 選取 AMI，然後選擇動作、啟用 AMI。可以選取多個 AMI，一次重新啟用多個 AMI。
5. 在啟用 AMI 視窗中，選擇啟用。

AWS CLI

重新啟用已停用的 AMI

使用 [enable-image](#) 命令並指定 AMI 的 ID。

```
aws ec2 enable-image --image-id ami-1234567890abcdef0
```

預期的輸出結果

```
{  
  "Return": "true"  
}
```

封存 AMI 快照

可封存與已停用的 EBS 後端 AMI 相關聯的快照。這有助於降低與需要長期保留但很少使用之 AMI 相關聯的儲存成本。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的封存 Amazon EBS 快照](#)。

封存與 AMI 相關聯的快照

1. [停用 AMI](#)。
2. [封存快照](#)。

當 AMI 已停用且其相關快照已封存時，您無法進行使用。

還原已停用的 AMI 及已封存的快照以供使用

1. [還原與 AMI 關聯的已封存快照](#)。
2. [啟用 AMI](#)。

取消註冊 (刪除) AMI

當您取消註冊 AMI 時，Amazon EC2 將其永久刪除。一旦取消註冊，您就無法使用 AMI 啟動新的執行個體。您可以考慮在使用完 AMI 後取消註冊 AMI。

為了防止 AMI 意外或惡意註銷，您可以開啟取[消註冊](#)保護。如果您不小心取消註冊 EBS 後的 AMI，只有在永久刪除之前在允許的時間內還原該 AMI 時，才可以使用[資源回收筒](#)還原該 AMI。

取消註冊 AMI 不會影響從 AMI 啟動的任何執行個體。您可以繼續使用這些執行個體。取消註冊 AMI 也不會影響 AMI 建立程序期間建立的任何快照集。您將繼續支付這些執行個體的使用費用，以及快照的儲存成本。因此，為避免產生不必要的費用，建議您終止任何執行個體，並刪除不需要的快照。如需詳細資訊，請參閱 [避免未使用資源的成本](#)。

目錄

- [考量事項](#)
- [取消註冊 AMI](#)
- [檢查上次使用 AMI 的時間](#)
- [保護 AMI 免於取消註冊](#)
- [避免未使用資源的成本](#)

考量事項

- 您無法取消註冊不是您帳戶擁有的 AMI。
- 您無法使用 Amazon EC2 取消註冊由 AWS Backup 服務管理的 AMI。請改用 AWS Backup 來刪除備份儲存庫中對應的復原點。如需詳細資訊，請參閱《AWS Backup 開發人員指南》中的刪除備份。

取消註冊 AMI

使用下列任一方法來取消註冊 EBS 支援的 AMI 或 instance store-backed AMI。

Tip

為避免產生不必要的成本，您應該刪除不需要的任何資源。例如，對於 EBS 支援的 AMI，如果您不需要與已取消註冊 AMI 相關聯的快照，則應將其刪除。如需詳細資訊，請參閱 [避免未使用資源的成本](#)。

Console

若要取消註冊 AMI

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 AMIs (AMI)。
3. 在篩選列中，選擇 [我擁有] 以列出可用的 AMI，或選擇 [停用的影像] 以列出已停用的 AMI。
4. 選取要取消註冊的 AMI。
5. 選擇 Actions (動作)、Deregister AMI (取消註冊 AMI)。

- 當系統提示您進行確認時，請選擇「取消註冊 AMI」。

主控台從清單移除 AMI 可能需要幾分鐘的時間。選擇 Refresh (重新整理) 以重新整理狀態。

AWS CLI

若要取消註冊 AMI

使用取消註冊影像指令，並指定要取消註冊的 AMI ID。

```
aws ec2 deregister-image --image-id ami-0123456789example
```

Powershell

若要取消註冊 AMI

使用指 [Unregister-EC2Image](#) 令程式並指定要取消註冊之 AMI 的識別碼。

```
Unregister-EC2Image -ImageId ami-0123456789example
```

檢查上次使用 AMI 的時間

LastLaunchedTime 是一個時間戳記，表示您的 AMI 上次用於啟動執行個體的時間。最近未用於啟動執行個體的 AMI 可能就是取消註冊或 [待移除](#) 的項目。

Note

- 使用 AMI 啟動執行個體時，在報告該使用情況之前會有 24 小時延遲。
- lastLaunchedTime 資料從 2017 年 4 月開始提供。

Console

檢視 AMI 的上次啟動時間

- 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
- 在左側導覽窗格中選擇 AMI。

3. 從篩選條件列中，選擇 Owned by me (由我擁有)。
4. 選取 AMI，然後檢查 Last launched time (上次啟動時間) 欄位 (若您選取 AMI 旁的核取方塊，其位於 Details (詳細資料) 索引標籤)。此欄位顯示上次使用 AMI 啟動執行個體的日期和時間。

AWS CLI

您可以使用描述[影像](#)或[描述-影像屬性](#)命令來檢視 AMI 的上次啟動時間。

使用描述影像檢視 AMI 的上次啟動時間

使用 [describe-images](#) 命令並指定 AMI 的 ID。

```
aws ec2 describe-images --image-id ami-0123456789example
```

範例輸出

Note

此 LastLaunchedTime 欄位只會出現在您擁有的 AMI 的輸出中。

```
{
  "Images": [
    {
      ...
      "LastLaunchedTime": {
        "Value": "2024-04-02T02:03:18Z"
      },
      ...
    }
  ]
}
```

檢視 AMI 的上次啟動時間

使用[描述-圖像屬性](#)命令並指定。--attribute lastLaunchedTime 您必須是 AMI 的擁有者才能執行此命令。

```
aws ec2 describe-image-attribute \
```

```
--image-id ami-0123456789example \  
--attribute lastLaunchedTime
```

範例輸出

```
{  
  "ImageId": "ami-1234567890example",  
  "LastLaunchedTime": {  
    "Value": "2022-02-10T02:03:18Z"  
  }  
}
```

保護 AMI 免於取消註冊

您可以在 AMI 上開啟取消註冊保護，以防止意外或惡意刪除。當您開啟取消註冊保護時，任何使用者都無法取消註冊 AMI，不論其 IAM 許可為何。如果您想要取消註冊 AMI，必須先關閉其上的取消註冊保護。

當您在 AMI 上開啟取消註冊保護時，您可以選擇包含 24 小時的冷卻時間。這個冷卻時間是指停止註冊保護在您關閉後保持有效的時間。在此冷卻時間內，AMI 無法取消註冊。當冷卻時間結束時，AMI 可以撤銷註冊。

在所有現有和新的 AMI 上，預設會關閉取消註冊保護。

開啟取消註冊保護

使用下列任何一種方法來開啟 AMI 的取消註冊保護。要做到這一點，你必須是 AMI 的所有者。

Console

在 AMI 上開啟取消註冊保護

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 AMIs (AMI)。
3. 在篩選列中，選擇 [我擁有] 以列出可用的 AMI，或選擇 [停用的影像] 以列出已停用的 AMI。
4. 選取您要開啟取消註冊保護的 AMI，然後選擇 [動作]、[管理 AMI 取消註冊保護]。
5. 在 [管理 AMI 取消註冊保護] 對話方塊中，您可以開啟取消註冊保護 (無論是否有冷卻時間)。請選擇下列其中一個選項：

- 啟用 24 小時的冷卻時間 — 使用冷卻時間後，當註銷保護關閉時，AMI 無法在 24 小時內取消註冊。
- 無需冷卻時間即可啟用 — 在沒有冷卻時間的情況下，當停用註冊保護關閉時，AMI 可以立即取消註冊。

6. 選擇儲存。

AWS CLI

在 AMI 上開啟取消註冊保護

使用 [啟用映像-取消註冊保護命令並指定 AMI ID](#)。若要包含可選的 24 小時冷卻時間，請將 `--with-cooldown` 設定為 `true`。若要排除冷卻時間，請省略 `--with-cooldown` 參數。

```
aws ec2 enable-image-deregistration-protection \  
  --image-id ami-0123456789example \  
  --with-cooldown true
```

關閉取消註冊保護

使用下列任何一種方法來關閉 AMI 的取消註冊保護。要做到這一點，你必須是 AMI 的所有者。

Note

如果您選擇在啟用 AMI 的取消註冊保護時包含 24 小時的冷卻時間，則當您關閉註銷保護時，您將無法立即取消註冊 AMI。冷卻時間是 24 小時的時間段，即使在您關閉註冊保護之後，註銷保護仍然生效。在此冷卻時間內，AMI 無法取消註冊。冷卻時間結束後，AMI 可以撤銷註冊。

Console

關閉 AMI 的取消註冊保護

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 AMIs (AMI)。
3. 在篩選列中，選擇 [我擁有] 以列出可用的 AMI，或選擇 [停用的影像] 以列出已停用的 AMI。
4. 選取 AMI 以關閉取消註冊保護，然後選擇 [動作]、[管理 AMI 取消註冊保護]。

5. 在 [管理 AMI 取消註冊保護] 對話方塊中，選擇 [停用]。
6. 選擇儲存。

AWS CLI

關閉 AMI 的取消註冊保護

使用 [停用影像-取消註冊保護命令](#)，並指定 AMI ID。

```
aws ec2 disable-image-deregistration-protection --image-id ami-0123456789example
```

避免未使用資源的成本

取消註冊 AMI 時，不會刪除與 AMI 相關聯的資源。這些資源包括 EBS 支援 AMI 的快照，以及適用於執行個體商店支援 AMI 的 Amazon S3 中的檔案。當您取消註冊 AMI 時，也不會終止或停止從 AMI 啟動的任何執行個體。

您將繼續支付儲存快照和檔案的成本，而且任何執行中的執行個體都會產生費用。如需詳細資訊，請參閱 [收費方式](#)。

為避免產生這些類型的不必要成本，建議您刪除不需要的任何資源。

若要判斷您的 AMI 是 EBS 支援還是執行個體存放區支援，請參閱 [判定 AMI 的根設備類型](#)

刪除與您的 Amazon EBS-backed AMI 聯的資源

使用下列任何一種方法刪除與 EBS 支援 AMI 相關聯的資源。

Console

若要刪除與 EBS 支援 AMI 相關聯的資源

1. [取消註冊 AMI。](#)

記下 AMI ID — 這可協助您在下一個步驟中找到要刪除的快照。

2. [刪除不需要的快照。](#)

關聯 AMI 的識別碼會顯示在「快照」畫面的「說明」欄中。

3. [終止您不需要的執行個體。](#)

AWS CLI

若要刪除與 EBS 支援 AMI 相關聯的資源

1. 使用取消註冊影像指令來取[消註冊](#) AMI。

```
aws ec2 deregister-image --image-id ami-0123456789example
```

2. 使用刪除快照指令[刪除不需要的快照](#)。

```
aws ec2 delete-snapshot --snapshot-id snap-0123456789example
```

3. 通過使用終端實例命令[終止不需要的實例](#)。

```
aws ec2 terminate-instances --instance-ids i-0123456789example
```

PowerShell

若要刪除與 EBS 支援 AMI 相關聯的資源

1. 使用[Unregister-EC2Image](#)指令程式取消註冊 AMI。

```
Unregister-EC2Image -ImageId ami-0123456789example
```

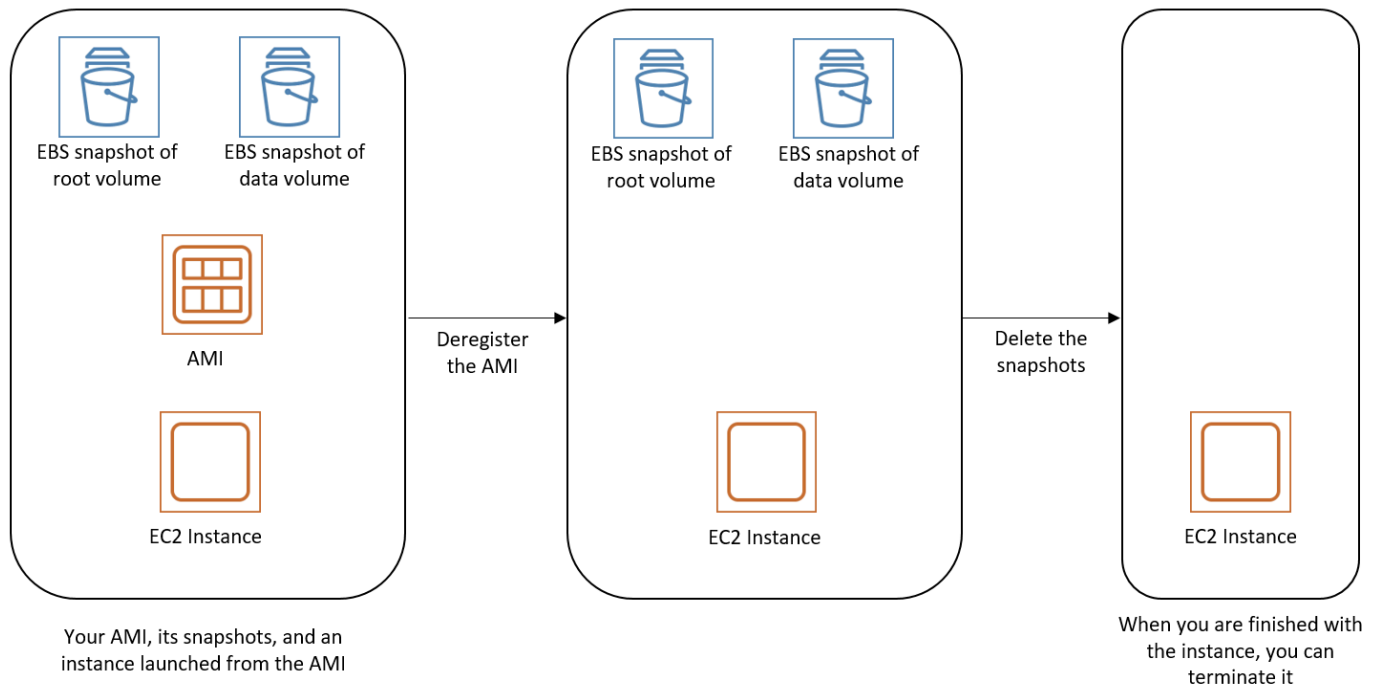
2. 使用[Remove-EC2Snapshot](#)指令程式刪除不需要的快照。

```
Remove-EC2Snapshot -SnapshotId snap-0123456789example
```

3. 使用[Remove-EC2Instance](#)指令程式終止不需要的執行個體。

```
Remove-EC2Instance -InstanceId i-0123456789example
```

下圖說明刪除與 EBS 支援 AMI 相關聯之資源的流程。



刪除與 instance store-backed AMI 相關聯的資源

使用下列方法刪除與 instance store-backed AMI 相關聯的資源。

刪除與 instance store-backed AMI 相關聯的資源

1. 使用取消註冊影像指令來取[消註冊](#) AMI。

```
aws ec2 deregister-image --image-id ami-0123456789example
```

2. 使用 [ec2-delete-bundle](#) (AMI 工具) 命令刪除 Amazon S3 中的套裝軟體。

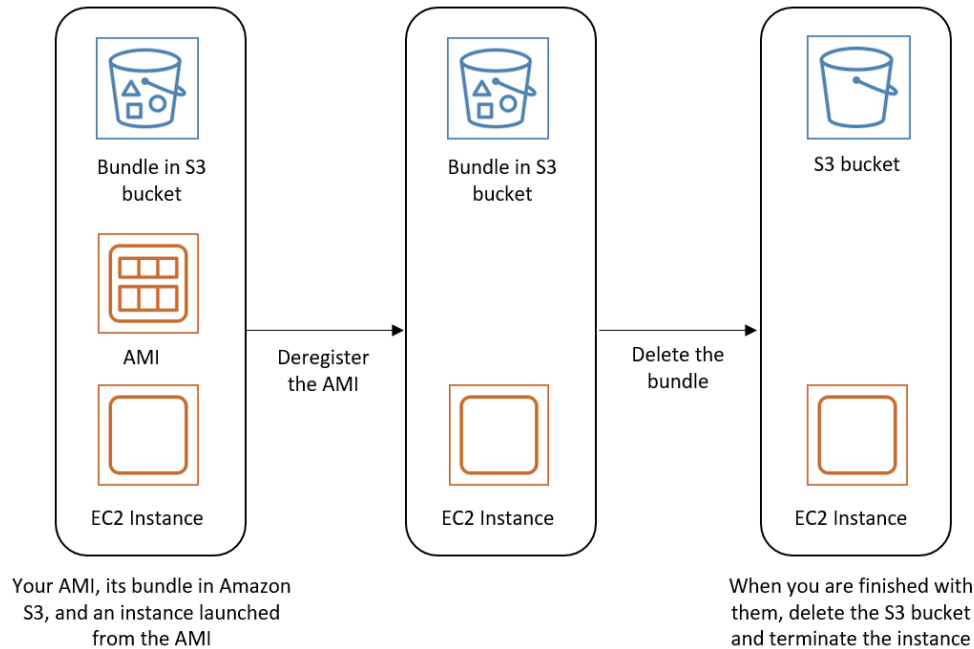
```
ec2-delete-bundle -b myawsbucket/myami -a your_access_key_id -s your_secret_access_key -p image
```

3. 通過使用終端實例命令終止不需要的實例。

```
aws ec2 terminate-instances --instance-ids i-0123456789example
```

4. 如果您已完成將服務包上傳到的 Amazon S3 儲存貯體，則可以刪除該儲存貯體。若要刪除 Amazon S3 儲存貯體，請開啟 Amazon S3 主控台，選取儲存貯體，然後依序選取 Actions (動作) 和 Delete (刪除)。

下圖說明刪除與 instance store-backed AMI 相關聯之資源的流程。



自動化 EBS 後端 AMI 生命週期

您可以使用 Amazon Data Lifecycle Manager 來自動建立、保留、複製、取代和取消註冊 Amazon EBS 後端 AMI 及其備份快照。如需詳細資訊，請參閱 [Amazon Data Lifecycle Manager](#)。

對 EBS 後端 AMI 使用加密

以 Amazon EBS 快照為後端的 AMI 可以利用 Amazon EBS 加密。資料和根磁碟區的快照可以經過加密再連接至 AMI。您可以啟動執行個體並複製映像，而且包括完整 EBS 加密支援在內。所有可用的區域都支援這些作業的 AWS KMS 加密參數。

具有 EBS 磁碟區的 EC2 執行個體從 AMIs 啟動的方式，如同其他執行個體一樣。此外，從以未加密 EBS 快照為後端的 AMI 啟動執行個體時，您可以在啟動期間加密部分或所有磁碟區。

與 EBS 磁碟區一樣，AMI 中的快照可以依預設加密 AWS KMS key，也可以加密至您指定的客戶管理金鑰。無論是哪種情況，您都必須擁有許可才能使用所選取的 KMS 金鑰。

具有加密快照的 AMI 可以跨 AWS 帳戶共用。如需詳細資訊，請參閱 [共享 AMI](#)。

對 EBS 後端 AMI 加密主題

- [執行個體啟動案例](#)

- [映像複製案例](#)

執行個體啟動案例

Amazon EC2 執行個體是透過區塊裝置對應提供的參數的RunInstances動作，透過 AWS Management Console 或直接使用 Amazon EC2 API 或 CLI 來啟動 Amazon EC2 執行個體。如需詳細資訊，請參閱 [區塊型儲存設備對應](#)。如需控制區塊裝置對應的範例 AWS CLI，請參閱[啟動、列出和終止 EC2 執行個體](#)。

根據預設，如果沒有明確的加密參數，從 AMI 的來源快照還原 EBS 磁碟區時，RunInstances 動作會維護快照的現有加密狀態。如果預設啟用加密，則從 AMI 建立的所有磁碟區 (無論是從加密或未加密的快照) 都會加密。如果預設未啟用加密，則執行個體會維持 AMI 的加密狀態。

您也可以啟動執行個體並提供加密參數，以同時將新的加密狀態套用到產生的磁碟區。因此，可看到有下列行為：

在無加密參數的情況下啟動

- 未加密快照會還原到未加密的磁碟區，除非預設加密已啟用，在此情況下，所有新建立的磁碟區都加密。
- 您擁有的加密快照會還原到以相同 KMS 金鑰所加密的磁碟區。
- 您不擁有的加密快照 (例如，AMI 會與您共用) 會還原至由您 AWS 帳戶預設 KMS 金鑰加密的磁碟區。

您可以提供加密參數來覆寫預設行為。可用的參數是 Encrypted 和 KmsKeyId。只設定 Encrypted 參數會產生下列行為：

已設定 Encrypted 但未指定 KmsKeyId 的執行個體啟動行為

- 未加密快照會還原到以您的 AWS 帳戶的預設 KMS 金鑰所加密的 EBS 磁碟區。
- 您擁有的加密快照會還原到以相同 KMS 金鑰所加密的 EBS 磁碟區。(換言之，Encrypted 參數沒有效果。)
- 您不擁有的加密快照 (也就是說，AMI 會與您共用) 還原至由您 AWS 帳戶預設 KMS 金鑰加密的磁碟區。(換言之，Encrypted 參數沒有效果。)

同時設定 Encrypted 和 KmsKeyId 參數可讓您為加密操作指定非預設 KMS 金鑰。產生下列行為：

Encrypted 和 KmsKeyId 都設定的執行個體

- 未加密快照會還原到以指定的 KMS 金鑰 所加密的 EBS 磁碟區。
- 加密快照會還原到不是以原始 KMS 金鑰 所加密的 EBS 磁碟區，而是以指定的 KMS 金鑰 所加密的 EBS 磁碟區。

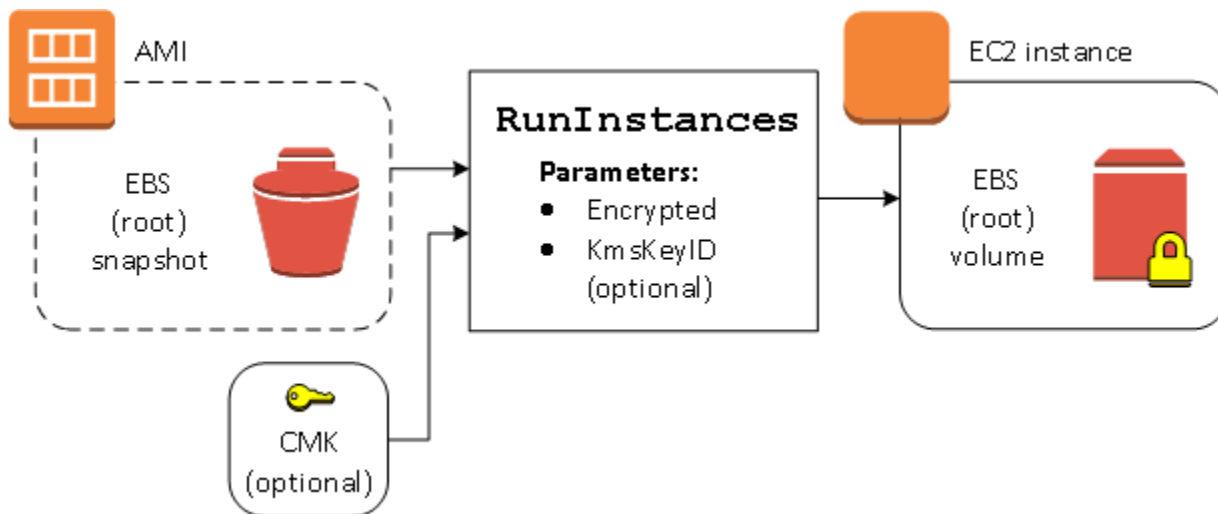
提交 KmsKeyId 但未同時設定 Encrypted 參數會導致錯誤。

下列各節提供從 AMI 使用非預設加密參數來啟動執行個體的範例。在每個案例中，提供給 RunInstances 動作的參數會導致從快照還原磁碟區期間變更加密狀態。

如需使用主控台從 AMI 啟動執行個體的資訊，請參閱 [啟動您的執行個體](#)。

啟動期間加密磁碟區

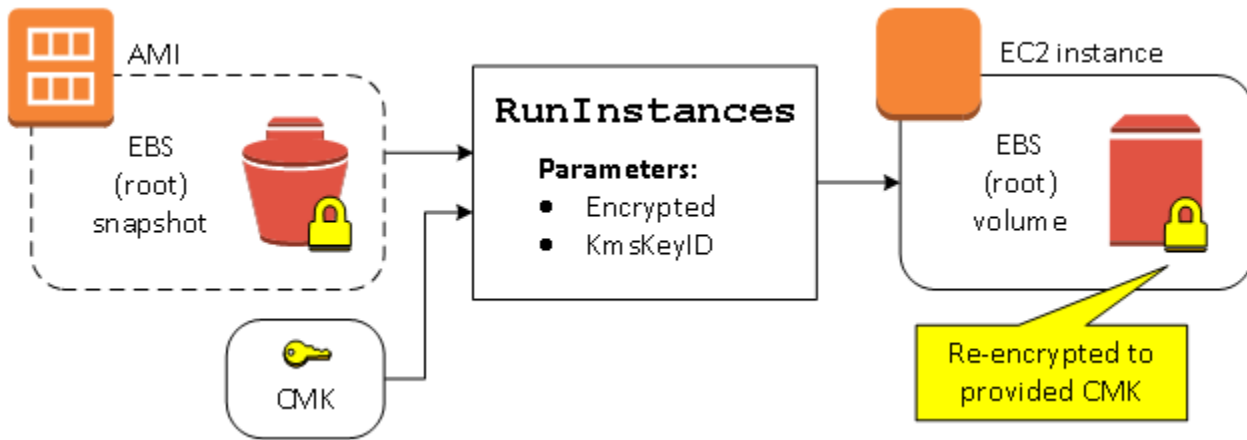
在此範例中，以未加密快照為後端的 AMI 用於啟動具有加密 EBS 磁碟區的 EC2 執行個體。



單獨 Encrypted 參數會導致加密此執行個體的磁碟區。提供 KmsKeyId 參數為選擇性。如果未指定 KMS 金鑰識別碼，則會使用 AWS 帳戶的預設 KMS 金鑰來加密磁碟區。若要以您擁有的不同 KMS 金鑰來加密磁碟區，請提供 KmsKeyId 參數。

啟動期間重新加密磁碟區

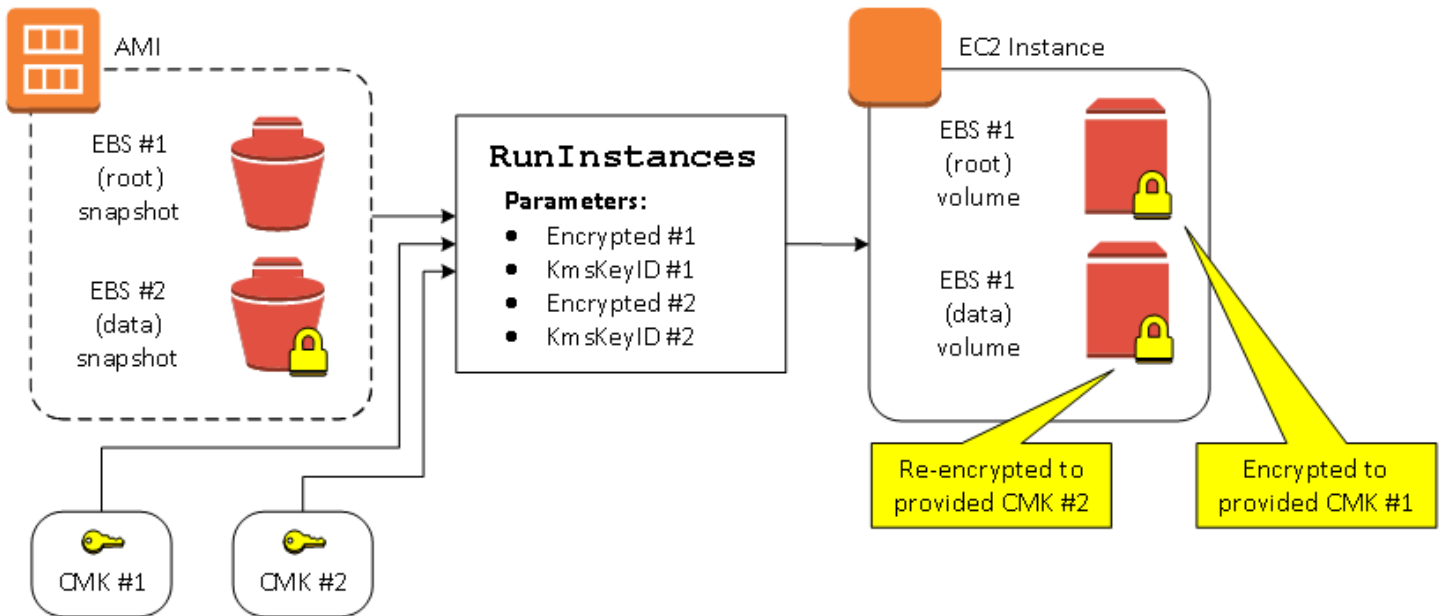
在此範例中，受加密快照支援的 AMI 用於啟動 EC2 執行個體，而此執行個體具有以新的 KMS 金鑰所加密的 EBS 磁碟區。



如果您擁有 AMI 但未提供加密參數，則產生的執行個體具有使用與快照相同的 KMS 金鑰所加密的磁碟區。如果 AMI 是共享，而非您所擁有，而且您未提供加密參數，則會以您的預設 KMS 金鑰來加密磁碟區。按上圖所示提供加密參數，磁碟區會以指定的 KMS 金鑰加密。

啟動期間變更多個磁碟區的加密狀態

在這個較複雜的範例中，以多個快照 (各有其自己的加密狀態) 為後端的 AMI 用於啟動 EC2 執行個體，而此執行個體具新加密的磁碟區和重新加密的磁碟區。



在此案例中，提供每個來源快照的加密參數給 **RunInstances** 動作。指定所有可能的加密參數時，無論您是否擁有 AMI，產生的執行個體都相同。

映像複製案例

透過 AWS Management Console 或直接使用 Amazon EC2 API 或 CLI，使用 CopyImage 動作來複製 Amazon EC2 AMI。

根據預設，如果沒有明確的加密參數，在複製期間，CopyImage 動作會維護 AMI 的來源快照的現有加密狀態。您也可以啟動 AMI 並提供加密參數，以同時將新的加密狀態套用到其相關聯的 EBS 快照。因此，可看到有下列行為：

在無加密參數的情況下複製

- 未加密快照會複製到另一個未加密的快照，除非預設加密已啟用，在此情況下，所有新建立的快照都加密。
- 您擁有的加密快照會複製到以相同 KMS 金鑰 加密的快照。
- 您不擁有的加密快照 (也就是說，AMI 會與您共用) 複製到由您 AWS 帳戶的預設 KMS 金鑰加密的快照集。

您可以提供加密參數來覆寫所有這些預設行為。可用的參數是 Encrypted 和 KmsKeyId。只設定 Encrypted 參數會產生下列行為：

已設定 **Encrypted** 但未指定 **KmsKeyId** 的複製映像行為

- 未加密快照會複製到以 AWS 帳戶的預設 KMS 金鑰所加密的快照。
- 加密快照會複製到以相同 KMS 金鑰 所加密的快照。(換言之，Encrypted 參數沒有效果。)
- 您不擁有的加密快照 (也就是說，AMI 會與您共用) 複製到由您 AWS 帳戶預設 KMS 金鑰加密的磁碟區。(換言之，Encrypted 參數沒有效果。)

同時設定 Encrypted 和 KmsKeyId 參數可讓您為加密操作指定客戶受管 KMS 金鑰。產生下列行為：

Encrypted 和 **KmsKeyId** 都設定的複製映像行為

- 未加密快照會複製到以指定的 KMS 金鑰 所加密的快照。
- 加密快照會複製到不是以原始 KMS 金鑰 所加密的快照，而是以指定的 KMS 金鑰 加密的快照。

提交 KmsKeyId 但未同時設定 Encrypted 參數會導致錯誤。

下一節提供使用非預設加密參數來複製 AMI 的範例，結果導致變更加密狀態。

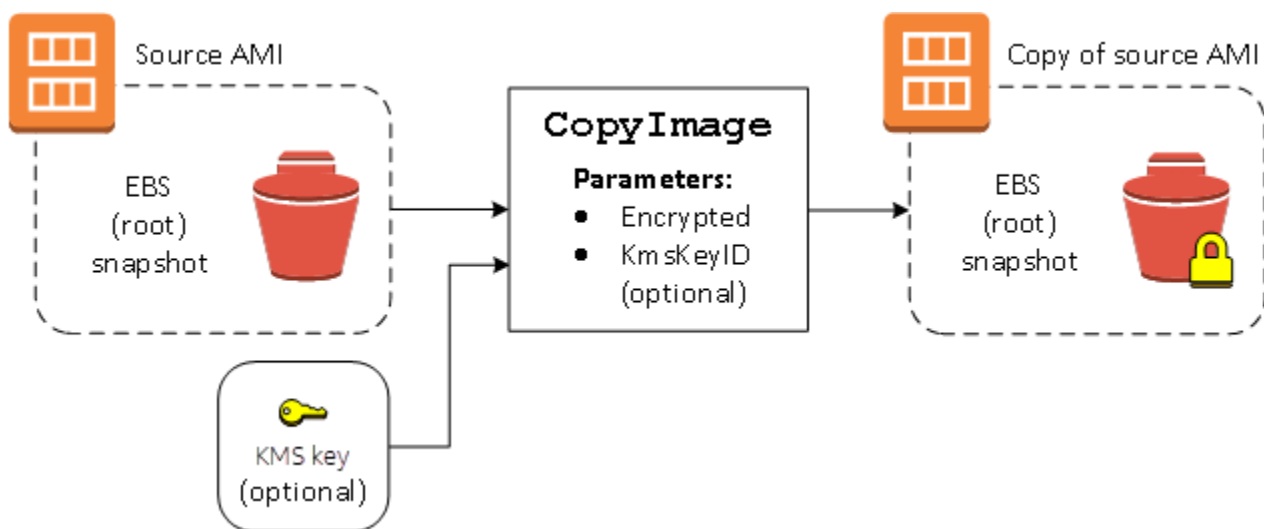
如需使用主控台的詳細說明，請參閱 [複製 AMI](#)。

複製期間加密未加密的映像

在此案例中，未加密根快照所支援的 AMI 將複製到具有加密根快照的 AMI。透過兩個加密參數叫用 CopyImage 動作，包括客戶受管金鑰。因此，根快照的加密狀態會變更，目標 AMI 會受到包含與來源快照相同資料的根快照支援，但使用指定的金鑰進行加密。您需要為兩個 AMI 中的快照支付儲存成本，以及從任一 AMI 啟動的任何執行個體的費用。

Note

依預設啟用加密，與將 Encrypted true 參數設定為 AMI 中所有快照的效果相同。



設定 Encrypted 參數會加密此執行個體的單一快照。若您未指定 KmsKeyId 參數，則會使用預設的客戶受管金鑰來加密快照複本。

Note

您也可以複製具有多個快照的映像，並個別設定各快照的加密狀態。

使用 Amazon 監控 AMI 事件 EventBridge

當 Amazon 機器映像 (AMI) 的狀態發生變化時，Amazon EC2 會生成一個發送到 Amazon 的事件 EventBridge (以前稱為 Amazon CloudWatch 活動)。您可以使用 Amazon EventBridge 偵測這些事

件並做出回應。您可以透過在其中建立規則 EventBridge 來觸發動作以回應事件來執行此操作。例如，您可以建立偵測 AMI 建立程序何時完成的 EventBridge 規則，然後叫用 Amazon SNS 主題以傳送電子郵件通知給您。

當 AMI 進入下列任何狀態時，Amazon EC2 會產生事件：

- available
- failed
- deregistered
- disabled

下表列出 AMI 操作和 AMI 可以進入的狀態。在該表中，是表示當對應操作執行時 AMI 可進入的狀態。

AMI 操作	available	failed	deregistered	disabled
CopyImage	是	是		
CreateImage	是	是		
CreateRes toreImageTask	是	是		
DeregisterImage			是	
DisableImage				是
EnableImage	是			
RegisterImage	是	是		

盡可能產生事件。

主題

- [AMI 事件](#)
- [創建 Amazon EventBridge 規則](#)

AMI 事件

有四個 EC2 AMI State Change 事件：

- [available](#)
- [failed](#)
- [deregistered](#)
- [disabled](#)

事件會以 JSON 格式傳送至預設 EventBridge 事件匯流排。

事件中的下列欄位可用於建立觸發動作的規則：

```
"source": "aws.ec2"
```

識別該事件是來自 Amazon EC2。

```
"detail-type": "EC2 AMI State Change"
```

識別事件名稱。

```
"detail": { "ImageId": "ami-0123456789example", "State": "available", }
```

提供下列資訊：

- AMI ID - 如果要追蹤特定 AMI。
- AMI 的狀態 (available、failed、deregistered 或 disabled)。

available

以下是 CreateImage、CopyImage、RegisterImage、CreateRestoreImageTask 或 EnableImage 操作成功之後，AMI 進入 available 狀態時，Amazon EC2 產生的事件範例。

"State": "available" 表示操作成功。

```
{
  "version": "0",
  "id": "example-9f07-51db-246b-d8b8441bcdf0",
  "detail-type": "EC2 AMI State Change",
  "source": "aws.ec2",
  "account": "012345678901",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-east-1",
```



```
"resources": ["arn:aws:ec2:us-east-1::image/ami-0123456789example"],
"detail": {
  "RequestId": "example-9dcc-40a6-aa77-7ce457d5442b",
  "ImageId": "ami-0123456789example",
  "State": "available",
  "ErrorMessage": ""
}
}
```

failed

以下是 CreateImage、CopyImage、RegisterImage 或 CreateRestoreImageTask 操作失敗之後，AMI 進入 failed 狀態時，Amazon EC2 產生的事件範例。

下列欄位會提供相關資訊：

- "State": "failed" - 表示操作失敗。
- "ErrorMessage": "" - 提供操作失敗的原因。

```
{
  "version": "0",
  "id": "example-9f07-51db-246b-d8b8441bcdf0",
  "detail-type": "EC2 AMI State Change",
  "source": "aws.ec2",
  "account": "012345678901",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-east-1",
  "resources": ["arn:aws:ec2:us-east-1::image/ami-0123456789example"],
  "detail": {
    "RequestId": "example-9dcc-40a6-aa77-7ce457d5442b",
    "ImageId": "ami-0123456789example",
    "State": "failed",
    "ErrorMessage": "Description of failure"
  }
}
```

deregistered

以下是在 DeregisterImage 操作成功之後，AMI 進入 deregistered 狀態時，Amazon EC2 產生的事件範例。如果操作失敗，則不會產生任何事件。因為 DeregisterImage 是一個同步操作，所以會立刻發現所有失敗。

"State": "deregistered" 表示 DeregisterImage 操作成功。

```
{
  "version": "0",
  "id": "example-9f07-51db-246b-d8b8441bcdf0",
  "detail-type": "EC2 AMI State Change",
  "source": "aws.ec2",
  "account": "012345678901",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-east-1",
  "resources": ["arn:aws:ec2:us-east-1::image/ami-0123456789example"],
  "detail": {
    "RequestId": "example-9dcc-40a6-aa77-7ce457d5442b",
    "ImageId": "ami-0123456789example",
    "State": "deregistered",
    "ErrorMessage": ""
  }
}
```

disabled

以下是在 DisableImage 操作成功之後，AMI 進入 disabled 狀態時，Amazon EC2 產生的事件範例。如果操作失敗，則不會產生任何事件。因為 DisableImage 是一個同步操作，所以會立刻發現所有失敗。

"State": "disabled" 表示 DisableImage 操作成功。

```
{
  "version": "0",
  "id": "example-9f07-51db-246b-d8b8441bcdf0",
  "detail-type": "EC2 AMI State Change",
  "source": "aws.ec2",
  "account": "012345678901",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-east-1",
  "resources": ["arn:aws:ec2:us-east-1::image/ami-0123456789example"],
  "detail": {
    "RequestId": "example-9dcc-40a6-aa77-7ce457d5442b",
    "ImageId": "ami-0123456789example",
    "State": "disabled",
    "ErrorMessage": ""
  }
}
```

```
}
```

創建 Amazon EventBridge 規則

您可以建立 Amazon EventBridge [規則](#)，以指定在 EventBridge 收到與規則中[事件模式相符的事件](#)時要採取的動作。當事件相符時，EventBridge 會將事件傳送至指定的[目標](#)，並觸發規則中定義的動作。

事件模式擁有與其相符事件相同的結構。事件模式與一個事件相符或不相符。

為 AMI 狀態變更事件建立規則時，可以在事件模式中包含下列欄位：

```
"source": "aws.ec2"
```

識別該事件是來自 Amazon EC2。

```
"detail-type": "EC2 AMI State Change"
```

識別事件名稱。

```
"detail": { "ImageId": "ami-0123456789example", "State": "available", }
```

提供下列資訊：

- AMI ID - 如果要追蹤特定 AMI。
- AMI 的狀態 (available、failed、deregistered 或 disabled)。

範例：建立 EventBridge 規則以傳送通知

下列範例會建立 EventBridge 規則，以便在 CreateImage 作業順利完成之後，任何 AMI 處於 available 狀態時傳送電子郵件、文字訊息或行動推播通知。

在建立 EventBridge 規則之前，您必須為電子郵件、文字訊息或行動推播通知建立 Amazon SNS 主題。

若要建立 EventBridge 規則，以便在建立 AMI 且處於 **available** 狀態時傳送通知

1. 在以下位置打開 Amazon EventBridge 控制台 <https://console.aws.amazon.com/events/>。
2. 選擇建立規則。
3. 針對 Define rule detail (定義規則詳細資訊) 執行下列動作：
 - a. 輸入規則的 Name (名稱)，或者輸入描述。

在同一個區域和同一個事件匯流排上，規則不能與另一個規則同名。

- b. 針對 Event bus (事件匯流排) 選擇 default (預設值)。當您帳戶中的 AWS 服務產生事件時，一律會前往您帳戶的預設事件匯流排。
 - c. 針對 Rule type (規則類型) 選擇 Rule with an event pattern (具有事件模式的規則)。
 - d. 選擇下一步。
4. 針對 Build event pattern (建置事件模式) 執行下列動作：
 - a. 對於事件來源，請選擇AWS 事件或 EventBridge合作夥伴事件。
 - b. 針對此範例的 Event pattern (事件模式)，您需指定下列事件模式，以便當 AMI 進入 available 狀態時與產生的任何 EC2 AMI State Change 事件相符：

```
{
  "source": ["aws.ec2"],
  "detail-type": ["EC2 AMI State Change"],
  "detail": {"State": ["available"]}
}
```

若要新增事件模式，您可以透過選擇 Event pattern form (事件模式表單) 使用範本，或者透過選擇 Custom pattern (JSON editor) (自定義模式 (JSON 編輯器)) 指定自有模式，如下：

- i. 若要使用範本建立事件模式，請執行下列動作：
 - A. 選擇 Event pattern form (事件模式表單)。
 - B. 在 Event source (事件來源) 欄位中，選擇 AWS services (服務)。
 - C. 在 AWS Service (服務) 欄位中，選擇 EC2。
 - D. 針對 Event type (事件類型) 選擇 EC2 AMI State Change (EC2 AMI 狀態變更)。
 - E. 若要自訂範本，請選擇 Edit pattern (編輯模式) 並進行變更以與範例事件模式相符。
 - ii. 若要指定自訂事件模式，請執行下列動作：
 - A. 選擇 Custom pattern (JSON editor) (自訂模式 (JSON 編輯器))。
 - B. 在 Event pattern (事件模式) 方塊中，為此範例新增事件模式。
 - c. 選擇下一步。
5. 針對 Select target(s) (選取目標) 執行下列動作：
 - a. 在 Target types (目標類型) 欄位中，選擇 AWS service (服務)。
 - b. 針對 Select a target (選取目標) 選取 SNS topic (SNS 主題)，以便在事件發生時傳送電子郵件、簡訊或行動裝置推送通知。

- c. 針對 Topic (主題)，請選擇現有的主題。您首先需要使用 Amazon SNS 主控台建立 Amazon SNS 主題。如需詳細資訊，請參閱 [Amazon 簡單通知服務開發人員指南中的使用 Amazon SNS 進行 application-to-person \(A2P\) 簡訊](#)。
 - d. (選用) 在 Additional settings (其他設定) 下，您可以選擇性地設定其他設定。如需詳細資訊，請參閱 [Amazon EventBridge 使用者指南中的建立對事件做出反應的 Amazon EventBridge 規則](#) (步驟 16)。
 - e. 選擇下一步。
6. (選用) 針對 Tags (標籤)，您可以選擇性地將一或多個標籤指派給您的規則，然後選擇 Next (下一步)。
 7. 針對 Review and create (檢閱和建立) 執行下列動作：
 - a. 檢閱規則的詳細資訊，然後視需求進行修改。
 - b. 選擇建立規則。

如需詳細資訊，請參閱 Amazon EventBridge 使用者指南中的以下主題：

- [Amazon EventBridge 活動](#)
- [Amazon EventBridge 事件模式](#)
- [Amazon EventBridge 規則](#)

如需如何建立 Lambda 函數和執行 Lambda 函數的 EventBridge 規則的教學課程，請參閱 [教學課程：使用 AWS Lambda 開發人員指南 EventBridge 中的使用記錄 Amazon EC2 執行個體的狀態](#)。

了解 AMI 帳單資訊

啟動執行個體時，有多種 Amazon Machine Image (AMI) 可供選擇，而且它們支援各種作業系統平台和功能。若要瞭解您在啟動執行個體時選擇的 AMI 如何影響帳 AWS 單的底線，您可以研究相關的作業系統平台和帳單資訊。在啟動任何隨需或 競價型執行個體，或購買 預留執行個體 之前執行此動作。

以下兩個例子說明事先研究 AMI 如何幫助您選擇最符合需求的 AMI：

- 針對 競價型執行個體，您可以使用 AMI 平台詳細資訊來確認 競價型執行個體 是否支援 AMI。
- 購買 預留執行個體 時，您可確保對應 AMI 平台詳細資訊選取操作系統平台 (平台)。

如需執行個體定價的詳細資訊，請參閱 [Amazon EC2 定價](#)。

內容

- [AMI 帳單資訊欄位](#)
- [尋找 AMI 帳單和用量詳細資訊](#)
- [驗證帳單上的 AMI 費用](#)

AMI 帳單資訊欄位

下列欄位提供與 AMI 相關的帳單資訊：

平台詳細資料

與 AMI 帳單代碼相關的平台詳細資訊。例如：Red Hat Enterprise Linux。

使用操作

Amazon EC2 執行個體的操作以及與 AMI 相關聯的帳單代碼。例如：RunInstances:0010。使用量作業與「[成本與用途報表](#)」(CUR) 與「[AWS 價目表 API](#)」中的「[生產線料號/作業](#)」欄位相對 [AWS 應](#)。

您可以在 Amazon EC2 主控台的執行個體或 AMI 頁面上檢視這些欄位，或在[描述影像或命令](#)傳回的回應中檢視這些欄位。[Get-EC2Image](#)

範例資料：使用操作 (依平台)

下表列出可在 Amazon EC2 主控台的執行個體或 AMI 頁面上顯示的一些平台詳細資訊和使用情況操作值，或在[描述影像或命令](#)傳回的回應中顯示。[Get-EC2Image](#)

平台詳細資料	使用操作 ²
Linux/UNIX	RunInstances
Red Hat BYOL Linux	RunInstances:00g0 ³
Red Hat Enterprise Linux	RunInstances:0010
Red Hat Enterprise Linux with HA	RunInstances:1010

平台詳細資料	使用操作 ²
Red Hat Enterprise Linux with SQL Server Standard and HA	RunInstances:1014
Red Hat Enterprise Linux with SQL Server Enterprise and HA	RunInstances:1110
Red Hat Enterprise Linux with SQL Server Standard	RunInstances:0014
Red Hat Enterprise Linux with SQL Server Web	RunInstances:0210
Red Hat Enterprise Linux with SQL Server Enterprise	RunInstances:0110
SQL Server Enterprise	RunInstances:0100
SQL Server Standard	RunInstances:0004
SQL Server Web	RunInstances:0200
SUSE Linux	RunInstances:000g
Ubuntu Pro	RunInstances:0g00
Windows	RunInstances:0002
Windows BYOL	RunInstances:0800
Windows with SQL Server Enterprise ¹	RunInstances:0102
Windows with SQL Server Standard ¹	RunInstances:0006
Windows with SQL Server Web ¹	RunInstances:0202

¹ 如果兩個軟體授權與 AMI 相關聯，平台詳細資料欄位會同時顯示兩者。

² 如果您正在執行 Spot 執行個體，則「AWS 成本和使用量報告」[lineitem/Operation](#)上的「使用量」作業值可能與此處列出的使用量作業值不同。例如，如果[lineitem/Operation](#)顯示RunInstances:0010:SV006，表示 Amazon EC2 在區域 6 的美國東部 (維吉尼亞北部) 執行 RHEL Spot 執行個體小時數。

³ 這會顯示RunInstances (Linux/UNIX)在您的使用情況報告中。

尋找 AMI 帳單和用量詳細資訊

在 Amazon EC2 主控台中，您可以從 AMI 頁面或執行個體頁面檢視 AMI 帳單資訊。您也可以使用 AWS CLI 或執行個體中繼資料服務尋找帳單資訊。

下列欄位可協助您驗證帳單上的 AMI 費用：

- 平台詳細資料
- 使用操作
- AMI ID

尋找 AMI 帳單資訊 (主控台)

請按照下列步驟在 Amazon EC2 主控台中檢視 AMI 帳單資訊：

從 AMI 頁面查詢 AMI 帳單資訊

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選取 AMI，然後選取一個 AMI。
3. 在 Details (詳細資料) 索引標籤上，檢查 Platform details (平台詳細資料) 和 Usage operation (使用操作) 的值。

從 Instances (執行個體) 頁面查詢 AMI 帳單資訊

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取一個執行個體。
3. 在 Details (詳細資訊) 索引標籤 (如果您使用的是舊版主控台，則是 Description (說明) 索引標籤) 上，查看 Platform details (平台詳細資訊) 和 Usage operation (使用操作) 的值。

尋找 AMI 帳單資訊 (AWS CLI)

若要使用尋找 AMI 帳單資訊 AWS CLI，您必須知道 AMI 識別碼。若不知道 AMI ID，可以使用 [describe-instances](#) 命令來從執行個體取得。

尋找 AMI ID

如果知道執行個體 ID，可以使用 [describe-instances](#) 命令來為執行個體取得 AMI ID。

```
aws ec2 describe-instances --instance-ids i-123456789abcde123
```

在輸出中，AMI ID 會在 ImageId 欄位中指定。

```
... "Instances": [  
  {  
    "AmiLaunchIndex": 0,  
    "ImageId": "ami-0123456789EXAMPLE",  
    "InstanceId": "i-123456789abcde123",  
    ...  
  }  
]
```

尋找 AMI 帳單資訊

如果知道 AMI ID，可以使用 [describe-images](#) 命令取得 AMI 平台和使用操作詳細資訊：

```
$ aws ec2 describe-images --image-ids ami-0123456789EXAMPLE
```

下列範例輸出顯示 PlatformDetails 和 UsageOperation 欄位。在此範例中，ami-0123456789EXAMPLE 平台是 Red Hat Enterprise Linux，而使用操作和計費代碼為 RunInstances:0010。

```
{  
  "Images": [  
    {  
      "VirtualizationType": "hvm",  
      "Description": "Provided by Red Hat, Inc.",  
      "Hypervisor": "xen",  
      "EnaSupport": true,  
      "SriovNetSupport": "simple",  
      "ImageId": "ami-0123456789EXAMPLE",  
      "State": "available",  
      "BlockDeviceMappings": [  

```

```
        {
            "DeviceName": "/dev/sda1",
            "Ebs": {
                "SnapshotId": "snap-111222333444aaabb",
                "DeleteOnTermination": true,
                "VolumeType": "gp2",
                "VolumeSize": 10,
                "Encrypted": false
            }
        }
    ],
    "Architecture": "x86_64",
    "ImageLocation": "123456789012/RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-
GP2",
    "RootDeviceType": "ebs",
    "OwnerId": "123456789012",
    "PlatformDetails": "Red Hat Enterprise Linux",
    "UsageOperation": "RunInstances:0010",
    "RootDeviceName": "/dev/sda1",
    "CreationDate": "2019-05-10T13:17:12.000Z",
    "Public": true,
    "ImageType": "machine",
    "Name": "RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-GP2"
}
]
}
```

驗證帳單上的 AMI 費用

為了確保您不會產生意外費用，您可以驗證 AWS 成本和用量報告 (CUR) 中執行個體的帳單資訊是否與您用來啟動執行個體的 AMI 相關聯的帳單資訊相符。

若要確認帳單資訊，請在 CUR 中尋找執行個體 ID，然後檢查 [lineitem/Operation](#) 欄中對應的值。該值應該符合 AMI 相關聯使用操作的值。

例如，AMI `ami-0123456789EXAMPLE` 具有下列帳單資訊：

- 平台詳細資訊 = Red Hat Enterprise Linux
- 使用操作 = RunInstances:0010

如果使用此 AMI 啟動執行個體，可在 CUR 中找到該執行個體 ID，然後檢查 [lineitem/Operation](#) 欄中的相應值。在此範例中，此值應為 RunInstances:0010。

AMI 配額

下列配額適用於建立及共用 AMI。這些配額適用於每個 AWS 區域。

配額名稱	描述	每個區域的預設配額
AMI	每個區域允許的公有和私有 AMI 的最大數量。其中包括可用、待定和已停用的 AMI，以及資源回收筒中的 AMI。	50,000
公有 AMI	每個區域允許的公有 AMI 數目上限，包括資源回收筒中的公有 AMI。	5
AMI 共用	可在區域中與 AMI 共用的實體 (組織、組織單位 (OU) 和帳戶) 的最大數目。請注意，如果您與組織或 OU 共用 AMI，則組織或 OU 中的帳戶數目不會計入配額中。	1,000

如果超出配額且想要建立或共用更多 AMI，則您可以執行以下操作：

- 如果超過總 AMI 配額或公有 AMI 配額，則請考慮取消註冊未使用的映像。
- 如果超過公有 AMI 配額，則請考慮將一個或多個公有 AMI 設為私有。
- 如果超過 AMI 共用配額，則請考慮與組織或 OU 共用您的 AMI，而不是與個別帳戶共用。
- 請求增加 AMI 的配額。

請求增加 AMI 的配額

如果您需要超過 AMI 的預設配額，則可以請求增加配額。

若要請求增加 AMI 的配額

1. 開啟 Service Quotas 主控台，網址為 <https://console.aws.amazon.com/servicequotas/>。

2. 在導覽窗格中，選擇 AWS services (AWS 服務)。
3. 從清單中選擇 Amazon Elastic Compute Cloud (Amazon EC2)，或在搜尋方塊中輸入服務名稱。
4. 選擇 AMI 配額以請求增加。您可以選取的 AMI 配額為：
 - AMI
 - 公有 AMI
 - AMI 共用
5. 選擇 Request quota increase (請求增加配額)。
6. 對於 Change quota value (變更配額值)，輸入新的配額值，然後選擇 Request (請求)。

若要檢視任何擱置或最近解決的請求，請從瀏覽窗格中選擇 Dashboard (儀表板)。對於擱置的請求，請選擇請求狀態以開啟請求回條。請求的初始狀態為 Pending (待定)。狀態變更為 Quota requested (已請求配額) 後，您會在 Support Center case number (支援中心案例編號) 中看到案例編號。選擇案例編號，為請求開啟票證。

解決請求之後，配額的 Applied quota value (套用的配額值) 會設定為新值。

如需詳細資訊，請參閱 [Service Quotas 使用者指南](#)。

Amazon EC2 執行個體

在您啟動生產環境之前，必須先回答下列問題。

問：哪一種執行個體類型最能符合我的需求？

Amazon EC2 提供不同的執行個體類型，供您選擇需要用來執行應用程式的 CPU、記憶體、儲存體和聯網容量。如需詳細資訊，請參閱 [Amazon EC2 執行個體類型](#)。

問：哪一種購買選項最能符合我的需求？

Amazon EC2 支援 隨需執行個體 (預設)、競價型執行個體 和 預留執行個體。如需詳細資訊，請參閱 [執行個體購買選項](#)。

問：哪一種根磁碟區類型符合我的需求？

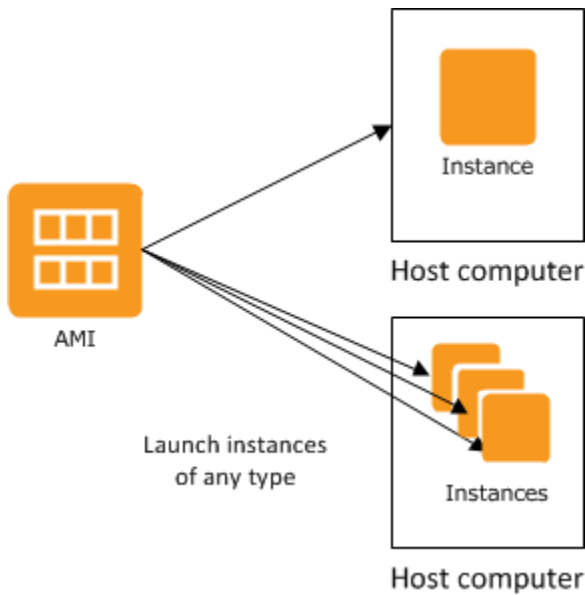
每一個執行個體若非 Amazon EBS 後端執行個體，即為執行個體存放區後端執行個體。根據您需要的根磁碟區類型來選取 AMI。如需詳細資訊，請參閱 [根設備儲存](#)。

問：我是否可以遠端管理混合環境中的 EC2 執行個體機群和機器？

AWS Systems Manager 可讓您在混合式環境中遠端安全地管理 Amazon EC2 執行個體以及現場部署執行個體和虛擬機器 (VM) 的組態，包括來自其他雲端供應商的 VM。如需詳細資訊，請參閱 [AWS Systems Manager 使用者指南](#)。

執行個體與 AMI

Amazon Machine Image (AMI) 是一種範本，其中包含軟體組態 (例如作業系統、應用程式伺服器 and 應用程式)。您可以從 AMI 啟動執行個體，執行個體是 AMI 的複本，在雲端中以虛擬伺服器的形式執行。您可以啟動 AMI 的多個執行個體，如下圖所示。



執行個體會持續執行運作，直到您將其停止、休眠或終止，或直到故障。如果執行個體故障，您可以從 AMI 啟動新的執行個體。

執行個體

執行個體是雲端虛擬伺服器。其在啟動時會設定 AMI 的複本，而此 AMI 是您啟動執行個體時所指定的。

您可以從單一 AMI 啟動不同類型的執行個體。執行個體類型基本上可決定執行個體所用主機電腦的硬體。每個執行個體類型皆提供不同的運算與記憶體功能。針對您計畫在執行個體類型上所執行的應用程式或軟體，根據其所需的記憶體數量和運算能力，來選取執行個體的類型。如需詳細的執行個體類型規格，請參閱 Amazon EC2 執行個體類型指南中的[規格](#)。如需定價資訊，請參閱 [Amazon EC2 隨需定價](#)。

執行個體在啟動之後看起來會像是傳統主機，您可以像使用任何電腦一樣地與其互動。您可以完全控制執行個體；您可以使用 `sudo` 來執行需要根權限的指令。

您的 AWS 帳戶對您可以執行的執行個體數量有限制。如需關於此限制的詳細資訊，以及如何申請提高上限，請參閱 Amazon EC2 一般常見問答集中的[我可以在 Amazon EC2 中執行多少執行個體](#)。

執行個體的儲存體

執行個體的根設備包含了用來啟動執行個體的映像。根裝置為 Amazon Elastic Block Store (Amazon EBS) 磁碟區或執行個體存放磁碟區。如需詳細資訊，請參閱 [Amazon EC2 執行個體根磁碟區](#)。

執行個體可以包含本機儲存磁碟區 (稱為執行個體存放磁碟區)，您可在啟動時使用區塊型設備映射來設定此磁碟區。如需詳細資訊，請參閱 [區塊型儲存設備對應](#)。當這些磁碟區新增並對應到執行個體之後，即可提供掛載和使用。如果執行個體故障，或是停止或終止，則這些磁碟區上的資料會遺失；因此，這些磁碟區最好用來儲存暫時性的資料。若要安全地保存重要資料，您應採用複寫到多個執行個體的策略，或是將持久性資料存放於 Amazon S3 或 Amazon EBS 磁碟區中。如需詳細資訊，請參閱 [Amazon EC2 執行個體的儲存選項](#)。

安全最佳實務

- 使用 AWS Identity and Access Management (IAM) 控制對 AWS 資源的存取，包括執行個體。如需詳細資訊，請參閱 [Amazon EC2 的 Identity and Access Management](#)。
- 透過只讓受信任的主機或網路存取執行個體上的通訊埠，來限制存取。例如，您可以藉由限制 22 號埠上的傳入流量，來限制 SSH 的存取。如需詳細資訊，請參閱 [適用於 Amazon EC2 執行個體的亞馬遜 EC2 安全群組](#)。
- 定期檢視安全群組中的規則，並確定實行最低權限的規則 (只開放需要的權限)。您也可以建立不同的安全群組，來處理具有不同安全要求的執行個體。請考慮建立只允許外部登入的堡壘安全群組，並且讓群組中其他的執行個體不允許外部登入。
- 停用使用密碼登入從 AMI 啟動的執行個體。密碼可能會遭到竊取或破解，而帶來安全的風險。如需詳細資訊，請參閱 [停用根使用者的密碼型態遠端登入](#)。如需安全共用 AMI 的相關詳細資訊，請參閱 [共享 AMI](#)。

停止和終止執行個體

您可以隨時停止或終止執行中的執行個體。

停止執行個體

當執行個體停止時，會進行正常的關閉動作，然後轉換為 stopped 狀態。該執行個體所有的 Amazon EBS 磁碟區仍會保持連結，您可以在之後再次啟動該執行個體。

當執行個體處於停止狀態時，將不會向您收取額外的執行個體使用費。從停止狀態轉換為執行中狀態的每次轉換都需支付費用。如果執行個體類型在執行個體停止時變更，則在執行個體啟動後，系統會向您收取新執行個體類型的費率。您也需要支付執行個體相關聯的 Amazon EBS 儲存費用，包括根裝置磁碟區。

當執行個體處於停止狀態時，您可以連結 Amazon EBS 磁碟區或中斷其連結。您也可以從執行個體建立 AMI，和變更核心、RAM 磁碟及執行個體類型。

終止執行個體

當執行個體終止時，會進行正常的關閉動作。根據預設，根設備磁碟區會刪除，但任何連結的 Amazon EBS 磁碟區會保留，由每個磁碟區的 `deleteOnTermination` 屬性設定決定。執行個體本身也會刪除，而且之後無法再次啟動該執行個體。

若要預防意外終止，您可以停用執行個體終止。若要這麼做，請確定將執行個體的 `disableApiTermination` 屬性設定為 `true`。若要控制執行個體關閉的行為 (例如在 Linux 中的 `shutdown -h` 或是在 Windows 中的 `shutdown`)，請根據需要，將 `instanceInitiatedShutdownBehavior` 執行個體屬性設定為 `stop` 或 `terminate`。使用 Amazon EBS 磁碟區做為根設備執行個體預設為 `stop`；使用執行個體存放區根設備的執行個體，在執行個體關閉時則一律會終止。

如需詳細資訊，請參閱 [執行個體生命週期](#)。

Note

某些 AWS 資源 (例如 Amazon EBS 磁碟區和彈性 IP 地址) 會產生費用，無論執行個體的狀態為何。如需詳細資訊，請參閱 AWS Billing 使用者指南中的 [避免非預期的費用](#)。如需 Amazon EBS 成本的詳細資訊，請參閱 [Amazon EBS 定價](#)。

AMI

Amazon Web Services (AWS) 發佈包含供公眾使用的常見軟體組態的 Amazon 機器映像 (AMI)。此外，AWS 開發人員社群的成員也發佈了自己的自訂 AMI。您也可以建立自己的自訂 AMI；這樣做可讓您快速輕鬆地啟動擁有所需一切的新執行個體。例如，如果您的應用程式是網站或 web 服務，則 AMI 可包含 web 伺服器、相關的靜態內容和動態網頁的程式碼。因此，在從這個 AMI 啟動執行個體後，Web 伺服器會啟動，您的應用程式也就緒可接受請求。

所有的 AMI 可分類為以 Amazon EBS 為後端和以執行個體存放區為後端，前者表示從 AMI 啟動的執行個體根設備為 Amazon EBS 磁碟區，後者表示從 AMI 啟動的執行個體根設備，是以儲存於 Amazon S3 中的範本所建立的執行個體存放磁碟區。

AMI 的描述會指出根設備的類型 (`ebs` 或 `instance store`)。這點十分重要，因為每種 AMI 的用途有極大的差異。如需這些差異的詳細資訊，請參閱 [根設備儲存](#)。

您可以在完成使用之後取消註冊 AMI。在您取消註冊 AMI 之後，您便無法用它來啟動新的執行個體。從 AMI 啟動的現有執行個體不會受影響。因此，如果您也完成了從這些 AMI 啟動的執行個體，即應該終止這些執行個體。

Amazon EC2 執行個體類型

啟動執行個體時，您指定的執行個體類型將決定執行個體所用的主機電腦硬體。每個執行個體類型均提供不同的運算、記憶體和儲存功能，且會依照這些功能分組為不同的執行個體系列。請根據您計劃在執行個體上執行之應用程式或軟體的需求選取執行個體類型。

Amazon EC2 會將主機電腦的部分資源 (例如 CPU、記憶體和執行個體儲存體) 專用於特定執行個體。Amazon EC2 在執行個體間共用主機電腦的其他資源，例如網路和磁碟子系統。如果主機電腦上的每個執行個體嘗試對其中一項共用資源作最大的利用，該資源會平均分配給每個執行個體。但是，當某一資源使用率較低時，執行個體便能使用該資源較高的比例 (需有資源可用)。

每個執行個體類型從共同資源獲得的最低效能可能更高或更低。例如，具有高 I/O 效能的執行個體類型會分配到較多的共用資源。分配較多的共用資源，也可減少 I/O 效能的變動。對多數的應用程式來說，中等的 I/O 效能即堪使用。但是，對於需要更高或更一致 I/O 效能的應用程式，請考慮使用具有較高 I/O 效能的執行個體類型。

內容

- [可用的執行個體類型](#)
- [硬體規格](#)
- [AMI 虛擬化類型](#)
- [尋找 Amazon EC2 執行個體類型](#)
- [取得執行個體類型的建議](#)
- [變更執行個體類型](#)
- [爆量效能執行個體](#)
- [GPU 執行個體的效能加速](#)

可用的執行個體類型

Amazon EC2 提供各式各樣的最佳化執行個體類型，以滿足不同的使用案例。執行個體類型包含 CPU、記憶體、儲存體和聯網功能的各種組合，供您靈活選擇適用於應用程式的適當資源組合。每個執行個體類型都包含一或多個執行個體大小，可讓您根據目標工作負載的需求擴展資源。如需有關功能和使用案例的詳細資訊，請參閱 [Amazon EC2 執行個體類型詳細資訊](#)。

實例類型命名慣例

名稱取決於執行個體系列、層代、處理器系列、功能和大小。如需詳細資訊，請參閱 Amazon EC2 執行個體類型指南中的 [命名慣例](#)。

尋找執行個體類型

若要判斷哪些執行個體類型符合您的需求，例如支援的區域、運算資源或儲存資源，請參閱 [Amazon EC2 執行個體類型指南中的尋找 Amazon EC2 執行個體類型](#) 和 [Amazon EC2 執行個體類型規格](#)。

最新一代的執行個體

- 一般用途：M5 | M5Ad | 兆廣播公司 | 兆六廣播公司 | M6gD | M6gd | M6i | M6i | M6-彈性 | M6-彈性 | M6 英寸 | M7a | 麥克 7 克 | 麥克 M2Pro | T2 | T3 | T4G
- 計算最佳化：C5 | C5Ad | C5 毫升 | C5 毫升 | C5 安 | C6 克重 | 氯化碳 | 氯化碳 | C6 英寸 | C6 英寸 | C7 克 | 氯化乙烯 | 氯化碳 | 氯化碳
- 記憶體最佳化：R5 | R5 安培 | R5 倍 | R5dN | R5dn | R5n | R6g | R6gd | R6 億 | R6 英寸 | U 型 | U-12 TB1 | U-18 兆比 1 | U-24 兆比 1 | 烏七至十二 TB | 烏 7 英寸 -16 TB | 烏 7 英寸 -32 TB | X2 國際化數據分佈 | X2 國際化 DN | X2iedn | X2eZn | X1e
- 儲存最佳化：D2 | D3 | 第 3 天 | H1 | I3 | I3en | I4G | I4i | IM4GN | i4GN
- 加速運算：DL1 | DL2Q | F1 | 四倍 | G4dN | G5 | G5G | G6 | GR6 | 基因 1 | 二級二 | P3 | P3dN | P4dN | P4dE | P4de | P5 | 中一田一號 | VT1
- 高效能運算：高效能運算 | HPC6a | HPC7g

上一代執行個體

- 一般用途：A1 | M1 | M2 | 立方米 | M4
- 運算最佳化：C1 | C3
- 記憶體最佳化：
- 儲存最佳化：I2
- 加速運算：G3

硬體規格

如需詳細的執行個體類型規格，請參閱 [Amazon EC2 執行個體類型指南中的規格](#)。如需定價資訊，請參閱 [Amazon EC2 隨需定價](#)。

為了決定最符合您需求的執行個體類型，我們建議您啟動執行個體並使用您自己的基準參考應用程式。由於您必須為執行個體按秒數付費，因此在作決定前先測試多個執行個體類型是比較方便且省錢的方式。假如您在作決定後需要變更，也可稍後再變更執行個體類型。如需詳細資訊，請參閱 [變更執行個體類型](#)。

Intel 處理器功能

在 Intel 處理器上執行的 Amazon EC2 執行個體可能包含下列功能。並非所有下列處理器功能都受到所有執行個體類型的支援。如需每種執行個體類型可用功能的詳細資訊，請參閱 [Amazon EC2 執行個體類型](#)。

- Intel AES New Instructions (AES-NI) – Intel AES-NI 加密指令集改進了原先 Advanced Encryption Standard (AES) 的演算法，可以提供更快的資料保護和更佳的安全性。所有最新一代的 EC2 執行個體都支援此處理器功能。
- Intel Advanced Vector Extensions (Intel AVX、Intel AVX2 和 Intel AVX-512) – Intel AVX 和 Intel AVX2 是 256 位元，而 Intel AVX-512 是 512 位元指令集延伸，專為浮點 (FP) 密集型應用程式所設計。Intel AVX instructions 可提升如圖片和音效/影片處理、科學模擬、財務分析、3D 模組化與分析等應用程式的效能。只有以 HVM AMI 啟動的執行個體可使用這些功能。
- Intel Turbo Boost Technology (Intel 渦輪加速技術) – Intel Turbo Boost Technology (Intel 渦輪加速技術) 處理器會自動執行核心，比基礎操作頻率更快。
- Intel Deep Learning Boost (Intel DL Boost) – 可加速 AI 深度學習使用案例。第二代 Intel Xeon 可擴充處理器透過新的向量神經網路指令 (VNNI/INT8) 擴展 Intel AVX-512，相較於前一代 Intel Xeon 可擴充處理器 (搭配 FP32)，大幅提升了影像辨識 / 切割、物件偵測、語音辨識、語言翻譯、推薦系統、強化學習等的深度學習推斷效能。VNNI 可能無法與所有 Linux 發行版相容。

下列執行個體支援 VNNI：M5n、R5n、M5dn、M5zn、R5b、R5dn、D3、D3en 和 C6i。C5 和 C5d 執行個體僅針對 12xlarge、24xlarge 以及 metal 執行個體支援 VNNI。

可能會因業界對 64 位元 CPU 的命名慣例而產生混淆。自晶片製造商 Advanced Micro Devices (AMD) 成功推出首款以 Intel x86 指令集為基礎的商用 64 位元架構後，該架構自此便廣泛統稱為 AMD64，無論產品來自哪一間晶片製造商。Windows 和多個 Linux 發行版本也遵照此作法。這說明了為何執行 Ubuntu 或 Windows 之執行個體上的內部系統資訊會顯示 CPU 架構為 AMD64，即使該執行個體是在 Intel 硬體上執行。

AWS 重力子處理器

[AWS Graviton](#) 是一系列處理器，旨在為您在 Amazon EC2 執行個體上執行的工作負載提供最優惠的價格效能。

如需詳細資訊，請參閱 [重力子入門](#)。

AWS 草屬

由 [AWS Trainium](#) 提供支援的執行個體專為高效能、具成本效益的深度學習訓練而打造。您可以使用這些執行個體來訓練自然語言處理、電腦視覺，以及廣泛應用程式所使用的推薦模型，例如語音辨識、建議、詐騙偵測，以及影像和影片分類。在常見的 ML 架構中使用您現有的工作流程，例如 PyTorch 和 TensorFlow。

AWS 推論

由 [AWS Inferentia](#) 提供支援的執行個體專為加速機器學習而設計。它們提供高效能和低延遲的機器學習推論。這些執行個體已進行最佳化，適用於部署不同應用的深度學習 (DL) 模型，例如自然語言處理、物件偵測與分類、內容個人化與篩選，以及語音辨識等應用。

您可以透過多種方式開始使用：

- 使用完全受管理的服務 SageMaker，是開始使用機器學習模型的最簡單方法。如需詳細資訊，請參閱 [Amazon 開 SageMaker 發人員指南 SageMaker 中的入門使用](#)。
- 使用深度學習 AMI 啟動 Inf1 或 Inf2 執行個體。如需詳細資訊，請參閱 [AWS Deep Learning AMI 開發人員指南中的具有 DLAMI 的 AWS Inferentia](#)。
- 使用您自己的 AMI 啟動 Inf1 或 Inf2 執行個體，並安裝 [AWS Neuron SDK](#)，以便能夠編譯、執行和分析 AWS Inferentia 的深度學習模型。
- 使用 Inf1 或 Inf2 執行個體和最佳化 Amazon ECS 的 AMI 啟動容器執行個體。若要取得更多資訊，請參閱 [Amazon Elastic Container Service Developer Guide 中的 Amazon Linux 2 \(Inferentia\) AMI](#)。
- 建立具有執行 Inf1 執行個體之節點的 Amazon EKS 叢集。如需詳細資訊，請參閱 [Amazon EKS 使用者指南中的 Inferentia 支援](#)。

AMI 虛擬化類型

執行個體的虛擬化類型由用來啟動執行個體的 AMI 決定。最新一代的執行個體類型僅支援硬體虛擬機器 (HVM)。上一代的某些執行個體類型支援半虛擬 (PV)，有些 AWS 區域支援 PV 執行個體。如需詳細資訊，請參閱 [AMI 虛擬化類型](#)。

為獲得最高效能，我們建議您使用 HVM AMI。此外，需要 HVM AMI 才能充分利用增強型聯網。HVM 虛擬化使用平台提供的 AWS 硬體輔助技術。透過 HVM 虛擬化，訪客 VM 便能如同在原生硬體平台上一般執行，只是仍是使用全虛擬化 (PV) 網路和儲存驅動程式來提升效能。

尋找 Amazon EC2 執行個體類型

您必須先選取要使用的執行個體類型，才能啟動執行個體。您選擇的執行個體類型可能取決於您的工作負載所需的資源，如運算、記憶體或儲存資源。識別數種可能適合您的工作負載的執行個體類型，並評估其在測試環境中的效能，可能是有利的做法。沒有任何替代方法可以測量您的應用程式在負載條件下的效能。

如果您已經有執行 EC2 執行個體，您可以使用 AWS Compute Optimizer 來取得執行個體類型的建議，以提升效能、節省成本或兩者兼而有之。如需詳細資訊，請參閱 [the section called “對於現有工作負載”](#)。

任務

- [使用主控台尋找執行個體類型](#)
- [使用尋找執行個體類型 AWS CLI](#)

使用主控台尋找執行個體類型

您可以使用 Amazon EC2 主控台來尋找符合您需求的執行個體類型。

使用主控台尋找執行個體類型

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 從導覽列選取要在其中啟動執行個體的區域。無論您的位置為何，皆可選取任何可用的區域。
3. 在導覽窗格中，選擇 Instance Types (執行個體類型)。
4. (選用) 選取偏好設定 (齒輪) 圖示來選取要顯示的執行個體類型屬性，例如隨需 Linux 定價，然後選取 Confirm (確認)。或者，選取執行個體類型的名稱，以開啟其詳細資訊頁面，並透過主控台檢視可用的所有屬性。主控台不會顯示透過 API 或命令列可用的所有屬性。
5. 使用執行個體類型屬性，將顯示的執行個體類型清單篩選為僅符合您需求的執行個體類型。例如，您可以篩選下列屬性：
 - Availability zones (可用區域) – 可用區域、本機區域或 Wavelength 區域的名稱。如需詳細資訊，請參閱 [the section called “區域 \(Region\) 和區域 \(Zone\)”](#)。
 - vCPUs 或 Cores (核心) – vCPU 或核心的數量。
 - Memory (GiB) (記憶體 (GiB)) – 記憶體大小 (以 GiB 為單位)。
 - Network performance (網路效能) – 網路效能 (以 GB 為單位)。

- Local instance storage (本機執行個體儲存體) – 表示執行個體類型是否具有本機執行個體儲存體 (true | false)。
6. (選擇性) 若要查看 side-by-side 比較，請選取多個執行個體類型的核取方塊。比較結果顯示在螢幕畫面的底部。
 7. (選用) 若要將執行個體類型清單儲存為逗號分隔值 (.csv) 檔案以供進一步檢閱，請選擇 Actions (動作)、Download list CSV (下載清單 CSV)。檔案包括符合您所設定篩選的所有執行個體類型。
 8. (選用) 若要使用滿足您需求的執行個體類型來啟動執行個體，請選取執行個體類型的核取方塊，然後選取 Actions (動作)、Launch instance (啟動執行個體)。如需詳細資訊，請參閱 [使用新的啟動執行個體精靈啟動執行個體](#)。

使用尋找執行個體類型 AWS CLI

您可以使用 Amazon EC2 的 AWS CLI 命令尋找符合您需求的執行個體類型。

若要使用尋找例證類型 AWS CLI

1. 如果尚未安裝，請安裝「如 AWS CLI 需詳細資訊，請參閱 [《AWS Command Line Interface 使用者指南》](#)」。
2. 使用 [describe-instance-types](#) 命令，根據執行個體屬性篩選執行個體類型。例如，您可以使用下列命令，僅顯示包含 64 GiB (65536 MiB) 記憶體在目前世代執行個體類型。

```
aws ec2 describe-instance-types --filters "Name=current-generation,Values=true"
"Name=memory-info.size-in-mib,Values=65536" --query "InstanceTypes[*].
[InstanceType]" --output text | sort
```

3. 使用 [describe-instance-type-offerings](#) 命令來依位置 (區域 (Region) 或區域 (Zone)) 篩選提供的執行個體類型。例如，您可以使用下列命令來顯示指定的區域中提供的執行個體類型。

```
aws ec2 describe-instance-type-offerings --location-type "availability-
zone" --filters Name=location,Values=us-east-2a --region us-east-2 --query
"InstanceTypeOfferings[*].[InstanceType]" --output text | sort
```

4. 找到符合您需求的執行個體類型後，請儲存清單，以便在啟動執行個體時可以使用這些執行個體類型。如需詳細資訊，請參閱 AWS Command Line Interface 使用者指南中的 [啟動您的執行個體](#)。

取得執行個體類型的建議

下列工具可協助您為新工作負載或現有工作負載選取最佳執行個體類型：

- 新工作負載 — EC2 執行個體類型尋找器會考量您的使用案例、工作負載類型、CPU 製造商偏好設定，以及您如何設定價格和效能的優先順序，以及您可以指定的其他參數。接著，它會使用這些資料，針對最適合您的新工作負載的 Amazon EC2 執行個體類型提供建議和指導。
- 現有工作負載 — AWS Compute Optimizer 分析現有執行個體規格和使用率指標。然後它會使用編譯的資料，針對您現有的工作負載，建議哪些 Amazon EC2 執行個體類型要進行成本和/或效能優化。

取得執行個體類型建議：

- [取得新工作負載的執行個體類型建議](#)
- [取得現有工作負載的執行個體類型建議](#)

取得新工作負載的執行個體類型建議

EC2 執行個體類型尋找器會考量您的使用案例、工作負載類型、CPU 製造商偏好設定，以及您如何設定價格和效能的優先順序，以及您可以指定的其他參數。接著，它會使用這些資料，針對最適合您的新工作負載的 Amazon EC2 執行個體類型提供建議和指導。

如此眾多的執行個體類型可供使用，尋找適合您工作負載的執行個體類型可能非常耗時且複雜。透過使用 EC2 執行個體類型尋找工具，您可以保持最新的執行個體類型，並為工作負載達到最佳價格效能。

本主題概述如何透過 Amazon EC2 主控台取得 EC2 執行個體類型的建議和指導。您也可以直接前往 Amazon Q 尋求執行個體類型建議。如需詳細資訊，請參閱 [Amazon Q 開發人員使用者指南](#)。

如果您正在尋找適用於現有工作負載的執行個體類型建議，請使用 AWS Compute Optimizer。如需詳細資訊，請參閱 [取得現有工作負載的執行個體類型建議](#)。

使用 EC2 執行個體類型尋找工具

在 Amazon EC2 主控台中，您可以從啟動執行個體精靈中的 EC2 執行個體類型尋找工具、建立啟動範本時，或在 [執行個體類型] 頁面上取得執行個體類型建議。

使用下列指示，在 Amazon EC2 主控台中使用 EC2 執行個體類型尋找工具取得 EC2 執行個體類型的建議和指導。若要檢視步驟的動畫，請參閱 [檢視動畫：使用 EC2 執行個體類型尋找工具取得執行個體類型建議](#)。

使用 EC2 執行個體類型尋找工具取得執行個體類型建議

1. 使用以下任何一種方式開始您的流程：

- 請遵循該程序來[啟動執行個體](#)。在執行個體類型旁，選擇取得建議連結。

- 依照程序建立[啟動範本](#)。在執行個體類型旁，選擇取得建議連結。
 - 在瀏覽窗格中，選擇「實體類型」，然後選擇「實體類型尋找工具」按鈕。
2. 在「取得執行個體類型選取建議」畫面中，執行下列動作：
 - a. 選取工作負載類型、使用案例、優先順序和 CPU 製造商的選項，以指定執行個體類型需求。
 - b. (選擇性) 若要為工作負載指定更詳細的需求，請執行下列動作：
 - i. 展開 [進階參數]。
 - ii. 若要加入參數，請選取參數，選擇「新增」，然後指定參數值。對您要加入的每個其他參數重複上述步驟。若要指示沒有最小值或最大值，請將欄位保留空白。
 - iii. 若要在新增參數後移除參數，請選擇參數旁邊的 X。
 - c. 選擇取得執行個體類型建議。

Amazon EC2 為您提供符合指定需求的執行個體系列建議。

3. 若要檢視建議的執行個體系列中每個執行個體類型的詳細資料，請選擇 [檢視建議的執行個體系列
4. 選取符合您需求的執行個體類型，然後選擇「動作」、「啟動執行個體」或「動作」、「建立啟動範本」。

或者，如果您是在啟動執行個體精靈或啟動範本頁面中啟動程序，而且想要回到原始流程，請記下您要使用的執行個體類型。接著，在啟動執行個體精靈或啟動範本中，針對「執行個體類型」選擇執行個體類型，並完成啟動執行個體或建立啟動範本的程序。

檢視動畫：使用 EC2 執行個體類型尋找工具取得執行個體類型建議

The screenshot displays the AWS Management Console interface. On the left is a navigation sidebar with categories like 'Instances', 'Images', 'Elastic Block Store', and 'Network & Security'. The main content area is divided into several panels:

- Resources:** A table showing EC2 resources in the US East (N. Virginia) Region. The table lists: Instances (running) - 2, Auto Scaling Groups - 0, Dedicated Hosts - 0, Elastic IPs - 0, Instances - 2, Key pairs - 0, Load balancers - 0, Placement groups - 0, Security groups - 12, Snapshots - 3, and Volumes - 2.
- Launch instance:** A section with a 'Launch Instance' button and a 'Migrate a server' link. A note states: 'Note: Your instances will launch in the US East (N. Virginia) Region'.
- Service health:** A section titled 'AWS Health Dashboard' showing the status for the US East (N. Virginia) Region as 'This service is operating normally'.
- Account attributes:** A section showing 'Default VPC' (vpc-92304aeb) and various settings like 'Data protection and security', 'Zones', 'EC2 Serial Console', 'Default credit specification', and 'EC2 console preferences'.
- Explore AWS:** A section with a close button and promotional text: 'Get Up to 40% Better Price Performance' and 'Enable Best Price-Performance with AWS Graviton2'.

取得現有工作負載的執行個體類型建議

AWS Compute Optimizer 提供 Amazon EC2 執行個體建議，協助您改善效能、節省金錢或兩者兼而有之。您可以使用這些建議來決定是否要移至新的執行個體類型。

若要提出建議，Compute Optimizer 會分析您現有的執行個體規格和使用率指標。然後，編譯的資料會用來建議哪些 Amazon EC2 執行個體類型最適合處理現有的工作負載。建議會連同每小時執行個體定價一起傳回。

本主題概述如何透過 Amazon EC2 主控台檢視建議。如需詳細資訊，請參閱 [《AWS Compute Optimizer 使用者指南》](#)。

Note

若要從 Compute Optimizer 取得建議，您必須先選擇加入 Compute Optimizer。如需詳細資訊，請參閱 AWS Compute Optimizer 使用者指南中的 [AWS Compute Optimizer 入門](#)。

如果您正在尋找新工作負載的執行個體類型建議，請使用 Amazon Q EC2 執行個體類型選取器。如需詳細資訊，請參閱 [取得新工作負載的執行個體類型建議](#)。

目錄

- [限制](#)
- [問題清單](#)
- [檢視建議](#)
- [評估建議的考量事項](#)
- [其他資源](#)

限制

Compute Optimizer 目前會產生 C、D、H、I、M、R、T、X 和 z 執行個體類型的建議。Compute Optimizer 不會考慮其他執行個體類型。如果您使用的是其他執行個體類型，它們將不會列在 Compute Optimizer 建議檢視中。如需有關支援和不支援的執行個體類型的詳細資訊，請參閱《AWS Compute Optimizer 使用者指南》中的 [Amazon EC2 執行個體需求](#)。

問題清單

Compute Optimizer 會將其 EC2 執行個體的發現結果分類如下：

- Under-provisioned (佈建不足)–當您的執行個體至少有一個規格 (例如 CPU、記憶體或網路) 不符合工作負載的效能需求時，EC2 執行個體即視為佈建不足。佈建不足的 EC2 執行個體可能會導致應用程式效能不佳。
- Over-provisioned (過度佈建)–當執行個體至少一個規格可以縮小 (例如 CPU、記憶體或網路)，同時仍然符合工作負載的效能需求時，以及當沒有規格佈建不足時，EC2 執行個體即視為過度佈建。過度佈建的 EC2 執行個體可能會導致不必要的基礎設施成本。
- Optimized (最佳化)–當執行個體的所有規格 (例如 CPU、記憶體和網路) 都符合工作負載的效能需求，且執行個體未過度佈建時，EC2 執行個體即視為最佳化。最佳化的 EC2 執行個體會以最佳效能和基礎設施成本執行您的工作負載。對於最佳化的執行個體，Compute Optimizer 有時可能會建議新一代執行個體類型。
- None (無)–沒有此執行個體的建議。如果您已在 12 小時內選擇加入 Compute Optimizer，或是執行個體執行時間不到 30 小時，或 Compute Optimizer 不支援執行個體類型，就可能發生這種情況。如需詳細資訊，請參閱前一節中的 [限制](#)。

檢視建議

選擇加入 Compute Optimizer 之後，您可以在 EC2 主控台中檢視 Compute Optimizer 為 EC2 執行個體所產生的發現結果。然後，您可以存取 Compute Optimizer 主控台來檢視建議。如果您最近選擇加入，發現結果可能需要長達 12 小時才會反映在 EC2 主控台中。

透過 EC2 主控台檢視 EC2 執行個體的建議

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體，然後選擇執行個體 ID。
3. 在執行個體摘要頁面上，在頁面底部附近的 AWS Compute Optimizer 橫幅中，選擇檢視詳細資訊。

執行個體會 Compute Optimizer 中開啟，並標示為 Current (目前) 執行個體。最多可提供三種不同的執行個體類型建議，標示為 Option 1 (選項 1)、Option 2 (選項 2) 和 Option 3 (選項 3)。視窗下半部顯示目前執行處理的最近 CloudWatch 測量結果資料：CPU 使用率、記憶體使用率、網路輸入和網路輸出。

4. (選擇性) 在 Compute Optimizer 主控台中，選擇設定



以變更表格中的可見資料欄，或檢視目前和建議執行個體類型不同購買選項的公開定價資訊。


Note

如果您購買的是預留執行個體，隨需執行個體可能會以預留執行個體方式計費。變更目前的執行個體類型之前，請先評估對預留執行個體使用率和涵蓋範圍的影響。

決定您是否要使用其中一項建議。決定是否要最佳化以達到效能提升、降低成本，還是兩者的組合。如需詳細資訊，請參閱 AWS Compute Optimizer 使用者指南中的 [檢視資源建議](#)。

透過 Compute Optimizer 主控台檢視所有區域中所有 EC2 執行個體的建議

1. 在 <https://console.aws.amazon.com/compute-optimizer/> 開啟運算最佳化工具主控台。
2. 選擇 View recommendations for all EC2 instances (檢視所有 EC2 執行個體的建議)。
3. 您可以在建議頁面上執行下列動作：
 - a. 若要篩選一或多個「區 AWS 域」的建議，請在「依一或多個區域篩選」文字方塊中輸入「地區」的名稱，或在出現的下拉式清單中選擇一或多個「區域」。

- b. 若要檢視其他帳戶中的資源建議，請選取 Account (帳戶)，然後選取不同的帳戶 ID。
只有當您登入組織的管理帳戶，且已選擇組織內的所有成員帳戶時，才能使用此選項。
- c. 若要清除選取的篩選條件，請選取 Clear filters (清除篩選條件)。
- d. 若要變更目前和建議執行個體類型顯示的購買選項，請選擇設定 )，然後選擇隨需執行個體、預留執行個體、標準 1 年無預付或預留執行個體 (標準 3 年無預付)。
- e. 若要檢視詳細資訊，例如其他建議和使用率指標的比較，請選擇所需執行個體旁所列的發現結果 (Under-provisioned (佈建不足)、Over-provisioned (過度佈建) 或 Optimized (最佳化))。如需詳細資訊，請參閱 AWS Compute Optimizer 使用者指南中的 [檢視資源詳細資訊](#)。

評估建議的考量事項

變更執行個體類型之前，請考慮下列事項：

- 建議不會預測您的使用量。建議是根據您最近 14 天期間的歷史使用情況而定。請務必選擇符合未來資源需求的執行個體類型。
- 著重於圖形指標，以判斷實際使用量是否低於執行個體容量。您也可以在 CloudWatch 中檢視指標資料 (平均值、峰值、百分位數)，以進一步評估 EC2 執行個體建議。例如，注意 CPU 百分比指標在白天的變化，以及是否有需要應付的尖峰時期。如需詳細資訊，請參閱 [Amazon CloudWatch 使用者指南中的檢視可用指標](#)。
- Compute Optimizer 可能會提供高載效能執行個體的建議，亦即 T3、T3a 和 T2 執行個體。如果您定期爆增超過基準線，請確定您可以根據新執行個體類型的 vCPU 繼續執行此作業。如需詳細資訊，請參閱 [爆量效能執行個體的重要概念和定義](#)。
- 如果您購買的是預留執行個體，隨需執行個體可能會以預留執行個體方式計費。變更目前的執行個體類型之前，請先評估對預留執行個體使用率和涵蓋範圍的影響。
- 盡可能考慮轉換為較新一代的執行個體。
- 移轉至不同的執行個體系列時，請確定目前的執行個體類型和新的執行個體類型是相容的，例如在虛擬化、架構或網路類型方面。如需詳細資訊，請參閱 [變更執行個體類型的相容性](#)。
- 最後，請考慮針對每個建議所提供的效能風險評分。效能風險指出您可能需要花費的精力，以驗證建議的執行個體類型是否符合工作負載的效能需求。我們也建議在進行任何變更之前和之後進行嚴格的負載和效能測試。

調整 EC2 執行個體大小時還有其他考量事項。如需詳細資訊，請參閱 [變更執行個體類型](#)。

其他資源

如需詳細資訊：

- [Amazon EC2 執行個體類型](#)
- [AWS Compute Optimizer 使用者指南](#)

變更執行個體類型

隨著您的需求變更，您可能會發現您的執行個體過度利用 (執行個體類型太小)，或利用不足 (執行個體類型太大)。若出現此情況，您可以透過變更執行個體類型來調整執行個體的大小。例如，如果您的 t2.micro 執行個體對於其工作負載來說太小，則您可以透過將其變更為較大的 T2 執行個體類型 (例如 t2.large) 來增加其大小。或者，您可以將其變更為另一個執行個體類型，例如 m5.large。您可能也會希望將先前世代的執行個體類型變更為目前世代的執行個體類型，以利用某些功能，例如 IPv6 的支援。

如果想要獲得有關最能處理現有工作負載的執行個體類型建議，可以使用 AWS Compute Optimizer。如需詳細資訊，請參閱 [取得現有工作負載的執行個體類型建議](#)。

若變更執行個體類型，您將開始支付新執行個體類型的費率。如需所有執行個體類型的隨需費率，請參閱 [Amazon EC2 隨需定價](#)。

若要在不變更執行個體類型的情況下為執行個體新增額外儲存空間，請將 EBS 磁碟區新增至執行個體。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的將 Amazon EBS 磁碟區連接至執行個體](#)。

要遵循哪些指示？

變更執行個體類型有不同的指示。要使用的指示取決於執行個體的根磁碟區，以及執行個體類型是否與執行個體的目前組態相容。如需如何判斷相容性的相關資訊，請參閱 [變更執行個體類型的相容性](#)。

使用下表判斷要遵循哪些指示。

根磁碟區	相容性	使用這些指示
EBS	相容	變更 EBS 支援執行個體的執行個體類型
EBS	不相容	透過啟動新執行個體來變更執行個體類型
執行個體存放區	不適用	變更執行個體存放區後端執行個體的執行個體類型

相容執行個體類型的考量事項

變更現有執行個體的執行個體類型時，請考慮下列事項：

- 您必須先停止 Amazon EBS 為後端之執行個體，才能變更其執行個體類型。請確認您已為執行個體停止時的停機時間做好規劃。停止執行個體並變更其執行個體類型可能會花費數分鐘，重新啟動您的執行個體所需要花費的時間則會根據您應用程式的啟動指令碼而有所不同。如需詳細資訊，請參閱 [停止並啟動 Amazon EC2 執行個體](#)。
- 當您停止並啟動執行個體時，我們會將執行個體移至新硬體。若您的執行個體具有公有 IPv4 地址，我們會釋出地址並給予您的執行個體新的公有 IPv4 地址。如果您需要不會變更的公有 IPv4 地址，請使用 [彈性 IP 地址](#)。
- 您無法變更 [Spot 執行個體](#) 的執行個體類型。
- [Windows 執行個體] 建議您在變更執行個體類型之前更新 AWS PV 驅動程式套件。如需詳細資訊，請參閱 [the section called “升級 PV 驅動程式”](#)。
- 若您的執行個體位於 Auto Scaling 群組中，Amazon EC2 Auto Scaling 服務會將停止的執行個體標記為運作狀態不良，且可能會終止它並啟動取代用執行個體。若要避免這種情況，您可以在變更執行個體類型時，暫停群組的擴展程序。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南中的 [暫停及繼續擴展程序](#)。
- 當您變更具有 NVMe 執行個體存放區磁碟區之執行個體的執行個體類型時，更新後的執行個體可能具有其他執行個體存放區磁碟區，因為即使未在 AMI 或執行個體區塊型設備映射中加以指定，所有 NVMe 執行個體存放區磁碟區都可用。否則，更新後的執行個體通常會有與您在啟動原始執行個體時指定之執行個體儲存體磁碟區相同數目的執行個體儲存體磁碟區。
- 可連接到執行個體的 Amazon EBS 磁碟區數目上限，取決於執行個體類型和執行個體大小。如果執行個體類型或執行個體大小不支援已連接至執行個體的磁碟區數量，您就無法變更為這樣的執行個體類型或執行個體大小。如需詳細資訊，請參閱 [執行個體磁碟區限制](#)。

變更 EBS 支援執行個體的執行個體類型

如果您想要的執行個體類型與執行個體的目前組態相容，請使用下列指示來變更 EBS 後端執行個體的執行個體類型。

變更 Amazon EBS 為後端之執行個體的執行個體類型

1. (選用) 若新執行個體類型需要的驅動程式並未安裝於現有的執行個體，您必須先連接您的執行個體並安裝驅動程式。如需詳細資訊，請參閱 [變更執行個體類型的相容性](#)。

2. [Windows 執行個體] 如果您將 Windows 執行個體設定為使用靜態 IP 位址，而且從不支援增強型聯網的執行個體類型變更為支援增強型聯網的執行個體類型，則當您重新設定靜態 IP 位址位址時，可能會收到有關潛在 IP 位址衝突的警告。為避免出現警告，請在執行個體的網路介面上啟用 DHCP，然後再變更執行個體類型。在執行個體中開啟 Network and Sharing Center (網路和共用中心)，開啟網路介面的 Internet Protocol Version 4 (TCP/IPv4) Properties (網際網路通訊協定第 4 版 (TCP/IPv4) 內容)，然後選擇 Obtain an IP address automatically (自動取得 IP 地址)。在網路介面上變更執行個體類型，然後重新設定靜態 IP 地址。
3. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
4. 在導覽窗格中，選擇 Instances (執行個體)。
5. 選取執行個體，並選取 Instance state (執行個體狀態)、Stop instance (停止執行個體)。出現確認提示時，請選擇 Stop (停止)。停止執行個體可能需要幾分鐘。
6. 在仍然選取執行個體的情況下，選取 Actions (動作)、Instance settings (執行個體設定)、Change instance type (變更執行個體類型)。如果執行個體狀態不是 stopped，此選項會呈現灰色。
7. 在 Change instance type (變更執行個體類型) 頁面上，執行下列動作：
 - a. 針對 Instance type (執行個體類型)，選取您想要的執行個體類型。

如果執行個體類型不在清單中，表示其與您執行個體的組態不相容。請改用下列指示：[透過啟動新執行個體來變更執行個體類型](#)。

- b. (選用) 如果您選取的執行個體類型支援 EBS 最佳化，請選取 EBS-optimized (EBS 最佳化) 以啟用 EBS 最佳化，或取消選取 EBS-optimized (EBS 最佳化) 以停用 EBS 最佳化。如果您選取的執行個體類型預設為 EBS 最佳化，則 EBS-optimized (EBS 最佳化) 便會處於選取狀態，而且您無法取消選取。
 - c. 選擇 Apply (套用) 以接受新的設定。
8. 若要啟動執行個體，請選取執行個體，然後選取 Instance state (執行個體狀態)、Start instance (啟動執行個體)。執行個體進入 running 狀態可能需要數分鐘。若您的執行個體未啟動，請參閱[對變更執行個體類型進行疑難排解](#)。
9. [視窗執行個體] 如果您的執行個體使用 EC2Launch v1 執行 Windows 伺服器 2016 或 Windows 伺服器 2019，請連線至您的 Windows 執行個體，並執行下列 EC2Launch PowerShell 指令碼，在執行個體類型變更後設定執行個體。

⚠ Important

當您啟用初始化執行個體 EC2 Launch 指令碼時，管理員密碼將重設。您可以修改組態檔來停用管理員密碼重設，方法為在初始化任務的設定中指定它。如需如何停用密碼重設的步驟，請參閱[設定初始化工作 \(EC2Launch\)](#) 或[變更設定 \(EC2Launch v2\)](#)。

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 - Schedule
```

透過啟動新執行個體來變更執行個體類型

如果 EBS 後端執行個體目前的組態與您想要的新執行個體類型不相容，則您無法變更原始執行個體的執行個體類型。反之，您必須搭配與您想要之新執行個體類型相容的組態，啟動新的執行個體，然後將您的應用程式遷移到新的執行個體。例如，如果您從 PV AMI 啟動原始執行個體，但想要變更為需要 HVM AMI 的目前一代執行個體類型，則需要從 HVM AMI 啟動新執行個體。如需如何判斷相容性的相關資訊，請參閱[變更執行個體類型的相容性](#)。

若要將您的應用程式遷移到新的執行個體，請執行下列動作：

- 備份原始執行個體上的資料。
- 搭配與您想要之新執行個體類型相容的組態，啟動新的執行個體，並連接任何已連接至原始執行個體的 EBS 磁碟區。
- 在新的執行個體上安裝您的應用程式及任何軟體。
- 還原任何資料。
- 如果原始執行個體具有彈性 IP 地址，而且您想要確保您的使用者可以繼續在新的執行個體上使用應用程式而不會中斷，則您必須將彈性 IP 地址與新的執行個體建立關聯。如需詳細資訊，請參閱[彈性 IP 地址](#)。

變更新執行個體組態的執行個體類型

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 備份您需要保留的資料，如下所示：
 - 對於執行個體存放區磁碟區上的資料，請將資料備份至持久性儲存。

- 對於 EBS 磁碟區上的資料，請建立磁碟區快照或將磁碟區與執行個體分離，以便稍後將它們附加到新的執行個體。
3. 在導覽窗格中，選擇執行個體。
 4. 選擇啟動執行個體。設定執行個體時，請執行下列動作：
 - a. 選取將支援您想要之執行個體類型的 AMI。請注意，目前世代執行個體類型需要 HVM AMI。
 - b. 選取您想要的新執行個體類型。如果您想要的執行個體類型無法使用，表示該類型與您選取的 AMI 組態不相容。
 - c. 如果您是使用彈性 IP 地址，請選取原始執行個體目前執行所在的 VPC。
 - d. 若您希望允許相同的流量觸達新執行個體，請選取與原始執行個體相關聯的安全群組。
 - e. 完成設定新的執行個體後，請完成步驟來選取金鑰對並啟動您的執行個體。執行個體進入 running 狀態可能需要數分鐘。
 5. 如有需要，根據您建立的快照，連接任何新的 EBS 磁碟區，或將任何您從原始執行個體分離的 EBS 磁碟區連接到新的執行個體。
 6. 在新的執行個體上安裝您的應用程式及任何必要的軟體。
 7. 還原您從原始執行個體的執行個體儲存體磁碟區備份的任何資料。
 8. 如果您是使用彈性 IP 地址，請將其指派給新的執行個體，如下所示：
 - a. 在導覽窗格中，選擇 Elastic IPs (彈性 IP)。
 - b. 選取與原始執行個體相關聯的彈性 IP 地址，選取 Actions (動作)，然後選取 Disassociate Elastic IP address (與彈性 IP 地址取消關聯)。出現確認提示時，請選擇 Disassociate (取消關聯)。
 - c. 在彈性 IP 地址仍在選取狀態的情況下，選取 Actions (動作)，然後選取 Associate Elastic IP address (與彈性 IP 地址建立關聯)。
 - d. 對於 Resource type (資源類型)，選擇 Instance (執行個體)。
 - e. 針對 Instance (執行個體)，選擇要與彈性 IP 地址建立關聯的新執行個體。
 - f. (選用) 對於 Private IP address (私人 IP 地址)，指定要與彈性 IP 地址建立關聯的私人 IP 地址。
 - g. 選擇 Associate (關聯)。
 9. (選用) 若您不再需要原始執行個體，則可將其終止。選取執行個體，確認您要終止的是原始執行個體，而非新的執行個體 (例如，檢查名稱或啟動時間)，然後選取 Instance state (執行個體狀態)、Terminate instance (終止執行個體)。

變更執行個體類型的相容性

只在執行個體的目前組態與您想要的執行個體類型相容時，您才能變更執行個體類型。如果您想要的執行個體類型與執行個體的目前狀態不相容，您必須搭配與執行個體類型相容的組態，啟動新的執行個體，然後將您的應用程式遷移到新的執行個體。

[Linux 執行個體] 您可以使用 `AWSsupport-MigrateXenToNitroLinux` 執行本，將相容的 Linux 執行個體從 Xen 執行個體類型移轉為 Nitro 執行個體類型。如需詳細資訊，請參閱 [AWS Systems Manager Automation Runbook 參考](#) 中的 [AWSsupport-MigrateXenToNitroLinux runbook](#)。

[Windows 執行個體] 如需將相容 Windows 執行個體從 Xen 執行個體類型移轉至 Nitro 執行個體類型的其他指引，請參閱 [移轉至最新一代執行個體類型](#)。

相容性的判斷方式如下：

虛擬化類型

Linux AMI 會使用兩種虛擬化類型的其中之一：全虛擬化 (PV) 或硬體虛擬機器 (HVM)。如果已從 PV AMI 啟動執行個體，則您無法將其變更為僅限 HVM 的執行個體類型。如需詳細資訊，請參閱 [AMI 虛擬化類型](#)。若要檢查您執行個體的虛擬化類型，請檢查 Amazon EC2 主控台中 Instances (執行個體) 畫面詳細資訊窗格上的 Virtualization (虛擬化) 值。

架構

AMI 是處理器特有架構，所以您選取的執行個體類型，必須和目前的執行個體類型有相同的處理器架構。例如：

- 如果目前執行個體類型具有 Arm 架構處理器，您會受限於支援 Arm 架構處理器的執行個體類型，例如 C6g 和 M6g。
- 下列執行個體類型是唯一支援 32 位元 AMIs 的執行個體類型：t2.nano、t2.micro、t2.small、t2.medium、c3.large、t1.micro、m1.small、m1.medium 和 c1.medium。如果您要變更 32 位元執行個體的執行個體類型，則會受限於這些執行個體類型。

網路介面卡

如果您從某個網路介面卡的驅動程式切換到另一個網路介面卡，該網路介面卡設定會在作業系統建立新的介面卡時重設。若要重新進行設定，您可能需要存取具有系統管理員許可的本機帳戶。以下是從某個網路介面卡移至另一個網路介面卡的範例：

- AWS 光伏 (T2 實例) 到英特爾 82599 VF (M4 實例)
- Intel 82599 VF (大多數 M4 執行個體) 至 ENA (M5 執行個體)

- ENA (M5 執行個體) 至高頻寬 ENA (M5n 執行個體)

網路卡

某些執行個體類型支援多個[網路卡](#)。您必須選取一個執行個體類型，它支援與目前執行個體類型相同數量的網路卡。

增強型網路

支援[增強型聯網](#)的執行個體類型需要安裝必要的驅動程式。例如，在 [AWS Nitro 系統上建置的執行個體](#)需要安裝彈性網路介面卡 (ENA) 驅動程式的 EBS 支援 AMI。若要將不支援增強型聯網的執行個體類型變更為支援增強型聯網的執行個體類型，您必須視需要在執行個體上安裝 [ENA 驅動程式](#)或 [ixgbevf 驅動程式](#)。

Note

當您在啟用 ENA Express 的情況下調整執行個體大小時，新的執行個體類型也必須支援 ENA Express。如需支援 ENA Express 的執行個體類型清單，請參閱 [ENA Express 支援的執行個體類型](#)。

若要將支援 ENA Express 的執行個體類型變更為不支援其的執行個體類型，請確定目前在調整執行個體大小之前，ENA Express 未啟用。

NVMe

EBS 磁碟區會在 [AWS Nitro 系統上建置的執行個體](#)上公開為 NVMe 區塊裝置。如果將不支援 NVMe 的執行個體類型變更為支援 NVMe 的執行個體類型，您必須先在執行個體上安裝 NVMe 驅動程式。此外，您在區塊裝置對應中指定之裝置的裝置名稱也會使用 NVMe 裝置名稱 (/dev/nvme[0-26]n1) 來重新命名。

[Linux 實例] 因此，要在啟動時使用掛載文件系統/etc/fstab，您必須使用 UUID/Label 而不是設備名稱。

磁碟區限制

可連接到執行個體的 Amazon EBS 磁碟區數目上限，取決於執行個體類型和執行個體大小。如需詳細資訊，請參閱 [執行個體磁碟區限制](#)。

只有執行個體類型或執行個體大小支援的磁碟區數量，等於或大於目前連接至執行個體的磁碟區數量，您才能變更為這樣的執行個體類型或執行個體大小。如果您變更為不支援目前連接磁碟區數量的執行個體類型或執行個體大小，則該請求會失敗。例如，如果您從具有 32 個連接磁碟區的 m7i.4xlarge 執行個體變更為最多支援 27 個磁碟區的 m6i.4xlarge，則該請求會失敗。

對變更執行個體類型進行疑難排解

請使用以下資訊來協助您診斷並修正在變更執行個體類型時可能遇到的問題。

在變更執行個體類型之後執行個體不會啟動

可能原因：未符合新執行個體類型的需求

如果您的執行個體未啟動，則原因可能是未符合新執行個體類型的其中一個需求。如需詳細資訊，請參閱[為何我的 Linux 執行個體在我變更其類型之後未啟動？](#)

可能原因：AMI 不支援執行個體類型

如果您使用 EC2 主控台變更執行個體類型，則只能使用所選 AMI 支援的執行個體類型。不過，如果您使用啟動執行個體，您可以指定不相容的 AMI 和執行個體類型。AWS CLI 如果 AMI 與執行個體類型不相容，則執行個體無法啟動。如需詳細資訊，請參閱[變更執行個體類型的相容性](#)。

可能原因：執行個體位於集群放置群組中

如果您的執行個體位於[集群放置群組](#)中，並在變更執行個體類型之後，執行個體無法啟動，請嘗試下列動作：

1. 停止集群放置群組中的所有執行個體。
2. 變更受影響執行個體的執行個體類型。
3. 啟動集群放置群組中的所有執行個體。

在變更執行個體類型之後，無法從網際網路連線到應用程式或網站

可能原因：公有 IPv4 地址已釋出

變更執行個體類型時，您必須先停止執行個體。當您停止執行個體時，我們會釋出公有 IPv4 地址，並給予您的執行個體新的公有 IPv4 地址。

若要在執行個體停止和啟動之間保留公有 IPv4 地址，建議您使用彈性 IP 地址，只要您的執行個體正在執行，就不需要額外費用。如需詳細資訊，請參閱[彈性 IP 地址](#)。

變更執行個體存放區後端執行個體的執行個體類型

執行個體存放區後端執行個體是具有執行個體存放區根磁碟區的執行個體。您無法變更具具有執行個體存放區根磁碟區之執行個體的執行個體類型。反之，您必須從執行個體建立 AMI、從這個 AMI 啟動新的

原始執行個體，並選取您想要的執行個體類型，然後將您的應用程式遷移到新的執行個體。請注意，您想要的執行個體類型必須與您建立的 AMI 相容。如需如何判斷相容性的相關資訊，請參閱 [變更執行個體類型的相容性](#)。

程序概觀

- 備份原始執行個體上的資料。
- 從原始執行個體建立 AMI
- 從這個 AMI 啟動新的執行個體，然後選取您想要的執行個體類型。
- 在新的執行個體上安裝應用程式。
- 如果原始執行個體具有彈性 IP 地址，而且您想要確保您的使用者可以繼續在新的執行個體上使用應用程式而不會中斷，則您必須將彈性 IP 地址與新的執行個體建立關聯。如需詳細資訊，請參閱 [彈性 IP 地址](#)。

變更執行個體存放區後端執行個體的執行個體類型

1. 備份您需要保留的資料，如下所示：
 - 對於執行個體存放區磁碟區上的資料，請將資料備份至持久性儲存。
 - 對於 EBS 磁碟區上的資料，請建立磁碟區快照或將磁碟區與執行個體分離，以便稍後將其附加至新的執行個體。
2. 透過滿足先決條件並遵循 [建立執行個體存放區後端 Linux AMI](#) 中的程序，從您的執行個體建立 AMI。在您完成從您的執行個體建立 AMI 後，返回此步驟。
3. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
4. 在導覽窗格中，選擇 AMIs (AMI)。從篩選條件清單中，選取 Owned by me (由我擁有)，然後選取您在步驟 2 中建立的映像。注意，AMI Name (AMI 名稱) 為您在註冊映像時指定的名稱，Source (來源) 則為您的 Amazon S3 儲存貯體。

Note

如果未看到您在步驟 2 中建立的 AMI，請確定您已選取您在其中建立 AMI 的區域。

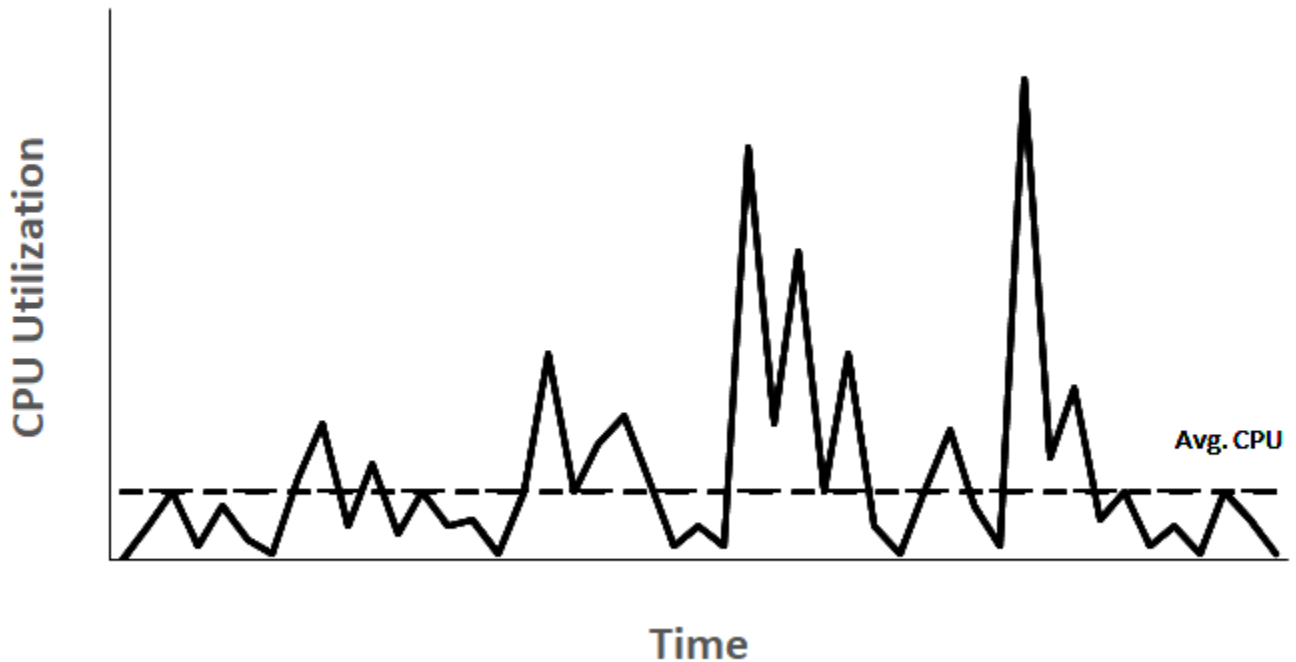
5. 選取 AMI 後，選取 Launch instance from image (從映像啟動執行個體)。設定執行個體時，請執行下列動作：
 - a. 選取您想要的新執行個體類型。如果您想要的執行個體類型無法使用，表示該類型與您建立的 AMI 組態不相容。如需詳細資訊，請參閱 [變更執行個體類型的相容性](#)。

- b. 如果您是使用彈性 IP 地址，請選取原始執行個體目前執行所在的 VPC。
 - c. 若您希望允許相同的流量觸達新執行個體，請選取與原始執行個體相關聯的安全群組。
 - d. 完成設定新的執行個體後，請完成步驟來選取金鑰對並啟動您的執行個體。執行個體進入 running 狀態可能需要數分鐘。
6. 如有需要，根據您建立的快照，連接任何新的 EBS 磁碟區，或將任何您從原始執行個體分離的 EBS 磁碟區連接到新的執行個體。
 7. 在新的執行個體上安裝您的應用程式及任何必要的軟體。
 8. 如果您是使用彈性 IP 地址，請將其指派給新的執行個體，如下所示：
 - a. 在導覽窗格中，選擇 Elastic IPs (彈性 IP)。
 - b. 選取與原始執行個體相關聯的彈性 IP 地址，選取 Actions (動作)，然後選取 Disassociate Elastic IP address (與彈性 IP 地址取消關聯)。出現確認提示時，請選擇 Disassociate (取消關聯)。
 - c. 在彈性 IP 地址仍在選取狀態的情況下，選取 Actions (動作)，然後選取 Associate Elastic IP address (與彈性 IP 地址建立關聯)。
 - d. 對於 Resource type (資源類型)，選擇 Instance (執行個體)。
 - e. 針對 Instance (執行個體)，選擇要與彈性 IP 地址建立關聯的新執行個體。
 - f. (選用) 對於 Private IP address (私人 IP 地址)，指定要與彈性 IP 地址建立關聯的私人 IP 地址。
 - g. 選擇 Associate (關聯)。
 9. (選用) 若您不再需要原始執行個體，則可將其終止。選取執行個體，確認您要終止的是原始執行個體，而非新的執行個體 (例如，檢查名稱或啟動時間)，然後選取 Instance state (執行個體狀態)、Terminate instance (終止執行個體)。

爆量效能執行個體

許多一般用途工作負載平均起來並不忙碌，而且不需要高級別的持續 CPU 效能。下圖說明客戶目前在 AWS 雲端執行的許多常見工作負載的 CPU 使用率。

Many common workloads look like this



這些 low-to-moderate CPU 使用率工作負載會導致 CPU 週期浪費，因此您需要支付超過使用量的費用。為了克服此問題，您可以利用低成本爆量一般用途執行個體，也就是 T 執行個體。

T 執行個體系列提供基準 CPU 效能，能夠隨時長時間超越基準高效能執行。基準 CPU 的定義是為了滿足大多數一般用途工作負載的需求，包括大規模微服務、Web 伺服器、中小型資料庫、資料記錄、程式碼儲存庫、虛擬桌面、開發和測試環境，以及業務關鍵應用程式。T 執行個體可在運算、記憶體和網路資源之間取得平衡，並提供最具成本效益的方式，讓您執行各種具有 low-to-moderate CPU 使用率的一般用途應用程式。與 M 執行個體相比，這些執行個體最多可節省 15% 的成本，並且透過更小、更經濟的執行個體大小可節省更多成本，提供最低 2 個 vCPU 和 0.5 GiB 的記憶體。較小的 T 執行個體大小 (例如超微型、微型、小型和中型) 非常適合需要少量記憶體且不需要高 CPU 使用率的工作負載。

Note

本主題說明爆量 CPU。如需有關爆量網路效能的詳細資訊，請參閱 [Amazon EC2 執行個體網路頻寬](#)。

EC2 爆量執行個體類型

EC2 爆量執行個體包含 T4g、T3a 和 T3 執行個體類型以及前一代 T2 執行個體類型。

T4g 執行個體類型是最新一代的爆量執行個體。它們提供最佳的效能價格，並為您提供所有 EC2 執行個體類型中最低的成本。T4G 執行個體類型由 ARM 架構的 [AWS Graviton2](#) 處理器提供支援，並提供來自作業系統廠商、獨立軟體廠商以及熱門服務和應用程式的廣泛生態系統支援。AWS

下表摘要列出爆量執行個體類型之間的主要差異。

類型	描述	處理器系列
最新一代		
T4g	成本最低的 EC2 執行個體類型，與 T3 相比價格/效能高 40%，成本低 20%	AWS 重力 2 處理器配備臂 NEOVERSE N1 核心
T3a	成本最低的 x86 型執行個體，成本比 T3 執行個體低 10%	AMD 第一代 EPYC 處理器
T3	適用於 x86 工作負載的最佳峰值價格/效能，比上一代 T2 執行個體的價格/效能低 30%	Intel Xeon 可擴充處理器 (Skylake、Cascade Lake 處理器)
上一代		
T2	上一代爆量執行個體	Intel Xeon 處理器

如需執行個體定價和其他規格的相關資訊，請參閱 [Amazon EC2 定價](#) 和 [Amazon EC2 執行個體類型](#)。如需有關爆量網路效能的詳細資訊，請參閱 [Amazon EC2 執行個體網路頻寬](#)。

如果您的帳戶剛建立不到 12 個月，可以在一定的使用限額內免費使用 t2.micro 執行個體 (或 t3.micro 無法使用的區域中的 t2.micro 執行個體)。如需詳細資訊，請參閱 [AWS 免費方案](#)。

T 執行個體支援的購買選項

- On-Demand Instances
- Reserved Instances
- 專用執行個體 (僅限 T3)

- 專用執行個體 (僅限 T3，僅限 standard 模式)
- Spot 執行個體

如需詳細資訊，請參閱 [執行個體購買選項](#)。

內容

- [最佳實務](#)
- [爆量效能執行個體的重要概念和定義](#)
- [爆量效能執行個體的無限制模式](#)
- [爆量效能執行個體的標準模式](#)
- [使用爆量效能執行個體](#)
- [監控爆量效能執行個體的 CPU 額度](#)

最佳實務

請遵循這些最佳實務，讓高載效能執行個體發揮最大的效益。

- 請確定您所選擇的執行個體容量，高於作業系統和應用程式的最低記憶體容量需求。使用圖形化使用者介面的作業系統 (例如 Windows) 可能會耗用大量記憶體和 CPU 資源，因此在許多使用案例中可能需要 t3.micro 或更大的執行個體大小。隨著工作負載對記憶體和 CPU 的需求隨時間增加，使用 T 執行個體可獲得一定的靈活性，您可以擴展到相同執行個體類型的較大執行個體大小，或選取其他執行個體類型。
- 為您的帳戶啟用 [AWS Compute Optimizer](#)，並檢閱工作負載的 Compute Optimizer 建議。Compute Optimizer 可協助評估執行個體是否應該擴大規模以提高效能，還是應縮小規模以節省成本。Compute Optimizer 也可能會根據您的情況建議不同的執行個體類型。如需詳細資訊，請參閱《AWS Compute Optimizer 使用者指南》中的 [檢視 EC2 執行個體建議](#)。

爆量效能執行個體的重要概念和定義

傳統的 Amazon EC2 執行個體類型提供固定的 CPU 資源；爆量效能執行個體則提供 CPU 基準使用率，並可大幅提升 CPU 使用率至超越基準水準。這可確保您只需支付基準 CPU 外加任何額外的爆量 CPU 使用的費用，進而降低運算成本。基準使用率與大幅提升效能的能力，取決於 CPU 的額度。高載效能執行個體是唯一透過額度來使用 CPU 的執行個體類型。

當每個爆量效能執行個體維持在 CPU 基準以下時，會持續獲得額度，並在爆量超出基準時持續消耗額度。獲得或消耗的額度取決於執行個體的 CPU 使用率：

- 如果 CPU 使用率低於基準，則獲得的額度大於消耗的額度。
- 如果 CPU 使用率等於基準，則獲得的額度等於消耗的額度。
- 如果 CPU 使用率高於基準，則消耗的額度大於獲得的額度。

當獲得的額度大於消耗的額度時，差額稱為累積額度，它稍後可用於超出基準 CPU 使用率。同樣地，如果消耗的額度大於獲得的額度，則執行個體行為取決於額度組態模式 – 標準模式或無限制模式。

在標準模式中，當消耗的額度超過獲得的額度時，執行個體會使用累積額度以超出基準 CPU 使用率。如果沒有剩餘的累積額度，則執行個體會逐漸降至基準 CPU 使用率，並且無法超出基準線，直到累積更多額度為止。

在無限制模式中，如果執行個體超出基準 CPU 使用率，則執行個體會先使用累積額度進行超出。如果沒有剩餘的累積額度，則執行個體會消耗剩餘額度進行超出。當 CPU 使用率掉到基準以下，該執行個體會使用自己先前所獲得的 CPU 額度，來償還先前用掉的剩餘額度。獲得 CPU 額度來償還剩餘額度的能力，讓 Amazon EC2 能夠在 24 小時的期間，實現平均的執行個體 CPU 使用率。如果在 24 小時的期間，CPU 平均使用量超過基準，則會以每 vCPU 小時的[其他固定費率](#)，向執行個體收取其他使用量的費用。

目錄

- [重要概念和定義](#)
- [獲得 CPU 額度](#)
- [CPU 額度獲得率](#)
- [CPU 額度累積限額](#)
- [累積的 CPU 額度壽命](#)
- [基準使用率](#)

重要概念和定義

下列重要概念和定義適用於爆量效能執行個體。

CPU 使用率

CPU 使用率是執行個體上目前正在使用的已分配的 EC2 運算單位的百分比。此指標測量執行個體正在使用的已分配的 CPU 週期的百分比。「CPU 使用率」 CloudWatch 指標顯示每個執行個體的 CPU 使用率，而非每個核心的 CPU 使用率。執行個體的基準 CPU 規格也以每個執行個體的 CPU 使用率為基礎。若要使用 AWS Management Console 或來測量 CPU 使用率 AWS CLI，請參閱[取得特定執行個體的統計資料](#)。

CPU 額度

vCPU 時間單位。

範例：

1 個 CPU 額度 = 1 個 vCPU * 100% 使用率 * 1 分鐘。

1 個 CPU 額度 = 1 個 vCPU * 50% 使用率 * 2 分鐘。

1 個 CPU 額度 = 2 個 vCPU * 25% 使用率 * 2 分鐘。

基準使用率

基準使用率是在獲得的 CPU 額度符合使用的 CPU 額度時，可用於零的淨額度餘額的 CPU 水準。基準使用率也稱為基準。基準使用率表示為 vCPU 使用率百分比，其計算方式如下：基準使用率百分比 = (獲得的額度/vCPU 數)/60 分鐘。

如需每個爆量效能執行個體類型的基準使用率，請參閱[點數表](#)。

獲得的點數

執行個體在執行時持續獲得的額度。

每小時獲得的額度 = 基準使用率百分比 * vCPU 數量 * 60 分鐘

範例：

具有 2 個 vCPU 和 5% 基準使用率的 t3.nano 每小時可獲得 6 個額度，其計算方式如下：

2 個 vCPU * 5% 基準使用率 * 60 分鐘 = 每小時 6 個額度

消耗或使用的額度

執行個體在執行時持續使用的額度。

每分鐘消耗的 CPU 額度 = vCPU 數量 * CPU 使用率 * 1 分鐘

累積額度

當執行個體使用的額度少於基準使用率所需的額度時，未消耗的 CPU 額度。換句話說，累積額度 = (獲得的額度 - 使用的額度) 低於基準。

範例：

如果在 2% CPU 使用率的情況下執行 t3.nano (低於一小時 5% 的基準)，則累積額度的計算方式如下：

累積的 CPU 額度 = (每小時獲得的額度 - 每小時使用的額度) = 6 - 2 個 vCPU * 2% CPU 使用率 * 60 分鐘 = 6 - 2.4 = 每小時 3.6 個累積額度

額度累積限額

它取決於執行個體大小，但通常等於 24 小時內獲得的最大額度。

範例：

對於 t3.nano，額度累積限額 = 24 * 6 = 144 個額度

啟動額度

僅適用於為標準模式設定的 T2 執行個體。啟動額度是分配給新 T2 執行個體的有限 CPU 額度，因此在標準模式下啟動時，它可以超出基準線。

剩餘額度

執行個體耗盡其累積額度餘額之後所消耗的額度。剩餘額度是專為爆量執行個體所設計，以便長時間維持高效能，並且僅用於無限制模式。剩餘額度餘額用於決定執行個體在無限制模式下爆量使用了多少額度。

標準模式

額度組態模式，透過消耗掉額度餘額中所累積的額度，此模式可讓執行個體超出基準。

無限制模式

額度組態模式，透過在任何期間皆隨需提供高 CPU 使用率，此模式可讓執行個體超出基準。如果在累計 24 小時的期間或執行個體的生命週期期間 (以較短者為準)，執行個體的 CPU 平均使用率等於或低於基準，則執行個體的每小時定價將自動涵蓋所有 CPU 使用量峰值。如果執行個體需要長時間以更高的 CPU 使用率執行，也可使用每 vCPU 小時的[其他固定費率](#)來執行。

下表摘要列出爆量執行個體類型之間的主要額度差異。

類型	支援的 CPU 額度類型	額度組態模式	執行個體啟動和停止之間累積的 CPU 額度的使用壽命
最新一代			
T4g	獲得的額度、累積的額度、消耗的額度、	標準、無限制 (預設)	7 天 (額度會在執行個體停止後保留 7 天)

類型	支援的 CPU 額度類型	額度組態模式	執行個體啟動和停止之間累積的 CPU 額度的使用壽命
	剩餘額度 (僅限無限制模式)		
T3a	獲得的額度、累積的額度、消耗的額度、剩餘額度 (僅限無限制模式)	標準、無限制 (預設)	7 天 (額度會在執行個體停止後保留 7 天)
T3	獲得的額度、累積的額度、消耗的額度、剩餘額度 (僅限無限制模式)	標準、無限制 (預設)	7 天 (額度會在執行個體停止後保留 7 天)
上一代			
T2	獲得的額度、累積的額度、消耗的額度、啟動額度 (僅限標準模式)、剩餘額度 (僅限無限制模式)	標準 (預設)、無限制	0 天 (當執行個體停止時，額度會消失)

Note

專用執行個體上啟動的 T3 執行個體不支援無限制模式。

獲得 CPU 額度

視執行個體的大小而定，每個爆量效能執行個體每小時會持續獲得 (以毫秒級的解析度) 固定比率的 CPU 額度。管理額度累積或使用的會計程序，也會以毫秒級的解析度進行，因此您無需擔心 CPU 額度超支；CPU 的短時間突發只會使用一小部分的 CPU 額度。

如果爆量效能執行個體所使用的 CPU 資源比基準使用率所需的還少 (例如在閒置時)，則未使用的 CPU 額度會累計到 CPU 額度餘額。如果爆量效能執行個體的效能需要突增到超過基準使用率，將會用掉累

積的額度。爆量效能執行個體所累積的額度愈多，在需要更多的 CPU 使用率時，能夠突增超過基準使用率的時間也愈長。

下表列出了爆量效能執行個體類型、每小時獲得的 CPU 額度、執行個體可累計獲得的 CPU 額度上限、每個執行個體的 vCPU 數量，以及使用單一 vCPU 時的基準使用率 (以完整核心的百分比表示)。

執行個體類型	每小時獲得的 CPU 額度	可累積的額度上限*	vCPU***	每個 vCPU 的基準使用率
T2				
t2.nano	3	72	1	5%
t2.micro	6	144	1	10%
t2.small	12	288	1	20%
t2.medium	24	576	2	20%**
t2.large	36	864	2	30%**
t2.xlarge	54	1296	4	22.5%**
t2.2xlarge	81.6	1958.4	8	17%**
T3				
t3.nano	6	144	2	5%**
t3.micro	12	288	2	10%**
t3.small	24	576	2	20%**
t3.medium	24	576	2	20%**
t3.large	36	864	2	30%**
t3.xlarge	96	2304	4	40%**
t3.2xlarge	192	4608	8	40%**
T3a				

執行個體類型	每小時獲得的 CPU 額度	可累積的額度上限*	vCPU***	每個 vCPU 的基準使用率
t3a.nano	6	144	2	5%**
t3a.micro	12	288	2	10%**
t3a.small	24	576	2	20%**
t3a.medium	24	576	2	20%**
t3a.large	36	864	2	30%**
t3a.xlarge	96	2304	4	40%**
t3a.2xlarge	192	4608	8	40%**
T4g				
t4g.nano	6	144	2	5%**
t4g.micro	12	288	2	10%**
t4g.small	24	576	2	20%**
t4g.medium	24	576	2	20%**
t4g.large	36	864	2	30%**
t4g.xlarge	96	2304	4	40%**
t4g.2xlarge	192	4608	8	40%**

* 可累計的額度等於在 24 小時期間可獲得的額度。

** 表格中所顯示的是每個 vCPU 的百分比基準使用率。在中 CloudWatch，會顯示每個 vCPU 的 CPU 使用率。例如，在基準層次運作之 t3.large 執行個體的 CPU 使用率，在 CloudWatch CPU 測量結果中會顯示為 30%。如需如何計算基準使用率的資訊，請參閱 [基準使用率](#)。

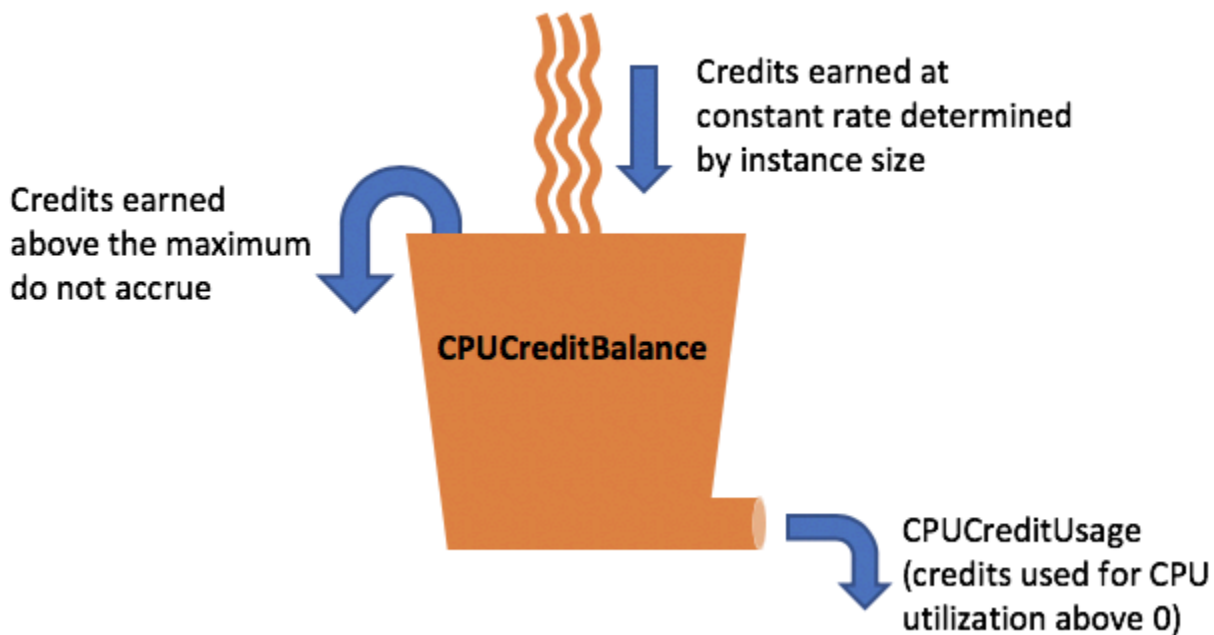
*** 每個 vCPU 都是 Intel Xeon 核心或 AMD EPYC 核心的一個執行緒，但 T2 和 T4g 執行個體除外。

CPU 額度獲得率

每小時所獲得的 CPU 額度，是由執行個體的大小所決定。例如，t3.nano 每小時可獲得 6 點，而 t3.small 每小時可獲得 24 點。上表列出所有執行個體的額度獲得率。

CPU 額度累積限額

執行中的執行個體所獲得的額度，雖然永遠不會過期，但是執行個體可累積獲得的額度有其上限。此額度上限取決於 CPU 額度餘額的限額。在達到上限之後，任何新獲得的額度都會遭到捨棄，如下圖所示。全滿的水桶代表 CPU 額度餘額的限額，溢出的水代表新獲得的額度已超過限額。



每種執行個體容量大小各有不同的 CPU 額度餘額限制。例如，t3.micro 執行個體可在 CPU 額度餘額中，累積最多 288 點獲得的 CPU 額度。上面的表格列出了每個執行個體的可累積獲得的額度上限。

T2 標準執行個體也會獲得啟動額度。啟動額度不會列入 CPU 額度餘額限額的計算。如果 T2 執行個體並未用掉其啟動額度，而且在 24 小時的期間維持閒置，同時累計獲得的額度，則其 CPU 額度餘額會顯示為超過限額。如需詳細資訊，請參閱 [啟動額度](#)。

T4g、T3a 和 T3 執行個體不會獲得啟動額度。這些執行個體會根據預設啟動為 unlimited，因此啟動時可立即大幅提升效能，而不需任何啟動額度。依預設，在專用執行個體啟動上啟動的 T3 執行個體為 standard；專用執行個體上的 T3 執行個體不支援 unlimited 模式。

累積的 CPU 額度壽命

執行中的執行個體，其 CPU 額度不會過期。

若為 T2，CPU 額度餘額不會在執行個體停止與啟動之間持續保存。如果您停止 T2 執行個體，則執行個體會失去其所有累積的額度。

對於 T4g、T3a 和 T3，CPU 額度餘額會在執行個體停止之後持續保存七天，之後額度便會消失。如果您在七天內啟動執行個體，則不會失去任何額度。

如需詳細資訊，請參閱 [CloudWatch 測量結果表格CPUCreditBalance](#) 中的。

基準使用率

基準使用率是在獲得的 CPU 額度符合使用的 CPU 額度時，可用於零的淨額度餘額的 CPU 水準。基準使用率也稱為基準。

基準使用率是以 vCPU 使用率百分比表示，其計算方式如下：

$$\text{(number of credits earned/number of vCPUs)/60 minutes} = \% \text{ baseline utilization}$$

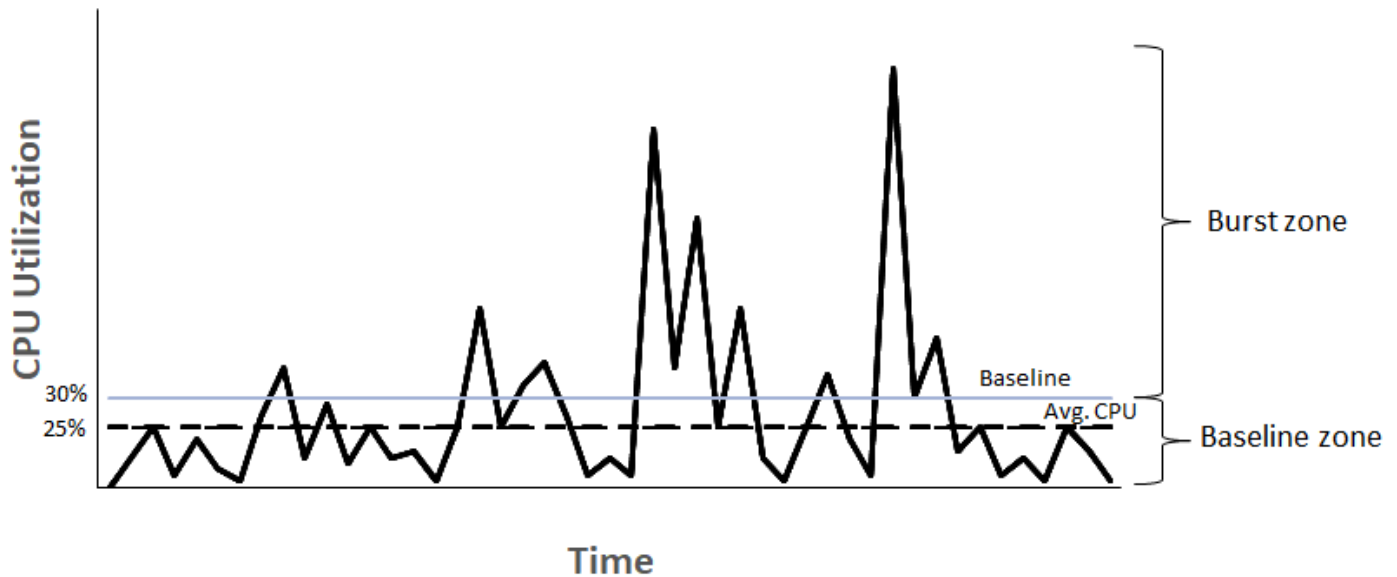
例如，具有 2 個 vCPU 的 t3.nano 執行個體每小時可獲得 6 點，因此會產生 5% 的基準使用率，其計算方式如下：

$$\text{(6 credits earned/2 vCPUs)/60 minutes} = 5\% \text{ baseline utilization}$$

具有 2 個 vCPUs 的 t3.large 執行個體每小時可獲得 36 個積分，因此基準使用率為 30% ()
(36/2)/60。

下圖提供平均 CPU 使用率低於基準線的範例。t3.large

Example of t3.large



爆量效能執行個體的無限制模式

設定為 `unlimited` 的高載效能執行個體，在任何期間皆可隨需提供高 CPU 使用率。如果在累計 24 小時的期間或執行個體的生命週期期間 (以較短者為準)，執行個體的 CPU 平均使用率等於或低於基準，則執行個體的每小時定價將自動涵蓋所有 CPU 使用量峰值。

針對絕大多數的一般用途工作負載，設定為 `unlimited` 的執行個體可提供足夠效能，而不需支付任何額外費用。如果執行個體需要長時間以更高的 CPU 使用率執行，也可使用每 vCPU 小時的其他固定費率來執行。如需有關定價的資訊，請參閱 [Amazon EC2 定價](#) 與 [T2/T3/T4 無限制模式定價](#)。

如果您使用 [AWS 免費方案](#) 中的 `t2.micro` 或 `t3.micro` 執行個體，並在 `unlimited` 模式下使用，則當累計 24 小時期間內的平均使用率超過該執行個體的 [基準使用率](#)，就可能產生費用。

T4G、T3a 和 T3 執行個體 `unlimited` 依預設啟動 (除非您 [變更](#) 預設值)。如果 24 小時期間內的平均 CPU 使用量超過基準，則剩餘額度會產生費用。如果您以 `unlimited` 形式啟動 Spot 執行個體，並計劃立即持續使用它們一段短暫的時間，且沒有產生 CPU 額度的閒置時間，則剩餘額度可能會產生費用。建議您在 [標準](#) 模式下啟動 Spot 執行個體，以免需要支付更高的費用。如需詳細資訊，請參閱 [剩餘額度可能會產生費用](#) 及 [爆量效能執行個體](#)。

Note

依預設，在專用執行個體啟動上啟動的 T3 執行個體為 standard；專用執行個體上的 T3 執行個體不支援 unlimited 模式。

目錄

- [無限制模式概念](#)
 - [無限制爆量效能執行個體如何運作](#)
 - [使用無限制模式與固定 CPU 的時機](#)
 - [剩餘額度可能會產生費用](#)
 - [T2 無限制執行個體沒有啟動額度](#)
 - [啟用無限制模式](#)
 - [在無限制與標準之間切換時，額度會有什麼變化](#)
 - [監控額度用量](#)
- [無限制模式範例](#)
 - [範例 1：說明 T3 Unlimited 的額度使用](#)
 - [範例 2：說明 T2 Unlimited 的額度使用](#)

無限制模式概念

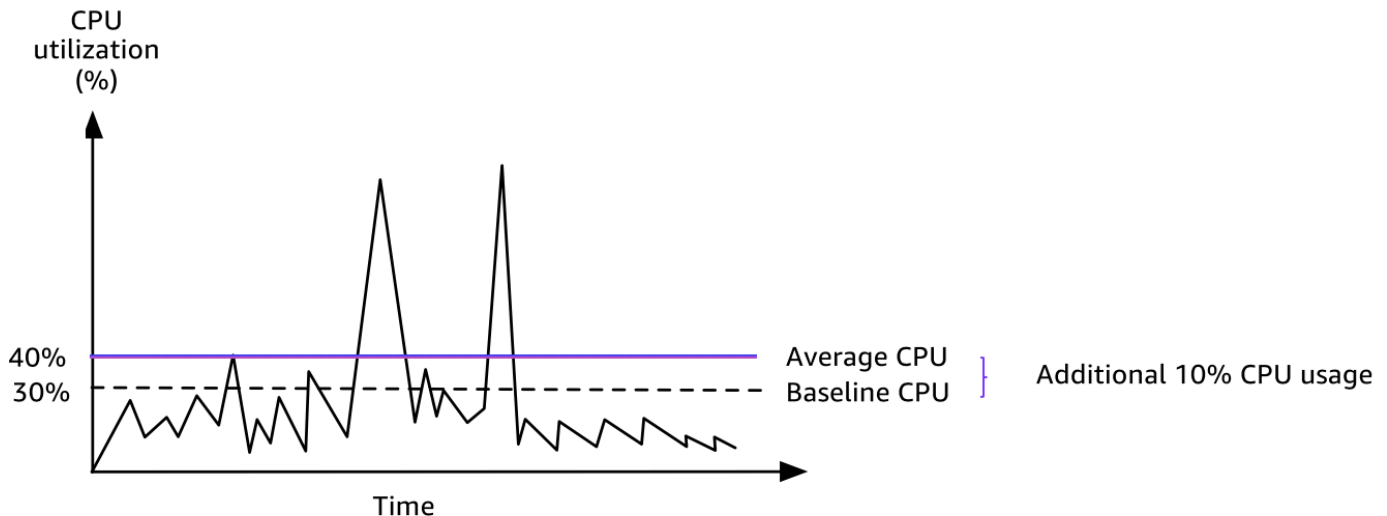
unlimited 模式是爆量效能執行個體的額度組態選項。對於執行中或已停止的執行個體，隨時都可以啟用或停用它。您可以在每個 AWS 區域、每個高載績效執行環境系列的帳戶層次[設 unlimited 定為預設信用選項](#)，以便使用預設信用選項來啟動帳戶中的所有新高載績效執行環境。

無限制爆量效能執行個體如何運作

如果設定為 unlimited 的爆量效能執行個體將其 CPU 額度餘額用完了，則可以使用多餘的額度來大幅提升效能以超過[基準](#)。當 CPU 使用率掉到基準以下，該執行個體會使用自己先前所獲得的 CPU 額度，來償還先前用掉的剩餘額度。獲得 CPU 額度來償還剩餘額度的能力，讓 Amazon EC2 能夠在 24 小時的期間，實現平均的執行個體 CPU 使用率。如果在 24 小時的期間，CPU 平均使用量超過基準，則會以每 vCPU 小時的[其他固定費率](#)，向執行個體收取其他使用量的費用。

下圖顯示 t3.large 的 CPU 使用量。t3.large 的基準 CPU 使用率為 30%。如果在 24 小時的期間，執行個體以平均 30% 或更少 CPU 使用率執行，則沒有額外費用，因為執行個體每小時定價已涵

蓋成本。不過，如果在 24 小時的期間，執行個體以平均 40% CPU 使用率執行 (如圖所示)，則會以每 vCPU 小時的[其他固定費率](#)，向執行個體收取額外 10% CPU 使用率的費用。



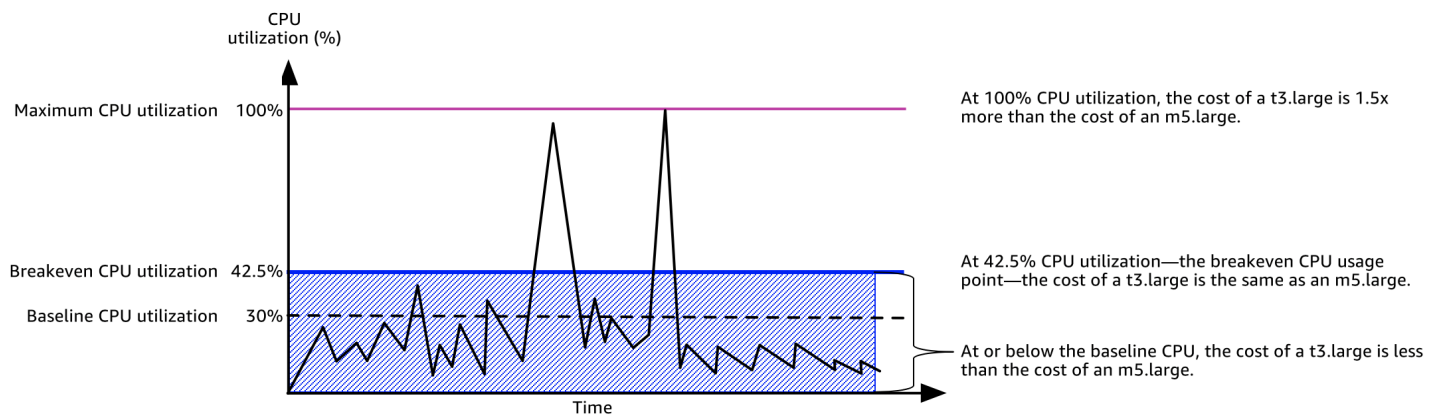
如需每一個執行個體類型的每 vCPU 基準使用率的詳細資訊，以及每一個執行個體類型可以賺取多少額度，請參閱[額度表](#)。

使用無限制模式與固定 CPU 的時機

判斷您應該在 unlimited 模式中使用爆量效能執行個體 (例如 T3)，還是固定效能執行個體 (例如 M5) 時，您需要判定損益兩平的 CPU 使用量。爆量效能執行個體的損益兩平 CPU 使用量，即是爆量效能執行個體成本與固定效能執行個體相同的那個點。損益兩平的 CPU 使用量可協助您判斷下列事項：

- 如果在 24 小時的期間，平均 CPU 使用量為等於或低於損益兩平的 CPU 使用量，請在 unlimited 模式下使用爆量效能執行個體，讓您能夠得益於爆量效能執行個體的更低價格，同時又能得到與固定效能執行個體相同的效能。
- 如果在 24 小時的期間，平均 CPU 使用量為超過損益兩平的 CPU 使用量，則爆量效能執行個體的成本將超過同等大小的固定效能執行個體。如果 T3 執行個體持續以 100% CPU 爆量，則您最終付出的成本將是同等大小 M5 執行個體的 1.5 倍。

下圖顯示損益兩平的 CPU 使用量點，其中 t3.large 的成本與 m5.large 相同。t3.large 的損益兩平 CPU 使用量點為 42.5%。如果平均 CPU 使用量為 42.5%，則執行 t3.large 的成本與 m5.large 相同，但若平均 CPU 使用量超過 42.5%，則更為昂貴。如果工作負載需要的平均 CPU 使用量小於 42.5%，則您可以得益於 t3.large 的更低價格，同時又可得到與 m5.large 相同的效能。



下表顯示如何計算損益兩平的 CPU 使用量閾值，讓您可以在 unlimited 模式下使用爆量效能執行個體，或使用固定效能執行個體時，判斷何者更便宜。資料表中的資料欄標示為 A 到 K。

執行個體類型	vCPU	T3 價格*/小時	M5 價格*/小時	價格差異	每個 vCPU 的 T3 基準使用率 (%)	剩餘額度的每 vCPU 小時費用	每分鐘 vCPU 費用	每 vCPU 可額外爆量分鐘	其他可用的 CPU %	損益兩平 CPU %
A	B	C	D	E = D - C	F	G	H = G / 60	I = E / H	J = (I / 60) / B	K = F + J
t3.large	2	0.0835 USD	0.096 USD	0.0125 USD	30%	0.05 USD	0.000833 USD	15	12.5%	42.5%

* 價格是以 us-east-1 和 Linux OS 為基礎。

資料表提供下列資訊：

- 資料欄 A 顯示執行個體類型，即 t3.large。
- 資料欄 B 顯示 t3.large 的 vCPU 數目。

- 資料欄 C 顯示 t3.large 每小時的價格。
- 資料欄 D 顯示 m5.large 每小時的價格。
- 資料欄 E 顯示 t3.large 與 m5.large 的價格差異。
- 資料欄 F 顯示 t3.large 的每 vCPU 基準使用率，即 30%。根據基準，執行個體的每小時成本涵蓋 CPU 使用量的成本。
- 資料欄 G 顯示每 vCPU 小時的 [其他固定費率](#)，若執行個體在耗盡其賺取的額度之後爆量為 100% CPU，即會以此費率向執行個體收費。
- 資料欄 H 顯示每 vCPU 分鐘的 [其他固定費率](#)，若執行個體在耗盡其賺取的額度之後爆量為 100% CPU，即會以此費率向執行個體收費。
- 資料欄 I 顯示 t3.large 每小時可以額外爆量為 100% CPU 幾分鐘，同時又可支付與 m5.large 相同的每小時價格。
- 資料欄 J 顯示超過基準的額外 CPU 使用量 (以 % 表示)，而執行個體可以爆量為這個使用量，同時又可支付與 m5.large 相同的每小時價格。
- 資料欄 K 顯示損益兩平的 CPU 使用量 (以 % 表示)，t3.large 可以爆量為這個使用量，而支付的費用不會超過 m5.large。除此之外，t3.large 的成本超過 m5.large。

下表顯示相較於類似大小的 M5 執行個體類型，T3 執行個體類型的損益兩平 CPU 使用量 (以 % 表示)。

T3 執行個體類型	相較於 M5，T3 的損益兩平 CPU 使用量 (以 % 表示)
t3.large	42.5%
t3.xlarge	52.5%
t3.2xlarge	52.5%

剩餘額度可能會產生費用

如果執行個體的 CPU 平均使用率等於或低於基準，執行個體不會產生額外的費用。由於執行個體在 24 小時的期間獲得了 [最高額度](#) (例如，t3.micro 執行個體可以在 24 小時的期間獲得最多 288 點)，因此可以用掉等於最高額度的剩餘額度，而不會產生費用。

不過，如果 CPU 使用率保持在基準以上，則執行個體就無法獲得足夠的額度，來償還先前用掉的剩餘額度。未償還的剩餘額度，會以每 vCPU 小時的其他固定費率來收費。如需有關費率的資訊，請參閱 [T2/T3/T4g 無限制模式定價](#)。

發生下列任一情況時，將會針對先前用掉的剩餘額度來收取費用。

- 支出剩餘額度超過執行個體在 24 小時期間可獲得的 [最大額度數量](#)。在小時結束時，將收取超過最大值的支出剩餘額度的費用。
- 執行個體已停止或終止。
- 執行個體從 unlimited 切換至 standard。

CloudWatch 度量會追蹤已花費的剩餘積分 `CPU Surplus Credit Balance`。計費的剩餘積分會由 CloudWatch 量度追蹤 `CPU Surplus Credits Charged`。如需詳細資訊，請參閱 [高載效能執行個體的其他 CloudWatch 指標](#)。

T2 無限制執行個體沒有啟動額度

T2 標準執行個體會收到 [啟動額度](#)，但 T2 無限制執行個體則不會收到。只要在累計 24 小時的期間或是其生命週期期間 (以較短者為準)，T2 無限制執行個體的 CPU 平均使用率等於或低於基準，此等執行個體即可隨時讓效能爆量超過基準水準，而不會產生額外的費用。因此，T2 Unlimited 執行個體不需要使用啟動額度，以在啟動之後立即達成高效能。

如果 T2 執行個體從 standard 切換為 unlimited，則在將剩下的 `CPU Credit Balance` 結轉過去之前，會先從 `CPU Credit Balance` 移除所有累積的啟動額度。

T4g、T3a 和 T3 執行個體永遠不會收到啟動額度，因為它們支援無限制模式。無限制模式額度組態可讓 T4g、T3a 和 T3 執行個體根據需要使用盡可能多的 CPU，以盡可能長時間地超出基準。

啟用無限制模式

在執行中或已停止的執行個體上，可以隨時從 unlimited 切換為 standardUnlimited，也可從 standard 切換為 unlimited。如需詳細資訊，請參閱 [將爆量效能執行個體啟動為無限制或標準](#) 及 [如何修改爆量效能執行個體的額度規格](#)。

您可以在每個 AWS 區域、每個高載績效執行環境系列的帳戶層次設 unlimited 定為預設信用選項，以便使用預設信用選項來啟動帳戶中的所有新高載績效執行環境。如需詳細資訊，請參閱 [設定帳戶的預設額度規格](#)。

您可以使用 Amazon EC2 主控台或 AWS CLI，檢查爆量效能執行個體是否已設定為 unlimited 或 standard。如需詳細資訊，請參閱 [檢視爆量效能執行個體的額度規格](#) 及 [檢視預設額度規格](#)。

在無限制與標準之間切換時，額度會有什麼變化

CPUCreditBalance 是追蹤執行個體累積的學分數目的 CloudWatch 量度。

CPUSurplusCreditBalance 是追蹤執行個體花費的剩餘積分數 CloudWatch 量的量度。

當您將設定為 unlimited 的執行個體變更為 standard，會發生下列情況：

- CPUCreditBalance 值會維持不變結轉過去。
- CPUSurplusCreditBalance 值會立即收取費用。

當 standard 執行個體切換為 unlimited 時，會發生下列情況：

- 將包含累積獲得之額度的 CPUCreditBalance 值結轉過去。
- 若為 T2 Standard 執行個體，從 CPUCreditBalance 值移除所有啟動額度，然後將剩下的 CPUCreditBalance 值 (包含累積獲得的額度) 結轉過去。

監控額度用量

若要查看執行個體花費的點數是否超過基準提供的額度，您可以使用 CloudWatch 指標追蹤使用情況，也可以設定每小時警示以收到點數用量通知。如需詳細資訊，請參閱 [監控爆量效能執行個體的 CPU 額度](#)。

無限制模式範例

下列範例說明設定為 unlimited 之執行個體的額度使用。

範例

- [範例 1：說明 T3 Unlimited 的額度使用](#)
- [範例 2：說明 T2 Unlimited 的額度使用](#)

範例 1：說明 T3 Unlimited 的額度使用

在此範例中，您會看到啟動為 t3.nano 之 unlimited 執行個體的 CPU 使用率，以及該執行個體如何用掉獲得的和多餘的額度，來維持 CPU 的使用率。

t3.nano 執行個體會在此期間獲得 144 點的 CPU 額度，這些額度可用來兌換 144 分鐘的 vCPU 使用。當它耗盡其 CPU 積分餘額 (以 CloudWatch 指標表示 CPUCreditBalance) 時，它可以花費剩餘的 CPU 積分 (尚未獲得)，以根據需要爆發。由於 t3.nano 執行個體在 24 小時期間

已獲得最高的 144 點，因此可以用掉等於此最高額度的剩餘額度，而不會立即產生費用。如果該執行個體用掉超過 144 點的 CPU 額度，則在該小時結束時，將會收取額度差額的費用。

此範例的目的 (如下列圖表所示)，是要說明即使用完 CPUCreditBalance，執行個體如何可利用剩餘額度來大幅提升效能。下列的工作流程參考了圖表上的編號點：

P1 – 在圖表上的第 0 小時，執行個體會啟動為 unlimited，並立即開始獲得額度。該執行個體從啟動後即維持閒置，其 CPU 使用率為 0%，而且沒有用掉額度。所有為使用的額度都會累計到額度餘額。前 24 小時，CPUCreditUsage 為 0，而 CPUCreditBalance 值達到其最大值 144。

P2 – 接下來 12 小時，CPU 使用率為 2.5%，低於 5% 基準。執行個體獲得的額度超出其用掉的額度，但 CPUCreditBalance 值不得超出其最大值，即 144 點。

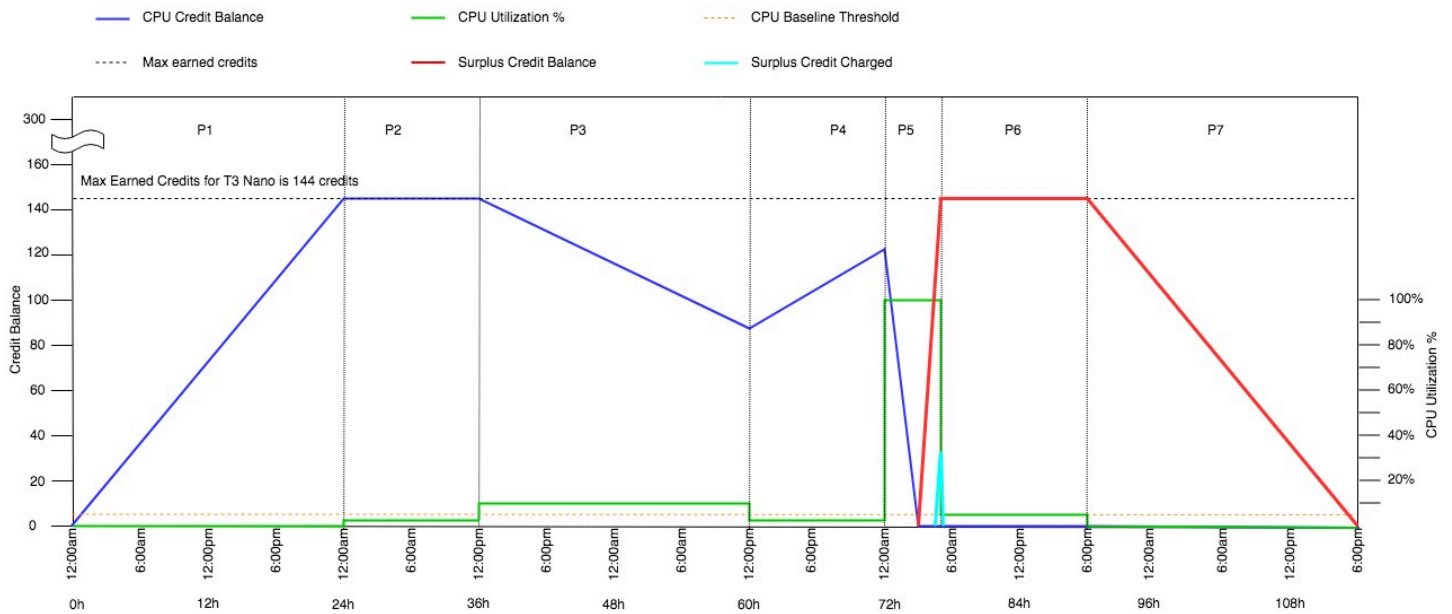
P3 – 接下來 24 小時，CPU 使用率為 7% (高於基準)，其需要用掉 57.6 點。執行個體用掉的額度超過獲得的額度，而且 CPUCreditBalance 值減少至 86.4 點。

P4 – 接下來 12 小時，CPU 使用率下降至 2.5% (低於基準)，其需要用掉 36 點。同時執行個體獲得 72 點。執行個體獲得的額度超過用掉的額度，而且 CPUCreditBalance 值增加至 122 點。

P5 – 接下來 5 小時，執行個體的 CPU 使用率爆量至 100%，並總共用掉 570 點來維持此爆量。在此期間大約一小時，執行個體耗盡其整個 CPUCreditBalance (即 122 點)，並開始用掉剩餘額度以維持高 CPU 使用率，在此期間總計剩餘 448 點 (570-122=448)。當 CPUSurplusCreditBalance 值達到 144 CPU 額度 (t3.nano 執行個體可在 24 小時期間獲得的最大額度) 時，獲得的額度無法抵銷任何用掉的剩餘額度。之後用掉的剩餘額度合計達 304 點 (448-144=304)，因此在該小時結束時，會對 304 點產生少許的額外費用。

P6 – 接下來 13 小時，CPU 使用率為 5% (基準)。執行個體獲得與其用掉一樣多的額度，不需額外支付 CPUSurplusCreditBalance。CPUSurplusCreditBalance 值仍為 144 點。

P7 – 在此範例的最後 24 小時，執行個體閒置且 CPU 使用率為 0%。在此期間，執行個體獲得 144 點，用來支付 CPUSurplusCreditBalance。



範例 2：說明 T2 Unlimited 的額度使用

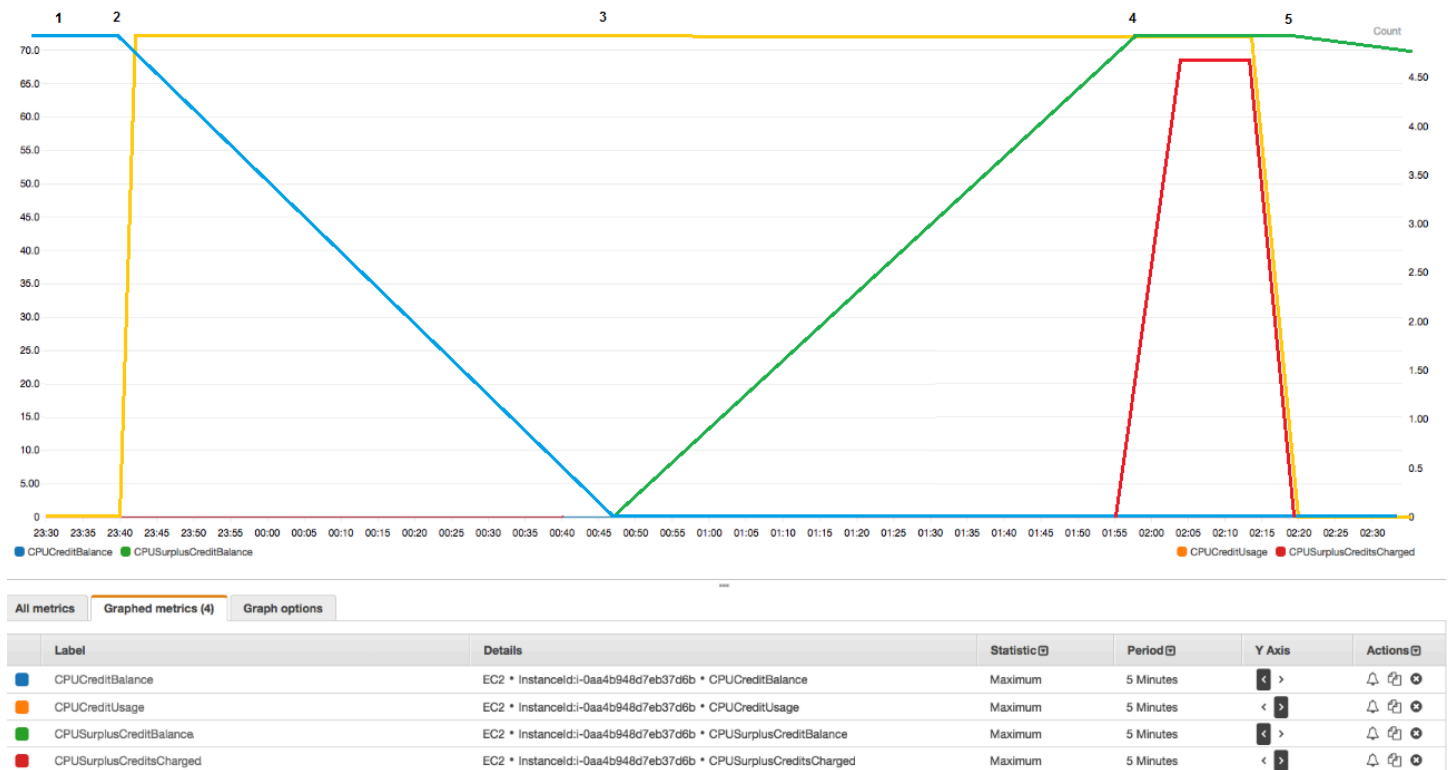
在此範例中，您會看到啟動為 `t2.nano` 之 `unlimited` 執行個體的 CPU 使用率，以及該執行個體如何用掉獲得的和多餘的額度，來維持 CPU 的使用率。

`t2.nano` 執行個體會在累計 24 小時的期間獲得 72 點的 CPU 額度，這些額度可用來兌換 72 分鐘的 vCPU 使用。當它耗盡其 CPU 積分餘額 (以 CloudWatch 指標表示 `CPUCreditBalance`) 時，它可以花費剩餘的 CPU 積分 (尚未獲得)，以根據需要爆發。由於 `t2.nano` 執行個體在 24 小時期間已獲得最高的 72 點，因此可以用掉等於此最高額度的剩餘額度，而不會立即產生費用。如果該執行個體用掉超過 72 點的 CPU 額度，則在該小時結束時，將會收取額度差額的費用。

此範例的目的 (如下列圖表所示)，是要說明即使用完 `CPUCreditBalance`，執行個體如何可利用剩餘額度來大幅提升效能。您可以假設，在圖表中時間線的開頭，執行個體已有累積的額度餘額，此餘額等於該執行個體在 24 小時期間內可以獲得的最高額度。下列的工作流程參考了圖表上的編號點：

- 1 – 在前 10 分鐘期間，`CPUCreditUsage` 為 0，而 `CPUCreditBalance` 維持在最大值 72 點。
- 2 – 在 23:40，隨著 CPU 的使用率增加，執行個體用掉了 CPU 額度，`CPUCreditBalance` 值減少。
- 3 – 在大約 00:47 時，執行個體用完整個 `CPUCreditBalance`，並且開始使用剩餘額度來維持 CPU 的高使用率。
- 4 – 剩餘額度一直用到 01:55，此時 `CPUSurplusCreditBalance` 到達 72 點的 CPU 額度。這等於 `t2.nano` 執行個體在 24 小時期間內可獲得的最高點數。之後，任何用掉的剩餘額度，都不能用 24 小時期間內所獲得的額度抵銷，因此在該小時結束時，會產生少許的額外費用。

5 – 執行個體繼續使用剩餘額度，直到大約 02:20。此時，CPU 的使用率掉到基準以下，執行個體開始以每小時 3 點的速度獲得額度 (或每 5 分鐘 0.25 點)，並使用這些額度來償還 CPU Surplus Credit Balance。在 CPU Surplus Credit Balance 值減少到 0 之後，執行個體開始以每 5 分鐘 0.25 點的速度，累積其 CPU Credit Balance 中的獲得額度。



計算帳單 (Linux 實例)

剩餘點數的費用為每個虛擬主機小時 0.05 USD。該執行個體在 01:55 到 02:20 之間，用掉了約 25 點剩餘額度，這等於 0.42 vCPU 小時。此執行個體的額外費用為 0.42 vCPU 小時 x 每小時 0.05 美元/ 虛擬 CPU 每小時 = 0.021 美元，四捨五入至 0.02 美元。下列是這個 T2 Unlimited 執行個體月底的帳單：

Amazon Elastic Compute Cloud running Linux/UNIX		
\$0.0058 per On Demand Linux t2.nano Instance Hour	720.000 Hrs	\$4.18
Amazon Elastic Compute Cloud T2 CPU Credits		
\$0.05 per vCPU-Hour of T2 CPU credits	0.420 vCPU-Hours	\$0.02

計算帳單 (視窗執行個體)

剩餘點數成本為每小時 0.096 美元。該執行個體在 01:55 到 02:20 之間，用掉了約 25 點剩餘額度，這等於 0.42 vCPU 小時。此執行個體的額外費用為 0.42 vCPU 小時 x 0.096/vCPU 小時 = 0.04032 美元，四捨五入至 0.04 USD。下列是這個 T2 Unlimited 執行個體月底的帳單：

Amazon Elastic Compute Cloud running Windows		
\$0.0081 per On Demand Windows t2.nano Instance Hour	720.000 Hrs	\$5.83

Amazon Elastic Compute Cloud T2 CPU Credits		
\$0.096 per vCPU-Hour of T2 CPU credits	0.420 vCPU-Hours	\$0.04

您可以設定帳單提醒，每小時通知任何累計的費用，並根據需要採取行動。

爆量效能執行個體的標準模式

設定為 standard 的高載效能標準執行個體，適用於平均 CPU 使用率持續低於執行個體之基準 CPU 使用率的工作負載。若要讓效能突增至超過基準，執行個體會用掉在 CPU 額度餘額中所累積的額度。如果執行個體累積的額度過低，CPU 使用率會逐漸降到基準水準，如此執行個體就不會在其累積的 CPU 額度餘額用完時，經歷效能急遽下降的狀況。如需詳細資訊，請參閱 [爆量效能執行個體的重要概念和定義](#)。

內容

- [標準模式概念](#)
 - [標準爆量效能執行個體如何運作](#)
 - [啟動額度](#)
 - [啟動額度限制](#)
 - [啟動額度與獲得額度之間的差異](#)
- [標準模式範例](#)
 - [範例 1：說明 T3 Standard 的額度使用](#)
 - [範例 2：說明 T2 Standard 的額度使用](#)
 - [期間 1：1 – 24 小時](#)
 - [期間 2：25 – 36 小時](#)
 - [期間 3：37 – 61 小時](#)
 - [期間 4：62 – 72 小時](#)
 - [期間 5：73 – 75 小時](#)
 - [期間 6：76 – 90 小時](#)

- [期間 7：91 – 96 小時](#)

標準模式概念

standard 模式是爆量效能執行個體的組態選項。對於執行中或已停止的執行個體，隨時都可以啟用或停用它。您可以在每個 AWS 區域、每個高載績效執行環境系列的帳戶層次[設 standard 定為預設信用選項](#)，以便使用預設信用選項來啟動帳戶中的所有新高載績效執行環境。

標準爆量效能執行個體如何運作

當設定為 standard 的爆量效能執行個體處於執行中狀態時，每小時會持續獲得 (以毫秒級的解析度) 固定比率的額度。若為 T2 Standard，當執行個體停止時，會失去所有已累積的額度，而且其額度餘額會重設為零。當此等執行個體重新啟動時，會獲得一組新的啟動額度，並且開始累積獲得的額度。對於 T4g、T3a 和 T3 標準執行個體，CPU 額度餘額會在執行個體停止之後持續保存七天，之後額度便會消失。如果您在七天內啟動執行個體，則不會失去任何額度。

T2 標準執行個體會獲得兩種 [CPU 額度](#)：獲得的額度和啟動額度。當 T2 Standard 執行個體處於執行狀態時，每小時會持續獲得 (以毫秒級的解析度) 固定比率的額度。一開始，執行個體尚未獲得額度，可能無法順暢啟動；因此，為了提供流暢啟動的體驗，執行個體在開始時會獲得啟動額度，可以在累積獲得額度的同時先用掉這些額度。

T4g、T3a 和 T3 執行個體不會收到啟動額度，因為它們支援無限制模式。無限制模式額度組態可讓 T4g、T3a 和 T3 執行個體根據需要使用盡可能多的 CPU，以盡可能長時間地超出基準。

啟動額度

在啟動或開始時，T2 Standard 執行個體會獲得每 vCPU 30 點的啟動額度，T1 Standard 執行個體可獲得 15 點的啟動額度。例如，t2.micro 執行個體擁有一個 vCPU，會獲得 30 點的啟動額度，而 t2.xlarge 執行個體擁有四個 vCPU，會獲得 120 點的啟動額度。啟動額度的設計旨在提供流暢的啟動體驗，讓執行個體能夠在尚未獲得累計的額度之前，先在啟動之後立即大幅提升效能。

啟動額度會比獲得的額度先用掉。未用掉的啟動額度會累計到 CPU 額度餘額，但不會列入 CPU 額度餘額限額的計算。例如，t2.micro 執行個體具有最多 144 點獲得額度的 CPU 額度餘額上限。如果該執行個體在啟動後保持閒置 24 小時，其 CPU 額度餘額會達到 174 點 (30 點的啟動額度 + 144 點的獲得額度)，超過了上限。不過，在該執行個體用掉 30 點的啟動額度之後，額度餘額不得超過 144 點。如需每種執行個體容量大小的 CPU 額度餘額限額詳細資訊，請參閱[額度表](#)。

下表列出了在啟動或開始時所獲得的初始 CPU 額度配額，以及 vCPU 的數量。

執行個體類型	啟動額度	vCPU
t1.micro	15	1
t2.nano	30	1
t2.micro	30	1
t2.small	30	1
t2.medium	60	2
t2.large	60	2
t2.xlarge	120	4
t2.2xlarge	240	8

啟動額度限制

T2 Standard 執行個體可以獲得啟動額度的次數是有限的。預設的限制是所有 T2 Standard 執行個體共 100 次啟動或開始 (每個帳戶、每個區域、每個累計 24 小時期間合計)。例如，當 1 個執行個體在 24 小時期間內停止和啟動 100 次、當 100 個執行個體在 24 小時期間內啟動，或是當等於 100 次啟動的其他組合動作發生時，就會到達限制的次數。新帳戶的限制值可能較低，而根據您的使用量，此限制會隨時間提高。

Tip

為確保工作負載隨時都能獲得所需的效能，請切換為 [爆量效能執行個體的無限制模式](#)，或考慮使用較大的執行個體大小。

啟動額度與獲得額度之間的差異

下表列出了啟動額度與獲得額度之間的差異。

	啟動額度	獲得的額度
額度獲得率	<p>在啟動或開始時，T2 Standard 執行個體會獲得每 vCPU 30 點的啟動額度。</p> <p>如果 T2 執行個體從 unlimited 切換為 standard，在切換時不會獲得啟動額度。</p>	<p>視執行個體的大小而定，每個 T2 執行個體每小時會持續獲得 (以毫秒級的解析度) 固定比率的 CPU 額度。如需每種執行個體容量大小所獲得的 CPU 額度詳細資訊，請參閱額度表。</p>
額度獲得限制	<p>獲得啟動額度的限制，是所有 T2 Standard 執行個體共 100 次啟動或開始 (每個帳戶、每個區域、每個累計 24 小時期間合計)。新帳戶的限制值可能較低，而根據您的使用量，此限制會隨時間提高。</p>	<p>T2 執行個體所累積的額度不能超過 CPU 額度餘額限額。如果 CPU 額度餘額到達其上限，則在到達限額之後所獲得的任何額度都將被捨棄。啟動額度不會列入此限額的計算。如需每種 T2 執行個體容量大小的 CPU 額度餘額限額詳細資訊，請參閱額度表。</p>
額度使用	<p>啟動額度會比獲得的額度先用掉。</p>	<p>獲得的額度只會在所有啟動額度用掉之後才使用。</p>
額度過期	<p>當 T2 Standard 執行個體正在執行時，啟動額度不會過期。如果 T2 Standard 執行個體停止，或是切換為 T2 Unlimited，所有的啟動額度都會消失。</p>	<p>當 T2 執行個體執行時，已經累積獲得的額度不會過期。當 T2 執行個體停止時，所有累積獲得的額度都會消失。</p>

CloudWatch 指標會追蹤累積的啟動積分和累積的已獲得積分數目 `CPUCreditBalance`。如需詳細資訊，請參閱[CloudWatch 測量結果表格CPUCreditBalance](#)中的。

標準模式範例

下列範例說明當執行個體設定為 standard 時的額度使用。

範例

- [範例 1：說明 T3 Standard 的額度使用](#)
- [範例 2：說明 T2 Standard 的額度使用](#)

範例 1：說明 T3 Standard 的額度使用

在此範例中，您會看到啟動為 t3.nano 的 standard 執行個體如何獲得、累積和用掉獲得的額度。您會看到額度餘額如何反映累積獲得的額度。

執行中的 t3.nano 執行個體每 24 小時會獲得 144 點。其額度餘額限制為獲得的 144 點。到達限額之後，獲得的新額度都將被捨棄。如需可獲得和累積的額度詳細資訊，請參閱[額度表](#)。

您可能會啟動 T3 Standard 執行個體，並立即使用它。或是您可能會啟動 T3 Standard 執行個體，然後讓該執行個體閒置幾天之後，才在其上執行應用程式。執行個體是否在使用中或處於閒置狀態，將決定額度是否會用掉或累積。如果執行個體在啟動後 24 小時的期間仍維持閒置，則額度餘額會達到其限制，即獲得的額度在累積後的上限。

此範例說明在啟動後 24 小時期間仍維持閒置的執行個體，並逐步說明 96 小時期間的 7 個階段、顯示獲得、累積、用掉和捨棄的比率，以及每個期間結束時的額度餘額值。

下列的工作流程參考了圖表上的編號點：

P1 – 在圖表上的第 0 小時，執行個體會啟動為 standard，並立即開始獲得額度。該執行個體從啟動後即維持閒置，其 CPU 使用率為 0%，而且沒有用掉額度。所有為使用的額度都會累計到額度餘額。前 24 小時，CPUCreditUsage 為 0，而 CPUCreditBalance 值達到其最大值 144。

P2 – 接下來 12 小時，CPU 使用率為 2.5%，低於 5% 基準。執行個體獲得的額度超出其用掉的額度，但 CPUCreditBalance 值不得超出其最大值，即 144 點。任何獲得的額度若超出限制，即會遭到捨棄。

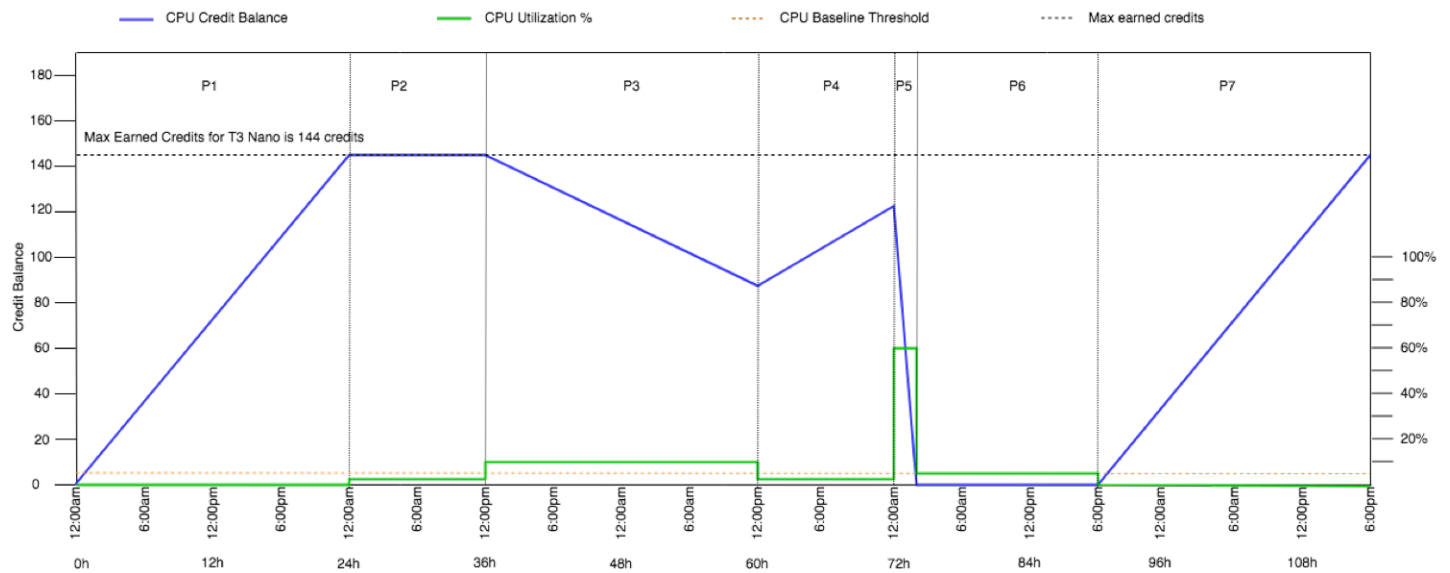
P3 – 接下來 24 小時，CPU 使用率為 7% (高於基準)，其需要用掉 57.6 點。執行個體用掉的額度超過獲得的額度，而且 CPUCreditBalance 值減少至 86.4 點。

P4 – 接下來 12 小時，CPU 使用率下降至 2.5% (低於基準)，其需要用掉 36 點。同時執行個體獲得 72 點。執行個體獲得的額度超過用掉的額度，而且 CPUCreditBalance 值增加至 122 點。

P5 – 接下來 2 小時，執行個體的 CPU 使用率爆量至 60%，並耗盡整個 CPUCreditBalance 值，即 122 點。在此期間結束時，由於 CPUCreditBalance 為零，CPU 使用率被迫下降至基準使用率，即 5%。處於基準時，執行個體獲得與其用掉一樣多的額度。

P6 – 接下來 14 小時，CPU 使用率為 5% (基準)。執行個體獲得與其用掉一樣多的額度。CPUCreditBalance 值仍為 0。

P7 – 在此範例的最後 24 小時，執行個體閒置且 CPU 使用率為 0%。在此期間，執行個體獲得 144 點，其會累積在 CPUCreditBalance。



範例 2：說明 T2 Standard 的額度使用

在此範例中，會示範做為 t2.nano 啟動的 standard 執行個體，如何獲得、累積和用掉啟動與獲得額度。範例中將會顯示額度餘額不只反映累積獲得的額度，也會反映累積的啟動額度。

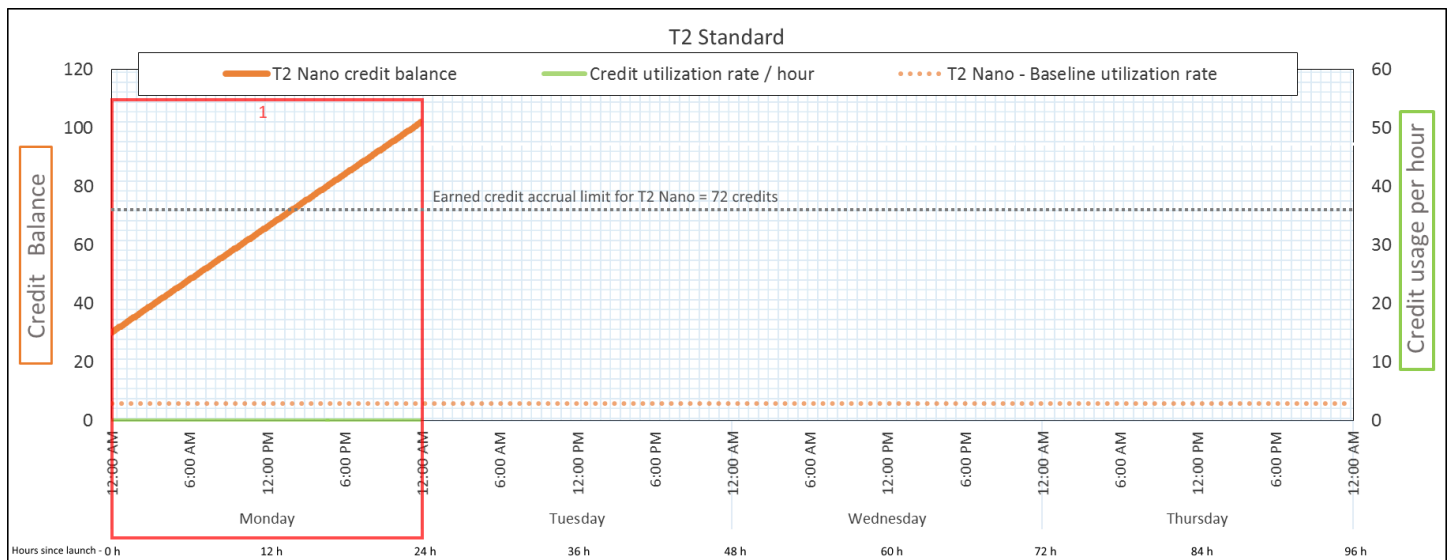
t2.nano 執行個體會是在啟動時獲得 30 點的啟動額度，而且每 24 小時可獲得 72 點。其額度餘額上限為 72 點獲得的額度；啟動額度不會列入此限額的計算。到達限額之後，獲得的新額度都將被捨棄。如需可獲得和累積的額度詳細資訊，請參閱[額度表](#)。如需限制的詳細資訊，請參閱[啟動額度限制](#)。

您可能會啟動 T2 Standard 執行個體，並立即使用它。或是您可能會啟動 T2 Standard 執行個體，然後讓該執行個體閒置幾天之後，才在其上執行應用程式。執行個體是否在使用中或處於閒置狀態，將決定額度是否會用掉或累積。如果執行個體在啟動後 24 小時的期間仍維持閒置，額度餘額會顯示為超過其限額，因為該餘額同時反映了累積獲得的額度和累積的啟動額度。不過，在使用 CPU 之後，會先用掉啟動額度。之後，此限額一律反映可累積獲得額度的上限。

此範例說明在啟動後 24 小時期間仍維持閒置的執行個體，並逐步說明 96 小時期間的 7 個階段、顯示獲得、累積、用掉和捨棄的比率，以及每個期間結束時的額度餘額值。

期間 1：1 – 24 小時

在圖表上的第 0 小時，T2 執行個體會啟動為 standard，並立即獲得 30 點的啟動額度。該執行個體會是在執行狀態中獲得額度。該執行個體從啟動後即維持閒置，其 CPU 使用率為 0%，而且沒有用掉額度。所有為使用的額度都會累計到額度餘額。在啟動後大約 14 小時，其額度餘額為 72 點 (30 點的啟動額度 + 42 點獲得的額度)，等於該執行個體可以在 24 小時中所獲得的額度。在啟動 24 小時後，額度餘額超過 72 點，因為有未使用的啟動額度累計到額度餘額—額度餘額為 102 點 (30 點的啟動額度 + 72 點獲得的額度)。



CPU 額度使用率	每 24 小時 0 點 (0% CPU 使用率)
額度獲得率	每 24 小時 72 點
額度捨棄率	每 24 小時 0 點
額度餘額	102 點 (30 點的啟動額度 + 72 點獲得的額度)

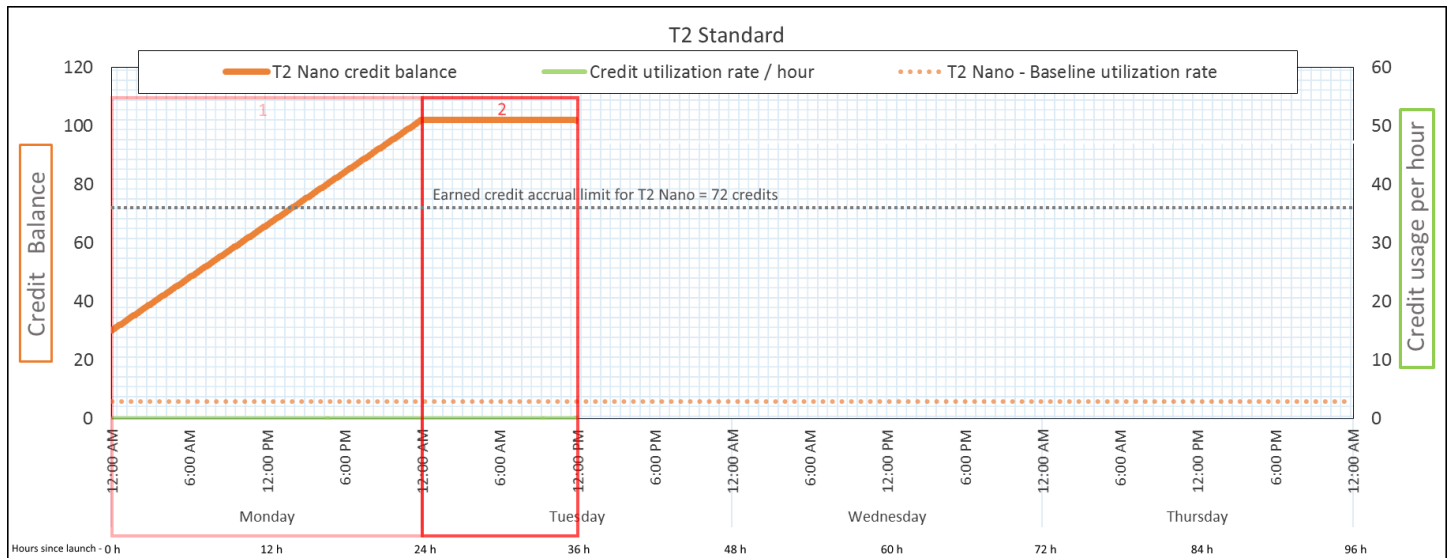
結論

如果啟動之後未使用 CPU，執行個體累積的額度，會比 24 小時中可獲得的更多 (30 點的啟動額度 + 72 點獲得的額度 = 102 點)。

在實際的情境中，EC2 執行個體會在啟動和執行時用掉少數的額度，這可防止餘額到達本範例中的理論值上限。

期間 2：25 – 36 小時

在接下來的 12 小時內，該執行個體會繼續保持閒置和獲得額度，但額度餘額不會增加。餘額會穩定地保持在 102 點 (30 點的啟動額度 + 72 點獲得的額度)。額度餘額已到達其上限 (72 點獲得的額度)，因此新獲得的額度將被捨棄。



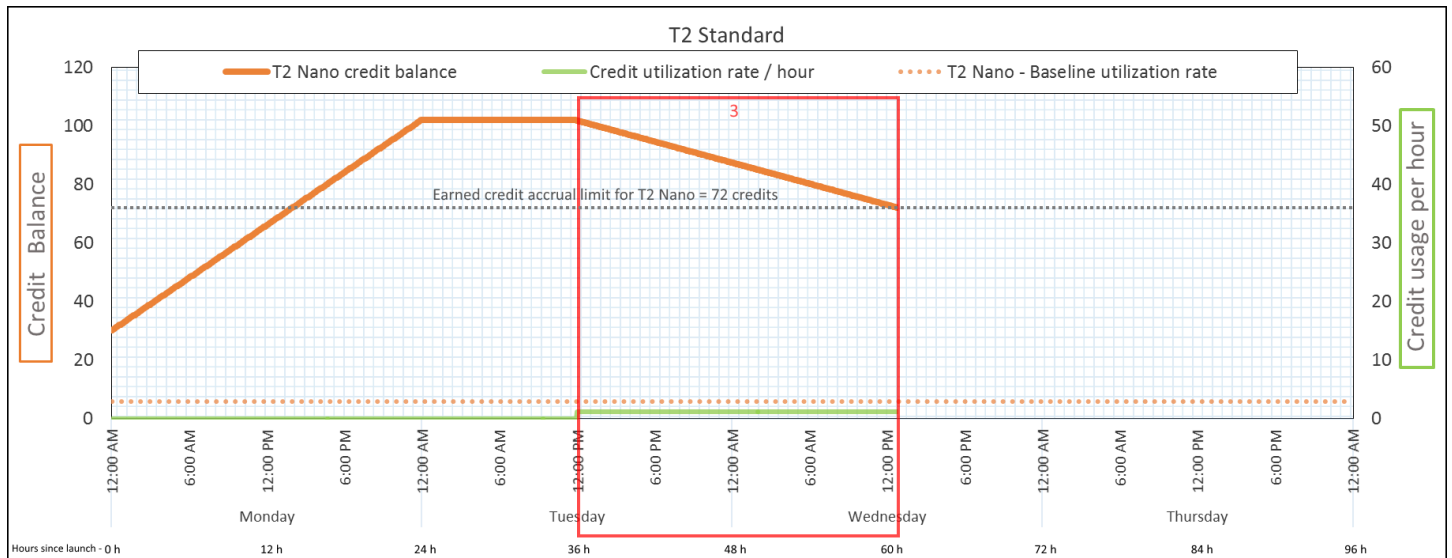
CPU 額度使用率	每 24 小時 0 點 (0% CPU 使用率)
額度獲得率	每 24 小時 72 點 (每小時 3 點)
額度捨棄率	每 24 小時 72 點 (100% 的額度獲得率)
額度餘額	102 點 (30 點的啟動額度 + 72 點獲得的額度) – 餘額未變

結論

執行個體會持續地獲得額度，但如果額度餘額已到達其上限，就不能再累積更多獲得的額度。到達限額之後，獲得的新額度都將被捨棄。啟動額度不會列入額度餘額限額的計算。如果餘額中包含累計的啟動額度，則餘額看起來會像是超過限額。

期間 3：37 – 61 小時

在接下來的 25 個小時期間，執行個體使用 2% 的 CPU，這需要 30 點。在同一個期間，執行個體獲得 75 點，但額度餘額減少了。餘額減少是由於累積的啟動額度會先用掉，而額度餘額已到達 72 點獲得額度的限制，因此新獲得的額度被捨棄。



CPU 額度使用率

每 24 小時 28.8 點 (每小時 1.2 點、2% CPU 使用率、40% 額度獲得率) – 25 小時 30 點

額度獲得率

每 24 小時 72 點

額度捨棄率

每 24 小時 72 點 (100% 的額度獲得率)

額度餘額

72 點 (30 點的啟動額度已用掉；72 點獲得的額度尚未用掉)

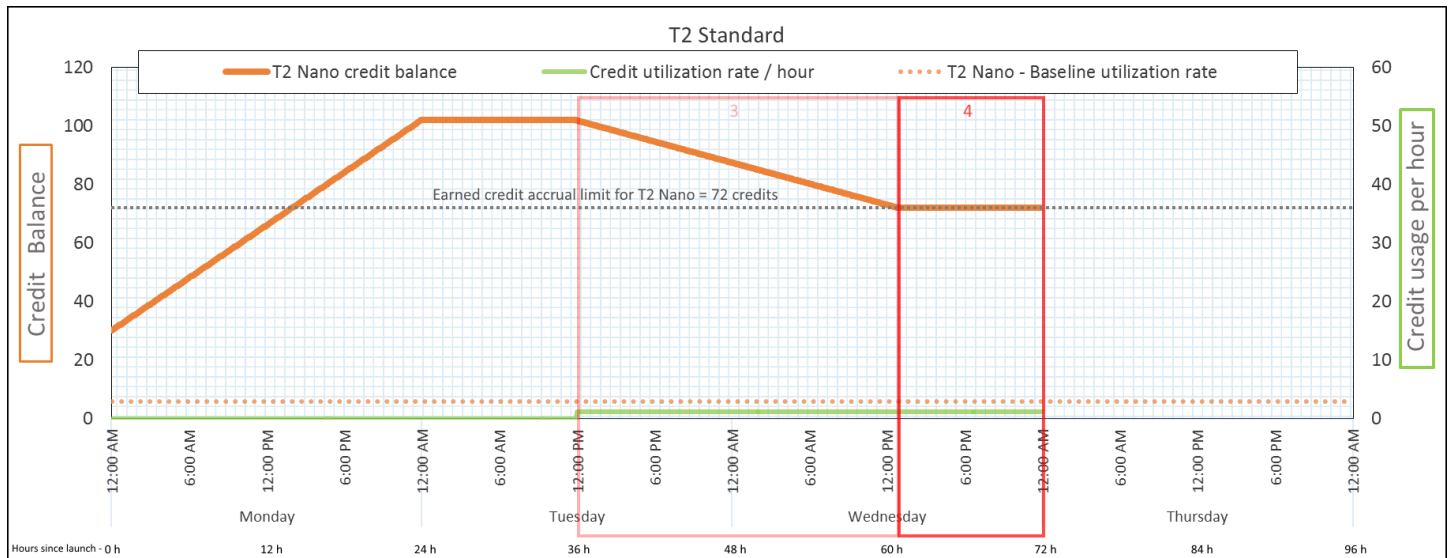
結論

執行個體會先用掉啟動額度，然後才會使用獲得的額度。啟動額度不會列入額度限額的計算。在啟動額度用掉之後，餘額就永遠不會高於 24 小時內可獲得的額度。此外，當執行個體執行時，就無法再獲得更多啟動額度。

期間 4：62 – 72 小時

在接下來的 11 個小時期間，執行個體使用 2% 的 CPU，這需要 13.2 點。這和前一個期間中的 CPU 使用率相同，但是餘額沒有減少，而是維持在 72 點。

餘額沒有減少是因為額度獲得率高於額度使用率。在此期間執行個體用掉 13.2 點，但也獲得了 33 點。但是，餘額的上限為 72 點，因此任何獲得的額度只要超過此限額，就會被捨棄。餘額會穩定地保持在 72 點，這跟期間 2 維持在 102 點不同，因為沒有累積的啟動額度。



CPU 額度使用率

每 24 小時 28.8 點 (每小時 1.2 點、2% CPU 使用率、40% 額度獲得率) – 11 小時 13.2 點

額度獲得率

每 24 小時 72 點

額度捨棄率

每 24 小時 43.2 點 (60% 的額度獲得率)

額度餘額

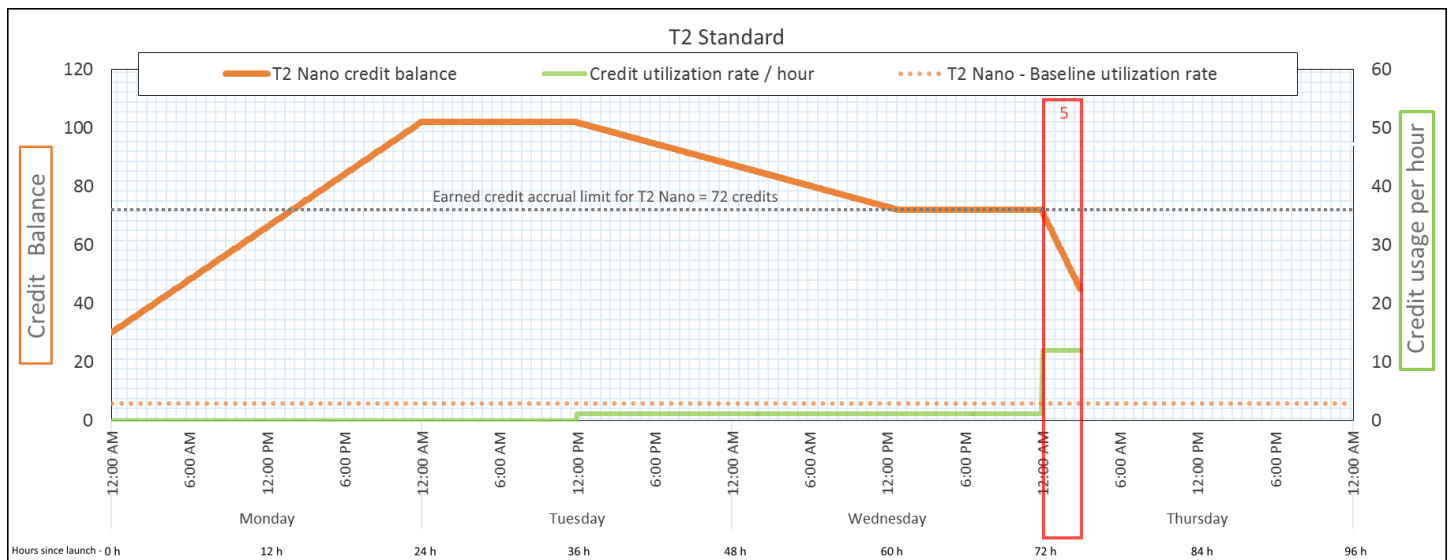
72 點 (0 點的啟動額度 + 72 點獲得的額度) – 餘額到達上限

結論

在啟動額度用掉之後，額度餘額的上限，會取決於執行個體在 24 小時內可獲得的額度。如果執行個體獲得的額度比用掉的更多，新獲得的額度只要超過限額，就會被捨棄。

期間 5：73 – 75 小時

在接下來的 3 個小時期間，執行個體運作的需求會突增為 20% 的 CPU 使用率，這需要 36 點。在同樣的 3 小時期間，執行個體會獲得 9 點，這會使得額度淨餘額減少 27 點。在 3 小時期間結束時，額度餘額為 45 點累積獲得的額度。



CPU 額度使用率

每 24 小時 288 點 (每小時 12 點、20% CPU 使用率、400% 額度獲得率) – 3 小時 36 點

額度獲得率

每 24 小時 72 點 (3 個小時 9 點)

額度捨棄率

每 24 小時 0 點

額度餘額

45 點 (先前的餘額 (72) - 用掉的額度 (36) + 獲得的額度 (9)) – 在每 24 小時獲得 216 點的額度獲得率之下，但額度餘額減少了，額度餘額減少 (使用率 288/24 + 獲得率 72/24 = 餘額減少率 216/24)

結論

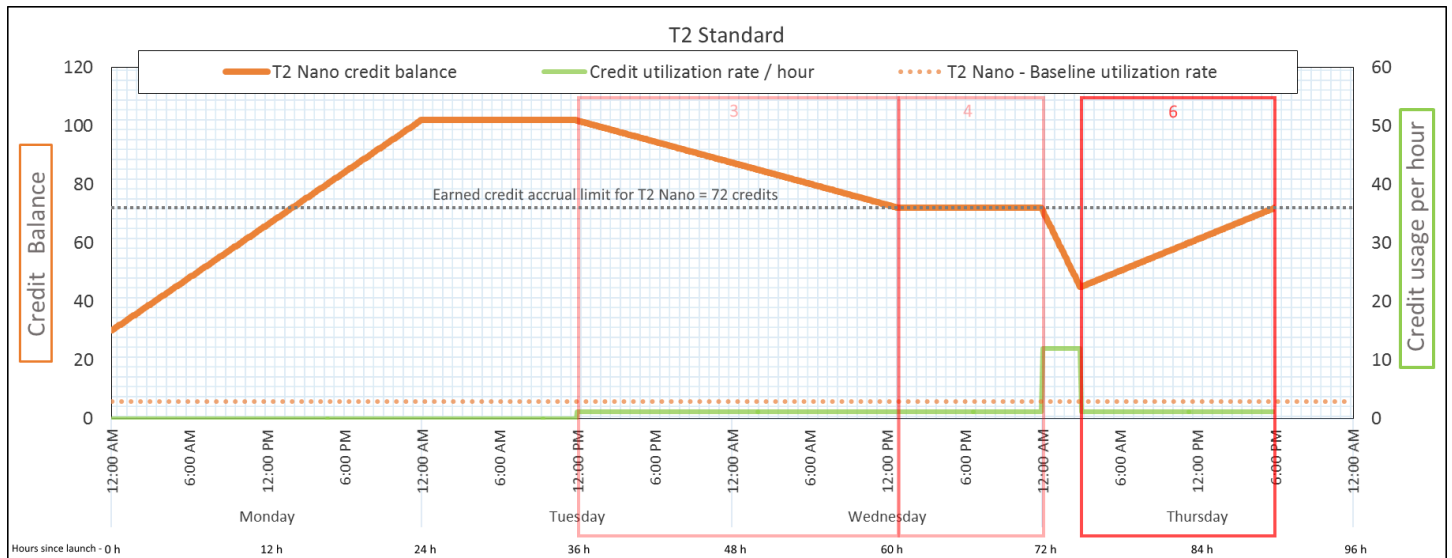
如果執行個體用掉的額度比獲得的多，其額度餘額會減少。

期間 6：76 – 90 小時

在接下來的 15 個小時期間，執行個體使用 2% 的 CPU，這需要 18 點。此期間的 CPU 使用率，和期間 3 與期間 4 的相同。不過，餘額在此期間是增加的，而期間 3 和期間 4 的餘額則分別是減少和持平。

在期間 3 中用掉了累積的啟動額度，並捨棄超過額度限額的所有獲得額度，因此使額度餘額減少。在期間 4 中，執行個體用掉的額度比獲得的少。捨棄超過額度限額的所有獲得額度，因此額度餘額持平，維持在 72 點的最高值。

在此期間中沒有累積的啟動額度，而餘額中累積獲得的額度低於限額。沒有任何獲得的額度被捨棄。此外，執行個體獲得的額度比用掉的多，因此使額度餘額增加。



CPU 額度使用率

每 24 小時 28.8 點 (每小時 1.2 點、2% CPU 使用率、40% 額度獲得率) – 15 小時 18 點

額度獲得率

每 24 小時 72 點 (15 個小時 45 點)

額度捨棄率

每 24 小時 0 點

額度餘額

72 點 (餘額以每小時 43.2 點的速度增加 – 變動率 = 使用率 28.8/24 + 獲得率 72/24)

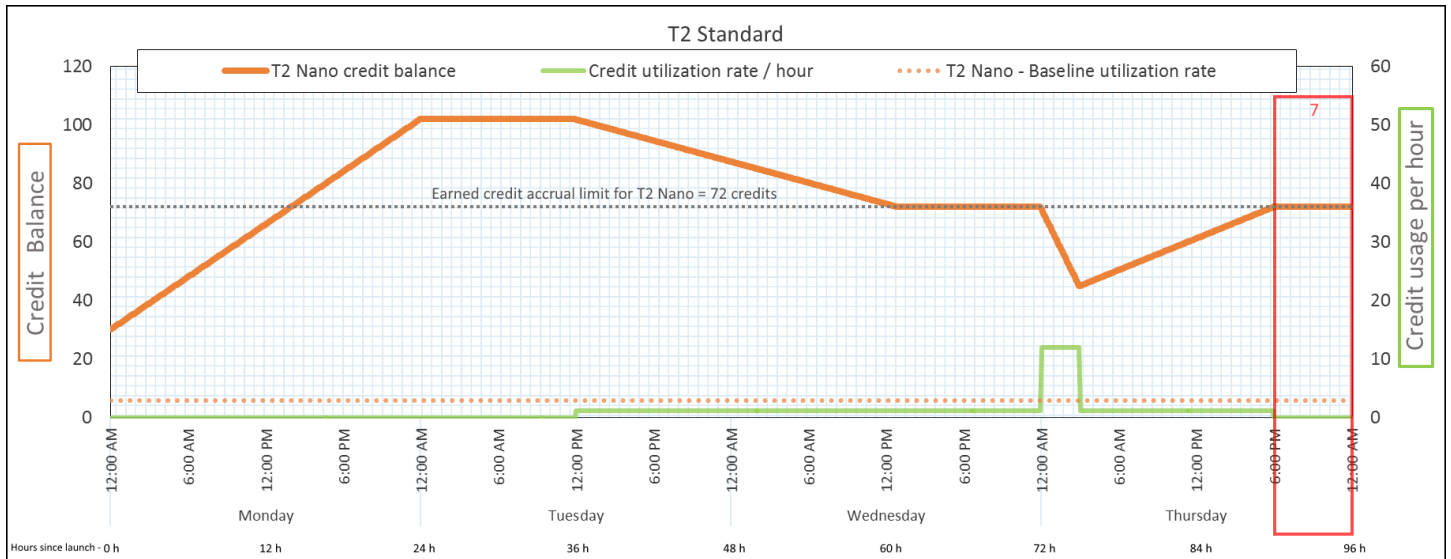
結論

如果執行個體用掉的額度比獲得的少，其額度餘額會增加。

期間 7：91 – 96 小時

接下來的 6 小時，執行個體會維持閒置—其 CPU 使用率為 0%—而且沒有用掉額度。此 CPU 使用率與期間 2 的相同，但餘額並未持平維持於 102 點—而是持平保持在 72 點，這是執行個體的額度餘額上限。

在期間 2 中，額度餘額包含 30 點累積的啟動額度。在期間 3 中用掉了啟動額度。執行中的執行個體無法再獲得更多的啟動額度。在到達其額度餘額上限後，任何獲得的額度只要超過此限額，就會被捨棄。



CPU 額度使用率

每 24 小時 0 點 (0% CPU 使用率)

額度獲得率

每 24 小時 72 點

額度捨棄率

每 24 小時 72 點 (100% 的額度獲得率)

額度餘額

72 點 (0 點的啟動額度、72 點獲得的額度)

結論

執行個體會持續地獲得額度，但如果額度餘額已到達其上限，就不能再累積更多獲得的額度。到達限額之後，獲得的新額度都將被捨棄。額度餘額的上限，會取決於執行個體在 24 小時內可獲得的額度。如需額度餘額上限的詳細資訊，請參閱[額度表](#)。

使用爆量效能執行個體

啟動、監視和修改高載效能執行個體 (T 執行個體) 的步驟類似。主要差異為它們啟動時的預設額度規格。

每個 T 執行個體系列都有下列預設信用規格：

- T4G、T3A 和 T3 執行個體的推出方式為 unlimited
- 專用執行個體中的 T3 執行個體只能以 standard 狀態啟動

- T2 執行個體啟動為 standard

可變更帳戶的[預設額度規格](#)。

目錄

- [將爆量效能執行個體啟動為無限制或標準](#)
- [使用 Auto Scaling 群組將爆量效能執行個體啟動為無限制](#)
- [檢視爆量效能執行個體的額度規格](#)
- [如何修改爆量效能執行個體的額度規格](#)
- [設定帳戶的預設額度規格](#)
- [檢視預設額度規格](#)

將爆量效能執行個體啟動為無限制或標準

您可以standard使用 Amazon EC2 主控台、開發 AWS 套件、命令列工具或自動擴展群組來啟動 T 執行個體。unlimited

下列程序說明如何使用 EC2 主控台或 AWS CLI。如需有關使用「Auto Scaling」群組的資訊，請參閱[使用 Auto Scaling 群組將爆量效能執行個體啟動為無限制](#)。

Console

將 T 執行個體啟動為無限制或標準

1. 請遵循該程序來[啟動執行個體](#)。
2. 在 Instance type (執行個體類型) 中，選取 T 執行個體類型。
3. 展開 Advanced details (進階詳細資訊)，並針對 Credit specification (額度規格)，選取額度規格。如果您沒有進行選取，則會使用預設值 (適用standard於 T2)，也就是 T4G、T3A 和 T3。unlimited
4. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。如需詳細資訊，請參閱[使用新的啟動執行個體精靈啟動執行個體](#)。

AWS CLI

將 T 執行個體啟動為無限制或標準

使用 [run-instances](#) 命令來啟動您的執行個體。使用 `--credit-specification CpuCredits=` 參數來指定額度規格。有效的點數規格為 `unlimited` 和 `standard`。

- 對於 T4G、T3A 和 T3，如果您不包含 `--credit-specification` 參數，則執行個體會依 `unlimited` 預設啟動。
- 若為 T2，如果未包含 `--credit-specification` 參數，執行個體預設會啟動為 `standard`。

```
aws ec2 run-instances \  
  --image-id ami-abc12345 \  
  --count 1 \  
  --instance-type t3.micro \  
  --key-name MyKeyPair \  
  --credit-specification "CpuCredits=unlimited"
```

使用 Auto Scaling 群組將爆量效能執行個體啟動為無限制

啟動或啟動 T 執行個體時，它們需要 CPU 積分才能獲得良好的啟動載入體驗。如果您使用 Auto Scaling 群組來啟動您的執行個體，我們建議將您的執行個體設定為 `unlimited`。如此在由 Auto Scaling 群組自動啟動或重新啟動時，這些執行個體即可使用剩餘額度。請使用剩餘額度來防止效能受限。

建立啟動範本

您必須使用「啟動範本」，在 Auto Scaling 群組中將執行個體啟動為 `unlimited`。啟動組態不支援將執行個體啟動為 `unlimited`。

Note

`unlimited` 模式不支援在專用執行個體上啟動的 T3 執行個體。

Console

建立以「無限制」狀態啟動執行個體的啟動範本

1. 遵循 Amazon EC2 自動擴展使用者指南中的使用進階設定建立啟動範本程序。
2. 在啟動範本內容中，針對執行個體類型，選擇執行個體大小。

3. 若要在 Auto Scaling 群組中以 unlimited 的形式啟動執行個體，請在進階詳細資料下，針對額度規格選擇無限制。
4. 在您完成定義啟動範本參數時，選擇 Create launch template (建立啟動範本)。

AWS CLI

建立以「無限制」狀態啟動執行個體的啟動範本

使用 [create-launch-template](#) 命令，並將 unlimited 指定為額度規格。

- 對於 T4G、T3A 和 T3，如果您未包含該 `CreditSpecification={CpuCredits=unlimited}` 值，則執行個體會依 unlimited 預設啟動。
- 若為 T2，如果未包含 `CreditSpecification={CpuCredits=unlimited}` 值，執行個體預設會啟動為 standard。

```
aws ec2 create-launch-template \  
  --launch-template-name MyLaunchTemplate \  
  --version-description FirstVersion \  
  --launch-template-data  
  ImageId=ami-8c1be5f6,InstanceType=t3.medium,CreditSpecification={CpuCredits=unlimited}
```

使用 Auto Scaling 群組與啟動範本建立關聯

若要建立啟動範本與 Auto Scaling 群組的關聯，請使用啟動範本來建立 Auto Scaling 群組，或是將啟動範本加入現有的 Auto Scaling 群組。

Console

使用啟動範本建立 Auto Scaling 群組

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在畫面上方的導覽列中，選取與您用來建立啟動範本相同的區域。
3. 在導覽窗格中，選擇 Auto Scaling Groups (AS 群組) 與 Create Auto Scaling group (建立 AS 群組)。
4. 選取 Launch Template (啟動範本)、選取您的啟動範本，然後選取 Next Step (下一步)。

5. 填寫 Auto Scaling 群組的欄位。檢閱完您在 Review page (檢閱頁面) 上的組態設定後，請選擇 Create Auto Scaling group (建立 Auto Scaling 群組)。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南 中的 [使用啟動範本建立 Auto Scaling 群組](#)。

AWS CLI

使用啟動範本建立 Auto Scaling 群組

使用 [建立-自動縮放群組指令](#) AWS CLI 並指定參數。--launch-template

Console

將啟動範本新增至現有的 Auto Scaling 群組

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在畫面上方的導覽列中，選取與您用來建立啟動範本相同的區域。
3. 在導覽窗格中，選擇 Auto Scaling Groups (AS 安全群組)。
4. 從 Auto Scaling 群組清單中，選取 Auto Scaling 群組，然後選取 Actions (動作)、Edit (編輯)。
5. 在 Details (詳細資訊) 標籤上，對於 Launch Template (啟動範本)，選擇一個啟動範本，然後選擇 Save (儲存)。

AWS CLI

將啟動範本新增至現有的 Auto Scaling 群組

使用 [更新自動縮放群組 AWS CLI 命令並指定](#) 參數。--launch-template

檢視爆量效能執行個體的額度規格

您可以檢視執行中unlimited或已停止 T 執行處理的信用規格 (或standard)。

Console

若要檢視 T 執行環境的信用規格，請執行下

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽窗格中選擇 Instances (執行個體)。

3. 選取實例。
4. 選擇詳細資訊，然後檢視額度規格欄位。此值為 `unlimited` 或 `standard`。

AWS CLI

若要說明 T 執行個體的信用規格

使用 [describe-instance-credit-specifications](#) 命令。如果您未指定一或多個執行個體 ID，則會傳回具有 `unlimited` 額度規格的所有執行個體，以及先前設定為具有 `unlimited` 額度規格的執行個體。例如，如果您將 T3 執行個體的大小重新調整為 M4 執行個體，並且其設定為 `unlimited` 時，Amazon EC2 會傳回 M4 執行個體。

```
aws ec2 describe-instance-credit-specifications --instance-id i-1234567890abcdef0
```

範例輸出

```
{
  "InstanceCreditSpecifications": [
    {
      "InstanceId": "i-1234567890abcdef0",
      "CpuCredits": "unlimited"
    }
  ]
}
```

如何修改爆量效能執行個體的額度規格

您可以在 `unlimited` 和之間隨時切換執行中或已停止的 T 執行個體的信用規格 `standard`。

請注意，在 `unlimited` 模式中，執行個體可能會使用剩餘抵用金，這可能會產生額外費用。如需詳細資訊，請參閱 [剩餘額度可能會產生費用](#)。

Console

若要修改 T 執行環境的信用規格，請執行下

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽窗格中選擇 Instances (執行個體)。

3. 選取實例。若要一次修改多個執行個體的額度規格，請選取所有適用的執行個體。
4. 選擇動作、執行個體設定、變更額度規格。只有在選取 T 例證時，才會啟用此選項。
5. 若要將額度規格變更為 `unlimited`，請選取執行個體 ID 旁的核取方塊。若要將額度規格變更為 `standard`，請清除執行個體 ID 旁的核取方塊。

AWS CLI

若要修改 T 執行環境的信用規格，請執行下

使用 [modify-instance-credit-specification](#) 命令。使用 `--instance-credit-specification` 參數來指定執行個體及其額度規格。有效的額度規格為 `unlimited` 和 `standard`。

```
aws ec2 modify-instance-credit-specification \
  --region us-east-1 \
  --instance-credit-specification
  "InstanceId=i-1234567890abcdef0,CpuCredits=unlimited"
```

範例輸出

```
{
  "SuccessfulInstanceCreditSpecifications": [
    {
      "InstanceId": "i- 1234567890abcdef0"
    }
  ],
  "UnsuccessfulInstanceCreditSpecifications": []
}
```

設定帳戶的預設額度規格

每個 T 執行個體系列都有一個[預設信用規格](#)。您可以在每個 AWS 區域的科目層次，變更每個 T 執行環境系列的預設信用規格。

如果您使用 EC2 主控台中的啟動執行個體精靈來啟動執行個體，您針對額度規格選取的值則會覆寫帳戶層級的預設額度規格。如果您使用 AWS CLI 來啟動執行環境，則會使用預設信用規格來啟動帳戶中的所有新 T 執行環境。現有執行中或已停止執行個體的額度規格不受影響。

考量事項

執行個體系列的預設額度規格只能在連續 5 分鐘的期間內修改一次，而在連續 24 小時內最多可修改四次。

Console

在每個區域的帳戶層級設定預設額度規格

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 若要變更 AWS 區域，請使用頁面右上角的「地區」選取器。
3. 在左側導覽窗格中，選擇 EC2 Dashboard (EC2 儀表板)。
4. 在 Account attributes (帳戶屬性) 中，選擇 Default credit specification (預設額度規格)。
5. 選擇 Manage (管理)。
6. 針對每個執行個體系列選擇 Unlimited (無限制) 或 Standard (標準)，然後選擇 Update (更新)。

AWS CLI

在帳戶層次設定預設額度規格 (AWS CLI)

使用 [modify-default-credit-specification](#) 命令。使用 `--cpu-credits` 參數指定 AWS 區域、執行個體系列和預設額度規格。有效的預設額度規格為 `unlimited` 和 `standard`。

```
aws ec2 modify-default-credit-specification \  
  --region us-east-1 \  
  --instance-family t2 \  
  --cpu-credits unlimited
```

檢視預設額度規格

您可以在每個 AWS 區域的科目層次，檢視 T 執行環境系列的預設信用規格。

Console

若要在科目層次檢視預設的信用規格，請執行下

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 若要變更 AWS 區域，請使用頁面右上角的「地區」選取器。
3. 在左側導覽窗格中，選擇 EC2 Dashboard (EC2 儀表板)。

4. 在 Account attributes (帳戶屬性) 中，選擇 Default credit specification (預設額度規格)。

AWS CLI

若要在科目層次檢視預設的信用規格，請執行下

使用 [get-default-credit-specification](#) 命令。指定 AWS 區域和執行個體系列。

```
aws ec2 get-default-credit-specification --region us-east-1 --instance-family t2
```

監控爆量效能執行個體的 CPU 額度

EC2 將指標發送到 Amazon CloudWatch。您可以在 CloudWatch 主控台的 Amazon EC2 每個執行個體指標中查看 CPU 積分指標，或使用列出 AWS CLI 每個執行個體的指標。如需詳細資訊，請參閱 [使用主控台列出指標](#) 及 [使用列出量度 AWS CLI](#)。

目錄

- [高載效能執行個體的其他 CloudWatch 指標](#)
- [計算 CPU 額度用量](#)

高載效能執行個體的其他 CloudWatch 指標

高載效能執行個體具有下列額外 CloudWatch 指標，每五分鐘更新一次：

- CPUCreditUsage – 在衡量期間用掉的 CPU 額度。
- CPUCreditBalance – 執行個體已累積的 CPU 額度。當 CPU 效能瞬間突增，以及 CPU 額度用掉的速度比獲得的快時，此額度餘額會用完。
- CPUSurplusCreditBalance – 當 CPUCreditBalance 值為 0 時，為維持 CPU 使用率所用掉的多餘 CPU 額度。
- CPUSurplusCreditsCharged – 超過 24 小時期間可獲得[最高 CPU 額度](#)的多餘 CPU 額度，因此會產生額外的費用。

最後兩種指標只適用於設定為 unlimited 的執行個體。

下表說明高載效能執行處理的 CloudWatch 測量結果。如需詳細資訊，請參閱 [列出執行個體的可用 CloudWatch 指標](#)。

指標	描述
CPUCreditUsage	<p>執行個體為 CPU 使用率花費的 CPU 額度數量。一個 CPU 額度等於一個 vCPU 以 100% 使用率執行 1 分鐘，或同等的 vCPU、使用率與時間的組合 (例如，一個 vCPU 以 50% 使用率執行 2 分鐘，或兩個 vCPU 以 25% 使用率執行 2 分鐘)。</p> <p>CPU 額度指標僅提供 5 分鐘頻率。如果您要指定大於 5 分鐘的期間，請使用 Sum 統計資訊代替 Average 統計資訊。</p> <p>單位：額度 (vCPU-分鐘)</p>
CPUCreditBalance	<p>自執行個體啟動或開始後，累積獲得的 CPU 額度數量。如果是 T2 Standard，CPUCreditBalance 也包含已產生的啟動額度。</p> <p>獲得額度後，額度會在額度餘額中累積，並在支付額度時，從額度餘額中移出。額度餘額有最大值限制，它取決於執行個體大小。到達限制之後，任何獲得的新額度都會遭到捨棄。如果是 T2 Standard，啟動額度不會計入此限制。</p> <p>CPUCreditBalance 中的額度可供執行個體支付以大幅提升並超越基準 CPU 使用率。</p> <p>當執行個體執行時，CPUCreditBalance 中的額度不會過期。當 T4G、T3A 或 T3 執行個體停止時，該 CPUCredit Balance 值會持續七天。因此，所有累積的額度都會消失。當 T2 執行個體停止時，CPUCreditBalance 值不會持續保存，而且所有累積的額度都將消失。</p> <p>CPU 額度指標僅提供 5 分鐘頻率。</p> <p>單位：額度 (vCPU-分鐘)</p>
CPUSurplusCreditBalance	<p>當 unlimited 執行個體的 CPUCreditBalance 值為 0 時，該執行個體已支出的剩餘額度數量。</p>

指標	描述
	<p>CPUSurplusCreditBalance 值由獲得的 CPU 額度支付。如果剩餘額度超過執行個體在 24 小時期間可獲得的最大額度數量，超過最大值的支出剩餘額度將必須負擔額外的費用。</p> <p>單位：額度 (vCPU-分鐘)</p>
CPUSurplusCreditsCharged	<p>若支出剩餘額度數量未由獲得的 CPU 額度付清，會產生額外的費用。</p> <p>發生以下任何情況時，將收取支出剩餘額度的費用。</p> <ul style="list-style-type: none"> • 支出剩餘額度超過執行個體在 24 小時期間可獲得的最大額度數量。在小時結束時，將收取超過最大值的支出剩餘額度的費用。 • 執行個體已停止或終止。 • 執行個體從 unlimited 切換至 standard。 <p>單位：額度 (vCPU-分鐘)</p>

計算 CPU 額度用量

執行處理的 CPU 積分使用狀況是使用上表所述的執行處理 CloudWatch 測量結果來計算。

Amazon EC2 將指標發送到 CloudWatch 每五分鐘一次。在任何時點參考指標的 prior 值，即表示該指標先前的值 (五分鐘前傳送的值)。

計算標準執行個體的 CPU 額度使用

- 當用掉的額度，少於先前五分鐘間隔中所獲得的額度時，如果 CPU 使用率低於基準，則 CPU 額度餘額會增加。
- 當用掉的額度，多於先前五分鐘間隔中所獲得的額度時，如果 CPU 使用率高於基準，則 CPU 額度餘額會減少。

在數學上，這可以透過下列方程式計算：

Example

```
CPUCreditBalance = prior CPUCreditBalance + [Credits earned per hour * (5/60) - CPUCreditUsage]
```

執行個體的大小，決定了執行個體每小時可獲得的額度，以及執行個體額度餘額可累積獲得額度的上限。如需每小時所獲得的額度、每種執行個體大小的額度餘額限額資訊，請參閱[額度表](#)。

範例

此範例使用 t3.nano 執行個體。若要計算執行個體的 CPUCreditBalance 值，請使用前述的方程式，如下所示：

- CPUCreditBalance – 要計算的最新額度餘額。
- prior CPUCreditBalance – 5 分鐘前的額度餘額。在此範例中，執行個體累積了 2 種額度。
- Credits earned per hour – 一個 t3.nano 執行個體每小時會獲得 6 點。
- 5/60— 代表 CloudWatch 量度發佈之間的五分鐘間隔。將每小時獲得的額度乘上 5/60 (5 分鐘)，以算出該執行個體在過去 5 分鐘內獲得的額度。t3.nano 執行個體每 5 分鐘會獲得 0.5 點。
- CPUCreditUsage – 在過去 5 分鐘內執行個體所用掉的額度。在此範例中，執行個體過去 5 分鐘內用掉了 1 點。

使用這些值，可以計算出 CPUCreditBalance 值：

Example

```
CPUCreditBalance = 2 + [0.5 - 1] = 1.5
```

計算無限制執行個體的 CPU 額度使用

當爆量效能執行個體的效能需要爆發超過基準時，一律會先用掉累積的額度，然後再使用剩餘額度。當該執行個體累積的 CPU 額度餘額用完時，可以使用剩餘額度，讓 CPU 在所需的期間大幅提升。如果 CPU 的使用率掉到基準以下，執行個體一律會先償還剩餘額度，然後才能累積獲得的額度。

在下列的方程式中，我們使用 Adjusted balance 一詞，來代表這個 5 分鐘間隔內所發生的動作。我們使用此值來取得 CPUCreditBalance 和 CPUSurplusCreditBalance CloudWatch 量度的值。

Example

```
Adjusted balance = [prior CPUCreditBalance - prior CPUSurplusCreditBalance] + [Credits earned per hour * (5/60) - CPUCreditUsage]
```

0 的 Adjusted balance 值代表該執行個體為了突增的效能，已經用完所有獲得的額度，而且未使用剩餘額度。因此，CPUCreditBalance 和 CPUSurplusCreditBalance 都會設定為 0。

正的 Adjusted balance 值表示該執行個體已累積獲得的額度，而且已償還先前的剩餘額度 (如果有的話)。因此，Adjusted balance 值會指派給 CPUCreditBalance，而 CPUSurplusCreditBalance 會設定為 0。執行個體的大小，會決定執行個體所能累積的[額度上限](#)。

Example

```
CPUCreditBalance = min [max earned credit balance, Adjusted balance]  
CPUSurplusCreditBalance = 0
```

負的 Adjusted balance 值代表執行個體已經用完所有累積獲得的額度，而且也因為突增的效能用掉了剩餘額度。因此，Adjusted balance 值會指派給 CPUSurplusCreditBalance，而 CPUCreditBalance 會設為 0。再說一次，執行個體的大小，會決定執行個體所能累積的[額度上限](#)。

Example

```
CPUSurplusCreditBalance = min [max earned credit balance, -Adjusted balance]  
CPUCreditBalance = 0
```

如果用掉的剩餘額度超過執行個體可累積的最高額度，則剩餘額度餘額會設定為最高額度，如先前的方程式中所示。剩下的剩餘額度會收取費用 (以 CPUSurplusCreditsCharged 指標表示)。

Example

```
CPUSurplusCreditsCharged = max [-Adjusted balance - max earned credit balance, 0]
```

最後，當執行個體終止時，透過 CPUSurplusCreditBalance 追蹤的所有剩餘額度都會計費。如果執行個體從 unlimited 切換為 standard，也會針對所有剩下的 CPUSurplusCreditBalance 收費。

GPU 執行個體的效能加速

GPU 執行個體可讓您存取含數千個運算核心的 NVIDIA GPU。您可以使用這些執行個體，藉由運用 CUDA 或開放運算語言 (OpenCL) 平行運算架構來加速科學、工程和算圖應用程式。您也可以將它們用於圖形應用程式，包括遊戲串流、3D 應用程式串流和其他圖形工作負載。

您必須先安裝適當的驅動程式，才能啟動或最佳化 GPU 執行個體，如下所示：

- 若要在具有連接 NVIDIA GPU 的執行個體 (例如 P3 或 G4dn 執行個體) 上安裝 NVIDIA 驅動程式，請參閱。[安裝 NVIDIA 驅動程式](#)
- 若要在具有連接 AMD GPU 的執行個體 (例如 G4ad 執行個體) 上安裝 AMD 驅動程式，請參閱。[安裝 AMD 驅動程式](#)

目錄

- [在您的 Amazon EC2 GPU 執行個體上啟用 NVIDIA 網格虛擬應用程式](#)
- [優化 Amazon EC2 執行個體上的 GPU 設定](#)
- [在 G4ad 執行個體上設定雙 4K 顯示器](#)
- [開始使用適用於 Linux 的 P5 執行個體](#)

在您的 Amazon EC2 GPU 執行個體上啟用 NVIDIA 網格虛擬應用程式

若要在具有 NVIDIA GPU 的 GPU 執行個體上啟用 GRID 虛擬應用程式 (預設為啟用 NVIDIA GRID 虛擬工作站)，您必須定義驅動程式的產品類型，如下所示。

在 Linux 執行個體上啟用 GRID 虛擬應用

1. 從提供的範本檔案建立 `/etc/nvidia/gridd.conf` 檔案。

```
[ec2-user ~]$ sudo cp /etc/nvidia/gridd.conf.template /etc/nvidia/gridd.conf
```

2. 在您偏好的文字編輯器中開啟 `/etc/nvidia/gridd.conf` 檔案。
3. 找到 `FeatureType` 行，並將其設定為等於 `0`。然後新增一行 `IgnoreSP=TRUE`。

```
FeatureType=0 IgnoreSP=TRUE
```

4. 儲存檔案並結束。
5. 重新開機執行個體，套用新的組態。

```
[ec2-user ~]$ sudo reboot
```

啟用網格虛擬應用程式

啟用網格虛擬應用程式

1. 開啟 regedit.exe 來開啟登錄編輯程式。
2. 導覽至 HKEY_LOCAL_MACHINE\SOFTWARE\NVIDIA Corporation\Global\GridLicensing。
3. 開啟右窗格中的內容選單 (按一下滑鼠右鍵)，然後選擇 New (新增)、DWORD。
4. 對於「名稱」，輸入 FeatureType 並鍵入 Enter。
5. 開啟上下文 (按一下右鍵) 功能表，FeatureType 然後選擇「修改」。
6. 針對 Value data (值資料)，請為 NVIDIA GRID 虛擬應用程式輸入 0，並選擇 OK (確定)。
7. 開啟右窗格中的內容選單 (按一下滑鼠右鍵)，然後選擇 New (新增)、DWORD。
8. 針對 Name (名稱)，輸入 IgnoreSP 然後輸入 Enter。
9. 開啟 IgnoreSP 上的內容選單 (按一下滑鼠右鍵)，然後選擇 Modify (修改)。
10. 針對 Value data (值資料)，輸入 1 並選擇 OK (確定)。
11. 關閉登錄編輯器。

優化 Amazon EC2 執行個體上的 GPU 設定

您可以執行數種 GPU 設定最佳化，讓 NVIDIA GPU 執行個體發揮最佳效能。透過其中一些執行個體類型，NVIDIA 驅動程式會使用 Autoboot 功能，而這會改變 GPU 時脈速度。您可以停用 Autoboot 並將 GPU 時脈速度設為頻率上限，即可持續發揮 GPU 執行個體的最大效能。

最佳化 GPU 設定

1. 設定 GPU 以保有持續性。此命令可能需要幾分鐘的執行時間。

```
[ec2-user ~]$ sudo nvidia-persistenced
```

2. [僅限 G3 和 P2 執行個體] 停用執行個體上所有 GPU 的自動加速功能。

```
[ec2-user ~]$ sudo nvidia-smi --auto-boost-default=0
```

3. 將所有 GPU 時脈速度設為頻率上限。使用下列命令中指定的記憶體和圖形時脈速度。

某些版本的 NVIDIA 驅動程式不支援設定應用程式時脈速度，會顯示錯誤 "Setting applications clocks is not supported for GPU..."，您可以忽略此錯誤。

- G3 執行個體：

```
[ec2-user ~]$ sudo nvidia-smi -ac 2505,1177
```

- G4dn 執行個體：

```
[ec2-user ~]$ sudo nvidia-smi -ac 5001,1590
```

- G5 執行個體：

```
[ec2-user ~]$ sudo nvidia-smi -ac 6250,1710
```

- G6 和 GR6 執行個體：

```
[ec2-user ~]$ sudo nvidia-smi -ac 6251,2040
```

- P2 執行個體：

```
[ec2-user ~]$ sudo nvidia-smi -ac 2505,875
```

- P3 和 P3dn 執行個體：

```
[ec2-user ~]$ sudo nvidia-smi -ac 877,1530
```

- P4d 執行個體：

```
[ec2-user ~]$ sudo nvidia-smi -ac 1215,1410
```

- P4de 執行個體：

```
[ec2-user ~]$ sudo nvidia-smi -ac 1593,1410
```

- P5 執行個體：

```
[ec2-user ~]$ sudo nvidia-smi -ac 2619,1980
```

最佳化視窗上的 GPU 設定

1. 開啟視 PowerShell 窗並導覽至 NVIDIA 安裝資料夾。

```
cd "C:\Windows\System32\DriverStore\FileRepository\nv_dispswi.inf_*\"
```

2. [僅限 G3 和 P2 執行個體] 停用執行個體上所有 GPU 的自動加速功能。

```
.\nvidia-smi --auto-boost-default=0
```

3. 將所有 GPU 時脈速度設為頻率上限。使用下列命令中指定的記憶體和圖形時脈速度。

某些版本的 NVIDIA 驅動程式不支援設定應用程式時脈速度，會顯示錯誤 "Setting applications clocks is not supported for GPU..."，您可以忽略此錯誤。

- G3 執行個體：

```
.\nvidia-smi -ac "2505,1177"
```

- G4dn 執行個體：

```
.\nvidia-smi -ac "5001,1590"
```

- G5 執行個體：

```
.\nvidia-smi -ac "6250,1710"
```

- G6 和 GR6 執行個體：

```
.\nvidia-smi -ac "6251,2040"
```

- P2 執行個體：

```
.\nvidia-smi -ac "2505,875"
```

- P3 和 P3dn 執行個體：

```
.\nvidia-smi -ac "877,1530"
```

- P4d 執行個體：


```
[ec2-user ~]$ sudo nvidia-smi -ac 1215,1410
```

- P4de 執行個體：

```
[ec2-user ~]$ sudo nvidia-smi -ac 1593,1410
```

- P5 執行個體：

```
[ec2-user ~]$ sudo nvidia-smi -ac 2619,1980
```

在 G4ad 執行個體上設定雙 4K 顯示器

啟動 G4ad 執行個體

1. 連線到 Linux 執行個體以獲取您希望用於雙 4K (2x4k) 的目標 GPU PCI 匯流排地址：

```
lspci -vv | grep -i amd
```

您將會得到類似下列的輸出：

```
00:1e.0 Display controller: Advanced Micro Devices, Inc. [*AMD*/ATI] Device 7362 (rev c3)
Subsystem: Advanced Micro Devices, Inc. [AMD/ATI] Device 0a34
```

2. 注意上述輸出中 PCI 匯流排地址為 00:1e.0。建立名為 `/etc/modprobe.d/amdgpu.conf` 的檔案並新增：

```
options amdgpu virtual_display=0000:00:1e.0,2
```

3. 若要在 Linux 上安裝 AMD 驅動程式，請參閱[在您的 Amazon EC2 實例上安裝 AMD 驅動程序](#)。如果您已經安裝了 AMD GPU 驅動程式，則需要透過 DKMS 重建 amdgpu 核心模組。
4. 使用下面的 `xorg.conf` 檔案定義雙 (2x4k) 畫面拓撲，並將檔案儲存在 `/etc/X11/xorg.conf`：

```
~$ cat /etc/X11/xorg.conf
Section "ServerLayout"
    Identifier      "Layout0"
    Screen          0  "Screen0"
    Screen          1  "Screen1"
    InputDevice     "Keyboard0" "CoreKeyboard"
```

```
    InputDevice    "Mouse0" "CorePointer"
    Option         "Xinerama" "1"
EndSection
Section "Files"
    ModulePath    "/opt/amdgpu/lib64/xorg/modules/drivers"
    ModulePath    "/opt/amdgpu/lib/xorg/modules"
    ModulePath    "/opt/amdgpu-pro/lib/xorg/modules/extensions"
    ModulePath    "/opt/amdgpu-pro/lib64/xorg/modules/extensions"
    ModulePath    "/usr/lib64/xorg/modules"
    ModulePath    "/usr/lib/xorg/modules"
EndSection
Section "InputDevice"
    # generated from default
    Identifier    "Mouse0"
    Driver        "mouse"
    Option        "Protocol" "auto"
    Option        "Device" "/dev/psaux"
    Option        "Emulate3Buttons" "no"
    Option        "ZAxisMapping" "4 5"
EndSection
Section "InputDevice"
    # generated from default
    Identifier    "Keyboard0"
    Driver        "kbd"
EndSection

Section "Monitor"
    Identifier    "Virtual"
    VendorName    "Unknown"
    ModelName     "Unknown"
    Option        "Primary" "true"
EndSection

Section "Monitor"
    Identifier    "Virtual-1"
    VendorName    "Unknown"
    ModelName     "Unknown"
    Option        "RightOf" "Virtual"
EndSection

Section "Device"
    Identifier    "Device0"
    Driver        "amdgpu"
    VendorName    "AMD"
```

```
BoardName      "Radeon MxGPU V520"  
BusID          "PCI:0:30:0"  
EndSection  
  
Section "Device"  
Identifier     "Device1"  
Driver        "amdgpu"  
VendorName    "AMD"  
BoardName     "Radeon MxGPU V520"  
BusID        "PCI:0:30:0"  
EndSection  
  
Section "Extensions"  
Option        "DPMS" "Disable"  
EndSection  
  
Section "Screen"  
Identifier    "Screen0"  
Device       "Device0"  
Monitor      "Virtual"  
DefaultDepth 24  
Option       "AllowEmptyInitialConfiguration" "True"  
SubSection "Display"  
    Virtual   3840 2160  
    Depth     32  
EndSubSection  
EndSection  
  
Section "Screen"  
Identifier    "Screen1"  
Device       "Device1"  
Monitor      "Virtual"  
DefaultDepth 24  
Option       "AllowEmptyInitialConfiguration" "True"  
SubSection "Display"  
    Virtual   3840 2160  
    Depth     32  
EndSubSection  
EndSection
```

5. 請依照「設定 [互動式桌面](#)」中的指示來設定 DCV。
6. DCV 設定完成後，請重新啟動。
7. 確認驅動程式可正常運作：

```
dmesg | grep amdgpu
```

回應如下所示：

```
Initialized amdgpu
```

8. 您應該會在 `DISPLAY=:0 xrandr -q` 輸出中看到您已連接 2 個虛擬顯示器：

```
~$ DISPLAY=:0 xrandr -q
Screen 0: minimum 320 x 200, current 3840 x 1080, maximum 16384 x 16384
Virtual connected primary 1920x1080+0+0 (normal left inverted right x axis y axis)
 0mm x 0mm
 4096x3112  60.00
 3656x2664  59.99
 4096x2160  60.00
 3840x2160  60.00
 1920x1200  59.95
 1920x1080  60.00
 1600x1200  59.95
 1680x1050  60.00
 1400x1050  60.00
 1280x1024  59.95
 1440x900  59.99
 1280x960  59.99
 1280x854  59.95
 1280x800  59.96
 1280x720  59.97
 1152x768  59.95
 1024x768  60.00 59.95
 800x600   60.32 59.96 56.25
 848x480   60.00 59.94
 720x480   59.94
 640x480   59.94 59.94
Virtual-1 connected 1920x1080+1920+0 (normal left inverted right x axis y axis) 0mm x
 0mm
 4096x3112  60.00
 3656x2664  59.99
 4096x2160  60.00
 3840x2160  60.00
 1920x1200  59.95
 1920x1080  60.00
 1600x1200  59.95
```

```

1680x1050 60.00
1400x1050 60.00
1280x1024 59.95
1440x900 59.99
1280x960 59.99
1280x854 59.95
1280x800 59.96
1280x720 59.97
1152x768 59.95
1024x768 60.00 59.95
800x600 60.32 59.96 56.25
848x480 60.00 59.94
720x480 59.94
640x480 59.94 59.94

```

9. 當您連接到 DCV 時，將解析度變更為 2x4k，確認 DCV 已註冊雙顯示器支援。



開始使用適用於 Linux 的 P5 執行個體

P5 執行個體提供 8 個 NVIDIA H100 GPU 以及 640 GB 的高頻寬 GPU 記憶體。它們配備有第三代 AMD EPYC 處理器，並提供 2 TB 的系統記憶體、30 TB 的本機 NVMe 執行個體儲存空間、3,200 個 Gbps 聚合網路頻寬，以及 GPUDirect RDMA 支援。P5 執行個體也支援 Amazon EC2 UltraCluster 技術，使用 EFA 提供更低的延遲和改善的網路效能。

如下資料表提供 p5.48xlarge 規格的摘要。

vCPU	系統記憶體	GPU	記憶體	網路頻寬	GPUDirect RDMA	GPU 点对点	執行個體儲存體
192	2 TiB	8 NVIDIA H100 GPUs	640 GB HBM3	3,200 Gbps with EFAv2	支援	900 GB/s NVSw	8 x 3,800 GB NVMe SSD 磁碟區

軟體組態

開始使用 P5 執行個體的最簡單方法是使用已針對所有必要軟體預先設定的 AWS Deep Learning AMI 啟動執行個體。如需與 P5 執行個體搭配使用的最新 AWS Deep Learning AMI 資訊，請參閱[AWS 深度學習基礎 GPU AMI \(Ubuntu 20.04\)](#)。

如果您需要建置自訂 AMI 以搭配 P5 執行個體使用，建議您安裝下列最低軟體版本：

- NVIDIA 驅動程式 535.54.03 或更新版本
- CUDA 12.1 或更新版本
- NVIDIA GDRCopy 2.3 或更新版本
- EFA 安裝程式 1.24.1 或更新版本
- NCCL 2.18.3 或更新版本
- aws-ofi-nccl 插件 1.7.2-AWS 或更高版本

我們也建議您將執行個體設定為不使用深層的 C-states。如需詳細資訊，請參閱 Amazon Linux 2 使用者指南中的[限制更深 C 狀態的高效能和低延遲](#)。最新的 AWS 深度學習基礎 GPU AMI 已預先設定為不使用更深層的 C 狀態。

Ubuntu 20.04 具體建議

下列有關 Ubuntu 20.04 的建議有助於避免開機時出現無法預期的介面命名方式：

- 確保您正在執行 systemd 245.4-4ubuntu3.19，或稍後使用以下命令：

```
systemd --version
```

- 請確認您已設定 GRUB：
 - 在文字編輯器中開啟 `/etc/default/grub` 組態檔案。
 - 編輯 `GRUB_CMDLINE_LINUX_DEFAULT` 項目，以便包含 `net.naming-scheme=v247`。
 - 執行 `sudo update-grub` 以重新啟動執行個體。

聯網與 EFA 組態

P5 執行個體使用多個 EFA 介面來提供 3200 Gbps 的聯網頻寬。P5 執行個體支援 32 個網路卡。我們建議您為每張網路卡定義單一 EFA 網路介面。若要在啟動時設定這些介面，我們建議您進行下列設定：

- 對於網路介面 `0`，指定裝置索引 `0`

- 對於網路介面 1 至 31，指定裝置索引 1

如需如何針對 EFA 設定 P5 執行個體的詳細資訊，請參閱 [開始使用 P5 執行個體和 EFA](#)。

Amazon EC2 Mac 執行個體

Amazon EC2 Mac 執行個體原生支援 macOS 作業系統。

- EC2 x86 Mac 執行個體 (mac1.metal) 是以 2018 Mac mini 硬體為基礎，並搭載 3.2 GHz Intel 第八代 (Coffee Lake) Core i7 處理器。
- EC2 M1 Mac 執行個體 (mac2.metal) 是建置在由 Apple 晶片 M1 處理器驅動的 2020 Mac mini 硬體上。
- EC2 M2 Mac 執行個體 (mac2-m2.metal) 是建置在由 Apple 晶片 M2 處理器驅動的 2023 Mac mini 硬體上。
- EC2 M2 Pro Mac 執行個體 (mac2-m2pro.metal) 是建置在由 Apple 晶片 M2 Pro 處理器驅動的 2023 Mac mini 硬體上。

EC2 Mac 執行個體適用於開發、建置、測試和簽署 Apple 平台的應用程式，這些平台如 iPhone、iPad、iPod、Mac、Vision Pro、Apple Watch、Apple TV 和 Safari。您可以使用 SSH 或 Apple 遠端桌面 (ARD) 連線至 Mac 執行個體。

Note

以專用執行個體為計費單位。在該主機上執行的執行個體並無額外費用。

目錄

- [考量事項](#)
- [執行個體整備](#)
- [EC2 macOS AMI](#)
- [EC2 macOS Init](#)
- [Amazon EC2 系統監視器](#)
- [相關資源](#)
- [啟動 Mac 執行個體](#)

- [連接至您的 Mac 執行個體](#)
- [更新 Mac 實例上的操作系統和軟件](#)
- [增加 Mac 執行個體上 EBS 磁碟區的大小](#)
- [停止並終止您的 Mac 執行個體](#)
- [為您的專用主機尋找支援的 macOS 版本](#)
- [訂閱 macOS AMI 通知](#)
- [Amazon EC2](#)

考量事項

Mac 執行個體適用下列考量：

- Mac 執行個體只能做為 [專用執行個體](#) 上的裸機執行個體使用，最小配置期間為 24 小時，這段期間過後您才可以發佈專用執行個體。每個 專用執行個體 可以啟動一個 Mac 執行個體。您可以與組織內的 AWS 帳戶或組織單位或整個 AWS 組 AWS 織共用專用主機。
- Mac 實例有不同的可用 AWS 區域。如需各個 Mac 執行個體可用性的清單 AWS 區域，請參閱各 [區域的 Amazon EC2 執行個體類型](#)。
- Mac 執行個體只能做為 隨需執行個體 使用，不可做為 競價型執行個體 或 預留執行個體 使用。您可以購買 [Savings Plan](#)，節省 Mac 執行個體的費用。
- Mac 執行個體可以執行下列其中一個作業系統：
 - macOS Mojave (版本 10.14) (僅限 x86 Mac 執行個體)
 - macOS Catalina (版本 10.15) (僅限 x86 Mac 執行個體)
 - macOS Big Sur (版本 11) (x86 和 M1 Mac 執行個體)
 - macOS Monterey (版本 12) (x86 和 M1 Mac 執行個體)
 - macOS Ventura (13 版) (所有 Mac 執行個體、M2 和 M2 Pro Mac 執行個體支援 macOS Ventura 13.2 版或更新版本)
 - macOS Sonoma (版本 14) (所有 Mac 執行個體)
- EBS支持熱插拔。
- AWS 不管理或支持蘋果硬件上的內部 SSD。我們強烈建議您改用 Amazon EBS 磁碟區。EBS磁碟區在 Mac 執行個體上提供與其他 EC2 執行個體相同的彈性、可用性和耐久性優勢。
- 我們建議使用一般用途 SSD (gp2和gp3) 和佈建 IOPS SSD (io1和io2) 搭配 Mac 執行個體，以獲得最佳EBS效能。
- [Mac 執行個體支援 Amazon EC2 Auto Scaling。](#)

- 在 x86 Mac 執行個體上，自動軟體更新會停用。我們建議您在將執行個體投入實際執行環境前套用更新，並在執行個體上測試更新。如需詳細資訊，請參閱 [更新 Mac 實例上的操作系統和軟件](#)。
- 當您停止或終止 Mac 執行個體時，系統會在專用主機上執行清理工作流程。如需詳細資訊，請參閱 [停止並終止您的 Mac 執行個體](#)。

Warning

請勿使用 FileVault。啟用 FileVault 將導致主機因為分割區被鎖定而無法開機。如果需要資料加密，請使用 Amazon EBS 加密以避免開機問題和效能影響。使用 Amazon EBS 加密時，會在託管執行個體的伺服器上進行加密操作，以確保執行個體 data-at-rest 及其連接 EBS 儲存 data-in-transit 之間以及之間的安全性。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的 Amazon EBS 加密](#)

執行個體整備

啟動 Mac 執行個體之後，您將需要等待執行個體準備就緒後，才能夠與其連接。對於具有 x86 Mac 執行個 AWS 體或蘋果矽晶 Mac 執行個體的供貨 AMI，啟動時間可以從大約 6 分鐘到 20 分鐘不等。根據選擇的 Amazon EBS 磁碟區大小、使用者資料中包含的額外指令、或自訂 macOS AMI 上的其他載入軟體，啟動時間可能會增加。

您可以使用如下所示的小型 shell 指令碼來輪詢 describe-instance-status API，以瞭解執行個體何時可以連線到。將以下命令中的範例執行個體 ID 取代為您自己的 ID：

```
for i in $(seq 1 200); do aws ec2 describe-instance-status --instance-ids=i-0123456789example \
  --query='InstanceStatuses[0].InstanceStatus.Status'; sleep 5; done;
```

EC2 macOS AMI

Amazon EC2 macOS 設計能為在 Amazon EC2 Mac 執行個體上執行的開發人員工作負載提供穩定、安全且高效能的環境。EC2 macOS AMI 包含可輕鬆整合的套件 AWS，例如啟動設定工具和熱門的程 AWS 式庫和工具。

如需 EC2 macOS AMI 的詳細資訊，請參閱 [Amazon EC2](#)。

AWS 定期提供更新的 EC2 macOS AMI，其中包括對所擁有套件的更新，以 AWS 及最新的完整測試 macOS 版本。此外，AWS 也會在更新的 AMI 經過完整測試和驗證後，提供具有最新次要版本更新或

主要版本更新的 AMI。如果您無須保留 Mac 執行個體的資料或自訂，您可以使用目前的 AMI 啟動新的執行個體，然後終止上一個執行個體，以取得最新的更新。否則您可以選擇哪些更新要套用到您的 Mac 執行個體。

如需有關如何訂閱 macOS AMI 通知的詳細資訊，請參閱[訂閱 macOS AMI 通知](#)。

EC2 macOS Init

EC2macOS 初始化用於在啟動時初始化 EC2 Mac 執行個體。它會使用優先順序群組，來同時執行任務的邏輯群組。

launchd plist 檔案是 `/Library/LaunchDaemons/com.amazon.ec2.macos-init.plist`。EC2 macOS Init 的檔案位於 `/usr/local/aws/ec2-macos-init`。

如需詳細資訊，請參閱 <https://github.com/aws/ec2-macos-init>。

Amazon EC2 系統監視器

適用於 macOS 的 Amazon EC2 系統監視器為 Amazon 提供 CPU 使用率指標 CloudWatch。它會在 1 分鐘內 CloudWatch 透過自訂序列裝置傳送這些指標。您可以啟用或停用此代理程式，如下所示。依預設會啟用此功能。

```
sudo setup-ec2monitoring [enable | disable]
```

Note

蘋果矽晶片 Mac 執行個體目前不支援適用於 macOS 的 Amazon EC2 系統監視器。

相關資源

如需定價的資訊，請參閱 [定價](#)。

如需 Mac 執行個體的詳細資訊，請參閱 [Amazon EC2 Mac 執行個體](#)。

如需 Mac 執行個體的硬體規格和網路效能的詳細資訊，請參閱 [一般用途執行個體](#)。

啟動 Mac 執行個體

EC2 Mac 執行個體需要[專用執行個體](#)。您首先需要將主機配置到您的帳戶，然後在主機上啟動執行個體。

您可以使用 AWS Management Console 或啟動 Mac 執行個體 AWS CLI。

使用主控台啟動 Mac 執行個體

在 專用主機 上啟動 Mac 實體啟動

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 配置專用執行個體，如下所示：
 - a. 在導覽窗格中，選擇 專用執行個體。
 - b. 選擇 Allocate 專用執行個體 (配置 專用執行個體)，然後執行下列動作：
 - i. 針對執行個體系列，請選擇 mac1、mac2、mac2-m2 或 mac2-m2pro。如果 Instance family (執行個體系列) 未在清單中出現，則表示目前所選的區域不支援。
 - ii. 針對執行個體類型，請根據您選取的執行個體系列，選取 mac1.metal、mac2.metal、mac2-m2.metal 或 mac2-m2pro.metal。
 - iii. 在 Availability Zone (可用區域) 中，選擇 專用執行個體 的可用區域。
 - iv. 在 Quantity (數量) 中，保留 1。
 - v. 選擇 Allocate (配置)。
3. 在主機上啟動執行個體，如下所示：
 - a. 選取您建立的 專用執行個體，然後執行下列動作：
 - i. 選擇 Actions (動作)、Launch instance(s) onto host (在主機上啟動執行個體)。
 - ii. 在 Application and OS Images (Amazon Machine Image) (應用程式和作業系統映像 (Amazon Machine Image)) 下，選取 macOS AMI。
 - iii. 在執行個體類型下，選取適當的執行個體類型 (mac1.metal、mac2.metal、mac2-m2.metal 或 mac2-m2pro.metal)。
 - iv. 在 Advanced details (進階詳細資訊) 下，確認已根據您建立的專用執行個體預先設定 Tenancy (租用)、Tenancy host by (租用執行個體者) 和 Tenancy host ID (租用執行個體 ID)。視需要更新 Tenancy affinity (租用親和性)。
 - v. 視需要完成精靈、指定 EBS 磁碟區、安全群組和金鑰對。
 - vi. 在 Summary (摘要) 面板中，選擇 Launch instance (啟動執行個體)。
 - b. 會有確認頁面讓您知道您的執行個體正在啟動。選擇 View all instances (檢視所有執行個體)，以關閉確認頁面並返回主控台。執行個體的初始狀態為 pending。當其狀態變更為 running 並通過狀態檢查時，則表示該執行狀態就緒。

啟動 Mac 執行個體，使用 AWS CLI

配置專用執行個體

使用下列 [allocate-hosts](#) 命令為您的 Mac 執行個體配置專用執行個體，將 `instance-type` 取代為 `mac1.metal`、`mac2.metal`、`mac2-m2.metal` 或 `mac2-m2pro.metal`，並將 `region` 和 `availability-zone` 取代為適合您環境的項目。

```
aws ec2 allocate-hosts --region us-east-1 --instance-type mac1.metal --availability-zone us-east-1b --auto-placement "on" --quantity 1
```

在主機上啟動執行個體

使用下列 [run-instances](#) 命令來啟動 Mac 執行個體，再次將其 `instance-type` 取代為 `mac1.metal`、`mac2.metal`、`mac2-m2.metal` 或 `mac2-m2pro.metal`，並將 `region` 和 `availability-zone` 取代為之前使用的執行個體。

```
aws ec2 run-instances --region us-east-1 --instance-type mac1.metal --placement Tenancy=host --image-id ami_id --key-name my-key-pair
```

執行個體的初始狀態為 `pending`。當其狀態變更為 `running` 並通過狀態檢查時，則表示該執行狀態就緒。使用以下 [describe-instance-status](#) 命令顯示您執行個體的状态資訊。

```
aws ec2 describe-instance-status --instance-ids i-017f8354e2dc69c4f
```

以下是執行中並通過狀態檢查之執行個體的範例輸出。

```
{
  "InstanceStatuses": [
    {
      "AvailabilityZone": "us-east-1b",
      "InstanceId": "i-017f8354e2dc69c4f",
      "InstanceState": {
        "Code": 16,
        "Name": "running"
      },
      "InstanceStatus": {
        "Details": [
          {
            "Name": "reachability",
```

```
        "Status": "passed"
      }
    ],
    "Status": "ok"
  },
  "SystemStatus": {
    "Details": [
      {
        "Name": "reachability",
        "Status": "passed"
      }
    ],
    "Status": "ok"
  }
}
]
```

連接至您的 Mac 執行個體

您可以使用 SSH 或圖形使用者介面連線至 Mac 執行個體。

使用 SSH 連線至您的執行個體

Important

多個使用者可以同時存取作業系統。由於連接埠 5900 上提供內建螢幕共享服務，所以通常有一個 1:1 user:GUI 工作階段。在 macOS 中使用 SSH 支援多個工作階段，最多可達 `sshd_config` 檔案中的 "Max Sessions" (工作階段上限) 限制。

Amazon EC2 Mac 執行個體預設不允許遠端根 SSH。密碼驗證也將停用，以避免密碼遭到暴力破解攻擊。ec2-user 帳戶設為使用 SSH 進行遠端登入。ec2-user 用戶也擁有 sudo 權限。連線到您的執行個體後，您可以新增其他使用者。

若要支援使用 SSH 連線到您的執行個體，請使用允許 SSH 存取的金鑰對和安全群組啟動執行個體，並確保執行該個體具有網際網路連線能力。連線到執行個體時，您可以提供金鑰對的 .pem 檔案。

請使用下列程序使用 SSH 用戶端連線至您的 Mac 執行個體。如果您在嘗試連線至執行個體時遇到錯誤，請參閱 [連線到 Linux 執行個體的疑難排解](#)。

使用 SSH 連線到您的執行個體

1. 在命令列輸入 `ssh`，以確認您的本機電腦已安裝 SSH 用戶端。如果您的電腦無法辨識命令，請搜尋適用於作業系統的 SSH 用戶端並安裝該用戶端。
2. 取得執行個體的公有 DNS 名稱。您可以使用 Amazon EC2 主控台尋找 Details (詳細資訊) 和 Networking (網路) 標籤上的公有 DNS 名稱。使用 AWS CLI，您可以使用 [描述執行個體](#) 命令尋找公用 DNS 名稱。
3. 找到您在啟動執行個體時所指定金鑰對的 `.pem` 檔案。
4. 使用以下 `ssh` 命令、指定執行個體的公有 DNS 名稱和 `.pem` 檔案，以連線到您的執行個體。

```
ssh -i /path/key-pair-name.pem ec2-user@instance-public-dns-name
```

連線至您的執行個體的圖形使用者介面 (GUI)

請使用下列程序，藉助 VNC、Apple 遠端桌面 (ARD) 或 Apple 螢幕共享應用程式 (隨附於 macOS) 連線至您的執行個體 GUI。

Note

macOS 10.14 及以上版本只允許控制是否透過 [System Preferences \(系統偏好設定\)](#) 啟用 Screen Sharing (螢幕共用)。

使用 ARD 用戶端或 VNC 用戶端連線到您的執行個體

1. 確認您的本機電腦已安裝 ARD 用戶端，或支援已安裝 ARD 的 VNC 用戶端。在 macOS 上，您可以利用內建的「螢幕共用」應用程式。否則請搜尋適用於您作業系統的 ARD，然後安裝該用戶端。
2. 從您的本機電腦，[使用 SSH 連線到您的執行個體](#)。
3. 如下所示使用 `passwd` 命令設定 `ec2-user` 帳戶的密碼。

```
[ec2-user ~]$ sudo passwd ec2-user
```

4. 使用下列命令，安裝並啟動 macOS Screen Sharing。

```
[ec2-user ~]$ sudo launchctl enable system/com.apple.screensharing  
sudo launchctl load -w /System/Library/LaunchDaemons/com.apple.screensharing.plist
```

5. 輸入 `exit` 並按 `Enter`，斷開與執行個體的連線。
6. 從您的電腦，使用以下 `ssh` 命令連線到執行個體。除了上一節所述的選項之外，使用 `-L` 選項也可啟用連接埠轉送，並將本機連接埠 5900 的所有流量轉送至執行個體上的 ARD 伺服器。

```
ssh -L 5900:localhost:5900 -i /path/key-pair-name.pem ec2-user@instance-public-dns-name
```

7. 從您的本機電腦，使用支援 ARD 的 ARD 用戶端或 VNC 用戶端連線至 `localhost:5900`。例如，使用 macOS 上的「螢幕共用」應用程式，如下所示：
 - a. 開啟 Finder，並選取開始。
 - b. 選取連線至伺服器。
 - c. 在伺服器地址欄位中，輸入 `vnc://localhost:5900`。
 - d. 依提示使用 `ec2-user` 您為 `ec2-user` 帳戶建立的使用者名稱和密碼進行登入。

修改 Mac 執行個體上的 macOS 螢幕解析度

使用 ARD 或支援 ARD 的 VNC 用戶端連線至 EC2 Mac 執行個體後，您就可以使用任何公開可用的 macOS 工具或公用程式，例如 [displayplacer](#) 來修改 macOS 環境的螢幕解析度。

若要使用 `displayplacer` 修改螢幕解析度

1. 安裝 `displayplacer`。

```
[ec2-user ~]$ brew tap jakehilborn/jakehilborn && brew install displayplacer
```

2. 顯示目前的螢幕資訊和可能的螢幕解析度。

```
[ec2-user ~]$ displayplacer list
```

3. 套用所需的螢幕解析度。

```
[ec2-user ~]$ displayplacer "id:<screenID> res:<width>x<height> origin:(0,0) degree:0"
```

例如：

```
RES="2560x1600"
```

```
displayplacer "id:69784AF1-CD7D-B79B-E5D4-60D937407F68 res:${RES} scaling:off
origin:(0,0) degree:0"
```

更新 Mac 實例上的操作系統和軟件

Warning

只有 Amazon EC2 M1 Mac 執行個體才能安裝 Beta 版或預覽版 macOS 版本。Amazon EC2 不符合 Beta 版或預覽版 macOS 版本的資格，且無法確保執行個體在更新至生產前 macOS 版本後仍能正常運作。

嘗試在 Amazon EC2 x86 Mac 執行個體上安裝 Beta 版或預覽版 macOS 版本，會導致停止或終止執行個體時，EC2 Mac 專用執行個體降級，並且會阻止您在該專用執行個體上開始或啟動新的執行個體。

在 x86 Mac 執行個體和 Apple 晶片 Mac 執行個體上更新軟體的步驟。

- [在 x86 Mac 執行個體上更新軟體](#)
- [在 Apple 晶片 Mac 執行個體上更新軟體](#)

在 x86 Mac 執行個體上更新軟體

在 x86 Mac 執行個體上，您可以使用 `softwareupdate` 指令從 Apple 安裝作業系統更新。

若要在 x86 Mac 執行個體上從 Apple 安裝作業系統更新

1. 使用以下命令列出具有可用更新的套件。

```
[ec2-user ~]$ softwareupdate --list
```

2. 安裝所有更新或僅特定更新。若要安裝特定更新，請使用下列命令。

```
[ec2-user ~]$ sudo softwareupdate --install label
```

若要安裝所有更新，請使用下列命令。

```
[ec2-user ~]$ sudo softwareupdate --install --all --restart
```


系統管理員可以用 AWS Systems Manager 來在 x86 Mac 執行個體上推出預先核准的作業系統更新。如需詳細資訊，請參閱 [AWS Systems Manager 使用者指南](#)。

您可以使用 Homebrew 在 EC2 macOS AMI 中安裝套件的更新，以便在執行個體上擁有這些套件的最新版本。您也可以使用 Homebrew 在 Amazon EC2 macOS 上安裝和執行常用的 macOS 應用程式。如需詳細資訊，請參閱 [Homebrew 文件](#)。

使用 Homebrew 安裝更新

1. 使用以下命令更新 Homebrew。

```
[ec2-user ~]$ brew update
```

2. 使用以下命令列出具有可用更新的套件。

```
[ec2-user ~]$ brew outdated
```

3. 安裝所有更新或僅特定更新。若要安裝特定更新，請使用下列命令。

```
[ec2-user ~]$ brew upgrade package name
```

若要安裝所有更新，請使用下列命令。

```
[ec2-user ~]$ brew upgrade
```

在 Apple 晶片 Mac 執行個體上更新軟體

考量事項

彈性網路介面卡 (ENA) 驅動程式

由於網路驅動程式組態中的更新，ENA 驅動程式 1.0.2 版與 macOS 13.3 或更新版本不相容。如果您要安裝 Beta 版、預覽版或生產 macOS 13.3 版或更新版本，且尚未安裝最新的 ENA 驅動程式，請使用下列程序安裝新的驅動程式版本。

安裝 ENA 驅動程式的新版本

1. 在「終端機」視窗中，使用 [SSH](#) 連線至 Apple 晶片 Mac 執行個體。
2. 使用下列命令將 ENA 應用程式下載到 Applications 檔案中。

```
[ec2-user ~]$ brew install amazon-ena-ethernet-dext
```

疑難排解秘訣

如果您收到警告 No available formula with the name amazon-ena-ethernet-dext，則請執行下列命令。

```
[ec2-user ~]$ brew update
```

3. 輸入 exit 並按 return，斷開與執行個體的連線。
4. 使用 VNC 用戶端啟動 ENA 應用程式。
 - a. 使用 [連線至您的執行個體的圖形使用者介面 \(GUI\)](#) 設定 VNC 用戶端。
 - b. 使用「螢幕共用」應用程式連線至執行個體後，請前往應用程式資料夾並開啟 ENA 應用程式。
 - c. 選擇啟用
 - d. 若要確認驅動程式已正確啟動，請在終端視窗中執行下列命令。命令輸出顯示舊驅動程式處於終止狀態，並且新驅動程式處於啟動狀態。

```
systemextensionsctl list;
```

- e. 重新啟動執行個體後，只會出現新的驅動程式。

Apple 晶片 Mac 執行個體上的軟體更新

在 Apple 晶片 Mac 執行個體上，您必須完成數個步驟才能執行就地作業系統更新。首先，使用具有 VNC (虛擬網路運算) 用戶端的 GUI 存取執行個體的內部磁碟。此程序使用 macOS Screen Sharing，即內建的 VNC 用戶端。然後，在 Amazon EBS 磁碟區上以 aws-managed-user 身分登入，將擁有權委派給管理使用者 (ec2-user)。

當您執行此程序時，會建立兩個密碼。一個密碼用於管理使用者 (ec2-user)，另一個密碼用於特殊管理使用者 (aws-managed-user)。請記住這些密碼，因為將在執行該程序時使用這些密碼。

Note

在 macOS Big Sur 上使用此程序，只能執行次要更新，如從 macOS Big Sur 11.7.3 更新為 macOS Big Sur 11.7.4。對於 macOS Monterey 或更高版本，您可以執行主要軟體更新。

存取內部磁碟

1. 從本機電腦的「終端機」中，透過下列命令使用 SSH 連線至 Apple 晶片 Mac 執行個體。如需詳細資訊，請參閱 [使用 SSH 連線至您的執行個體](#)。

```
ssh -i /path/key-pair-name.pem ec2-user@instance-public-dns-name
```

2. 使用下列命令，安裝並啟動 macOS Screen Sharing。

```
[ec2-user ~]$ sudo launchctl enable system/com.apple.screensharing  
sudo launchctl load -w /System/Library/LaunchDaemons/com.apple.screensharing.plist
```

3. 使用以下命令設定 ec2-user 的密碼。記住密碼，供以後使用。

```
[ec2-user ~]$ sudo /usr/bin/dscl . -passwd /Users/ec2-user
```

4. 輸入 exit 並按回車鍵，斷開與執行個體的連線。
5. 從本機電腦的「終端」中，使用下列命令透過 VNC 連接埠的 SSH 通道重新連線至執行個體。

```
ssh -i /path/key-pair-name.pem -L 5900:localhost:5900 ec2-user@instance-public-dns-name
```

Note

在完成下列 VNC 連線和 GUI 步驟之前，請勿結束此 SSH 工作階段。重新啟動執行個體時，連線會自動關閉。

6. 從您的本機電腦，使用下列步驟連線到 localhost:5900：
 - a. 開啟 Finder，並選取開始。
 - b. 選取連線至伺服器。
 - c. 在伺服器地址欄位中，輸入 vnc://localhost:5900。

7. 在 macOS 視窗中，使用您在[步驟 3](#) 中建立的密碼以 `ec2-user` 身分連線至 Apple 晶片 Mac 執行個體的遠端工作階段。
8. 使用下列其中一個選項存取名為 `InternalDisk` 的內部磁碟。
 - a. 對於 macOS Ventura 或更高版本：開啟系統設定，在左窗格中選取一般，然後選取窗格右下角的啟動磁碟。
 - b. 對於 macOS Monterey 或以下版本：開啟系統偏好設定，選取啟動磁碟，然後選擇視窗左下角的鎖定圖像來解鎖窗格。

疑難排解秘訣

如果您需要掛載內部磁碟，則請在終端中執行下列命令。

```
APFSVolumeName="InternalDisk" ; SSDContainer=$(diskutil list | grep  
"Physical Store disk0" -B 3 | grep "/dev/disk" | awk {'print $1'} ) ;  
diskutil apfs addVolume $SSDContainer APFS $APFSVolumeName
```

9. 選擇名為 `InternalDisk` 的內部磁碟，然後選取 [重新啟動]。出現提示時再次選取重新啟動。

Important

如果內部磁碟的名稱為 `Macintosh HD InternalDisk`，則您的執行個體必須停止並重新啟動，才能更新專用主機。如需詳細資訊，請參閱 [停止並終止您的 Mac 執行個體](#)。

使用下列程序將擁有權委派給管理使用者。當透過 SSH 重新連線至執行個體時，可以使用特殊管理使用者 (`aws-managed-user`) 從內部磁碟開機。`aws-managed-user` 的初始密碼為空白，因此您需要在第一次連線時覆寫它。然後，由於開機磁碟區已變更，因此您需要重複以下步驟來安裝和啟動 macOS Screen Sharing。

將擁有權委派給 Amazon EBS 磁碟區上的管理員

1. 從本機電腦的「終端機」中，使用下列命令連線至 Apple 晶片 Mac 執行個體。

```
ssh -i /path/key-pair-name.pem aws-managed-user@instance-public-dns-name
```

- 當您收到警告 `WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!` 時，請使用下列其中一個命令來解決此問題。

- 使用以下命令清除已知主機。然後，重複上一個步驟。

```
rm ~/.ssh/known_hosts
```

- 將下列命令新增到上一個步驟的 SSH 命令。

```
-o UserKnownHostsFile=/dev/null -o StrictHostKeyChecking=no
```

- 使用下列命令設定 `aws-managed-user` 的密碼。`aws-managed-user` 初始密碼為空白，因此您需要在第一次連線時覆寫它。

- ```
[aws-managed-user ~]$ sudo /usr/bin/dscl . -passwd /Users/aws-managed-user password
```

- 當您收到 `Permission denied. Please enter user's old password:` 提示時，請按 Enter 鍵。

#### 疑難排解秘訣

如果發生 `passwd: DS error: eDSAuthFailed` 錯誤，請使用下列命令。

```
[aws-managed-user ~]$ sudo passwd aws-managed-user
```

- 使用下列命令，安裝並啟動 macOS Screen Sharing。

```
[aws-managed-user ~]$ sudo launchctl enable system/com.apple.screensharing
sudo launchctl load -w /System/Library/LaunchDaemons/com.apple.screensharing.plist
```


- 輸入 `exit` 並按回車鍵，斷開與執行個體的連線。
- 從本機電腦的「終端」中，使用下列命令透過 VNC 連接埠的 SSH 通道重新連線至執行個體。

```
ssh -i /path/key-pair-name.pem -L 5900:localhost:5900 aws-managed-user@instance-public-dns-name
```

- 從您的本機電腦，使用下列步驟連線到 `localhost:5900`：


- 開啟 Finder，並選取開始。

- b. 選取連線至伺服器。
  - c. 在伺服器地址欄位中，輸入 `vnc://localhost:5900`。
8. 在 macOS 視窗中，使用您在 [步驟 3](#) 中建立的密碼以 `aws-managed-user` 身分連線至 Apple 晶片 Mac 執行個體的遠端工作階段。

 Note


當系統提示您使用 Apple ID 登入時，請選取稍後設定。

9. 使用下列其中一個選項，存取 Amazon EBS 磁碟區。
- a. 對於 macOS Ventura 或更新版本：開啟系統設定，在左窗格中選取一般，然後選取窗格右下角的啟動磁碟。
  - b. 對於 macOS Monterey 或較早的版本：開啟系統偏好設定，選取啟動磁碟，然後使用視窗左下角的鎖定圖像來解鎖窗格。

 Note

在重新開機之前，當系統提示輸入管理員密碼時，請使用您上面設定的 `aws-managed-user` 密碼。此密碼可能與您為 `ec2-user` 或執行個體上的預設管理員帳戶設定的密碼不同。下列說明指定何時使用執行個體的管理員密碼。

10. 選取 Amazon EBS 磁碟區 (未在「啟動磁碟」視窗 `InternalDisk` 中命名的磁碟區)，然後選擇「重新啟動」。

 Note

如果您將多個可開機的 Amazon EBS 磁碟區連接至 Apple 晶片 Mac 執行個體，則請務必為每個磁碟區使用唯一的名稱。

11. 確認重新啟動，然後在出現提示時選擇授權使用者。
12. 於在此磁碟區授權使用者窗格中，確認已選取管理使用者 (預設為 `ec2-user`)，然後選取授權。
13. 輸入您在上一個程序的 [步驟 3](#) 中建立的 `ec2-user` 密碼，然後選取繼續。
14. 出現提示時，請輸入特殊管理使用者的密碼 (`aws-managed-user`)。
15. 從本機電腦的「終端」中，使用 SSH 與使用者名稱 `ec2-user` 重新連線至執行個體。

**i** 疑難排解秘訣

如果收到警告 `WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!`，則請執行下列命令，然後使用 SSH 重新連線至執行個體。

```
rm ~/.ssh/known_hosts
```

16. 若要執行軟體更新，請使用 [在 x86 Mac 執行個體上更新軟體](#) 下的命令。

## 增加 Mac 執行個體上 EBS 磁碟區的大小

您可以增加 Mac 執行個體上 Amazon EBS 磁碟區的大小。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的 Amazon EBS 彈性磁碟區](#)。

增加磁碟區的大小後，您必須依照下列方式增加 APFS 容器的大小。

使增加的磁碟空間可供使用

1. 判斷是否需要重新啟動。如果您已調整正在執行的 Mac 執行個體上的現有 EBS 磁碟區的大小，則須 [重新啟動](#) 執行個體，以使得新的大小可用。如果在啟動期間修改磁碟空間，則不需要重新開機。

檢視磁碟大小的目前狀態：

```
[ec2-user ~]$ diskutil list external physical
/dev/disk0 (external, physical):
#: TYPE NAME SIZE IDENTIFIER
0: GUID_partition_scheme *322.1 GB disk0
1: EFI EFI 209.7 MB disk0s1
2: Apple_APFS Container disk2 321.9 GB disk0s2
```

2. 複製並貼上下列命令。

```
[ec2-user ~]$ PDISK=$(diskutil list physical external | head -n1 | cut -d" " -f1)
APFSCONT=$(diskutil list physical external | grep "Apple_APFS" | tr -s " " | cut -d" " -f8)
yes | sudo diskutil repairDisk $PDISK
```

3. 複製並貼上下列命令。

```
[ec2-user ~]$ sudo diskutil apfs resizeContainer $APFSCONT 0
```

## 停止並終止您的 Mac 執行個體

當您停止 Mac 執行個體時，該執行個體會進入 stopped 狀態之前維持約 15 分鐘的 stopping 狀態。

當您停止或終止 Mac 執行個體時，Amazon EC2 會在底層的專用執行個體上執行清理工作流程以清除內部 SSD、清除持續性 NVRAM 變數，並更新至最新的裝置韌體。這可確保 Mac 執行個體提供與其他 EC2 Nitro 執行個體相同的安全性和資料隱私權。它還允許您運行最新的 macOS AMI。在清理工作流程期間，專用執行個體會暫時進入 pending (待定) 狀態。在 x86 Mac 執行個體上，清理工作流程最多可能需要 50 分鐘才能完成。在 Apple 晶片 Mac 執行個體上，清理工作流程最多可能需要 110 分鐘才能完成。此外，在 x86 Mac 執行個體上，如果裝置韌體需要更新，清理工作流程最多可能需要 3 個小時才能完成。

直到清理工作流程完成，也即 專用執行個體 進入 available 狀態，您才能啟動已停止的 Mac 執行個體或啟動新的 Mac 執行個體。

專用執行個體進入 pending 狀態時，計量和計費會暫停。清理工作流程期間，您不需支付費用。

## 為您的 Mac 執行個體釋出 專用主機

當您完成 Mac 執行個體時，您可以釋出 專用執行個體。您必須先停止或終止 Mac 執行個體，才能發佈專用執行個體。至少在配置期間經過 24 小時後才能發佈主機。

### 發佈專用執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體並選取 Instance state (執行個體狀態)，然後選取 Stop instance (停止執行個體) 或 Terminate instance (終止執行個體)。
4. 在導覽窗格中，選擇 專用執行個體。
5. 選取 專用執行個體 並選取 Actions (動作)，Release host (釋出主機)。
6. 出現確認提示時，請選擇 Release (釋出)。



## 為您的專用主機尋找支援的 macOS 版本

您可以檢視 Amazon EC2 專用主機支援的最新 macOS 版本。透過此功能，您可以驗證專用主機是否支援使用偏好 macOS 版本的執行個體啟動。

每個 macOS 版本都需要基礎蘋果 Mac 上的最低固件版本才能成功啟動。如果分配的 Mac 專用主機長時間保持閒置狀態，或者其上有長時間運行的實例，則 Apple Mac 固件版本可能會過時。

為確保最新 macOS 版本的支援能力，您可以停止或終止已配置的 Mac 專用主機上的執行個體。這會觸發主機清理工作流程，並更新基礎 Apple Mac 上的韌體，以支援最新的 macOS 版本。當您停止或終止執行中的執行個體時，具有長時間執行執行個體的專用主機會自動更新。

如需拖曳工作流程的詳細資訊，請參閱[停止並終止您的 Mac 執行個體](#)。

如需啟動 Mac 執行個體的詳細資訊，請參閱[啟動 Mac 執行個體](#)。

您可以使用 Amazon EC2 主控台或檢視已配置的專用主機上支援的最新 macOS 版本的相關資訊 AWS CLI。

### Console

使用主控台檢視專用主機韌體資訊

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 專用執行個體。
3. 在專用主機詳細資料頁面的「最新受支援的 macOS 版本」下，您可以查看主機可支援的最新 macOS 版本。

### AWS CLI

使用檢視專用主機韌體資訊 AWS CLI

使用該[describe-mac-hosts](#)命令，region 用適當的替換 AWS 區域。

```
$ aws ec2 describe-mac-hosts --region us-east-1
{
 "MacHosts": [
 {
 "HostId": "h-07879acf49EXAMPLE",
 "MacOSLatestSupportedVersions": [
```

```
 "14.3",
 "13.6.4",
 "12.7.3"
]
 }
]
}
```

## 訂閱 macOS AMI 通知

若要在發行新的 AMI 或 BridgeOS 更新後收到通知，請使用 Amazon SNS 訂閱通知。

如需有關 EC2 macOS AMI 的詳細資訊，請參閱[Amazon EC2](#)。

### 訂閱 macOS AMI 通知

1. 在 <https://console.aws.amazon.com/sns/v3/home> 開啟 Amazon SNS 主控台。
2. 如有必要，請在導覽列中將「區域」變更為美國東部 (維吉尼亞北部)。您必須使用此區域，因為您要訂閱的 SNS 通知已在本區域中建立完成。
3. 在導覽窗格中，選擇 Subscriptions (訂閱)。
4. 選擇 Create subscription (建立訂閱)。
5. 針對 Create subscription (建立訂閱) 對話方塊，執行下列作業：
  - a. 針對 Topic ARN (主題 ARN)，複製並貼上下列 Amazon Resource Name (ARN) 之一：
    - **arn:aws:sns:us-east-1:898855652048:amazon-ec2-macos-ami-updates**
    - **arn:aws:sns:us-east-1:898855652048:amazon-ec2-bridgeos-updates**
  - b. 在「通訊協定」中，選擇下列其中一項：
    - 電子郵件：

針對 Endpoint (端點)，輸入可用於接收通知的電子郵件地址。建立訂閱後，您將收到主旨行為 AWS Notification - Subscription Confirmation 的確認訊息。開啟電子郵件並選擇 Confirm subscription (確認訂閱) 完成訂閱。
    - 簡訊：

對於 Endpoint (端點)，輸入可用於接收通知的電子郵件地址。
    - AWS Lambda，Amazon SQS，Amazon 數據 Firehose 件 (通知以 JSON 格式提供)：

在 Endpoint (端點) 中，請輸入 Lambda 函數、SQS 佇列或 Firehose 串流的 ARN，以供您接收通知。

- c. 選擇 Create subscription (建立訂閱)。

只要有 macOS AMI 發行，我們就會向 amazon-ec2-macos-ami-updates 主題的訂閱者傳送通知。只要 bridgeOS 有所變更，我們就會向 amazon-ec2-bridgeos-updates 主題的訂閱者傳送通知。如果您不想再接收這些通知，請使用下列程序來取消訂閱。

#### 取消訂閱 macOS AMI 通知

1. 在 <https://console.aws.amazon.com/sns/v3/home> 開啟 Amazon SNS 主控台。
2. 如有必要，請在導覽列中將「區域」變更為美國東部 (維吉尼亞北部)。您必須使用此區域，因為 SNS 通知已在本區域中建立完成。
3. 在導覽窗格中，選擇 Subscriptions (訂閱)。
4. 選取訂閱，接著選取 Actions (動作)、Delete subscriptions (刪除訂閱)，提示確認時請選取 Delete (刪除)。

## Amazon EC2

以下資訊提供 macOS AMI 中預設包含的套件詳細資訊，並摘要說明每個 EC2 macOS AMI 版本的變更。

如需有關如何訂閱 macOS AMI 通知的詳細資訊，請參閱 [訂閱 macOS AMI 通知](#)。

### Amazon EC2 中包含的默認軟件包

下表說明預設包含在 EC2 macOS AMI 中的套件。

| 套件             | 版本備註                                                                                                            |
|----------------|-----------------------------------------------------------------------------------------------------------------|
| EC2初 macOS 化   | <a href="https://github.com/aws/ec2-macos-init/tags">https://github.com/aws/ec2-macos-init/tags</a>             |
| EC2macOS 實用程序  | <a href="https://github.com/aws/ec2-macos-utils/tags">https://github.com/aws/ec2-macos-utils/tags</a>           |
| AmazonSSM 代理程式 | <a href="https://github.com/aws/amazon-ssm-agent/releases">https://github.com/aws/amazon-ssm-agent/releases</a> |

| 套件                                       | 版本備註                                                                                                                                          |
|------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| AWS Command Line Interface (AWS CLI) 第二版 | <a href="https://raw.githubusercontent.com/aws/aws-cli/v2/CHANGELOG.rst">https://raw.githubusercontent.com/aws/aws-cli/v2/CHANGELOG.rst</a>   |
| Xcode 的命令列工具                             | <a href="https://developer.apple.com/documentation/xcode-release-notes">https://developer.apple.com/documentation/xcode-release-notes</a>     |
| Homebrew                                 | <a href="https://github.com/Homebrew/brew/releases">https://github.com/Homebrew/brew/releases</a>                                             |
| EC2 執行個體連線                               | <a href="https://github.com/aws/aws-ec2-instance-connect-config/releases">https://github.com/aws/aws-ec2-instance-connect-config/releases</a> |
| Safari                                   | <a href="https://developer.apple.com/documentation/safari-release-notes">https://developer.apple.com/documentation/safari-release-notes</a>   |

## Amazon EC2 的 macOS AMI 更新

下表說明 EC2 macOS AMI 發行版本中包含的變更。請注意，某些變更會套用至所有 EC2 macOS AMI，而其他變更僅適用於這些 AMI 的子集。

### 卡斯 macOS AMI 更新

| 發行版本       | 改變                                                                                                                                                                                                                                                                                                                                                                                                                               |
|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2024.06.07 | <p>所有 AMI</p> <ul style="list-style-type: none"> <li>自製軟體更新至 4.3.1-1</li> <li>已更新aws-cli至 2 月 15 日</li> <li>已更新amazon-ssm-agent 至 3 月 3 日</li> </ul> <p>蘋果系 macOS 索諾瑪 14.5 的發布 ( 所有 Mac 實例 )</p> <ul style="list-style-type: none"> <li><a href="#">macOS 索諾瑪 14.5 的安全性內容</a></li> </ul> <p>蘋果系 macOS 文圖拉 13.6.7 的發布 ( 所有 Mac 實例 )</p> <ul style="list-style-type: none"> <li><a href="#">macOS 文圖拉 13.6.7 的安全性內容</a></li> </ul> |

| 發行版本       | 改變                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|            | <ul style="list-style-type: none"><li>更新野生動物園到 17.5<ul style="list-style-type: none"><li><a href="#">野生動物園 17.5 的安全內容</a></li></ul></li></ul> <p>蘋果系 macOS 蒙特雷 12.7.5 的發布 ( 所有 Mac 實例 )</p> <ul style="list-style-type: none"><li><a href="#">macOS 蒙特雷 12.7.5 的安全性內容</a></li><li>更新野生動物園到 17.5<ul style="list-style-type: none"><li><a href="#">野生動物園 17.5 的安全內容</a></li></ul></li></ul>                                                                                                                                                                                                                                                                                              |
| 2024.04.12 | <p>所有 AMI</p> <ul style="list-style-type: none"><li>已更新自製軟體至 4 月 16 日</li><li>已更新aws-cli至 2 月 15 日</li></ul> <p>蘋果系 macOS 索諾瑪 14.4.1 的發布 ( 所有 Mac 實例 )</p> <ul style="list-style-type: none"><li><a href="#">macOS 索諾瑪 14.4.1 的安全性內容</a></li></ul> <p>蘋果系 macOS 文圖拉 13.6.6 的發布 ( 所有 Mac 實例 )</p> <ul style="list-style-type: none"><li><a href="#">macOS 文圖拉 13.6.6 的安全性內容</a></li><li>更新野生動物園 17.4.1<ul style="list-style-type: none"><li><a href="#">野生動物園 17.4.1 的安全性內容</a></li></ul></li></ul> <p>對於 macOS 蒙特雷 ( 所有 Mac 實例 )</p> <ul style="list-style-type: none"><li>更新野生動物園 17.4.1<ul style="list-style-type: none"><li><a href="#">野生動物園 17.4.1 的安全性內容</a></li></ul></li></ul> |

# Amazon EBS – 最佳化執行個體

Amazon EBS 最佳化執行個體使用最佳化組態堆疊，並為 Amazon EBS I/O 提供額外專用容量。此最佳化透過減少 Amazon EBS I/O 與執行個體的其他流量之間的爭用情況，為您的 EBS 磁碟區提供最佳效能。

EBS 最佳化執行個體可為 Amazon EBS 提供專用頻寬。當連接至 EBS 最佳化執行個體時，一般用途 SSD (gp2 和 gp3) 磁碟區設計為能在給定年份 99% 的時間裡提供至少 90% 的佈建 IOPS 效能，佈建 IOPS SSD (io1 和 io2) 磁碟區則設計為能在給定年份 99.9% 的時間裡提供至少 90% 的佈建 IOPS 效能。輸送量最佳化 HDD (st1) 和冷 HDD (sc1) 在給定年份內完成 99% 時間的預期輸送量至少 90% 的效能。不相容的期間約為統一分佈，目標為每小時 99% 的預期總輸送量。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的 Amazon EBS 磁碟區類型](#)。

## Important

執行個體的 EBS 效能受限於執行個體效能上限，或其連接磁碟區的彙總效能 (以較小者為準)。若要達到最大 EBS 效能，執行個體必須具有連接磁碟區，這些磁碟區提供的組合效能必須等於或大於最大執行個體效能。例如，r6i.16xlarge 若要達到 80,000 IOPS，每個執行個體至少須佈建 5 gp3 個 16,000 IOPS 的磁碟區 (5 個磁碟區 x 16,000 IOPS = 80,000 IOPS)。

## 目錄

- [支援的執行個體類型](#)
- [獲得最大效能](#)
- [檢視支援 EBS 最佳化的執行個體類型](#)
- [在啟動時啟用 EBS 最佳化](#)
- [啟用現有執行個體的 EBS 最佳化](#)

## 支援的執行個體類型

下表顯示支援 EBS 最佳化的執行個體類型。包括 Amazon EBS 專屬頻寬、含串流讀取工作負載和 128 KiB I/O 大小的連線可達到的典型最大彙整傳輸量，以及使用 16 KiB I/O 大小時執行個體可支援的 IOPS 最大數量。

請選擇 EBS 最佳化執行個體，它提供的專屬 Amazon EBS 輸送量超過應用程式需求；否則，Amazon EBS 和 Amazon EC2 之間的連線將成為效能瓶頸。

## 目錄

- [預設為 EBS 最佳化](#)
- [支援 EBS 最佳化](#)

## 預設為 EBS 最佳化

下表列出支援 EBS 最佳化的執行個體類型，且預設為啟用 EBS 最佳化。不需要啟用 EBS 最佳化，且停用 EBS 最佳化不具任何效果。

### Note

您也可以使用程式設計方式檢視此資訊。AWS CLI 如需詳細資訊，請參閱 [檢視支援 EBS 最佳化的執行個體類型](#)。

## 主題

- [一般用途](#)
- [運算最佳化](#)
- [記憶體最佳化](#)
- [儲存最佳化](#)
- [加速運算](#)
- [高效能運算](#)

## 一般用途

### Important

- <sup>1</sup> 這些執行個體支援至少每 24 小時一次維持最佳效能 30 分鐘，然後就會回復至其基準效能。
- <sup>2</sup> 這些執行個體可以無限期維持其上述效能。如果您的工作負載需要維持超過 30 分鐘的最佳效能，請選取其中一個執行個體。

| 執行個體大小                   | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|--------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| a1.medium <sup>1</sup>   | 300         | 3500        | 37.50                    | 437.50                    | 2500                 | 20000                |
| a1.large <sup>1</sup>    | 525         | 3500        | 65.62                    | 437.50                    | 4000                 | 20000                |
| a1.xlarge <sup>1</sup>   | 800         | 3500        | 100.00                   | 437.50                    | 6000                 | 20000                |
| a1.2xlarge <sup>1</sup>  | 1750        | 3500        | 218.75                   | 437.50                    | 10000                | 20000                |
| a1.4xlarge <sup>2</sup>  |             | 3500        |                          | 437.5                     |                      | 20000                |
| a1.metal <sup>2</sup>    |             | 3500        |                          | 437.5                     |                      | 20000                |
| m4.large <sup>2</sup>    |             | 450         |                          | 56.25                     |                      | 3600                 |
| m4.xlarge <sup>2</sup>   |             | 750         |                          | 93.75                     |                      | 6000                 |
| m4.2xlarge <sup>2</sup>  |             | 1000        |                          | 125.0                     |                      | 8000                 |
| m4.4xlarge <sup>2</sup>  |             | 2000        |                          | 250.0                     |                      | 16000                |
| m4.10xlarge <sup>2</sup> |             | 4000        |                          | 500.0                     |                      | 32000                |
| m4.16xlarge <sup>2</sup> |             | 10000       |                          | 1250.0                    |                      | 65000                |
| m5.large <sup>1</sup>    | 650         | 4750        | 81.25                    | 593.75                    | 3600                 | 18750                |
| m5.xlarge <sup>1</sup>   | 1150        | 4750        | 143.75                   | 593.75                    | 6000                 | 18750                |



| 執行個體大小                   | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|--------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| m5.2xlarge <sub>1</sub>  | 2300        | 4750        | 287.50                   | 593.75                    | 12000                | 18750                |
| m5.4xlarge <sub>2</sub>  |             | 4750        |                          | 593.75                    |                      | 18750                |
| m5.8xlarge <sub>2</sub>  |             | 6800        |                          | 850.0                     |                      | 30000                |
| m5.12xlarge <sup>2</sup> |             | 9500        |                          | 1187.5                    |                      | 40000                |
| m5.16xlarge <sup>2</sup> |             | 13600       |                          | 1700.0                    |                      | 60000                |
| m5.24xlarge <sup>2</sup> |             | 19000       |                          | 2375.0                    |                      | 80000                |
| m5.metal <sup>2</sup>    |             | 19000       |                          | 2375.0                    |                      | 80000                |
| m5a.large <sub>1</sub>   | 650         | 2880        | 81.25                    | 360.00                    | 3600                 | 16000                |
| m5a.xlarge <sub>1</sub>  | 1085        | 2880        | 135.62                   | 360.00                    | 6000                 | 16000                |
| m5a.2xlarge <sub>1</sub> | 1580        | 2880        | 197.50                   | 360.00                    | 8333                 | 16000                |
| m5a.4xlarge <sup>2</sup> |             | 2880        |                          | 360.0                     |                      | 16000                |
| m5a.8xlarge <sup>2</sup> |             | 4750        |                          | 593.75                    |                      | 20000                |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| m5a.12xlarge <sup>2</sup>  |             | 6780        |                          | 847.5                     |                      | 30000                |
| m5a.16xlarge <sup>2</sup>  |             | 9500        |                          | 1187.5                    |                      | 40000                |
| m5a.24xlarge <sup>2</sup>  |             | 13750       |                          | 1718.75                   |                      | 60000                |
| m5ad.large <sup>1</sup>    | 650         | 2880        | 81.25                    | 360.00                    | 3600                 | 16000                |
| m5ad.xlarge <sup>1</sup>   | 1085        | 2880        | 135.62                   | 360.00                    | 6000                 | 16000                |
| m5ad.2xlarge <sup>1</sup>  | 1580        | 2880        | 197.50                   | 360.00                    | 8333                 | 16000                |
| m5ad.4xlarge <sup>2</sup>  |             | 2880        |                          | 360.0                     |                      | 16000                |
| m5ad.8xlarge <sup>2</sup>  |             | 4750        |                          | 593.75                    |                      | 20000                |
| m5ad.12xlarge <sup>2</sup> |             | 6780        |                          | 847.5                     |                      | 30000                |
| m5ad.16xlarge <sup>2</sup> |             | 9500        |                          | 1187.5                    |                      | 40000                |
| m5ad.24xlarge <sup>2</sup> |             | 13750       |                          | 1718.75                   |                      | 60000                |

| 執行個體大小                    | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|---------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| m5d.large <sup>1</sup>    | 650         | 4750        | 81.25                    | 593.75                    | 3600                 | 18750                |
| m5d.xlarge <sup>1</sup>   | 1150        | 4750        | 143.75                   | 593.75                    | 6000                 | 18750                |
| m5d.2xlarge <sup>1</sup>  | 2300        | 4750        | 287.50                   | 593.75                    | 12000                | 18750                |
| m5d.4xlarge <sup>2</sup>  |             | 4750        |                          | 593.75                    |                      | 18750                |
| m5d.8xlarge <sup>2</sup>  |             | 6800        |                          | 850.0                     |                      | 30000                |
| m5d.12xlarge <sup>2</sup> |             | 9500        |                          | 1187.5                    |                      | 40000                |
| m5d.16xlarge <sup>2</sup> |             | 13600       |                          | 1700.0                    |                      | 60000                |
| m5d.24xlarge <sup>2</sup> |             | 19000       |                          | 2375.0                    |                      | 80000                |
| m5d.metal <sup>2</sup>    |             | 19000       |                          | 2375.0                    |                      | 80000                |
| m5dn.large <sup>1</sup>   | 650         | 4750        | 81.25                    | 593.75                    | 3600                 | 18750                |
| m5dn.xlarge <sup>1</sup>  | 1150        | 4750        | 143.75                   | 593.75                    | 6000                 | 18750                |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| m5dn.2xlarge <sup>1</sup>  | 2300        | 4750        | 287.50                   | 593.75                    | 12000                | 18750                |
| m5dn.4xlarge <sup>2</sup>  |             | 4750        |                          | 593.75                    |                      | 18750                |
| m5dn.8xlarge <sup>2</sup>  |             | 6800        |                          | 850.0                     |                      | 30000                |
| m5dn.12xlarge <sup>2</sup> |             | 9500        |                          | 1187.5                    |                      | 40000                |
| m5dn.16xlarge <sup>2</sup> |             | 13600       |                          | 1700.0                    |                      | 60000                |
| m5dn.24xlarge <sup>2</sup> |             | 19000       |                          | 2375.0                    |                      | 80000                |
| m5dn.meta1 <sup>2</sup>    |             | 19000       |                          | 2375.0                    |                      | 80000                |
| m5n.large <sup>1</sup>     | 650         | 4750        | 81.25                    | 593.75                    | 3600                 | 18750                |
| m5n.xlarge <sup>1</sup>    | 1150        | 4750        | 143.75                   | 593.75                    | 6000                 | 18750                |
| m5n.2xlarge <sup>1</sup>   | 2300        | 4750        | 287.50                   | 593.75                    | 12000                | 18750                |
| m5n.4xlarge <sup>2</sup>   |             | 4750        |                          | 593.75                    |                      | 18750                |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| m5n.8xlarge <sup>2</sup>   | 6800        |             | 850.0                    |                           | 30000                |                      |
| m5n.12xlarge <sup>2</sup>  | 9500        |             | 1187.5                   |                           | 40000                |                      |
| m5n.16xlarge <sup>2</sup>  | 13600       |             | 1700.0                   |                           | 60000                |                      |
| m5n.24xlarge <sup>2</sup>  | 19000       |             | 2375.0                   |                           | 80000                |                      |
| m5n.metal <sup>2</sup>     | 19000       |             | 2375.0                   |                           | 80000                |                      |
| m5zn.large <sup>1</sup>    | 800         | 3170        | 100.00                   | 396.25                    | 3333                 | 13333                |
| m5zn.xlarge <sup>1</sup>   | 1564        | 3170        | 195.50                   | 396.25                    | 6667                 | 13333                |
| m5zn.2xlarge <sup>2</sup>  | 3170        |             | 396.25                   |                           | 13333                |                      |
| m5zn.3xlarge <sup>2</sup>  | 4750        |             | 593.75                   |                           | 20000                |                      |
| m5zn.6xlarge <sup>2</sup>  | 9500        |             | 1187.5                   |                           | 40000                |                      |
| m5zn.12xlarge <sup>2</sup> | 19000       |             | 2375.0                   |                           | 80000                |                      |

| 執行個體大小                    | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|---------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| m5zn.meta1 <sup>2</sup>   |             | 19000       |                          | 2375.0                    |                      | 80000                |
| m6a.large <sup>1</sup>    | 650         | 10000       | 81.25                    | 1250.00                   | 3600                 | 40000                |
| m6a.xlarge <sup>1</sup>   | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |
| m6a.2xlarge <sup>1</sup>  | 2500        | 10000       | 312.50                   | 1250.00                   | 12000                | 40000                |
| m6a.4xlarge <sup>1</sup>  | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |
| m6a.8xlarge <sup>2</sup>  |             | 10000       |                          | 1250.0                    |                      | 40000                |
| m6a.12xlarge <sup>2</sup> |             | 15000       |                          | 1875.0                    |                      | 60000                |
| m6a.16xlarge <sup>2</sup> |             | 20000       |                          | 2500.0                    |                      | 80000                |
| m6a.24xlarge <sup>2</sup> |             | 30000       |                          | 3750.0                    |                      | 120000               |
| m6a.32xlarge <sup>2</sup> |             | 40000       |                          | 5000.0                    |                      | 160000               |
| m6a.48xlarge <sup>2</sup> |             | 40000       |                          | 5000.0                    |                      | 240000               |

| 執行個體大小                    | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|---------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| m6a.metal <sub>2</sub>    | 40000       |             | 5000.0                   |                           | 240000               |                      |
| m6g.medium <sup>1</sup>   | 315         | 4750        | 39.38                    | 593.75                    | 2500                 | 20000                |
| m6g.large <sub>1</sub>    | 630         | 4750        | 78.75                    | 593.75                    | 3600                 | 20000                |
| m6g.xlarge <sub>1</sub>   | 1188        | 4750        | 148.50                   | 593.75                    | 6000                 | 20000                |
| m6g.2xlarge <sup>1</sup>  | 2375        | 4750        | 296.88                   | 593.75                    | 12000                | 20000                |
| m6g.4xlarge <sup>2</sup>  | 4750        |             | 593.75                   |                           | 20000                |                      |
| m6g.8xlarge <sup>2</sup>  | 9500        |             | 1187.5                   |                           | 40000                |                      |
| m6g.12xlarge <sup>2</sup> | 14250       |             | 1781.25                  |                           | 50000                |                      |
| m6g.16xlarge <sup>2</sup> | 19000       |             | 2375.0                   |                           | 80000                |                      |
| m6g.metal <sub>2</sub>    | 19000       |             | 2375.0                   |                           | 80000                |                      |
| m6gd.medium <sup>1</sup>  | 315         | 4750        | 39.38                    | 593.75                    | 2500                 | 20000                |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| m6gd.large <sup>1</sup>    | 630         | 4750        | 78.75                    | 593.75                    | 3600                 | 20000                |
| m6gd.xlarge <sup>1</sup>   | 1188        | 4750        | 148.50                   | 593.75                    | 6000                 | 20000                |
| m6gd.2xlarge <sup>1</sup>  | 2375        | 4750        | 296.88                   | 593.75                    | 12000                | 20000                |
| m6gd.4xlarge <sup>2</sup>  |             | 4750        |                          | 593.75                    |                      | 20000                |
| m6gd.8xlarge <sup>2</sup>  |             | 9500        |                          | 1187.5                    |                      | 40000                |
| m6gd.12xlarge <sup>2</sup> |             | 14250       |                          | 1781.25                   |                      | 50000                |
| m6gd.16xlarge <sup>2</sup> |             | 19000       |                          | 2375.0                    |                      | 80000                |
| m6gd.meta1 <sup>2</sup>    |             | 19000       |                          | 2375.0                    |                      | 80000                |
| m6i.large <sup>1</sup>     | 650         | 10000       | 81.25                    | 1250.00                   | 3600                 | 40000                |
| m6i.xlarge <sup>1</sup>    | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |
| m6i.2xlarge <sup>1</sup>   | 2500        | 10000       | 312.50                   | 1250.00                   | 12000                | 40000                |
| m6i.4xlarge <sup>1</sup>   | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |



| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| m6i.8xlarge <sup>2</sup>   |             | 10000       |                          | 1250.0                    |                      | 40000                |
| m6i.12xlarge <sup>2</sup>  |             | 15000       |                          | 1875.0                    |                      | 60000                |
| m6i.16xlarge <sup>2</sup>  |             | 20000       |                          | 2500.0                    |                      | 80000                |
| m6i.24xlarge <sup>2</sup>  |             | 30000       |                          | 3750.0                    |                      | 120000               |
| m6i.32xlarge <sup>2</sup>  |             | 40000       |                          | 5000.0                    |                      | 160000               |
| m6i.metal <sup>2</sup>     |             | 40000       |                          | 5000.0                    |                      | 160000               |
| m6id.large <sup>1</sup>    | 650         | 10000       | 81.25                    | 1250.00                   | 3600                 | 40000                |
| m6id.xlarge <sup>1</sup>   | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |
| m6id.2xlarge <sup>1</sup>  | 2500        | 10000       | 312.50                   | 1250.00                   | 12000                | 40000                |
| m6id.4xlarge <sup>1</sup>  | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |
| m6id.8xlarge <sup>2</sup>  |             | 10000       |                          | 1250.0                    |                      | 40000                |
| m6id.12xlarge <sup>2</sup> |             | 15000       |                          | 1875.0                    |                      | 60000                |

| 執行個體大小                      | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|-----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| m6id.16xlarge <sup>2</sup>  |             | 20000       |                          | 2500.0                    |                      | 80000                |
| m6id.24xlarge <sup>2</sup>  |             | 30000       |                          | 3750.0                    |                      | 120000               |
| m6id.32xlarge <sup>2</sup>  |             | 40000       |                          | 5000.0                    |                      | 160000               |
| m6id.meta1 <sup>2</sup>     |             | 40000       |                          | 5000.0                    |                      | 160000               |
| m6idn.large <sup>1</sup>    | 1562        | 25000       | 195.31                   | 3125.00                   | 6250                 | 100000               |
| m6idn.xlarge <sup>1</sup>   | 3125        | 25000       | 390.62                   | 3125.00                   | 12500                | 100000               |
| m6idn.2xlarge <sup>1</sup>  | 6250        | 25000       | 781.25                   | 3125.00                   | 25000                | 100000               |
| m6idn.4xlarge <sup>1</sup>  | 12500       | 25000       | 1562.50                  | 3125.00                   | 50000                | 100000               |
| m6idn.8xlarge <sup>2</sup>  |             | 25000       |                          | 3125.0                    |                      | 100000               |
| m6idn.12xlarge <sup>2</sup> |             | 37500       |                          | 4687.5                    |                      | 150000               |
| m6idn.16xlarge <sup>2</sup> |             | 50000       |                          | 6250.0                    |                      | 200000               |

| 執行個體大小                      | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|-----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| m6idn.24xlarge <sup>2</sup> | 75000       |             | 9375.0                   |                           | 300000               |                      |
| m6idn.32xlarge <sup>2</sup> | 100000      |             | 12500.0                  |                           | 400000               |                      |
| m6idn.metal <sup>2</sup>    | 100000      |             | 12500.0                  |                           | 400000               |                      |
| m6in.large <sup>1</sup>     | 1562        | 25000       | 195.31                   | 3125.00                   | 6250                 | 100000               |
| m6in.xlarge <sup>1</sup>    | 3125        | 25000       | 390.62                   | 3125.00                   | 12500                | 100000               |
| m6in.2xlarge <sup>1</sup>   | 6250        | 25000       | 781.25                   | 3125.00                   | 25000                | 100000               |
| m6in.4xlarge <sup>1</sup>   | 12500       | 25000       | 1562.50                  | 3125.00                   | 50000                | 100000               |
| m6in.8xlarge <sup>2</sup>   | 25000       |             | 3125.0                   |                           | 100000               |                      |
| m6in.12xlarge <sup>2</sup>  | 37500       |             | 4687.5                   |                           | 150000               |                      |
| m6in.16xlarge <sup>2</sup>  | 50000       |             | 6250.0                   |                           | 200000               |                      |
| m6in.24xlarge <sup>2</sup>  | 75000       |             | 9375.0                   |                           | 300000               |                      |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| m6in.32xlarge <sup>2</sup> |             | 100000      |                          | 12500.0                   |                      | 400000               |
| m6in.meta1 <sup>2</sup>    |             | 100000      |                          | 12500.0                   |                      | 400000               |
| m7a.medium <sup>1</sup>    | 325         | 10000       | 40.62                    | 1250.00                   | 2500                 | 40000                |
| m7a.large <sup>1</sup>     | 650         | 10000       | 81.25                    | 1250.00                   | 3600                 | 40000                |
| m7a.xlarge <sup>1</sup>    | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |
| m7a.2xlarge <sup>1</sup>   | 2500        | 10000       | 312.50                   | 1250.00                   | 12000                | 40000                |
| m7a.4xlarge <sup>1</sup>   | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |
| m7a.8xlarge <sup>2</sup>   |             | 10000       |                          | 1250.0                    |                      | 40000                |
| m7a.12xlarge <sup>2</sup>  |             | 15000       |                          | 1875.0                    |                      | 60000                |
| m7a.16xlarge <sup>2</sup>  |             | 20000       |                          | 2500.0                    |                      | 80000                |
| m7a.24xlarge <sup>2</sup>  |             | 30000       |                          | 3750.0                    |                      | 120000               |

| 執行個體大小                      | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|-----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| m7a.32xlarge <sup>2</sup>   | 40000       |             |                          | 5000.0                    |                      | 160000               |
| m7a.48xlarge <sup>2</sup>   | 40000       |             |                          | 5000.0                    |                      | 240000               |
| m7a.metal-48xl <sup>2</sup> | 40000       |             |                          | 5000.0                    |                      | 240000               |
| m7g.medium <sup>1</sup>     | 315         | 10000       | 39.38                    | 1250.00                   | 2500                 | 40000                |
| m7g.large <sup>1</sup>      | 630         | 10000       | 78.75                    | 1250.00                   | 3600                 | 40000                |
| m7g.xlarge <sup>1</sup>     | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |
| m7g.2xlarge <sup>1</sup>    | 2500        | 10000       | 312.50                   | 1250.00                   | 12000                | 40000                |
| m7g.4xlarge <sup>1</sup>    | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |
| m7g.8xlarge <sup>2</sup>    | 10000       |             |                          | 1250.0                    |                      | 40000                |
| m7g.12xlarge <sup>2</sup>   | 15000       |             |                          | 1875.0                    |                      | 60000                |
| m7g.16xlarge <sup>2</sup>   | 20000       |             |                          | 2500.0                    |                      | 80000                |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| m7g.metal <sup>2</sup>     |             | 20000       |                          | 2500.0                    |                      | 80000                |
| m7gd.medium <sup>1</sup>   | 315         | 10000       | 39.38                    | 1250.00                   | 2500                 | 40000                |
| m7gd.large <sup>1</sup>    | 630         | 10000       | 78.75                    | 1250.00                   | 3600                 | 40000                |
| m7gd.xlarge <sup>1</sup>   | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |
| m7gd.2xlarge <sup>1</sup>  | 2500        | 10000       | 312.50                   | 1250.00                   | 12000                | 40000                |
| m7gd.4xlarge <sup>1</sup>  | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |
| m7gd.8xlarge <sup>2</sup>  |             | 10000       |                          | 1250.0                    |                      | 40000                |
| m7gd.12xlarge <sup>2</sup> |             | 15000       |                          | 1875.0                    |                      | 60000                |
| m7gd.16xlarge <sup>2</sup> |             | 20000       |                          | 2500.0                    |                      | 80000                |
| M7 克. 金屬                   |             | 20000       |                          | 2500.0                    |                      | 80000                |
| m7i.large <sup>1</sup>     | 650         | 10000       | 81.25                    | 1250.00                   | 3600                 | 40000                |
| m7i.xlarge <sup>1</sup>    | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |

| 執行個體大小                       | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|------------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| m7i.2xlarge <sup>1</sup>     | 2500        | 10000       | 312.50                   | 1250.00                   | 12000                | 40000                |
| m7i.4xlarge <sup>1</sup>     | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |
| m7i.8xlarge <sup>2</sup>     |             | 10000       |                          | 1250.0                    |                      | 40000                |
| m7i.12xlarge <sup>2</sup>    |             | 15000       |                          | 1875.0                    |                      | 60000                |
| m7i.16xlarge <sup>2</sup>    |             | 20000       |                          | 2500.0                    |                      | 80000                |
| m7i.24xlarge <sup>2</sup>    |             | 30000       |                          | 3750.0                    |                      | 120000               |
| m7i.48xlarge <sup>2</sup>    |             | 40000       |                          | 5000.0                    |                      | 240000               |
| m7i.metal-24xl <sup>2</sup>  |             | 30000       |                          | 3750.0                    |                      | 120000               |
| m7i.metal-48xl <sup>2</sup>  |             | 40000       |                          | 5000.0                    |                      | 240000               |
| m7i-flex.large <sup>1</sup>  | 312         | 10000       | 39.06                    | 1250.00                   | 2500                 | 40000                |
| m7i-flex.xlarge <sup>1</sup> | 625         | 10000       | 78.12                    | 1250.00                   | 3600                 | 40000                |

| 執行個體大小                        | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|-------------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| m7i-flex.2xlarge <sup>1</sup> | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |
| m7i-flex.4xlarge <sup>1</sup> | 2500        | 10000       | 312.50                   | 1250.00                   | 12000                | 40000                |
| m7i-flex.8xlarge <sup>1</sup> | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |
| mac1.meta1 <sup>2</sup>       |             | 14000       |                          | 1750.0                    |                      | 80000                |
| mac2.meta1 <sup>2</sup>       |             | 10000       |                          | 1250.0                    |                      | 55000                |
| mac2-m2.metal <sup>2</sup>    |             | 8000        |                          | 1000.0                    |                      | 55000                |
| mac2-m2pro.metal <sup>2</sup> |             | 8000        |                          | 1000.0                    |                      | 55000                |
| t3.nano <sup>1</sup>          | 43          | 2085        | 5.38                     | 260.62                    | 250                  | 11800                |
| t3.micro <sup>1</sup>         | 87          | 2085        | 10.88                    | 260.62                    | 500                  | 11800                |
| t3.small <sup>1</sup>         | 174         | 2085        | 21.75                    | 260.62                    | 1000                 | 11800                |
| t3.medium <sup>1</sup>        | 347         | 2085        | 43.38                    | 260.62                    | 2000                 | 11800                |
| t3.large <sup>1</sup>         | 695         | 2780        | 86.88                    | 347.50                    | 4000                 | 15700                |
| t3.xlarge <sup>1</sup>        | 695         | 2780        | 86.88                    | 347.50                    | 4000                 | 15700                |
| t3.2xlarge <sup>1</sup>       | 695         | 2780        | 86.88                    | 347.50                    | 4000                 | 15700                |



| 執行個體大小                   | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|--------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| t3a.nano <sup>1</sup>    | 45          | 2085        | 5.62                     | 260.62                    | 250                  | 11800                |
| t3a.micro <sup>1</sup>   | 90          | 2085        | 11.25                    | 260.62                    | 500                  | 11800                |
| t3a.small <sup>1</sup>   | 175         | 2085        | 21.88                    | 260.62                    | 1000                 | 11800                |
| t3a.medium <sup>1</sup>  | 350         | 2085        | 43.75                    | 260.62                    | 2000                 | 11800                |
| t3a.large <sup>1</sup>   | 695         | 2780        | 86.88                    | 347.50                    | 4000                 | 15700                |
| t3a.xlarge <sup>1</sup>  | 695         | 2780        | 86.88                    | 347.50                    | 4000                 | 15700                |
| t3a.2xlarge <sup>1</sup> | 695         | 2780        | 86.88                    | 347.50                    | 4000                 | 15700                |
| t4g.nano <sup>1</sup>    | 43          | 2085        | 5.38                     | 260.62                    | 250                  | 11800                |
| t4g.micro <sup>1</sup>   | 87          | 2085        | 10.88                    | 260.62                    | 500                  | 11800                |
| t4g.small <sup>1</sup>   | 174         | 2085        | 21.75                    | 260.62                    | 1000                 | 11800                |
| t4g.medium <sup>1</sup>  | 347         | 2085        | 43.38                    | 260.62                    | 2000                 | 11800                |
| t4g.large <sup>1</sup>   | 695         | 2780        | 86.88                    | 347.50                    | 4000                 | 15700                |
| t4g.xlarge <sup>1</sup>  | 695         | 2780        | 86.88                    | 347.50                    | 4000                 | 15700                |
| t4g.2xlarge <sup>1</sup> | 695         | 2780        | 86.88                    | 347.50                    | 4000                 | 15700                |

## 運算最佳化

**⚠ Important**

- <sup>1</sup> 這些執行個體支援至少每 24 小時一次維持最佳效能 30 分鐘，然後就會回復至其基準效能。
- <sup>2</sup> 這些執行個體可以無限期維持其上述效能。如果您的工作負載需要維持超過 30 分鐘的最佳效能，請選取其中一個執行個體。

| 執行個體大小                  | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|-------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| c4.large <sup>2</sup>   |             | 500         |                          | 62.5                      |                      | 4000                 |
| c4.xlarge <sup>2</sup>  |             | 750         |                          | 93.75                     |                      | 6000                 |
| c4.2xlarge <sub>2</sub> |             | 1000        |                          | 125.0                     |                      | 8000                 |
| c4.4xlarge <sub>2</sub> |             | 2000        |                          | 250.0                     |                      | 16000                |
| c4.8xlarge <sub>2</sub> |             | 4000        |                          | 500.0                     |                      | 32000                |
| c5.large <sup>1</sup>   | 650         | 4750        | 81.25                    | 593.75                    | 4000                 | 20000                |
| c5.xlarge <sup>1</sup>  | 1150        | 4750        | 143.75                   | 593.75                    | 6000                 | 20000                |
| c5.2xlarge <sub>1</sub> | 2300        | 4750        | 287.50                   | 593.75                    | 10000                | 20000                |
| c5.4xlarge <sub>2</sub> |             | 4750        |                          | 593.75                    |                      | 20000                |
| c5.9xlarge <sub>2</sub> |             | 9500        |                          | 1187.5                    |                      | 40000                |

| 執行個體大小                    | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|---------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| c5.12xlarge <sup>2</sup>  |             | 9500        |                          | 1187.5                    |                      | 40000                |
| c5.18xlarge <sup>2</sup>  |             | 19000       |                          | 2375.0                    |                      | 80000                |
| c5.24xlarge <sup>2</sup>  |             | 19000       |                          | 2375.0                    |                      | 80000                |
| c5.metal <sup>2</sup>     |             | 19000       |                          | 2375.0                    |                      | 80000                |
| c5a.large <sup>1</sup>    | 200         | 3170        | 25.00                    | 396.25                    | 800                  | 13300                |
| c5a.xlarge <sup>1</sup>   | 400         | 3170        | 50.00                    | 396.25                    | 1600                 | 13300                |
| c5a.2xlarge <sup>1</sup>  | 800         | 3170        | 100.00                   | 396.25                    | 3200                 | 13300                |
| c5a.4xlarge <sup>1</sup>  | 1580        | 3170        | 197.50                   | 396.25                    | 6600                 | 13300                |
| c5a.8xlarge <sup>2</sup>  |             | 3170        |                          | 396.25                    |                      | 13300                |
| c5a.12xlarge <sup>2</sup> |             | 4750        |                          | 593.75                    |                      | 20000                |
| c5a.16xlarge <sup>2</sup> |             | 6300        |                          | 787.5                     |                      | 26700                |
| c5a.24xlarge <sup>2</sup> |             | 9500        |                          | 1187.5                    |                      | 40000                |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| c5ad.large <sup>1</sup>    | 200         | 3170        | 25.00                    | 396.25                    | 800                  | 13300                |
| c5ad.xlarge <sup>1</sup>   | 400         | 3170        | 50.00                    | 396.25                    | 1600                 | 13300                |
| c5ad.2xlarge <sup>1</sup>  | 800         | 3170        | 100.00                   | 396.25                    | 3200                 | 13300                |
| c5ad.4xlarge <sup>1</sup>  | 1580        | 3170        | 197.50                   | 396.25                    | 6600                 | 13300                |
| c5ad.8xlarge <sup>2</sup>  |             | 3170        |                          | 396.25                    |                      | 13300                |
| c5ad.12xlarge <sup>2</sup> |             | 4750        |                          | 593.75                    |                      | 20000                |
| c5ad.16xlarge <sup>2</sup> |             | 6300        |                          | 787.5                     |                      | 26700                |
| c5ad.24xlarge <sup>2</sup> |             | 9500        |                          | 1187.5                    |                      | 40000                |
| c5d.large <sup>1</sup>     | 650         | 4750        | 81.25                    | 593.75                    | 4000                 | 20000                |
| c5d.xlarge <sup>1</sup>    | 1150        | 4750        | 143.75                   | 593.75                    | 6000                 | 20000                |
| c5d.2xlarge <sup>1</sup>   | 2300        | 4750        | 287.50                   | 593.75                    | 10000                | 20000                |
| c5d.4xlarge <sup>2</sup>   |             | 4750        |                          | 593.75                    |                      | 20000                |

| 執行個體大小                    | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|---------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| c5d.9xlarge <sup>2</sup>  |             | 9500        |                          | 1187.5                    |                      | 40000                |
| c5d.12xlarge <sup>2</sup> |             | 9500        |                          | 1187.5                    |                      | 40000                |
| c5d.18xlarge <sup>2</sup> |             | 19000       |                          | 2375.0                    |                      | 80000                |
| c5d.24xlarge <sup>2</sup> |             | 19000       |                          | 2375.0                    |                      | 80000                |
| c5d.metal <sup>2</sup>    |             | 19000       |                          | 2375.0                    |                      | 80000                |
| c5n.large <sup>1</sup>    | 650         | 4750        | 81.25                    | 593.75                    | 4000                 | 20000                |
| c5n.xlarge <sub>1</sub>   | 1150        | 4750        | 143.75                   | 593.75                    | 6000                 | 20000                |
| c5n.2xlarge <sup>1</sup>  | 2300        | 4750        | 287.50                   | 593.75                    | 10000                | 20000                |
| c5n.4xlarge <sup>2</sup>  |             | 4750        |                          | 593.75                    |                      | 20000                |
| c5n.9xlarge <sup>2</sup>  |             | 9500        |                          | 1187.5                    |                      | 40000                |
| c5n.18xlarge <sup>2</sup> |             | 19000       |                          | 2375.0                    |                      | 80000                |
| c5n.metal <sup>2</sup>    |             | 19000       |                          | 2375.0                    |                      | 80000                |
| c6a.large <sup>1</sup>    | 650         | 10000       | 81.25                    | 1250.00                   | 3600                 | 40000                |

| 執行個體大小                    | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|---------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| c6a.xlarge <sup>1</sup>   | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |
| c6a.2xlarge <sup>1</sup>  | 2500        | 10000       | 312.50                   | 1250.00                   | 12000                | 40000                |
| c6a.4xlarge <sup>1</sup>  | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |
| c6a.8xlarge <sup>2</sup>  |             | 10000       |                          | 1250.0                    |                      | 40000                |
| c6a.12xlarge <sup>2</sup> |             | 15000       |                          | 1875.0                    |                      | 60000                |
| c6a.16xlarge <sup>2</sup> |             | 20000       |                          | 2500.0                    |                      | 80000                |
| c6a.24xlarge <sup>2</sup> |             | 30000       |                          | 3750.0                    |                      | 120000               |
| c6a.32xlarge <sup>2</sup> |             | 40000       |                          | 5000.0                    |                      | 160000               |
| c6a.48xlarge <sup>2</sup> |             | 40000       |                          | 5000.0                    |                      | 240000               |
| c6a.metal <sup>2</sup>    |             | 40000       |                          | 5000.0                    |                      | 240000               |
| c6g.medium <sup>1</sup>   | 315         | 4750        | 39.38                    | 593.75                    | 2500                 | 20000                |
| c6g.large <sup>1</sup>    | 630         | 4750        | 78.75                    | 593.75                    | 3600                 | 20000                |

| 執行個體大小                    | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|---------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| c6g.xlarge <sup>1</sup>   | 1188        | 4750        | 148.50                   | 593.75                    | 6000                 | 20000                |
| c6g.2xlarge <sup>1</sup>  | 2375        | 4750        | 296.88                   | 593.75                    | 12000                | 20000                |
| c6g.4xlarge <sup>2</sup>  |             | 4750        |                          | 593.75                    |                      | 20000                |
| c6g.8xlarge <sup>2</sup>  |             | 9500        |                          | 1187.5                    |                      | 40000                |
| c6g.12xlarge <sup>2</sup> |             | 14250       |                          | 1781.25                   |                      | 50000                |
| c6g.16xlarge <sup>2</sup> |             | 19000       |                          | 2375.0                    |                      | 80000                |
| c6g.metal <sup>2</sup>    |             | 19000       |                          | 2375.0                    |                      | 80000                |
| c6gd.medium <sup>1</sup>  | 315         | 4750        | 39.38                    | 593.75                    | 2500                 | 20000                |
| c6gd.large <sup>1</sup>   | 630         | 4750        | 78.75                    | 593.75                    | 3600                 | 20000                |
| c6gd.xlarge <sup>1</sup>  | 1188        | 4750        | 148.50                   | 593.75                    | 6000                 | 20000                |
| c6gd.2xlarge <sup>1</sup> | 2375        | 4750        | 296.88                   | 593.75                    | 12000                | 20000                |
| c6gd.4xlarge <sup>2</sup> |             | 4750        |                          | 593.75                    |                      | 20000                |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| c6gd.8xlarge <sup>2</sup>  |             | 9500        |                          | 1187.5                    |                      | 40000                |
| c6gd.12xlarge <sup>2</sup> |             | 14250       |                          | 1781.25                   |                      | 50000                |
| c6gd.16xlarge <sup>2</sup> |             | 19000       |                          | 2375.0                    |                      | 80000                |
| c6gd.meta1 <sup>2</sup>    |             | 19000       |                          | 2375.0                    |                      | 80000                |
| c6gn.medium <sup>1</sup>   | 760         | 9500        | 95.00                    | 1187.50                   | 2500                 | 40000                |
| c6gn.large <sup>1</sup>    | 1235        | 9500        | 154.38                   | 1187.50                   | 5000                 | 40000                |
| c6gn.xlarge <sup>1</sup>   | 2375        | 9500        | 296.88                   | 1187.50                   | 10000                | 40000                |
| c6gn.2xlarge <sup>1</sup>  | 4750        | 9500        | 593.75                   | 1187.50                   | 20000                | 40000                |
| c6gn.4xlarge <sup>2</sup>  |             | 9500        |                          | 1187.5                    |                      | 40000                |
| c6gn.8xlarge <sup>2</sup>  |             | 19000       |                          | 2375.0                    |                      | 80000                |
| c6gn.12xlarge <sup>2</sup> |             | 28500       |                          | 3562.5                    |                      | 120000               |



| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| c6gn.16xlarge <sup>2</sup> |             | 38000       |                          | 4750.0                    |                      | 160000               |
| c6i.large <sup>1</sup>     | 650         | 10000       | 81.25                    | 1250.00                   | 3600                 | 40000                |
| c6i.xlarge <sup>1</sup>    | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |
| c6i.2xlarge <sup>1</sup>   | 2500        | 10000       | 312.50                   | 1250.00                   | 12000                | 40000                |
| c6i.4xlarge <sup>1</sup>   | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |
| c6i.8xlarge <sup>2</sup>   |             | 10000       |                          | 1250.0                    |                      | 40000                |
| c6i.12xlarge <sup>2</sup>  |             | 15000       |                          | 1875.0                    |                      | 60000                |
| c6i.16xlarge <sup>2</sup>  |             | 20000       |                          | 2500.0                    |                      | 80000                |
| c6i.24xlarge <sup>2</sup>  |             | 30000       |                          | 3750.0                    |                      | 120000               |
| c6i.32xlarge <sup>2</sup>  |             | 40000       |                          | 5000.0                    |                      | 160000               |
| c6i.metal <sup>2</sup>     |             | 40000       |                          | 5000.0                    |                      | 160000               |
| c6id.large <sup>1</sup>    | 650         | 10000       | 81.25                    | 1250.00                   | 3600                 | 40000                |
| c6id.xlarge <sup>1</sup>   | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| c6id.2xlarge <sup>1</sup>  | 2500        | 10000       | 312.50                   | 1250.00                   | 12000                | 40000                |
| c6id.4xlarge <sup>1</sup>  | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |
| c6id.8xlarge <sup>2</sup>  | 10000       |             | 1250.0                   |                           | 40000                |                      |
| c6id.12xlarge <sup>2</sup> | 15000       |             | 1875.0                   |                           | 60000                |                      |
| c6id.16xlarge <sup>2</sup> | 20000       |             | 2500.0                   |                           | 80000                |                      |
| c6id.24xlarge <sup>2</sup> | 30000       |             | 3750.0                   |                           | 120000               |                      |
| c6id.32xlarge <sup>2</sup> | 40000       |             | 5000.0                   |                           | 160000               |                      |
| c6id.metal <sup>2</sup>    | 40000       |             | 5000.0                   |                           | 160000               |                      |
| c6in.large <sup>1</sup>    | 1562        | 25000       | 195.31                   | 3125.00                   | 6250                 | 100000               |
| c6in.xlarge <sup>1</sup>   | 3125        | 25000       | 390.62                   | 3125.00                   | 12500                | 100000               |
| c6in.2xlarge <sup>1</sup>  | 6250        | 25000       | 781.25                   | 3125.00                   | 25000                | 100000               |
| c6in.4xlarge <sup>1</sup>  | 12500       | 25000       | 1562.50                  | 3125.00                   | 50000                | 100000               |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| c6in.8xlarge <sup>2</sup>  | 25000       |             | 3125.0                   |                           | 100000               |                      |
| c6in.12xlarge <sup>2</sup> | 37500       |             | 4687.5                   |                           | 150000               |                      |
| c6in.16xlarge <sup>2</sup> | 50000       |             | 6250.0                   |                           | 200000               |                      |
| c6in.24xlarge <sup>2</sup> | 75000       |             | 9375.0                   |                           | 300000               |                      |
| c6in.32xlarge <sup>2</sup> | 100000      |             | 12500.0                  |                           | 400000               |                      |
| c6in.metal <sub>2</sub>    | 100000      |             | 12500.0                  |                           | 400000               |                      |
| c7a.medium <sup>1</sup>    | 325         | 10000       | 40.62                    | 1250.00                   | 2500                 | 40000                |
| c7a.large <sup>1</sup>     | 650         | 10000       | 81.25                    | 1250.00                   | 3600                 | 40000                |
| c7a.xlarge <sub>1</sub>    | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |
| c7a.2xlarge <sup>1</sup>   | 2500        | 10000       | 312.50                   | 1250.00                   | 12000                | 40000                |
| c7a.4xlarge <sup>1</sup>   | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |
| c7a.8xlarge <sup>2</sup>   | 10000       |             | 1250.0                   |                           | 40000                |                      |

| 執行個體大小                      | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|-----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| c7a.12xlarge <sup>2</sup>   | 15000       |             |                          | 1875.0                    |                      | 60000                |
| c7a.16xlarge <sup>2</sup>   | 20000       |             |                          | 2500.0                    |                      | 80000                |
| c7a.24xlarge <sup>2</sup>   | 30000       |             |                          | 3750.0                    |                      | 120000               |
| c7a.32xlarge <sup>2</sup>   | 40000       |             |                          | 5000.0                    |                      | 160000               |
| c7a.48xlarge <sup>2</sup>   | 40000       |             |                          | 5000.0                    |                      | 240000               |
| c7a.metal-48xl <sup>2</sup> | 40000       |             |                          | 5000.0                    |                      | 240000               |
| c7g.medium <sup>1</sup>     | 315         | 10000       | 39.38                    | 1250.00                   | 2500                 | 40000                |
| c7g.large <sup>1</sup>      | 630         | 10000       | 78.75                    | 1250.00                   | 3600                 | 40000                |
| c7g.xlarge <sup>1</sup>     | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |
| c7g.2xlarge <sup>1</sup>    | 2500        | 10000       | 312.50                   | 1250.00                   | 12000                | 40000                |
| c7g.4xlarge <sup>1</sup>    | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |
| c7g.8xlarge <sup>2</sup>    | 10000       |             |                          | 1250.0                    |                      | 40000                |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| c7g.12xlarge <sup>2</sup>  |             | 15000       |                          | 1875.0                    |                      | 60000                |
| c7g.16xlarge <sup>2</sup>  |             | 20000       |                          | 2500.0                    |                      | 80000                |
| c7g.metal <sup>2</sup>     |             | 20000       |                          | 2500.0                    |                      | 80000                |
| c7gd.medium <sup>1</sup>   | 315         | 10000       | 39.38                    | 1250.00                   | 2500                 | 40000                |
| c7gd.large <sup>1</sup>    | 630         | 10000       | 78.75                    | 1250.00                   | 3600                 | 40000                |
| c7gd.xlarge <sup>1</sup>   | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |
| c7gd.2xlarge <sup>1</sup>  | 2500        | 10000       | 312.50                   | 1250.00                   | 12000                | 40000                |
| c7gd.4xlarge <sup>1</sup>  | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |
| c7gd.8xlarge <sup>2</sup>  |             | 10000       |                          | 1250.0                    |                      | 40000                |
| c7gd.12xlarge <sup>2</sup> |             | 15000       |                          | 1875.0                    |                      | 60000                |
| c7gd.16xlarge <sup>2</sup> |             | 20000       |                          | 2500.0                    |                      | 80000                |
| 金屬                         |             | 20000       |                          | 2500.0                    |                      | 80000                |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| c7gn.medium <sup>1</sup>   | 521         | 10000       | 65.12                    | 1250.00                   | 2083                 | 40000                |
| c7gn.large <sub>1</sub>    | 1042        | 10000       | 130.25                   | 1250.00                   | 4167                 | 40000                |
| c7gn.xlarge <sup>1</sup>   | 2083        | 10000       | 260.38                   | 1250.00                   | 8333                 | 40000                |
| c7gn.2xlarge <sup>1</sup>  | 4167        | 10000       | 520.88                   | 1250.00                   | 16667                | 40000                |
| c7gn.4xlarge <sup>1</sup>  | 8333        | 10000       | 1041.62                  | 1250.00                   | 33333                | 40000                |
| c7gn.8xlarge <sup>1</sup>  | 16667       | 20000       | 2083.38                  | 2500.00                   | 66667                | 80000                |
| c7gn.12xlarge <sup>1</sup> | 25000       | 30000       | 3125.00                  | 3750.00                   | 100000               | 120000               |
| c7gn.16xlarge <sup>1</sup> | 33333       | 40000       | 4166.62                  | 5000.00                   | 133333               | 160000               |
| 金屬                         | 33333       | 40000       | 4166.62                  | 5000.00                   | 133333               | 160000               |
| c7i.large <sup>1</sup>     | 650         | 10000       | 81.25                    | 1250.00                   | 3600                 | 40000                |
| c7i.xlarge <sup>1</sup>    | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |
| c7i.2xlarge <sub>1</sub>   | 2500        | 10000       | 312.50                   | 1250.00                   | 12000                | 40000                |
| c7i.4xlarge <sub>1</sub>   | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |

| 執行個體大小                      | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|-----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| c7i.8xlarge <sup>2</sup>    | 10000       |             |                          | 1250.0                    |                      | 40000                |
| c7i.12xlarge <sup>2</sup>   | 15000       |             |                          | 1875.0                    |                      | 60000                |
| c7i.16xlarge <sup>2</sup>   | 20000       |             |                          | 2500.0                    |                      | 80000                |
| c7i.24xlarge <sup>2</sup>   | 30000       |             |                          | 3750.0                    |                      | 120000               |
| c7i.48xlarge <sup>2</sup>   | 40000       |             |                          | 5000.0                    |                      | 240000               |
| c7i.metal-24xl <sup>2</sup> | 30000       |             |                          | 3750.0                    |                      | 120000               |
| c7i.metal-48xl <sup>2</sup> | 40000       |             |                          | 5000.0                    |                      | 240000               |
| C7-柔性.大1                    | 312         | 10000       | 39.06                    | 1250.00                   | 2500                 | 40000                |
| C7-柔性.大1                    | 625         | 10000       | 78.12                    | 1250.00                   | 3600                 | 40000                |
| C7-柔性版.2倍大1                 | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |
| C7-柔性.4倍大1                  | 2500        | 10000       | 312.50                   | 1250.00                   | 12000                | 40000                |
| C7-柔性.8倍大1                  | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |

## 記憶體最佳化

**⚠ Important**

<sup>1</sup> 這些執行個體支援至少每 24 小時一次維持最佳效能 30 分鐘，然後就會回復至其基準效能。

<sup>2</sup> 這些執行個體可以無限期維持其上述效能。如果您的工作負載需要維持超過 30 分鐘的最佳效能，請選取其中一個執行個體。

| 執行個體大小                   | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|--------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| r4.large <sup>2</sup>    |             | 425         |                          | 53.125                    |                      | 3000                 |
| r4.xlarge <sup>2</sup>   |             | 850         |                          | 106.25                    |                      | 6000                 |
| r4.2xlarge <sub>2</sub>  |             | 1700        |                          | 212.5                     |                      | 12000                |
| r4.4xlarge <sub>2</sub>  |             | 3500        |                          | 437.5                     |                      | 18750                |
| r4.8xlarge <sub>2</sub>  |             | 7000        |                          | 875.0                     |                      | 37500                |
| r4.16xlarge <sub>2</sub> |             | 14000       |                          | 1750.0                    |                      | 75000                |
| r5.large <sup>1</sup>    | 650         | 4750        | 81.25                    | 593.75                    | 3600                 | 18750                |
| r5.xlarge <sup>1</sup>   | 1150        | 4750        | 143.75                   | 593.75                    | 6000                 | 18750                |
| r5.2xlarge <sub>1</sub>  | 2300        | 4750        | 287.50                   | 593.75                    | 12000                | 18750                |
| r5.4xlarge <sub>2</sub>  |             | 4750        |                          | 593.75                    |                      | 18750                |



| 執行個體大小                    | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|---------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| r5.8xlarge <sub>2</sub>   |             | 6800        |                          | 850.0                     |                      | 30000                |
| r5.12xlarge <sub>2</sub>  |             | 9500        |                          | 1187.5                    |                      | 40000                |
| r5.16xlarge <sub>2</sub>  |             | 13600       |                          | 1700.0                    |                      | 60000                |
| r5.24xlarge <sub>2</sub>  |             | 19000       |                          | 2375.0                    |                      | 80000                |
| r5.metal <sup>2</sup>     |             | 19000       |                          | 2375.0                    |                      | 80000                |
| r5a.large <sup>1</sup>    | 650         | 2880        | 81.25                    | 360.00                    | 3600                 | 16000                |
| r5a.xlarge <sub>1</sub>   | 1085        | 2880        | 135.62                   | 360.00                    | 6000                 | 16000                |
| r5a.2xlarge <sub>1</sub>  | 1580        | 2880        | 197.50                   | 360.00                    | 8333                 | 16000                |
| r5a.4xlarge <sub>2</sub>  |             | 2880        |                          | 360.0                     |                      | 16000                |
| r5a.8xlarge <sub>2</sub>  |             | 4750        |                          | 593.75                    |                      | 20000                |
| r5a.12xlarge <sub>2</sub> |             | 6780        |                          | 847.5                     |                      | 30000                |
| r5a.16xlarge <sub>2</sub> |             | 9500        |                          | 1187.5                    |                      | 40000                |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| r5a.24xlarge <sup>2</sup>  |             | 13570       |                          | 1696.25                   |                      | 60000                |
| r5ad.large <sup>1</sup>    | 650         | 2880        | 81.25                    | 360.00                    | 3600                 | 16000                |
| r5ad.xlarge <sup>1</sup>   | 1085        | 2880        | 135.62                   | 360.00                    | 6000                 | 16000                |
| r5ad.2xlarge <sup>1</sup>  | 1580        | 2880        | 197.50                   | 360.00                    | 8333                 | 16000                |
| r5ad.4xlarge <sup>2</sup>  |             | 2880        |                          | 360.0                     |                      | 16000                |
| r5ad.8xlarge <sup>2</sup>  |             | 4750        |                          | 593.75                    |                      | 20000                |
| r5ad.12xlarge <sup>2</sup> |             | 6780        |                          | 847.5                     |                      | 30000                |
| r5ad.16xlarge <sup>2</sup> |             | 9500        |                          | 1187.5                    |                      | 40000                |
| r5ad.24xlarge <sup>2</sup> |             | 13570       |                          | 1696.25                   |                      | 60000                |
| r5b.large <sup>1</sup>     | 1250        | 10000       | 156.25                   | 1250.00                   | 5417                 | 43333                |
| r5b.xlarge <sup>1</sup>    | 2500        | 10000       | 312.50                   | 1250.00                   | 10833                | 43333                |
| r5b.2xlarge <sup>1</sup>   | 5000        | 10000       | 625.00                   | 1250.00                   | 21667                | 43333                |

| 執行個體大小                    | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|---------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| r5b.4xlarge <sup>2</sup>  |             | 10000       |                          | 1250.0                    |                      | 43333                |
| r5b.8xlarge <sup>2</sup>  |             | 20000       |                          | 2500.0                    |                      | 86667                |
| r5b.12xlarge <sup>2</sup> |             | 30000       |                          | 3750.0                    |                      | 130000               |
| r5b.16xlarge <sup>2</sup> |             | 40000       |                          | 5000.0                    |                      | 173333               |
| r5b.24xlarge <sup>2</sup> |             | 60000       |                          | 7500.0                    |                      | 260000               |
| r5b.metal <sup>2</sup>    |             | 60000       |                          | 7500.0                    |                      | 260000               |
| r5d.large <sup>1</sup>    | 650         | 4750        | 81.25                    | 593.75                    | 3600                 | 18750                |
| r5d.xlarge <sup>1</sup>   | 1150        | 4750        | 143.75                   | 593.75                    | 6000                 | 18750                |
| r5d.2xlarge <sup>1</sup>  | 2300        | 4750        | 287.50                   | 593.75                    | 12000                | 18750                |
| r5d.4xlarge <sup>2</sup>  |             | 4750        |                          | 593.75                    |                      | 18750                |
| r5d.8xlarge <sup>2</sup>  |             | 6800        |                          | 850.0                     |                      | 30000                |
| r5d.12xlarge <sup>2</sup> |             | 9500        |                          | 1187.5                    |                      | 40000                |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| r5d.16xlarge <sup>2</sup>  |             | 13600       |                          | 1700.0                    |                      | 60000                |
| r5d.24xlarge <sup>2</sup>  |             | 19000       |                          | 2375.0                    |                      | 80000                |
| r5d.metal <sup>2</sup>     |             | 19000       |                          | 2375.0                    |                      | 80000                |
| r5dn.large <sup>1</sup>    | 650         | 4750        | 81.25                    | 593.75                    | 3600                 | 18750                |
| r5dn.xlarge <sup>1</sup>   | 1150        | 4750        | 143.75                   | 593.75                    | 6000                 | 18750                |
| r5dn.2xlarge <sup>1</sup>  | 2300        | 4750        | 287.50                   | 593.75                    | 12000                | 18750                |
| r5dn.4xlarge <sup>2</sup>  |             | 4750        |                          | 593.75                    |                      | 18750                |
| r5dn.8xlarge <sup>2</sup>  |             | 6800        |                          | 850.0                     |                      | 30000                |
| r5dn.12xlarge <sup>2</sup> |             | 9500        |                          | 1187.5                    |                      | 40000                |
| r5dn.16xlarge <sup>2</sup> |             | 13600       |                          | 1700.0                    |                      | 60000                |
| r5dn.24xlarge <sup>2</sup> |             | 19000       |                          | 2375.0                    |                      | 80000                |
| r5dn.metal <sup>2</sup>    |             | 19000       |                          | 2375.0                    |                      | 80000                |

| 執行個體大小                    | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|---------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| r5n.large <sup>1</sup>    | 650         | 4750        | 81.25                    | 593.75                    | 3600                 | 18750                |
| r5n.xlarge <sup>1</sup>   | 1150        | 4750        | 143.75                   | 593.75                    | 6000                 | 18750                |
| r5n.2xlarge <sup>1</sup>  | 2300        | 4750        | 287.50                   | 593.75                    | 12000                | 18750                |
| r5n.4xlarge <sup>2</sup>  |             | 4750        |                          | 593.75                    |                      | 18750                |
| r5n.8xlarge <sup>2</sup>  |             | 6800        |                          | 850.0                     |                      | 30000                |
| r5n.12xlarge <sup>2</sup> |             | 9500        |                          | 1187.5                    |                      | 40000                |
| r5n.16xlarge <sup>2</sup> |             | 13600       |                          | 1700.0                    |                      | 60000                |
| r5n.24xlarge <sup>2</sup> |             | 19000       |                          | 2375.0                    |                      | 80000                |
| r5n.metal <sup>2</sup>    |             | 19000       |                          | 2375.0                    |                      | 80000                |
| r6a.large <sup>1</sup>    | 650         | 10000       | 81.25                    | 1250.00                   | 3600                 | 40000                |
| r6a.xlarge <sup>1</sup>   | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |
| r6a.2xlarge <sup>1</sup>  | 2500        | 10000       | 312.50                   | 1250.00                   | 12000                | 40000                |
| r6a.4xlarge <sup>1</sup>  | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |

| 執行個體大小                    | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|---------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| r6a.8xlarge <sup>2</sup>  | 10000       |             |                          | 1250.0                    |                      | 40000                |
| r6a.12xlarge <sup>2</sup> | 15000       |             |                          | 1875.0                    |                      | 60000                |
| r6a.16xlarge <sup>2</sup> | 20000       |             |                          | 2500.0                    |                      | 80000                |
| r6a.24xlarge <sup>2</sup> | 30000       |             |                          | 3750.0                    |                      | 120000               |
| r6a.32xlarge <sup>2</sup> | 40000       |             |                          | 5000.0                    |                      | 160000               |
| r6a.48xlarge <sup>2</sup> | 40000       |             |                          | 5000.0                    |                      | 240000               |
| r6a.metal <sup>2</sup>    | 40000       |             |                          | 5000.0                    |                      | 240000               |
| r6g.medium <sup>1</sup>   | 315         | 4750        | 39.38                    | 593.75                    | 2500                 | 20000                |
| r6g.large <sup>1</sup>    | 630         | 4750        | 78.75                    | 593.75                    | 3600                 | 20000                |
| r6g.xlarge <sup>1</sup>   | 1188        | 4750        | 148.50                   | 593.75                    | 6000                 | 20000                |
| r6g.2xlarge <sup>1</sup>  | 2375        | 4750        | 296.88                   | 593.75                    | 12000                | 20000                |
| r6g.4xlarge <sup>2</sup>  | 4750        |             |                          | 593.75                    |                      | 20000                |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| r6g.8xlarge <sup>2</sup>   | 9500        |             | 1187.5                   |                           | 40000                |                      |
| r6g.12xlarge <sup>2</sup>  | 14250       |             | 1781.25                  |                           | 50000                |                      |
| r6g.16xlarge <sup>2</sup>  | 19000       |             | 2375.0                   |                           | 80000                |                      |
| r6g.metal <sup>2</sup>     | 19000       |             | 2375.0                   |                           | 80000                |                      |
| r6gd.medium <sup>1</sup>   | 315         | 4750        | 39.38                    | 593.75                    | 2500                 | 20000                |
| r6gd.large <sup>1</sup>    | 630         | 4750        | 78.75                    | 593.75                    | 3600                 | 20000                |
| r6gd.xlarge <sup>1</sup>   | 1188        | 4750        | 148.50                   | 593.75                    | 6000                 | 20000                |
| r6gd.2xlarge <sup>1</sup>  | 2375        | 4750        | 296.88                   | 593.75                    | 12000                | 20000                |
| r6gd.4xlarge <sup>2</sup>  | 4750        |             | 593.75                   |                           | 20000                |                      |
| r6gd.8xlarge <sup>2</sup>  | 9500        |             | 1187.5                   |                           | 40000                |                      |
| r6gd.12xlarge <sup>2</sup> | 14250       |             | 1781.25                  |                           | 50000                |                      |
| r6gd.16xlarge <sup>2</sup> | 19000       |             | 2375.0                   |                           | 80000                |                      |

| 執行個體大小                              | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|-------------------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| r6gd.meta <sub>1</sub> <sup>2</sup> | 19000       |             | 2375.0                   |                           | 80000                |                      |
| r6i.large <sup>1</sup>              | 650         | 10000       | 81.25                    | 1250.00                   | 3600                 | 40000                |
| r6i.xlarge <sup>1</sup>             | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |
| r6i.2xlarge <sub>1</sub>            | 2500        | 10000       | 312.50                   | 1250.00                   | 12000                | 40000                |
| r6i.4xlarge <sub>1</sub>            | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |
| r6i.8xlarge <sub>2</sub>            | 10000       |             | 1250.0                   |                           | 40000                |                      |
| r6i.12xlarge <sub>2</sub>           | 15000       |             | 1875.0                   |                           | 60000                |                      |
| r6i.16xlarge <sub>2</sub>           | 20000       |             | 2500.0                   |                           | 80000                |                      |
| r6i.24xlarge <sub>2</sub>           | 30000       |             | 3750.0                   |                           | 120000               |                      |
| r6i.32xlarge <sub>2</sub>           | 40000       |             | 5000.0                   |                           | 160000               |                      |
| r6i.metal <sub>2</sub>              | 40000       |             | 5000.0                   |                           | 160000               |                      |
| r6idn.large <sub>1</sub>            | 1562        | 25000       | 195.31                   | 3125.00                   | 6250                 | 100000               |
| r6idn.xlarge <sub>1</sub>           | 3125        | 25000       | 390.62                   | 3125.00                   | 12500                | 100000               |



| 執行個體大小                      | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|-----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| r6idn.2xlarge <sup>1</sup>  | 6250        | 25000       | 781.25                   | 3125.00                   | 25000                | 100000               |
| r6idn.4xlarge <sup>1</sup>  | 12500       | 25000       | 1562.50                  | 3125.00                   | 50000                | 100000               |
| r6idn.8xlarge <sup>2</sup>  |             | 25000       |                          | 3125.0                    |                      | 100000               |
| r6idn.12xlarge <sup>2</sup> |             | 37500       |                          | 4687.5                    |                      | 150000               |
| r6idn.16xlarge <sup>2</sup> |             | 50000       |                          | 6250.0                    |                      | 200000               |
| r6idn.24xlarge <sup>2</sup> |             | 75000       |                          | 9375.0                    |                      | 300000               |
| r6idn.32xlarge <sup>2</sup> |             | 100000      |                          | 12500.0                   |                      | 400000               |
| r6idn.metal <sup>2</sup>    |             | 100000      |                          | 12500.0                   |                      | 400000               |
| r6in.large <sup>1</sup>     | 1562        | 25000       | 195.31                   | 3125.00                   | 6250                 | 100000               |
| r6in.xlarge <sup>1</sup>    | 3125        | 25000       | 390.62                   | 3125.00                   | 12500                | 100000               |
| r6in.2xlarge <sup>1</sup>   | 6250        | 25000       | 781.25                   | 3125.00                   | 25000                | 100000               |
| r6in.4xlarge <sup>1</sup>   | 12500       | 25000       | 1562.50                  | 3125.00                   | 50000                | 100000               |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| r6in.8xlarge <sup>2</sup>  |             | 25000       |                          | 3125.0                    |                      | 100000               |
| r6in.12xlarge <sup>2</sup> |             | 37500       |                          | 4687.5                    |                      | 150000               |
| r6in.16xlarge <sup>2</sup> |             | 50000       |                          | 6250.0                    |                      | 200000               |
| r6in.24xlarge <sup>2</sup> |             | 75000       |                          | 9375.0                    |                      | 300000               |
| r6in.32xlarge <sup>2</sup> |             | 100000      |                          | 12500.0                   |                      | 400000               |
| r6in.metal <sup>2</sup>    |             | 100000      |                          | 12500.0                   |                      | 400000               |
| r6id.large <sup>1</sup>    | 650         | 10000       | 81.25                    | 1250.00                   | 3600                 | 40000                |
| r6id.xlarge <sup>1</sup>   | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |
| r6id.2xlarge <sup>1</sup>  | 2500        | 10000       | 312.50                   | 1250.00                   | 12000                | 40000                |
| r6id.4xlarge <sup>1</sup>  | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |
| r6id.8xlarge <sup>2</sup>  |             | 10000       |                          | 1250.0                    |                      | 40000                |
| r6id.12xlarge <sup>2</sup> |             | 15000       |                          | 1875.0                    |                      | 60000                |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| r6id.16xlarge <sup>2</sup> |             | 20000       |                          | 2500.0                    |                      | 80000                |
| r6id.24xlarge <sup>2</sup> |             | 30000       |                          | 3750.0                    |                      | 120000               |
| r6id.32xlarge <sup>2</sup> |             | 40000       |                          | 5000.0                    |                      | 160000               |
| r6id.metal <sup>2</sup>    |             | 40000       |                          | 5000.0                    |                      | 160000               |
| r7a.medium <sup>1</sup>    | 325         | 10000       | 40.62                    | 1250.00                   | 2500                 | 40000                |
| r7a.large <sup>1</sup>     | 650         | 10000       | 81.25                    | 1250.00                   | 3600                 | 40000                |
| r7a.xlarge <sub>1</sub>    | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |
| r7a.2xlarge <sub>1</sub>   | 2500        | 10000       | 312.50                   | 1250.00                   | 12000                | 40000                |
| r7a.4xlarge <sub>1</sub>   | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |
| r7a.8xlarge <sub>2</sub>   |             | 10000       |                          | 1250.0                    |                      | 40000                |
| r7a.12xlarge <sup>2</sup>  |             | 15000       |                          | 1875.0                    |                      | 60000                |
| r7a.16xlarge <sup>2</sup>  |             | 20000       |                          | 2500.0                    |                      | 80000                |

| 執行個體大小                      | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|-----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| r7a.24xlarge <sup>2</sup>   |             | 30000       |                          | 3750.0                    |                      | 120000               |
| r7a.32xlarge <sup>2</sup>   |             | 40000       |                          | 5000.0                    |                      | 160000               |
| r7a.48xlarge <sup>2</sup>   |             | 40000       |                          | 5000.0                    |                      | 240000               |
| r7a.metal-48xl <sup>2</sup> |             | 40000       |                          | 5000.0                    |                      | 240000               |
| r7g.medium <sup>1</sup>     | 315         | 10000       | 39.38                    | 1250.00                   | 2500                 | 40000                |
| r7g.large <sup>1</sup>      | 630         | 10000       | 78.75                    | 1250.00                   | 3600                 | 40000                |
| r7g.xlarge <sub>1</sub>     | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |
| r7g.2xlarge <sub>1</sub>    | 2500        | 10000       | 312.50                   | 1250.00                   | 12000                | 40000                |
| r7g.4xlarge <sub>1</sub>    | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |
| r7g.8xlarge <sub>2</sub>    |             | 10000       |                          | 1250.0                    |                      | 40000                |
| r7g.12xlarge <sup>2</sup>   |             | 15000       |                          | 1875.0                    |                      | 60000                |
| r7g.16xlarge <sup>2</sup>   |             | 20000       |                          | 2500.0                    |                      | 80000                |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| r7g.metal <sup>2</sup>     |             | 20000       |                          | 2500.0                    |                      | 80000                |
| r7gd.medium <sup>1</sup>   | 315         | 10000       | 39.38                    | 1250.00                   | 2500                 | 40000                |
| r7gd.large <sup>1</sup>    | 630         | 10000       | 78.75                    | 1250.00                   | 3600                 | 40000                |
| r7gd.xlarge <sup>1</sup>   | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |
| r7gd.2xlarge <sup>1</sup>  | 2500        | 10000       | 312.50                   | 1250.00                   | 12000                | 40000                |
| r7gd.4xlarge <sup>1</sup>  | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |
| r7gd.8xlarge <sup>2</sup>  |             | 10000       |                          | 1250.0                    |                      | 40000                |
| r7gd.12xlarge <sup>2</sup> |             | 15000       |                          | 1875.0                    |                      | 60000                |
| r7gd.16xlarge <sup>2</sup> |             | 20000       |                          | 2500.0                    |                      | 80000                |
| 金屬                         |             | 20000       |                          | 2500.0                    |                      | 80000                |
| r7i.large <sup>1</sup>     | 650         | 10000       | 81.25                    | 1250.00                   | 3600                 | 40000                |
| r7i.xlarge <sup>1</sup>    | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |
| r7i.2xlarge <sup>1</sup>   | 2500        | 10000       | 312.50                   | 1250.00                   | 12000                | 40000                |

| 執行個體大小                      | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|-----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| r7i.4xlarge <sup>1</sup>    | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |
| r7i.8xlarge <sup>2</sup>    | 10000       |             | 1250.0                   |                           | 40000                |                      |
| r7i.12xlarge <sup>2</sup>   | 15000       |             | 1875.0                   |                           | 60000                |                      |
| r7i.16xlarge <sup>2</sup>   | 20000       |             | 2500.0                   |                           | 80000                |                      |
| r7i.24xlarge <sup>2</sup>   | 30000       |             | 3750.0                   |                           | 120000               |                      |
| r7i.48xlarge <sup>2</sup>   | 40000       |             | 5000.0                   |                           | 240000               |                      |
| r7i.metal-24xl <sup>2</sup> | 30000       |             | 3750.0                   |                           | 120000               |                      |
| r7i.metal-48xl <sup>2</sup> | 40000       |             | 5000.0                   |                           | 240000               |                      |
| r7iz.large <sup>1</sup>     | 792         | 10000       | 99.00                    | 1250.00                   | 3600                 | 40000                |
| r7iz.xlarge <sup>1</sup>    | 1584        | 10000       | 198.00                   | 1250.00                   | 6667                 | 40000                |
| r7iz.2xlarge <sup>1</sup>   | 3168        | 10000       | 396.00                   | 1250.00                   | 13333                | 40000                |
| r7iz.4xlarge <sup>1</sup>   | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |

| 執行個體大小                        | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|-------------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| r7iz.8xlarge <sup>2</sup>     | 10000       |             |                          | 1250.0                    |                      | 40000                |
| r7iz.12xlarge <sup>2</sup>    | 19000       |             |                          | 2375.0                    |                      | 76000                |
| r7iz.16xlarge <sup>2</sup>    | 20000       |             |                          | 2500.0                    |                      | 80000                |
| r7iz.32xlarge <sup>2</sup>    | 40000       |             |                          | 5000.0                    |                      | 160000               |
| r7iz.meta1-16xl <sup>2</sup>  | 20000       |             |                          | 2500.0                    |                      | 80000                |
| r7iz.meta1-32xl <sup>2</sup>  | 40000       |             |                          | 5000.0                    |                      | 160000               |
| u-3tb1.56xlarge <sup>2</sup>  | 19000       |             |                          | 2375.0                    |                      | 80000                |
| u-6tb1.56xlarge <sup>2</sup>  | 38000       |             |                          | 4750.0                    |                      | 160000               |
| u-6tb1.112xlarge <sup>2</sup> | 38000       |             |                          | 4750.0                    |                      | 160000               |
| u-6tb1.metal <sup>2</sup>     | 38000       |             |                          | 4750.0                    |                      | 160000               |
| u-9tb1.112xlarge <sup>2</sup> | 38000       |             |                          | 4750.0                    |                      | 160000               |

| 執行個體大小                        | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|-------------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| u-9tb1.metal <sup>2</sup>     | 38000       |             |                          | 4750.0                    |                      | 160000               |
| u-12tb1.12xlarge <sup>2</sup> | 38000       |             |                          | 4750.0                    |                      | 160000               |
| u-12tb1.metal <sup>2</sup>    | 38000       |             |                          | 4750.0                    |                      | 160000               |
| u-18tb1.12xlarge <sup>2</sup> | 38000       |             |                          | 4750.0                    |                      | 160000               |
| u-18tb1.metal <sup>2</sup>    | 38000       |             |                          | 4750.0                    |                      | 160000               |
| u-24tb1.12xlarge <sup>2</sup> | 38000       |             |                          | 4750.0                    |                      | 160000               |
| u-24tb1.metal <sup>2</sup>    | 38000       |             |                          | 4750.0                    |                      | 160000               |
| 二十四千四百六十四                     | 60000       |             |                          | 7500.0                    |                      | 420000               |
| 超大                            | 100000      |             |                          | 12500.0                   |                      | 420000               |
| 二十四分之二                        | 100000      |             |                          | 12500.0                   |                      | 420000               |
| 二十四分之二                        | 100000      |             |                          | 12500.0                   |                      | 420000               |
| x1.16xlarge <sup>2</sup>      | 7000        |             |                          | 875.0                     |                      | 40000                |
| x1.32xlarge <sup>2</sup>      | 14000       |             |                          | 1750.0                    |                      | 80000                |



| 執行個體大小                      | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|-----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| x2gd.medium <sup>1</sup>    | 315         | 4750        | 39.38                    | 593.75                    | 2500                 | 20000                |
| x2gd.large <sup>1</sup>     | 630         | 4750        | 78.75                    | 593.75                    | 3600                 | 20000                |
| x2gd.xlarge <sup>1</sup>    | 1188        | 4750        | 148.50                   | 593.75                    | 6000                 | 20000                |
| x2gd.2xlarge <sup>1</sup>   | 2375        | 4750        | 296.88                   | 593.75                    | 12000                | 20000                |
| x2gd.4xlarge <sup>2</sup>   |             | 4750        |                          | 593.75                    |                      | 20000                |
| x2gd.8xlarge <sup>2</sup>   |             | 9500        |                          | 1187.5                    |                      | 40000                |
| x2gd.12xlarge <sup>2</sup>  |             | 14250       |                          | 1781.25                   |                      | 60000                |
| x2gd.16xlarge <sup>2</sup>  |             | 19000       |                          | 2375.0                    |                      | 80000                |
| x2gd.meta <sup>2</sup>      |             | 19000       |                          | 2375.0                    |                      | 80000                |
| x2idn.16xlarge <sup>2</sup> |             | 40000       |                          | 5000.0                    |                      | 173333               |
| x2idn.24xlarge <sup>2</sup> |             | 60000       |                          | 7500.0                    |                      | 260000               |

| 執行個體大小                       | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|------------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| x2idn.32xlarge <sup>2</sup>  | 80000       |             |                          | 10000.0                   |                      | 260000               |
| x2idn.metal <sup>2</sup>     | 80000       |             |                          | 10000.0                   |                      | 260000               |
| x2iedn.xlarge <sup>1</sup>   | 2500        | 20000       | 312.50                   | 2500.00                   | 8125                 | 65000                |
| x2iedn.2xlarge <sup>1</sup>  | 5000        | 20000       | 625.00                   | 2500.00                   | 16250                | 65000                |
| x2iedn.4xlarge <sup>1</sup>  | 10000       | 20000       | 1250.00                  | 2500.00                   | 32500                | 65000                |
| x2iedn.8xlarge <sup>2</sup>  | 20000       |             |                          | 2500.0                    |                      | 65000                |
| x2iedn.16xlarge <sup>2</sup> | 40000       |             |                          | 5000.0                    |                      | 130000               |
| x2iedn.24xlarge <sup>2</sup> | 60000       |             |                          | 7500.0                    |                      | 195000               |
| x2iedn.32xlarge <sup>2</sup> | 80000       |             |                          | 10000.0                   |                      | 260000               |
| x2iedn.metal <sup>2</sup>    | 80000       |             |                          | 10000.0                   |                      | 260000               |
| x2iezn.2xlarge <sup>2</sup>  | 3170        |             |                          | 396.25                    |                      | 13333                |

| 執行個體大小                       | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|------------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| x2iezn.4xlarge <sup>2</sup>  | 4750        |             | 593.75                   |                           | 20000                |                      |
| x2iezn.6xlarge <sup>2</sup>  | 9500        |             | 1187.5                   |                           | 40000                |                      |
| x2iezn.8xlarge <sup>2</sup>  | 12000       |             | 1500.0                   |                           | 55000                |                      |
| x2iezn.12xlarge <sup>2</sup> | 19000       |             | 2375.0                   |                           | 80000                |                      |
| x2iezn.metal <sup>2</sup>    | 19000       |             | 2375.0                   |                           | 80000                |                      |
| x1e.xlarge <sub>2</sub>      | 500         |             | 62.5                     |                           | 3700                 |                      |
| x1e.2xlarge <sup>2</sup>     | 1000        |             | 125.0                    |                           | 7400                 |                      |
| x1e.4xlarge <sup>2</sup>     | 1750        |             | 218.75                   |                           | 10000                |                      |
| x1e.8xlarge <sup>2</sup>     | 3500        |             | 437.5                    |                           | 20000                |                      |
| x1e.16xlarge <sup>2</sup>    | 7000        |             | 875.0                    |                           | 40000                |                      |
| x1e.32xlarge <sup>2</sup>    | 14000       |             | 1750.0                   |                           | 80000                |                      |
| z1d.large <sup>1</sup>       | 800         | 3170        | 100.00                   | 396.25                    | 3333                 | 13333                |

| 執行個體大小                    | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|---------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| z1d.xlarge <sup>1</sup>   | 1580        | 3170        | 197.50                   | 396.25                    | 6667                 | 13333                |
| z1d.2xlarge <sup>2</sup>  |             | 3170        |                          | 396.25                    |                      | 13333                |
| z1d.3xlarge <sup>2</sup>  |             | 4750        |                          | 593.75                    |                      | 20000                |
| z1d.6xlarge <sup>2</sup>  |             | 9500        |                          | 1187.5                    |                      | 40000                |
| z1d.12xlarge <sup>2</sup> |             | 19000       |                          | 2375.0                    |                      | 80000                |
| z1d.metal <sup>2</sup>    |             | 19000       |                          | 2375.0                    |                      | 80000                |

## 儲存最佳化

### Important

- <sup>1</sup> 這些執行個體支援至少每 24 小時一次維持最佳效能 30 分鐘，然後就會回復至其基準效能。
- <sup>2</sup> 這些執行個體可以無限期維持其上述效能。如果您的工作負載需要維持超過 30 分鐘的最佳效能，請選取其中一個執行個體。

| 執行個體大小                 | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| d2.xlarge <sup>2</sup> |             | 750         |                          | 93.75                     |                      | 6000                 |

| 執行個體大小                    | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|---------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| d2.2xlarge <sub>2</sub>   |             | 1000        |                          | 125.0                     |                      | 8000                 |
| d2.4xlarge <sub>2</sub>   |             | 2000        |                          | 250.0                     |                      | 16000                |
| d2.8xlarge <sub>2</sub>   |             | 4000        |                          | 500.0                     |                      | 32000                |
| d3.xlarge <sup>1</sup>    | 850         | 2800        | 106.25                   | 350.00                    | 5000                 | 15000                |
| d3.2xlarge <sub>1</sub>   | 1700        | 2800        | 212.50                   | 350.00                    | 10000                | 15000                |
| d3.4xlarge <sub>2</sub>   |             | 2800        |                          | 350.0                     |                      | 15000                |
| d3.8xlarge <sub>2</sub>   |             | 5000        |                          | 625.0                     |                      | 30000                |
| d3en.xlarge <sup>1</sup>  | 850         | 2800        | 106.25                   | 350.00                    | 5000                 | 15000                |
| d3en.2xlarge <sup>1</sup> | 1700        | 2800        | 212.50                   | 350.00                    | 10000                | 15000                |
| d3en.4xlarge <sup>2</sup> |             | 2800        |                          | 350.0                     |                      | 15000                |
| d3en.6xlarge <sup>2</sup> |             | 4000        |                          | 500.0                     |                      | 25000                |
| d3en.8xlarge <sup>2</sup> |             | 5000        |                          | 625.0                     |                      | 30000                |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| d3en.12xlarge <sup>2</sup> | 7000        |             | 875.0                    |                           | 40000                |                      |
| h1.2xlarge <sub>2</sub>    | 1750        |             | 218.75                   |                           | 12000                |                      |
| h1.4xlarge <sub>2</sub>    | 3500        |             | 437.5                    |                           | 20000                |                      |
| h1.8xlarge <sub>2</sub>    | 7000        |             | 875.0                    |                           | 40000                |                      |
| h1.16xlarge <sup>2</sup>   | 14000       |             | 1750.0                   |                           | 80000                |                      |
| i3.large <sup>2</sup>      | 425         |             | 53.125                   |                           | 3000                 |                      |
| i3.xlarge <sup>2</sup>     | 850         |             | 106.25                   |                           | 6000                 |                      |
| i3.2xlarge <sup>2</sup>    | 1700        |             | 212.5                    |                           | 12000                |                      |
| i3.4xlarge <sup>2</sup>    | 3500        |             | 437.5                    |                           | 16000                |                      |
| i3.8xlarge <sup>2</sup>    | 7000        |             | 875.0                    |                           | 32500                |                      |
| i3.16xlarge <sub>2</sub>   | 14000       |             | 1750.0                   |                           | 65000                |                      |
| i3.metal <sup>2</sup>      | 19000       |             | 2375.0                   |                           | 80000                |                      |
| i3en.large <sup>1</sup>    | 576         | 4750        | 72.10                    | 593.75                    | 3000                 | 20000                |
| i3en.xlarge <sub>1</sub>   | 1153        | 4750        | 144.20                   | 593.75                    | 6000                 | 20000                |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| i3en.2xlarge <sup>1</sup>  | 2307        | 4750        | 288.39                   | 593.75                    | 12000                | 20000                |
| i3en.3xlarge <sup>1</sup>  | 3800        | 4750        | 475.00                   | 593.75                    | 15000                | 20000                |
| i3en.6xlarge <sup>2</sup>  |             | 4750        |                          | 593.75                    |                      | 20000                |
| i3en.12xlarge <sup>2</sup> |             | 9500        |                          | 1187.5                    |                      | 40000                |
| i3en.24xlarge <sup>2</sup> |             | 19000       |                          | 2375.0                    |                      | 80000                |
| i3en.metal <sup>2</sup>    |             | 19000       |                          | 2375.0                    |                      | 80000                |
| i4g.large <sup>1</sup>     | 625         | 10000       | 78.12                    | 1250.00                   | 2500                 | 40000                |
| i4g.xlarge <sup>1</sup>    | 1250        | 10000       | 156.25                   | 1250.00                   | 5000                 | 40000                |
| i4g.2xlarge <sup>1</sup>   | 2500        | 10000       | 312.50                   | 1250.00                   | 10000                | 40000                |
| i4g.4xlarge <sup>1</sup>   | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |
| i4g.8xlarge <sup>2</sup>   |             | 10000       |                          | 1250.0                    |                      | 40000                |
| i4g.16xlarge <sup>2</sup>  |             | 20000       |                          | 2500.0                    |                      | 80000                |
| i4i.large <sup>1</sup>     | 625         | 10000       | 78.12                    | 1250.00                   | 2500                 | 40000                |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| i4i.xlarge <sup>1</sup>    | 1250        | 10000       | 156.25                   | 1250.00                   | 5000                 | 40000                |
| i4i.2xlarge <sup>1</sup>   | 2500        | 10000       | 312.50                   | 1250.00                   | 10000                | 40000                |
| i4i.4xlarge <sup>1</sup>   | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |
| i4i.8xlarge <sup>2</sup>   |             | 10000       |                          | 1250.0                    |                      | 40000                |
| i4i.12xlarge <sup>2</sup>  |             | 15000       |                          | 1875.0                    |                      | 60000                |
| i4i.16xlarge <sup>2</sup>  |             | 20000       |                          | 2500.0                    |                      | 80000                |
| i4i.24xlarge <sup>2</sup>  |             | 30000       |                          | 3750.0                    |                      | 120000               |
| i4i.32xlarge <sup>2</sup>  |             | 40000       |                          | 5000.0                    |                      | 160000               |
| i4i.metal <sup>2</sup>     |             | 40000       |                          | 5000.0                    |                      | 160000               |
| im4gn.large <sup>1</sup>   | 1250        | 10000       | 156.25                   | 1250.00                   | 5000                 | 40000                |
| im4gn.xlarge <sup>1</sup>  | 2500        | 10000       | 312.50                   | 1250.00                   | 10000                | 40000                |
| im4gn.2xlarge <sup>1</sup> | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |



| 執行個體大小                      | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|-----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| im4gn.4xlarge <sup>2</sup>  |             | 10000       |                          | 1250.0                    |                      | 40000                |
| im4gn.8xlarge <sup>2</sup>  |             | 20000       |                          | 2500.0                    |                      | 80000                |
| im4gn.16xlarge <sup>2</sup> |             | 40000       |                          | 5000.0                    |                      | 160000               |
| is4gen.medium <sup>1</sup>  | 625         | 10000       | 78.12                    | 1250.00                   | 2500                 | 40000                |
| is4gen.large <sup>1</sup>   | 1250        | 10000       | 156.25                   | 1250.00                   | 5000                 | 40000                |
| is4gen.xlarge <sup>1</sup>  | 2500        | 10000       | 312.50                   | 1250.00                   | 10000                | 40000                |
| is4gen.2xlarge <sup>1</sup> | 5000        | 10000       | 625.00                   | 1250.00                   | 20000                | 40000                |
| is4gen.4xlarge <sup>2</sup> |             | 10000       |                          | 1250.0                    |                      | 40000                |
| is4gen.8xlarge <sup>2</sup> |             | 20000       |                          | 2500.0                    |                      | 80000                |

## 加速運算

### Important

<sup>1</sup> 這些執行個體支援至少每 24 小時一次維持最佳效能 30 分鐘，然後就會回復至其基準效能。

<sup>2</sup> 這些執行個體可以無限期維持其上述效能。如果您的工作負載需要維持超過 30 分鐘的最佳效能，請選取其中一個執行個體。

| 執行個體大小                    | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|---------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| d1.24xlarge <sup>2</sup>  |             | 19000       |                          | 2375.0                    |                      | 80000                |
| d1.24xlarge <sup>2</sup>  |             | 19000       |                          | 2375.0                    |                      | 80000                |
| f1.2xlarge <sup>2</sup>   |             | 1700        |                          | 212.5                     |                      | 12000                |
| f1.4xlarge <sup>2</sup>   |             | 3500        |                          | 437.5                     |                      | 44000                |
| f1.16xlarge <sup>2</sup>  |             | 14000       |                          | 1750.0                    |                      | 75000                |
| g3.4xlarge <sup>2</sup>   |             | 3500        |                          | 437.5                     |                      | 20000                |
| g3.8xlarge <sup>2</sup>   |             | 7000        |                          | 875.0                     |                      | 40000                |
| g3.16xlarge <sup>2</sup>  |             | 14000       |                          | 1750.0                    |                      | 80000                |
| g4ad.xlarge <sup>1</sup>  | 400         | 3170        | 50.00                    | 396.25                    | 1700                 | 13333                |
| g4ad.2xlarge <sup>1</sup> | 800         | 3170        | 100.00                   | 396.25                    | 3400                 | 13333                |
| g4ad.4xlarge <sup>1</sup> | 1580        | 3170        | 197.50                   | 396.25                    | 6700                 | 13333                |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| g4ad.8xlarge <sup>2</sup>  | 3170        |             | 396.25                   |                           | 13333                |                      |
| g4ad.16xlarge <sup>2</sup> | 6300        |             | 787.5                    |                           | 26667                |                      |
| g4dn.xlarge <sup>1</sup>   | 950         | 3500        | 118.75                   | 437.50                    | 3000                 | 20000                |
| g4dn.2xlarge <sup>1</sup>  | 1150        | 3500        | 143.75                   | 437.50                    | 6000                 | 20000                |
| g4dn.4xlarge <sup>2</sup>  | 4750        |             | 593.75                   |                           | 20000                |                      |
| g4dn.8xlarge <sup>2</sup>  | 9500        |             | 1187.5                   |                           | 40000                |                      |
| g4dn.12xlarge <sup>2</sup> | 9500        |             | 1187.5                   |                           | 40000                |                      |
| g4dn.16xlarge <sup>2</sup> | 9500        |             | 1187.5                   |                           | 40000                |                      |
| g4dn.meta <sup>1,2</sup>   | 19000       |             | 2375.0                   |                           | 80000                |                      |
| g5.xlarge <sup>1</sup>     | 700         | 3500        | 87.50                    | 437.50                    | 3000                 | 15000                |
| g5.2xlarge <sup>1</sup>    | 850         | 3500        | 106.25                   | 437.50                    | 3500                 | 15000                |
| g5.4xlarge <sup>2</sup>    | 4750        |             | 593.75                   |                           | 20000                |                      |

| 執行個體大小                    | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|---------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| g5.8xlarge <sup>2</sup>   | 16000       |             |                          | 2000.0                    |                      | 65000                |
| g5.12xlarge <sup>2</sup>  | 16000       |             |                          | 2000.0                    |                      | 65000                |
| g5.16xlarge <sup>2</sup>  | 16000       |             |                          | 2000.0                    |                      | 65000                |
| g5.24xlarge <sup>2</sup>  | 19000       |             |                          | 2375.0                    |                      | 80000                |
| g5.48xlarge <sup>2</sup>  | 19000       |             |                          | 2375.0                    |                      | 80000                |
| g5g.xlarge <sup>1</sup>   | 1188        | 4750        | 148.50                   | 593.75                    | 6000                 | 20000                |
| g5g.2xlarge <sup>1</sup>  | 2375        | 4750        | 296.88                   | 593.75                    | 12000                | 20000                |
| g5g.4xlarge <sup>2</sup>  | 4750        |             |                          | 593.75                    |                      | 20000                |
| g5g.8xlarge <sup>2</sup>  | 9500        |             |                          | 1187.5                    |                      | 40000                |
| g5g.16xlarge <sup>2</sup> | 19000       |             |                          | 2375.0                    |                      | 80000                |
| g5g.metal <sup>2</sup>    | 19000       |             |                          | 2375.0                    |                      | 80000                |
| 大 <sup>1</sup>            | 1000        | 5000        | 125.00                   | 625.00                    | 4000                 | 20000                |
| 大                         | 2000        | 5000        | 250.00                   | 625.00                    | 8000                 | 20000                |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| 大尺寸                        | 8000        |             | 1000.0                   |                           | 32000                |                      |
| 大尺寸                        | 16000       |             | 2000.0                   |                           | 64000                |                      |
| 大                          | 20000       |             | 2500.0                   |                           | 80000                |                      |
| 大                          | 20000       |             | 2500.0                   |                           | 80000                |                      |
| 大型                         | 30000       |             | 3750.0                   |                           | 120000               |                      |
| 大                          | 60000       |             | 7500.0                   |                           | 240000               |                      |
| GR6.4 X 大                  | 8000        |             | 1000.0                   |                           | 32000                |                      |
| g 6.8 X 大                  | 16000       |             | 2000.0                   |                           | 64000                |                      |
| inf1.xlarge <sub>1</sub>   | 1190        | 4750        | 148.75                   | 593.75                    | 4000                 | 20000                |
| inf1.2xlarge <sub>1</sub>  | 1190        | 4750        | 148.75                   | 593.75                    | 6000                 | 20000                |
| inf1.6xlarge <sub>2</sub>  |             | 4750        |                          | 593.75                    |                      | 20000                |
| inf1.24xlarge <sub>2</sub> |             | 19000       |                          | 2375.0                    |                      | 80000                |
| inf2.xlarge <sub>1</sub>   | 1250        | 10000       | 156.25                   | 1250.00                   | 6000                 | 40000                |
| inf2.8xlarge <sub>2</sub>  |             | 10000       |                          | 1250.0                    |                      | 40000                |

| 執行個體大小                     | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| inf2.24xlarge <sup>2</sup> | 30000       |             | 3750.0                   |                           | 120000               |                      |
| inf2.48xlarge <sup>2</sup> | 60000       |             | 7500.0                   |                           | 240000               |                      |
| p2.xlarge <sup>2</sup>     | 750         |             | 93.75                    |                           | 6000                 |                      |
| p2.8xlarge <sub>2</sub>    | 5000        |             | 625.0                    |                           | 32500                |                      |
| p2.16xlarge <sup>2</sup>   | 10000       |             | 1250.0                   |                           | 65000                |                      |
| p3.2xlarge <sub>2</sub>    | 1750        |             | 218.75                   |                           | 10000                |                      |
| p3.8xlarge <sub>2</sub>    | 7000        |             | 875.0                    |                           | 40000                |                      |
| p3.16xlarge <sup>2</sup>   | 14000       |             | 1750.0                   |                           | 80000                |                      |
| p3dn.24xlarge <sup>2</sup> | 19000       |             | 2375.0                   |                           | 80000                |                      |
| p4d.24xlarge <sup>2</sup>  | 19000       |             | 2375.0                   |                           | 80000                |                      |
| p4de.24xlarge <sup>2</sup> | 19000       |             | 2375.0                   |                           | 80000                |                      |
| p5.48xlarge <sup>2</sup>   | 80000       |             | 10000.0                  |                           | 260000               |                      |

| 執行個體大小                      | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|-----------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| trn1.2xlarge <sup>1</sup>   | 5000        | 20000       | 625.00                   | 2500.00                   | 16250                | 65000                |
| trn1.32xlarge <sup>2</sup>  |             | 80000       |                          | 10000.0                   |                      | 260000               |
| trn1n.32xlarge <sup>2</sup> |             | 80000       |                          | 10000.0                   |                      | 260000               |
| vt1.3xlarge <sup>1</sup>    | 2375        | 4750        | 296.88                   | 593.75                    | 10000                | 20000                |
| vt1.6xlarge <sup>2</sup>    |             | 4750        |                          | 593.75                    |                      | 20000                |
| vt1.24xlarge <sup>2</sup>   |             | 19000       |                          | 2375.0                    |                      | 80000                |

## 高效能運算

### Important

<sup>1</sup> 這些執行個體支援至少每 24 小時一次維持最佳效能 30 分鐘，然後就會回復至其基準效能。

<sup>2</sup> 這些執行個體可以無限期維持其上述效能。如果您的工作負載需要維持超過 30 分鐘的最佳效能，請選取其中一個執行個體。

| 執行個體大小                       | 基準頻寬 (Mbps) | 最高頻寬 (Mbps) | 基準輸送量 (MB/s, 128 KB I/O) | 最大傳輸量 (MB/s, 128 KiB I/O) | 基準 IOPS (16 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|------------------------------|-------------|-------------|--------------------------|---------------------------|----------------------|----------------------|
| hpc6a.48xlarge <sup>1</sup>  | 87          | 2085        | 10.88                    | 260.62                    | 500                  | 11000                |
| hpc6id.32xlarge <sup>1</sup> | 87          | 2085        | 10.88                    | 260.62                    | 500                  | 11000                |
| hpc7a.12xlarge <sup>1</sup>  | 87          | 2085        | 10.88                    | 260.62                    | 500                  | 11000                |
| hpc7a.24xlarge <sup>1</sup>  | 87          | 2085        | 10.88                    | 260.62                    | 500                  | 11000                |
| hpc7a.48xlarge <sup>1</sup>  | 87          | 2085        | 10.88                    | 260.62                    | 500                  | 11000                |
| hpc7a.96xlarge <sup>1</sup>  | 87          | 2085        | 10.88                    | 260.62                    | 500                  | 11000                |
| hpc7g.4xlarge <sup>1</sup>   | 87          | 2085        | 10.88                    | 260.62                    | 500                  | 11000                |
| hpc7g.8xlarge <sup>1</sup>   | 87          | 2085        | 10.88                    | 260.62                    | 500                  | 11000                |
| hpc7g.16xlarge <sup>1</sup>  | 87          | 2085        | 10.88                    | 260.62                    | 500                  | 11000                |

## 支援 EBS 最佳化

下表列出支援 EBS 最佳化的執行個體類型，但預設不啟用 EBS 最佳化。當您啟動這些執行個體時或在它們執行之後，您可以啟用 EBS 最佳化。執行個體必須啟用 EBS 最佳化，才能達到所述的效能等級。要為未預設 EBS 最佳化的執行個體啟用 EBS 最佳化時，您需為專屬容量額外支付少許的每小時使用費。如需定價資訊，請參閱 [Amazon EC2 定價、隨需定價頁面](#) 中的 EBS 最佳化執行個體。



**Note**

您也可以使用程式設計方式檢視此資訊。AWS CLI如需詳細資訊，請參閱 [檢視支援 EBS 最佳化的執行個體類型](#)。

| 執行個體大小     | 最高頻寬 (Mbps) | 最大傳輸量 (MB/s、128 KiB I/O) | 最大 IOPS (16 KiB I/O) |
|------------|-------------|--------------------------|----------------------|
| c1.xlarge  | 1000        | 125.0                    | 8000                 |
| c3.xlarge  | 500         | 62.5                     | 4000                 |
| c3.2xlarge | 1000        | 125.0                    | 8000                 |
| c3.4xlarge | 2000        | 250.0                    | 16000                |
| i2.xlarge  | 500         | 62.5                     | 4000                 |
| i2.2xlarge | 1000        | 125.0                    | 8000                 |
| i2.4xlarge | 2000        | 250.0                    | 16000                |
| m1.large   | 500         | 62.5                     | 4000                 |
| m1.xlarge  | 1000        | 125.0                    | 8000                 |
| m2.2xlarge | 500         | 62.5                     | 4000                 |
| m2.4xlarge | 1000        | 125.0                    | 8000                 |
| m3.xlarge  | 500         | 62.5                     | 4000                 |
| m3.2xlarge | 1000        | 125.0                    | 8000                 |
| r3.xlarge  | 500         | 62.5                     | 4000                 |
| r3.2xlarge | 1000        | 125.0                    | 8000                 |
| r3.4xlarge | 2000        | 250.0                    | 16000                |

i2.8xlarge、c3.8xlarge 和 r3.8xlarge 執行個體不具備專屬 EBS 頻寬，因此不提供 EBS 最佳化。在這些執行個體上，網路流量和 Amazon EBS 流量將共用同一個 10 GB 網路介面。

## 獲得最大效能

您可以使用 `EBSIOBalance%` 和 `EBSByteBalance%` 指標來協助您判斷執行個體的大小是否正確。您可以在 CloudWatch 主控台中檢視這些指標，並設定根據您指定的閾值觸發的警示。這些指標以百分比表示。維持低平衡百分比的執行個體視為有待擴展規模。平衡百分比從未低於 100% 的執行個體視為有待縮減規模。如需詳細資訊，請參閱 [使用監控執行個體 CloudWatch](#)。

此記憶體密集執行個體專為在雲端執行大型記憶體內資料庫而設計，包括 SAP HANA 記憶體內資料庫的生產部署。若要將 EBS 效能最大化，請使用具有偶數數量 `io1` 或 `io2` 磁碟區且有完全相同佈建效能的高記憶體執行個體。例如，對於 IOPS 繁重的工作負載，使用具有 40,000 個佈建 IOPS 的四個 `io1` 或 `io2` 磁碟區，以獲得最大 160,000 個執行個體 IOPS。同樣地，對於輸送量繁重的工作負載，使用具有具有 48,000 佈建 IOPS 的六個 `io1` 或 `io2` 磁碟區，以獲得最大 4,750 MB/s 輸送量。如需其他建議，請參閱 [SAP HANA 的儲存組態](#)。

### 考量事項

- 在 2020 年 2 月 26 日之後啟動的 G4dn、I3en、Inf1、M5a、M5ad、R5a、R5ad、T3、T3a 和 Z1d 執行個體可提供上表所列的最大效能。若要從 2020 年 2 月 26 日之前啟動的執行個體獲得最大效能，請停止並啟動該執行個體。
- 2019 年 12 月 3 日之後推出的 C5、C5d、C5n、M5、M5d、M5n、M5dn、R5、R5d、R5n、R5dn 和 P3dn 執行個體可提供上表所列的最大效能。若要從 2019 年 12 月 3 日之前啟動的執行個體獲得最大效能，請停止並啟動該執行個體。
- 2020 年 3 月 12 日之後啟動的 `u-6tb1.metal`、`u-9tb1.metal` 和 `u-12tb1.metal` 執行個體可提供上表中的效能。2020 年 3 月 12 日之前推出的這些類型執行個體可能會提供較低的效能。若要從 2020 年 3 月 12 日之前啟動的執行個體獲得最大效能，請聯絡您的客戶團隊以免費升級該執行個體。

## 檢視支援 EBS 最佳化的執行個體類型

您可以使用檢 AWS CLI 視目前區域中支援 EBS 最佳化的執行個體類型。

若要檢視支援 EBS 最佳化且已預設啟用的執行個體類型

使用下列 [describe-instance-types](#) 命令。如果在 Windows 命令提示字元中執行此命令，請以 `^` 字元取代 `\` 行接續字元。

```
aws ec2 describe-instance-types \
--query 'InstanceTypes[].{InstanceType:InstanceType,"MaxBandwidth(Mb/
s)":EbsInfo.EbsOptimizedInfo.MaximumBandwidthInMbps,MaxIOPS:EbsInfo.EbsOptimizedInfo.MaximumIops):EbsInfo.EbsOptimizedInfo.MaximumThroughputInMBps}' \
--filters Name=ebs-info.ebs-optimized-support,Values=default --output=table
```

eu-west-1 的輸出範例：

```

| DescribeInstanceTypes |
+-----+-----+-----+-----+
| InstanceType | MaxBandwidth(Mb/s) | MaxIOPS | MaxThroughput(MB/s) |
+-----+-----+-----+-----+
m5dn.8xlarge	6800	30000	850.0
m6gd.xlarge	4750	20000	593.75
c4.4xlarge	2000	16000	250.0
r4.16xlarge	14000	75000	1750.0
m5ad.large	2880	16000	360.0
...			

```

若要檢視支援 EBS 最佳化但預設未啟用的執行個體類型

使用下列 [describe-instance-types](#) 命令。

```
aws ec2 describe-instance-types \
--query 'InstanceTypes[].{InstanceType:InstanceType,"MaxBandwidth(Mb/
s)":EbsInfo.EbsOptimizedInfo.MaximumBandwidthInMbps,MaxIOPS:EbsInfo.EbsOptimizedInfo.MaximumIops):EbsInfo.EbsOptimizedInfo.MaximumThroughputInMBps}' \
--filters Name=ebs-info.ebs-optimized-support,Values=supported --output=table
```

eu-west-1 的輸出範例：

```

| DescribeInstanceTypes |
+-----+-----+-----+-----+
| InstanceType | MaxBandwidth(Mb/s) | MaxIOPS | MaxThroughput(MB/s) |
+-----+-----+-----+-----+
i2.2xlarge	1000	8000	125.0
m2.4xlarge	1000	8000	125.0
m2.2xlarge	500	4000	62.5
c1.xlarge	1000	8000	125.0

```

|            |      |       |       |
|------------|------|-------|-------|
| i2.xlarge  | 500  | 4000  | 62.5  |
| m3.xlarge  | 500  | 4000  | 62.5  |
| m1.xlarge  | 1000 | 8000  | 125.0 |
| r3.4xlarge | 2000 | 16000 | 250.0 |
| r3.2xlarge | 1000 | 8000  | 125.0 |
| c3.xlarge  | 500  | 4000  | 62.5  |
| m3.2xlarge | 1000 | 8000  | 125.0 |
| r3.xlarge  | 500  | 4000  | 62.5  |
| i2.4xlarge | 2000 | 16000 | 250.0 |
| c3.4xlarge | 2000 | 16000 | 250.0 |
| c3.2xlarge | 1000 | 8000  | 125.0 |
| m1.large   | 500  | 4000  | 62.5  |

## 在啟動時啟用 EBS 最佳化

若要啟用執行個體的最佳化，請設定其 EBS 最佳化屬性。

使用主控台在執行個體啟動時啟用 Amazon EBS 最佳化

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 選擇 Launch Instance (啟動執行個體)。
3. 在 Step 1: Choose an Amazon Machine Image (AMI) (步驟 1：選取 Amazon Machine Image (AMI)) 中，選取 AMI。
4. 在 Step 2: Choose an Instance Type (步驟 2：選取執行個體類型) 中，選取列為支援 Amazon EBS 最佳化的執行個體類型。
5. 在 Step 3: Configure Instance Details (步驟 3：設定執行個體詳細資訊) 中，填寫需要的欄位，然後選擇 Launch as EBS-optimized instance (啟動為 EBS 最佳化執行個體)。如果您在先前步驟中選取的執行個體類型不支援 Amazon EBS 最佳化，則不會顯示此選項。若您選取的執行個體類型預設為 Amazon EBS 最佳化，此選項便會處於選取狀態，且您將無法取消選取。
6. 請依照指示完成精靈並啟動您的執行個體。

使用命令列在執行個體啟動時啟用 EBS 最佳化

您可以使用下列其中一個命令，搭配對應的選項。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [run-instances](#) 搭配 `--ebs-optimized` (AWS CLI)
- [New-EC2Instance](#) 與 `-EbsOptimized` ( AWS Tools for Windows PowerShell )

## 啟用現有執行個體的 EBS 最佳化

若要啟用或停用現有執行個體的最佳化，請修改其 Amazon EBS 最佳化執行個體屬性。如果執行個體正在執行，您必須先停止它。

### Warning

停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區的資料，請將資料備份至持久性儲存。

使用主控台為現有執行個體啟用 EBS 最佳化

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取該執行個體。
3. 若要停止執行個體，選擇 Actions (動作)、Instance state (執行個體狀態)、Stop instance (停止執行個體)。停止執行個體可能需要幾分鐘。
4. 在仍然選取執行個體的情況下，選取 Actions (動作)、Instance settings (執行個體設定)、Change instance type (變更執行個體類型)。
5. 針對 Change Instance Type (變更執行個體類型)，執行下列其中一項作業：
  - 如果您選取的執行個體類型預設為 Amazon EBS 最佳化，則 EBS-optimized (EBS 最佳化) 便會處於選取狀態，而且您無法變更它。您可選擇 Cancel (取消)，因為該執行個體已啟用 Amazon EBS 最佳化。
  - 若您執行個體的執行個體類型支援 Amazon EBS 最佳化，請選擇 EBS-optimized (EBS 最佳化)，然後選擇 Apply (套用)。
  - 若您執行個體的執行個體類型不支援 Amazon EBS 最佳化，您將無法選擇 EBS-optimized (EBS 最佳化)。您可從 Instance type (執行個體類型) 中選取支援 Amazon EBS 最佳化的執行個體類型、選取 EBS-optimized (EBS 最佳化)，然後選取 Apply (套用)。
6. 選擇 Instance state (執行個體狀態)、Start instance (啟動執行個體)。

使用命令列為現有執行個體啟用 EBS 最佳化

1. 如果執行個體正在執行，請使用下列其中一個命令來停止它：
  - [stop-instances](#) (AWS CLI)
  - [Stop-EC2Instance](#) (AWS Tools for Windows PowerShell)

2. 若要啟用 EBS 最佳化，請使用下列其中一個命令搭配對應的選項：

- [modify-instance-attribute](#) 搭配 `--ebs-optimized` (AWS CLI)
- [Edit-EC2InstanceAttribute](#) 與 `-EbsOptimized` ( AWS Tools for Windows PowerShell )

## 執行個體購買選項

Amazon EC2 提供下列購買選項，可讓您根據需求選擇最適合的成本：

- [隨需執行個體](#) – 對於啟動的執行個體，按秒數付費。
- [Savings Plans](#) – 對一致的使用量設定綁約來降低 Amazon EC2 成本，每小時以美金計價，為期一年或三年。
- [預留執行個體](#) – 透過設定綁約至一致的執行個體設定來降低 Amazon EC2 的成本，包含執行個體類型和區域，為期一年或三年。
- [Spot 執行個體](#) – 請求未使用的 EC2 執行個體，可大幅降低您的 Amazon EC2 成本。
- [專用執行個體](#) – 付費使用專供您執行執行個體的實體主機，並使用您現有的每個通訊端、每個核心或每個 VM 軟體的授權，以降低成本。
- [專用執行個體](#) – 依時數為單一租戶硬體上執行的執行個體付費。
- [容量保留](#) – 為特定可用區域中的 EC2 執行個體保留容量。

如果您無法對特定執行個體組態做出承諾，但可以承諾使用量，請購買 Savings Plans 以降低隨需執行個體的成本。如果您需要容量預留，請針對特定可用區域購買預留執行個體或容量預留。容量區塊可用於保留 GPU 執行個體叢集。如能彈性決定應用程式執行的時機以及應用程式是否可以中斷，則 Spot 執行個體會是一項具成本效益的選擇。專用執行個體或專用執行個體可協助您處理合規性要求，並使用您現有的伺服器軟體授權來減少成本。如需詳細資訊，請參閱 [Amazon EC2 定價](#)。

如需 Savings Plans 的詳細資訊，請參閱 [Savings Plans 使用者指南](#)。

內容

- [判斷執行個體生命週期](#)
- [On-Demand Instances](#)
- [Reserved Instances](#)
- [Spot 執行個體](#)
- [Dedicated Hosts](#)
- [專用執行個體](#)

- [Capacity Reservations](#)

## 判斷執行個體生命週期

執行個體啟動時即為生命週期開始時，終止後則生命週期結束。您選擇的購買選項將影響執行個體的生命週期。例如，隨需執行個體 啟動後即開始執行，終止後則結束。只要有容量可用，Spot 執行個體便會持續執行，因此您的最高價會高於 Spot 價格。

請用下列其中一個方法來確定執行個體的生命週期。

使用主控台判斷執行個體的生命週期

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取實例。
4. 在 Details (詳細資料) 標籤的 Instance details (執行個體詳細資料) 下，尋找 Lifecycle (生命週期)。若該值為 spot，表示該執行個體為 Spot 執行個體。若該值為 normal，表示該執行個體為隨需執行個體 或 預留執行個體。
5. 在 Details (詳細資料) 標籤的 Host and placement group (主機與置放群組) 下，尋找 Tenancy (租用戶)。若該值為 host，表示該執行個體在 專用執行個體 上執行。若該值為 dedicated，表示該執行個體在 專用執行個體 上執行。

若要使用 AWS CLI

使用下列 [describe-instances](#) 命令：

```
aws ec2 describe-instances --instance-ids i-1234567890abcdef0
```

如果執行個體在 專用執行個體 上執行，其輸出將包含下列資訊：

```
"Tenancy": "host"
```

如果執行個體為 專用執行個體，其輸出將包含下列資訊：

```
"Tenancy": "dedicated"
```

如果執行個體為 Spot 執行個體，其輸出將包含下列資訊：

```
"InstanceLifecycle": "spot"
```

否則，輸出將不包含 InstanceLifecycle。

## On-Demand Instances

透過隨需執行個體，您只需要按秒數支付運算容量開銷，無需簽訂長期合約。您可以完整控制執行個體的生命週期 — 您決定何時啟動、停止、休眠、開始、重新開機或終止它。

購買隨需執行個體時，無需簽訂長期合約。您僅需按隨需執行個體處於 running 狀態的秒數付費，最少 60 秒。執行中隨需執行個體的每秒小時價格是固定的，列在 [Amazon EC2 定價、隨需定價頁面](#) 上。

我們建議您將隨需執行個體用於具有短期、不定期但無法中斷之工作負載的應用程式。

若要大幅節省隨需執行個體，請使用 [AWS Savings Plans](#)、[Spot 執行個體](#) 或 [Reserved Instances](#)。

### 內容

- [隨需執行個體配額](#)
  - [監控隨需執行個體配額和用量](#)
  - [請求提高配額](#)
- [查詢隨需執行個體的價格](#)

### 隨需執行個體配額

每 AWS 帳戶 個區域的執行隨需執行個體數量都有配額。隨需執行個體配額是依照執行中隨需執行個體正在使用的虛擬中央處理單元 (vCPU) 的數量來進行管理，與執行個體類型無關。每個配額類型可指定一個或多個執行個體系列的 vCPU 數量上限。

您的帳戶包含下列隨需執行個體配額。配額僅適用於執行中的執行個體。如果您的執行個體處於擱置中、停止、停止或休眠狀態，則不會計入您的配額中。

| 名稱            | 預設 | 可調整               |
|---------------|----|-------------------|
| 執行中隨需 DL 執行個體 | 0  | <a href="#">是</a> |
| 執行中隨需 F 執行個體  | 0  | <a href="#">是</a> |



| 名稱                                | 預設 | 可調整               |
|-----------------------------------|----|-------------------|
| 執行中隨需 G 和 VT 執行個體                 | 0  | <a href="#">是</a> |
| 執行中隨需 HPC 執行個體                    | 0  | <a href="#">是</a> |
| 執行中隨需記憶體密集型執行個體                   | 0  | <a href="#">是</a> |
| 執行中隨需 Inf 執行個體                    | 0  | <a href="#">是</a> |
| 執行中隨需 P 執行個體                      | 0  | <a href="#">是</a> |
| 執行中隨需的標準 (A、C、D、H、I、M、R、T、Z) 執行個體 | 5  | <a href="#">是</a> |
| 執行中隨需 Trn 執行個體                    | 0  | <a href="#">是</a> |
| 執行中隨需 X 執行個體                      | 0  | <a href="#">是</a> |

如需不同執行個體系列、世代和大小的相關資訊，請參閱 [Amazon EC2 執行個體類型指南](#)。

只要 vCPU 的數量不超過您的帳戶配額，就可以啟動滿足不斷變化之應用程式需求的任意執行個體類型組合。例如，在 256 個 vCPU 的標準執行個體配額下，您可以啟動 32 個 m5.2xlarge 執行個體 (32 x 8 個 vCPU) 或 16 個 c5.4xlarge 執行個體 (16 x 16 個 vCPU)。如需詳細資訊，請參閱 [EC2 隨需執行個體限制](#)。

## 任務

- [監控隨需執行個體配額和用量](#)
- [請求提高配額](#)

### 監控隨需執行個體配額和用量

可以使用下列方法檢視和管理每個區域的隨需執行個體配額。

使用 Service Quotas 主控台檢視目前的配額

1. 開啟 Service Quotas 主控台，網址為 <https://console.aws.amazon.com/servicequotas/home/services/ec2/quotas/>。
2. 從導覽列上，選取一個區域。

3. 在篩選條件欄位中，輸入 **On-Demand**。
4. 套用的配額值欄會顯示帳戶中每個隨需執行個體配額類型的 vCPU 數目上限。

使用 AWS Trusted Advisor 主控台檢視目前的配額

在 AWS Trusted Advisor 主控台中開啟 [\[服務限制\] 頁面](#)。

若要設定 CloudWatch 鬧鐘

透過 Amazon CloudWatch 指標整合，您可以根據配額監控 EC2 使用量。您也可以設定警示，在接近配額時提出警告。如需詳細資訊，請參閱 [Service Quotas 使用指南中的 Service Quotas 和 Amazon CloudWatch 警示](#)。

請求提高配額

即使 Amazon EC2 根據使用量自動提高隨需執行個體配額，您也可以在必需要時要求提高配額。例如，如果您打算啟動的執行個體超過目前配額允許的數量，則可以使用或 [Amazon EC2 服務配額](#) 中所述的 Service Quotas 主控台來請求提高配額。

查詢隨需執行個體的價格

您可以使用價目表服務 API 或 AWS 價目表 API 來查詢隨需執行個體的價格。如需詳細資訊，請參閱 [使用指南中的「使 AWS Billing 用 AWS 價目表 API」](#)。

## Reserved Instances

### Important

我們建議 Savings Plans 而非預留執行個體。如同預留執行個體一樣，儲蓄計劃是最簡單且最靈活的方式，可節省 AWS 運算成本並提供更低的價格（最高可享隨需定價 72% 的費用）。但是，Savings Plans 與預留執行個體不同。使用預留執行個體時，您可以承諾使用特定的執行個體組態，而 Savings Plans 則可以彈性使用最符合您需求的執行個體組態。要使用 Savings Plans，您需要承諾一致的使用金額，以每小時美元為單位。如需詳細資訊，請參閱 [AWS Savings Plans 使用者指南](#)。

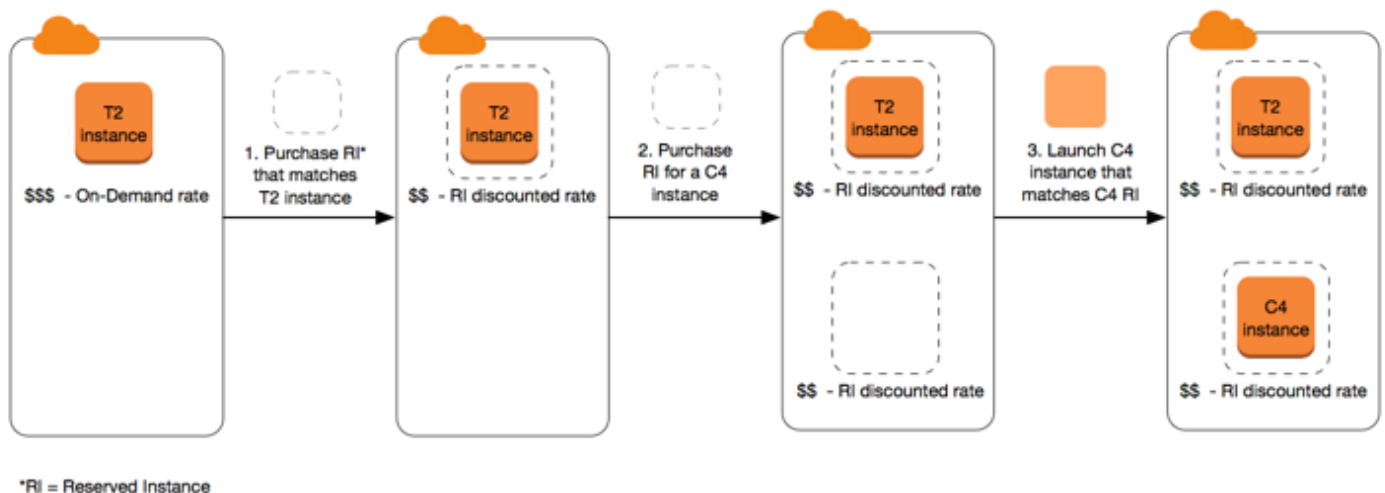
相較於隨需資料庫執行個體的定價，預留執行個體可大幅節省 Amazon EC2 成本。預留執行個體並非實體執行個體，而是一種套用到您帳戶中隨需執行個體用量的計費折扣。這些隨需執行個體必須符合特定的屬性，像是執行個體類型和區域，才能受益於計費折扣。

## 預留執行個體主題

- [預留執行個體 概觀](#)
- [決定預留執行個體定價的重要變數](#)
- [地區和區域預留執行個體 \(範圍\)](#)
- [預留執行個體類型 \(優惠類別\)](#)
- [預留執行個體的套用方式](#)
- [使用您的 預留執行個體](#)
- [計費方式](#)
- [購買預留執行個體](#)
- [在預留執行個體 Marketplace 銷售](#)
- [修改 預留執行個體](#)
- [交換 可轉換預留執行個體](#)
- [預留執行個體配額](#)

## 預留執行個體 概觀

下圖顯示購買及使用預留執行個體的基本概觀。



在此案例中，您的帳戶中會有一個執行中且目前以隨需費率付款的隨需執行個體 (T2)。您購買符合您執行中執行個體屬性的預留執行個體，即會立即套用計費利益。接下來，您會為 C4 執行個體購買預留執行個體。您的帳戶中沒有任何符合此預留執行個體屬性的執行中執行個體。在最終步驟中，您會啟動符合 C4 預留執行個體屬性的執行個體，並會立即套用計費利益。

## 決定預留執行個體定價的重要變數

預留執行個體定價取決於以下重要變數。

### 執行個體屬性

預留執行個體有四個執行個體屬性可決定其價格。

- **執行個體類型**：例如，`m4.large`。這是由執行個體系列 (例如 `m4`) 和執行個體大小 (例如 `large`) 所組成。
- **區域**：購買預留執行個體的區域。
- **租用**：您的執行個體是在共享 (預設) 或是單一租戶 (專用) 硬體上執行。如需詳細資訊，請參閱 [專用執行個體](#)。
- **平台**：作業系統，例如 Windows 或 Linux/Unix。如需詳細資訊，請參閱 [選擇平台](#)。

### 期限承諾

您可以購買一年或三年承諾的預留執行個體，三年承諾提供更大折扣。

- **一年**：一年定義為 31536000 秒 (365 天)。
- **三年**：三年定義為 94608000 秒 (1095 天)。

預留執行個體不會自動續訂。當它們到期時，您可以繼續使用 EC2 執行個體而無須中斷，但您將需要支付隨需費率的費用。在以上的範例中，當涵蓋 T2 和 C4 執行個體的預留執行個體到期時，您會重新開始支付隨需費率，直到您終止執行個體或購買符合執行個體屬性的新預留執行個體。

#### Important

在您購買預留執行個體之後，您便無法取消購買。但是，若您需要變更，您可以[修改](#)、[交換](#)或[販售](#)您的預留執行個體。

### 付款選項

預留執行個體有下列付款選項：

- **全預付款**：期間開始時便支付完整的款項，並在期間的剩餘部分不會產生其他成本或額外的每小時費用，無論使用多少小時。

- 部分預付款：必須預先支付其中一部分的成本，期間內剩餘的小時數則會以折扣後的每小時費率計費，無論是否有使用預留執行個體。
- 無預付款：您在期間中的每個小時都會以折扣後的每小時費率計費，無論是否有使用預留執行個體。不需要預付款。

#### Note

無預付款預留執行個體是以合約義務為基礎，針對保留的完整期間每月付款。基於此原因，在您購買無預付款預留執行個體前，將需要成功的帳單歷史記錄。

一般來說，為預留執行個體支付較高的預付款可以節省更多費用。您也可以預留執行個體 Marketplace 上尋找由第三方提供、價格較低且有效期間長度較短的預留執行個體。如需詳細資訊，請參閱 [在預留執行個體 Marketplace 銷售](#)。

#### 方案類別

若您的運算需求變更，根據優惠類別，您可能可以修改或交換您的預留執行個體。

- 標準：這些提供更多的折扣，但只能修改。標準版預留執行個體無法交換。
- 可轉換：這些提供比標準預留執行個體更少的折扣，但可以與具有不同執行個體屬性的另一個可轉換預留執行個體交換。可轉換預留執行個體亦可修改。

如需詳細資訊，請參閱 [預留執行個體類型 \(優惠類別\)](#)。

#### Important

在您購買預留執行個體之後，您便無法取消購買。但是，若您需要變更，您可以[修改](#)、[交換](#)或[販售](#)您的預留執行個體。

如需詳細資訊，請參閱 [Amazon EC2 預留執行個體定價頁面](#)。

#### 地區和區域預留執行個體 (範圍)

您購買預留執行個體時會決定預留執行個體的範圍。範圍是地區或區域。

- 地區：當您為某一地區購買預留執行個體時，它便稱為地區預留執行個體。

- **區域**：當您為某一特定可用區域購買預留執行個體時，它便稱為區域預留執行個體。

範圍不影響價格。您為區域或區域預留執行個體支付相同的價格。如需預留執行個體定價的詳細資訊，請參閱 [決定預留執行個體定價的重要變數](#) 和 [Amazon EC2 預留執行個體定價](#)。

如需有關如何指定預留執行個體範圍的詳細資訊，請參閱 [RI 屬性](#)，特別是可用區域項目符號。

### 地區與區域預留執行個體之間的差異

下表特別列出地區預留執行個體與區域預留執行個體之間的一些主要差異：

|           | 地區預留執行個體                                                                                                                       | 區域預留執行個體                                     |
|-----------|--------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------|
| 能夠預留容量    | 區域預留執行個體不會預留容量。                                                                                                                | 指定可用區域中的區域預留執行個體會預留容量。                       |
| 可用區域彈性    | 預留執行個體折扣會套用到指定區域內任何可用區域中的執行個體用量。                                                                                               | 無可用區域彈性 – 預留執行個體折扣僅套用到指定可用區域中的執行個體用量。        |
| 執行個體大小彈性  | 預留執行個體折扣會套用到執行個體系列中的執行個體用量，而無論大小。<br><br>只在使用預設租用的 Amazon Linux/Unix 預留執行個體上才支援。如需詳細資訊，請參閱 <a href="#">由正規化因素決定的執行個體大小彈性</a> 。 | 無執行個體大小彈性 – 預留執行個體折扣僅套用到指定之執行個體類型和大小的執行個體用量。 |
| 將購買項目放入佇列 | 您可以將區域預留執行個體的購買項目放入佇列。                                                                                                         | 您無法將區域預留執行個體的購買項目放入佇列。                       |

如需詳細資訊和範例，請參閱 [預留執行個體的套用方式](#)。

## 預留執行個體類型 (優惠類別)

預留執行個體 的產品類別為標準版或可轉換版。標準版的 預留執行個體 提供的折扣比可轉換版的 預留執行個體 更多，但您無法交換標準版的 預留執行個體。您可以交換可轉換版的 預留執行個體。您可以修改標準版和可轉換版的 預留執行個體。

預留執行個體 的組態會在期間內包含單一執行個體類型、平台、範圍和租用。若您的運算需求變更，您可能可以修改或交換您的 預留執行個體。

### 標準版與可轉換版 預留執行個體 的差異

以下為標準版和可轉換版 預留執行個體 的不同。

|                        | 標準預留執行個體                                        | Convertible Reserved Instance                                                                      |
|------------------------|-------------------------------------------------|----------------------------------------------------------------------------------------------------|
| 修改 預留執行個體              | 可以修改某些屬性。如需詳細資訊，請參閱 <a href="#">修改 預留執行個體</a> 。 | 可以修改某些屬性。如需詳細資訊，請參閱 <a href="#">修改 預留執行個體</a> 。                                                    |
| 交換預留執行個體               | 無法交換。                                           | 可使用新的屬性 (包含執行個體系列、執行個體類型、平台、範圍或租用) 在另一個 可轉換預留執行個體 的期間內交換。如需詳細資訊，請參閱 <a href="#">交換 可轉換預留執行個體</a> 。 |
| 在預留執行個體 Marketplace 銷售 | 可在預留執行個體 Marketplace 銷售。                        | 不可在預留執行個體 Marketplace 銷售。                                                                          |
| 在預留執行個體 Marketplace 購買 | 可在預留執行個體 Marketplace 購買。                        | 不可在預留執行個體 Marketplace 購買。                                                                          |

## 預留執行個體的套用方式

預留執行個體並非實體執行個體，而是一種套用到您帳戶中執行中隨需執行個體的計費折扣。隨需執行個體必須符合預留執行個體的特定規格，才能得利於計費折扣。

如果您購買預留執行個體並且已有符合預留執行個體規格的執行中預留執行個體，即會立即並自動套用計費折扣。您不需要重新啟動您的執行個體。如果您沒有符合資格的執行中隨需執行個體，請啟動具有與預留執行個體相同規格的隨需執行個體。如需詳細資訊，請參閱 [使用您的 預留執行個體](#)。

預留執行個體的方案類別 (標準或可轉換) 不會影響計費折扣的套用方式。

## 主題

- [區域預留執行個體的套用方式](#)
- [區域預留執行個體的套用方式](#)
- [執行個體大小彈性](#)
- [套用預留執行個體的範例](#)

## 區域預留執行個體的套用方式

為了在特定可用區域預留容量而購買的預留執行個體稱為區域預留執行個體。

- 預留執行個體折扣會套用到該可用區域中相符的執行個體用量。
- 執行中執行個體的屬性 (租用、平台、可用區域、執行個體類型和執行個體大小) 必須符合該預留執行個體的屬性。

例如，如果您為可用區域 us-east-1a 購買兩個 c4.xlarge 預設租用 Linux/Unix 標準預留執行個體，則最多能有兩個在可用區域 us-east-1a 中執行的 c4.xlarge 預設租用 Linux/Unix 執行個體獲得預留執行個體折扣帶來的利益。

## 區域預留執行個體的套用方式

為區域購買的預留執行個體稱為區域預留執行個體，並會提供可用區域和執行個體大小彈性。

- 預留執行個體折扣會套用到該區域內任何可用區域中的執行個體用量。
- 預留執行個體折扣會套用到執行個體系列中的執行個體用量，而無論大小—這稱為 [執行個體大小彈性](#)。

## 執行個體大小彈性

由於執行個體大小彈性的因素，執行個體專屬折扣適用於具相同 [系列、世代和屬性](#) 的執行個體使用。根據正規化因素，預留執行個體先套用至執行個體系列中最小的執行個體，接著逐步套用至最大的執行個體。有關如何套用預留執行個體折扣的範例，請參閱 [案例 2：使用正規化因素的單個帳戶中的預留執行個體](#)。



## 限制

- 支援：僅區域預留執行個體支援執行個體大小彈性。
- 不支援：下列預留執行個體不支援執行個體大小彈性：
  - 為特定可用區域購買的預留執行個體 (區域預留執行個體)
  - G4ad、G4dn、G5、G5G、InF1 和 In2 執行個體的預留執行個體
  - 預留執行個體 適用於 Windows Server、Windows Server with SQL Standard、Windows Server with SQL Server Enterprise、Windows Server with SQL Server Web、RHEL 和 SUSE Linux Enterprise Server
  - 搭配專用租用的預留執行個體

### 由正規化因素決定的執行個體大小彈性

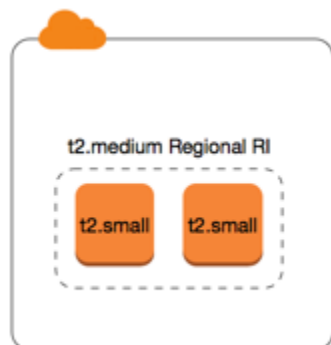
執行個體大小彈性取決於執行個體大小的正規化因素。在區域內的任何可用區域中，折扣可完整或部分套用到相同執行個體系列的執行中執行個體，而這取決於保留的執行個體大小而定。只要執行個體系列、租用和平台三個屬性符合即可。

下表列出執行個體系列中不同的大小，以及其對應的正規化因素。此規模會用於將預留執行個體折扣費率套用到執行個體系列的標準化用量。

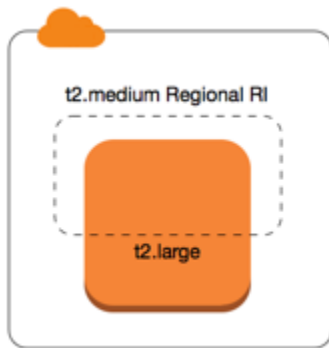
| 執行個體大小  | 正規化因素 |
|---------|-------|
| nano    | 0.25  |
| micro   | 0.5   |
| small   | 1     |
| medium  | 2     |
| large   | 4     |
| xlarge  | 8     |
| 2xlarge | 16    |
| 3xlarge | 24    |
| 4xlarge | 32    |

| 執行個體大小    | 正規化因素 |
|-----------|-------|
| 6xlarge   | 48    |
| 8xlarge   | 64    |
| 9xlarge   | 72    |
| 10xlarge  | 80    |
| 12xlarge  | 96    |
| 16xlarge  | 128   |
| 18xlarge  | 144   |
| 24xlarge  | 192   |
| 32xlarge  | 256   |
| 48xlarge  | 384   |
| 56xlarge  | 448   |
| 112xlarge | 896   |

舉例而言，t2.medium 執行個體的正規化因素為 2。若您在 US East (N. Virginia) 中購買 t2.medium 預設租用 Amazon Linux/Unix 預留執行個體，且在該區域中您的帳戶內有兩個執行中的 t2.small 執行個體，計費利益便會完整套用到兩個執行個體。



或者，若您在US East (N. Virginia) 區域中的帳戶有一個執行中的 t2.large 執行個體，計費利益便會套用到執行個體的 50% 用量。



正規化因數也會在修改預留執行個體時套用。如需詳細資訊，請參閱 [修改 預留執行個體](#)。

### 裸機執行個體的正規化因素

執行個體大小彈性也套用到執行個體系列中的裸機執行個體。如果您在裸機執行個體上有搭配共用租用的地區 Amazon Linux/Unix 預留執行個體，則可享受在相同執行個體系列中節省預留執行個體。反之亦然：如果您在與裸機執行個體相同的系列中有搭配共用租用的地區 Amazon Linux/Unix 預留執行個體，則可享受在裸機執行個體系列中節省預留執行個體。

metal 執行個體大小沒有單一正規化因素。裸機執行個體的正規化因素與相同執行個體系列中的對等虛擬化執行個體大小相同。例如，i3.metal 執行個體具有與 i3.16xlarge 執行個體相同的正規化因素。

| 執行個體大小                                                                                                                                              | 正規化因素 |
|-----------------------------------------------------------------------------------------------------------------------------------------------------|-------|
| a1.metal                                                                                                                                            | 32    |
| m5zn.metal   x2iezn.metal   z1d.metal                                                                                                               | 96    |
| c6g.metal   c6gd.metal   i3.metal   m6g.metal   m6gd.metal<br>  r6g.metal   r6gd.metal   x2gd.metal                                                 | 128   |
| c5n.metal                                                                                                                                           | 144   |
| c5.metal   c5d.metal   i3en.metal   m5.metal   m5d.metal  <br>m5dn.metal   m5n.metal   r5.metal   r5b.metal   r5d.metal  <br>r5dn.metal   r5n.metal | 192   |

| 執行個體大小                                                                      | 正規化因素 |
|-----------------------------------------------------------------------------|-------|
| c6i.metal   c6id.metal   m6i.metal   m6id.metal   r6d.metal<br>  r6id.metal | 256   |
| u-*.metal                                                                   | 896   |

例如，`i3.metal` 執行個體的正規化因素為 128。如果您在 US East (N. Virginia) 購買 `i3.metal` 預設租用 Amazon Linux/Unix 預留執行個體，則計費利益可套用如下：

- 如果您在該區域中的帳戶有一個執行中的 `i3.16xlarge`，計費利益會完全套用到 `i3.16xlarge` 執行個體 (`i3.16xlarge` 正規化因素 = 128)。
- 或者，如果您在該區域中的帳戶有兩個執行中的 `i3.8xlarge` 執行個體，計費利益會完全套用到這兩個 `i3.8xlarge` 執行個體 (`i3.8xlarge` 正規化因素 = 64)。
- 或者，如果您在該區域中的帳戶有四個執行中的 `i3.4xlarge` 執行個體，計費利益會完全套用到所有這四個 `i3.4xlarge` 執行個體 (`i3.4xlarge` 正規化因素 = 32)。

反之亦然。例如，若您在 US East (N. Virginia) 購買兩個 `i3.8xlarge` 預設租用 Amazon Linux/Unix 預留執行個體，而且您在該區域中有一個執行中的 `i3.metal` 執行個體，計費利益會完全套用到 `i3.metal` 執行個體。

### 套用預留執行個體的範例

下列案例涵蓋套用預留執行個體的方式。

- [案例 1：單一帳戶中的預留執行個體](#)
- [案例 2：使用正規化因素的單個帳戶中的預留執行個體](#)
- [案例 3：連結帳戶中的區域預留執行個體](#)
- [案例 4：連結帳戶中的區域預留執行個體](#)

#### 案例 1：單一帳戶中的預留執行個體

您正在帳戶 A 中執行下列隨需執行個體：

- 4 個 `m3.large` Linux，位於 `us-east-1a` 可用區域中的預設租用執行個體
- 2 個 `m4.xlarge` Amazon Linux，位於 `us-east-1b` 可用區域中的預設租用執行個體
- 1 個 `c4.xlarge` Amazon Linux，位於 `us-east-1c` 可用區域中的預設租用執行個體

您在帳戶 A 中購買下列預留執行個體：

- 4 個 m3.large Linux，位於 us-east-1a 可用區域中的預設租用預留執行個體 (容量已預留)
- 4 個 m4.large Amazon Linux，位於 us-east-1 區域中的預設租用預留執行個體
- 1 個 c4.large Amazon Linux，位於 us-east-1 區域中的預設租用預留執行個體

預留執行個體利益會以下列方式套用：

- 四個 m3.large 執行個體會使用四個 m3.large 區域預留執行個體的折扣和容量保留，因為它們之間的屬性 (執行個體大小、區域、平台、租用) 相符。
- m4.large 區域預留執行個體可提供可用區域和執行個體大小彈性，因為它們是使用預設租用的區域 Amazon Linux 預留執行個體。

m4.large 相當於 4 個標準化單位/小時。

您已購買四個 m4.large 區域預留執行個體，總共等於 16 個標準化單位/小時 (4x4)。帳戶 A 擁有兩個執行中的 m4.xlarge 執行個體，相當於 16 個標準化單位/小時 (2x8)。在此案例中，四個 m4.large 區域預留執行個體可為兩個 m4.xlarge 執行個體提供完整用量的計費利益。

- 位於 us-east-1 中的 c4.large 區域預留執行個體可提供可用區域和執行個體大小彈性，因為它是使用預設租用的區域 Amazon Linux 預留執行個體，並會套用到 c4.xlarge 執行個體。c4.large 執行個體相當於 4 個標準化單位/小時，c4.xlarge 則相當於 8 個標準化單位/小時。

在此案例中，c4.large 區域預留執行個體會為 c4.xlarge 用量提供部分利益。這是因為 c4.large 預留執行個體相當於 4 個標準化單位/小時用量，但 c4.xlarge 執行個體需要 8 個標準化單位/小時。因此，c4.large 預留執行個體計費折扣會套用到 50% 的 c4.xlarge 用量。剩餘的 c4.xlarge 用量會以隨需費率收費。

案例 2：使用正規化因素的單個帳戶中的預留執行個體

您正在帳戶 A 中執行下列隨需執行個體：

- 2 個 m3.xlarge Amazon Linux，位於可用區域 us-east-1a 中的預設租用執行個體
- 2 個 m3.large Amazon Linux，位於 us-east-1b 可用區域中的預設租用執行個體

您在帳戶 A 中購買下列預留執行個體：

- 1 個 m3.2xlarge Amazon Linux，位於 us-east-1 區域中的預設租用預留執行個體

預留執行個體利益會以下列方式套用：

- 位於 us-east-1 中的 m3.2xlarge 區域預留執行個體可提供可用區域和執行個體大小彈性，因為它是使用預設租用的區域 Amazon Linux 預留執行個體。它首先套用至 m3.large 執行個體，然後套用至 m3.xlarge 執行個體，因為它會根據正規化因素，先套用至執行個體系列中最小的執行個體，接著逐步套用至最大的執行個體。

m3.large 執行個體相當於 4 個標準化單位/小時。

m3.xlarge 執行個體相當於 8 個標準化單位/小時。

m3.2xlarge 執行個體相當於 16 個標準化單位/小時。

此利益適用於如下情況：

m3.2xlarge 區域預留執行個體為 2 x m3.large 用量提供全部利益，因為這些執行個體共佔用 8 個標準化單位/小時。這將留下 8 個標準化單位/小時來套用至 m3.xlarge 執行個體。

藉助剩餘的 8 個標準化單位/小時，m3.2xlarge 區域預留執行個體可為 1 x m3.xlarge 用量提供全部利益，因為每個 m3.xlarge 執行個體相當於 8 個標準化單位/小時。剩餘的 m3.xlarge 用量會以隨需費率收費。

### 案例 3：連結帳戶中的區域預留執行個體

預留執行個體會先套用到購買帳戶中的用量，之後則是組織內任何其他帳戶內的合格用量。如需詳細資訊，請參閱 [預留執行個體與合併帳單](#)。針對提供執行個體大小彈性的區域預留執行個體，利益會先套用至執行個體系列中最小的執行個體，接著逐步套用至最大的執行個體。

您正在帳戶 A (購買帳戶) 中執行下列隨需執行個體：

- 2 個 m4.xlarge Linux，位於 us-east-1a 可用區域中的預設租用執行個體
- 1 個 m4.2xlarge Linux，位於 us-east-1b 可用區域中的預設租用執行個體
- 2 個 c4.xlarge Linux，位於 us-east-1a 可用區域中的預設租用執行個體
- 1 個 c4.2xlarge Linux，位於 us-east-1b 可用區域中的預設租用執行個體

另一個客戶正在帳戶 B – 一個連結帳戶中執行下列隨需執行個體：

- 2 個 m4.xlarge Linux，位於 us-east-1a 可用區域中的預設租用執行個體

您在帳戶 A 中購買下列區域預留執行個體：

- 4 個 m4.xlarge Linux，位於 us-east-1 區域中的預設租用預留執行個體
- 2 個 c4.xlarge Linux，位於 us-east-1 區域中的預設租用預留執行個體

地區預留執行個體利益會以下列方式套用：

- 帳戶 A (購買帳戶) 中的兩個 m4.xlarge 執行個體及單一 m4.2xlarge 執行個體會使用四個 m4.xlarge 預留執行個體的折扣。三個執行個體全部符合屬性 (執行個體系列、區域、平台、租用)。折扣會先套用到購買帳戶 (帳戶 A) 中的執行個體，即使帳戶 B (連結帳戶) 有兩個也符合預留執行個體的 m4.xlarge 也一樣。沒有容量保留，因為預留執行個體是地區預留執行個體。
- 兩個 c4.xlarge 預留執行個體的折扣會套用至兩個 c4.xlarge 執行個體，因為這兩個執行個體的大小小於 c4.2xlarge 執行個體。沒有容量保留，因為預留執行個體是地區預留執行個體。

#### 案例 4：連結帳戶中的區域預留執行個體

一般而言，某帳戶擁有的預留執行個體會先套用到該帳戶中的用量。但是，若組織中其他帳戶內有合格且未使用的特定可用區域預留執行個體 (區域性預留執行個體)，便會在套用該帳戶擁有的區域預留執行個體前套用到帳戶。這是為了確保最高的預留執行個體使用率和較低的帳單。基於計費目的，組織中的所有帳戶都會視為一個帳戶處理。以下範例可協助說明。

您正在帳戶 A (購買帳戶) 中執行下列隨需執行個體：

- 1 個位於 us-east-1a 可用區域中的 m4.xlarge Linux 預設租用執行個體

一名客戶正在連結帳戶 B 中執行下列隨需執行個體：

- 1 個位於 us-east-1b 可用區域中的 m4.xlarge Linux 預設租用執行個體

您在帳戶 A 中購買下列區域預留執行個體：

- 1 個 m4.xlarge Linux，位於 us-east-1 區域中的預設租用預留執行個體

一名客戶也在連結帳戶 C 中購買下列區域預留執行個體：

- 1 個 m4.xlarge Linux，位於 us-east-1a 可用區域中的預設租用預留執行個體

預留執行個體利益會以下列方式套用：

- 帳戶 C 擁有的 m4.xlarge 區域性預留執行個體折扣會套用到帳戶 A 中的 m4.xlarge 用量。
- 帳戶 A 擁有的 m4.xlarge 區域預留執行個體折扣會套用到帳戶 B 中的 m4.xlarge 用量。
- 若帳戶 A 擁有的區域預留執行個體先套用到帳戶 A 的用量，則帳戶 C 擁有的區域性預留執行個體便會處於未使用的狀態，且帳戶 B 的用量會以隨需費率收費。

如需詳細資訊，請參閱 [Billing and Cost Management 報告中的預留執行個體](#)。

#### Note

區域預留執行個體預留容量僅適用於擁有的帳戶，無法與其他 AWS 帳戶共用。如果您需要與其他人士共用容量 AWS 帳戶，請使用 [On-Demand Capacity Reservations](#)。

## 使用您的 預留執行個體

預留執行個體會自動套用至執行中的隨需執行個體，只要其規格相符。若您沒有符合您預留執行個體規格的執行中隨需執行個體，預留執行個體會處於未使用的狀態，直到您使用必要的規格啟動執行個體為止。

如果您要啟動隨需執行個體，以利用預留執行個體的計費利益，請確定您在設定隨需執行個體時指定下列資訊：

### 平台

您必須指定符合您預留執行個體平台 (產品描述) 的 Amazon Machine Image (AMI)。例如，如果針對您的預留執行個體指定了 Linux/UNIX，您可以從 Amazon Linux AMI 或 Ubuntu AMI 啟動執行個體。

### 執行個體類型

如果購買了區域預留執行個體，您必須指定與預留執行個體相同的執行個體類型，例如 t3.large。如需詳細資訊，請參閱 [區域預留執行個體的套用方式](#)。

如果購買了區域預留執行個體，您必須從預留執行個體之執行個體之同一執行個體系列指定執行個體類型。例如，如果針對預留執行個體指定了 t3.xlarge，您必須從 T3 系列啟動執行個體，但可以指定任何大小，例如 t3.medium。如需詳細資訊，請參閱 [區域預留執行個體的套用方式](#)。



## 可用區域

如果針對特定可用區域購買了區域預留執行個體，您必須將執行個體啟動到相同的可用區域。

如果購買了區域預留執行個體，您可以將執行個體啟動到區域中針對預留執行個體指定的任何可用區域。

## 租用

執行個體的租用 (dedicated 或 shared) 必須符合預留執行個體的租用。如需詳細資訊，請參閱 [專用執行個體](#)。

如需預留執行個體如何套用到執行中隨需執行個體的範例，請參閱 [預留執行個體的套用方式](#)。如需詳細資訊，請參閱 [為什麼 Amazon EC2 預留執行個體無法按照我預期的方式套用到我的 AWS 帳單？](#)

您可以使用各種方法，啟動使用預留執行個體折扣的隨需執行個體。如需不同啟動方法的詳細資訊，請參閱 [啟動您的執行個體](#)。您也可以使用 Amazon EC2 Auto Scaling 來啟動執行個體。如需詳細資訊，請參閱 [Amazon EC2 Auto Scaling 使用者指南](#)。

## 計費方式

所有預留執行個體都可提供您相較於隨需定價的折扣。您可透過預留執行個體，為整個期間支付費用 (無論實際用量為何)。您可以選擇為您的預留執行個體預付款項、部分預付款項或每月付款，取決於為預留執行個體指定的 [付款選項](#)。

預留執行個體;到期時，將會根據 EC2 執行個體用量的隨需費率向您收費。最早可提前三年將預留執行個體放入佇列供購買。這可協助您確保您的涵蓋範圍不會間斷。如需詳細資訊，請參閱 [將您的購買項目放入佇列](#)。

AWS 免費方案適用於新 AWS 帳戶。若您正在使用 AWS 免費方案執行 Amazon EC2 執行個體，並且您已購買預留執行個體，將會根據標準定價準則向您收費。如需資訊，請參閱 [AWS 免費方案](#)。

## 內容

- [用量帳單](#)
- [檢視您的帳單](#)
- [預留執行個體 與合併帳單](#)
- [預留執行個體折扣定價方案](#)

## 用量帳單

預留執行個體會在您選取的期間內以每時鐘小時計價，無論執行個體是否正在執行中。每個鐘點小時都會於標準 24 小時制的整點 (每小時的零分和零秒) 開始。例如，1:00:00 至 1:59:59 是一點鐘。如需執行個體狀態的詳細資訊，請參閱[執行個體生命週期](#)。

預留執行個體計費利益可以每秒套用到執行中的執行個體。以秒計費適用於使用開放原始碼 Linux 發行版本 (例如 Amazon Linux 和 Ubuntu) 的執行個體。每小時計費用於商業 Linux 發行版本，例如 Red Hat Enterprise Linux 和 SUSE Linux Enterprise Server。

預留執行個體計費利益每個時鐘小時可套用到最多 3600 秒 (一小時) 的執行個體用量。您可以同時執行多個執行個體，但每個時鐘小時最多只能接收到 3600 秒的預留執行個體折扣利益；時鐘小時內超過 3600 秒的執行個體用量會以隨需費率計費。

例如，若您購買一個 m4.xlarge 預留執行個體，並同時執行四個 m4.xlarge 執行個體一個小時，一個執行個體會以預留執行個體用量收取一個小時的費用，其餘三個執行個體則會以隨需用量收取三個小時的費用。

但是，若您購買一個 m4.xlarge 預留執行個體，並在同一個小時內執行四個 m4.xlarge 執行個體，每一個執行 15 分鐘 (900 秒)，執行個體的總執行時間便為一個小時，其結果便是一個小時的預留執行個體用量和 0 個小時的隨需用量。

|            | 1:00 | 1:15 | 1:30 | 1:45 |
|------------|------|------|------|------|
| Instance 1 |      |      |      |      |
| Instance 2 |      |      |      |      |
| Instance 3 |      |      |      |      |
| Instance 4 |      |      |      |      |

若同時執行多個符合資格的執行個體，預留執行個體計費利益會在一個時鐘小時內同時套用到所有執行個體，最多達 3600 秒；在這之後，便會套用隨需費率。

|            | 1:00 | 1:15 | 1:30 | 1:45 |
|------------|------|------|------|------|
| Instance 1 |      |      |      |      |
| Instance 2 |      |      |      |      |
| Instance 3 |      |      |      |      |
| Instance 4 |      |      |      |      |

Uses Reserved Instance Rate for first 3600 seconds of use
Uses On-Demand Rate

[Billing and Cost Management](#) 主控台上的 Cost Explorer 可讓您根據執行中的隨需執行個體分析節省費用。[預留執行個體常見問答集](#) 包括清單值計算的範例。

如果您關閉 AWS 帳戶，資源的隨需計費就會停止。但是，若您的帳戶中有任何預留執行個體，您會繼續收到其帳單，直到它們到期為止。

### 檢視您的帳單

您可以透過檢視 [AWS Billing and Cost Management](#) 主控台了解您帳戶所需支付的費用。

- Dashboard (儀表板) 會顯示您帳戶的支出摘要。
- 在 Bills (帳單) 頁面上，於 Details (詳細資訊) 下方，展開 Elastic Compute Cloud 區段及區域，以取得預留執行個體的帳單資訊。

您可在線上檢視費用，或是下載 CSV 檔案。

您也可以使用 AWS 成本和用量報告追蹤預留執行個體使用率。如需詳細資訊，請參閱 AWS Billing 使用者指南中『成本及用量報告』下的[預留執行個體](#)。

### 預留執行個體 與合併帳單

若購買帳戶是在單一合併帳單付款人帳戶下一組帳戶的一部分，則會共用預留執行個體的計費利益。所有成員帳戶的執行個體用量每個月都會彙總到付款人的帳戶中。這通常對有不同功能團隊或群組的公司來說非常有用；然後，會套用標準預留執行個體邏輯計算帳單。如需詳細資訊，請參閱 [AWS Organizations](#) 的合併帳單。

如果您關閉購買預留執行個體的帳戶，系統會繼續為預留執行個體對付款人帳戶計費，直到預留執行個體到期。在關閉的帳戶於 90 天後永久刪除後，會員帳戶無法再從預留執行個體帳單折扣中受益。

#### Note

區域預留執行個體預留容量僅適用於擁有的帳戶，無法與其他 AWS 帳戶共用。如果您需要與其他入共用容量 AWS 帳戶，請使用 [On-Demand Capacity Reservations](#)。

### 預留執行個體折扣定價方案

若您的帳戶符合折扣定價方案的資格，從您在該方案層級中購買預留執行個體的時間點開始，預付及執行個體使用費便會自動接收到折扣。若要符合折扣的資格，您在區域中的預留執行個體清單價值必須為 500,000 USD (含) 以上。

適用的規定如下：

- 定價方案和相關折扣只會套用到購買的 Amazon EC2 標準預留執行個體。
- 定價方案不會套用到 Windows with SQL Server Standard、SQL Server Web 和 SQL Server Enterprise 的預留執行個體。
- 定價方案不會套用到 Linux with SQL Server Standard、SQL Server Web 和 SQL Server Enterprise 的預留執行個體。
- 定價等級折扣僅適用於從購買的商品 AWS。它們不會套用到購買的第三方預留執行個體。
- 折扣定價方案目前無法套用到可轉換預留執行個體購買。

主題

- [計算 預留執行個體 定價折扣](#)
- [使用折扣方案購買](#)
- [跨定價方案](#)
- [定價方案的合併帳單](#)

計算 預留執行個體 定價折扣

您可以藉由計算您在區域中所有預留執行個體的清單值，來決定帳戶的定價方案。將每個保留每小時的經常性價格，乘上期間內的總小時數，並加上在購買時間點的未折扣預付價格（也稱為固定價格）。因為清單值是以未折扣（公有）定價為基礎，無論您是否符合批發折扣，或是價格是否在您購買預留執行個體之後降低，都不會受到影響。

$$\text{List value} = \text{fixed price} + (\text{undiscounted recurring hourly price} * \text{hours in term})$$

例如，針對 1 年部分預付款的 t2.small 預留執行個體，假設預付價格為 60.00 USD，每小時的費率為 0.007 USD。這會提供清單值 121.32 USD。

$$121.32 = 60.00 + (0.007 * 8760)$$

New console

使用 Amazon EC2 主控台檢視預留執行個體的固定價格值

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)。

3. 要顯示預付價格列，請選擇右上角的設置



打開預付價格，然後選擇確認。

## Old console

使用 Amazon EC2 主控台檢視預留執行個體的固定價格值

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)。
3. 要顯示預付價格列，請選擇右上角的設置



選擇預付價格，然後選擇關閉。

使用命令列檢視預留執行個體的固定價格值

- [describe-reserved-instances](#) (AWS CLI)
- [Get-EC2ReservedInstance](#) (AWS Tools for Windows PowerShell)
- [DescribeReserved執行個體](#) (Amazon EC2 API)

## 使用折扣方案購買

當您購買預留執行個體時，Amazon EC2 會自動將任何折扣套用到您購買中位於折扣定價方案內的部分。您無須進行任何不同操作，並且您可以使用任何 Amazon EC2 工具購買預留執行個體。如需詳細資訊，請參閱 [購買預留執行個體](#)。

當您在區域中的作用中預留執行個體清單值進入折扣定價方案後，日後在該區域購買的任何預留執行個體都會以折扣費率計費。若您在區域中購買的單一預留執行個體超過折扣方案的閾值，則該購買超過價格閾值的部分便會以折扣費率收費。如需在購買程序中建立之暫時預留執行個體 ID 的詳細資訊，請參閱 [跨定價方案](#)。

若您的清單值低於該折扣定價方案的價格點 (例如：若您的某些預留執行個體到期)，則日後在該區域中購買的預留執行個體將不會有折扣。但是，您原先在折扣定價方案內購買的預留執行個體，仍會繼續套用折扣。

當您購買預留執行個體時，便會發生下列四種可能案例中的其中一種：

- 無折扣 – 您在區域內的購買仍低於折扣閾值。

- 部分折扣 – 您在區域內的購買已超過第一個折扣方案的閾值。不會套用折扣到一或多個保留，剩餘的保留則會套用折扣費率。
- 完整折扣 – 您在區域內的整個購買都已符合單一折扣方案，並已取得適當折扣。
- 雙折扣費率 – 您在區域內的購買已從較低的折扣方案升等至更高的折扣方案。會根據兩種不同費率向您收費：一或多個保留會以較低的折扣費率計費，其餘保留則會以更高的折扣費率計費。

## 跨定價方案

若您的購買進入折扣定價方案，您會看到多個該購買的項目：其中一個為該購買中以標準價格收費的部分，另外一個則為該購買中以適用折扣費率收費的部分。

預留執行個體服務會產生數個預留執行個體 ID，因為您的購買跨越自未折扣方案，或是從其中一種折扣方案跨越至另一種。方案中每一組保留都會有一個 ID。因此，您購買 CLI 命令或 API 動作傳回的 ID，將會與新預留執行個體的實際 ID 不同。

## 定價方案的合併帳單

合併帳單帳戶會彙總區域內成員帳戶的清單值。當合併帳單帳戶所有作用中預留執行個體的清單值達到折扣定價方案，合併帳單帳戶的任何成員帳戶在之後購買的預留執行個體都會以折扣費率計費 (只要合併帳戶的清單值持續高於折扣定價方案的閾值)。如需詳細資訊，請參閱 [預留執行個體 與合併帳單](#)。

## 購買預留執行個體

若要購買預留執行個體，請搜尋來自 AWS 和第三方賣家的預留執行個體供應項目，然後調整搜尋參數，直到找到所需的完全相符項目為止。

當您在尋找要購買的預留執行個體時，您會收到傳回優惠的成本報價。當您繼續進行購買時，AWS 會自動對購買價格設定限制價格。您預留執行個體的總成本不會超過向您報價的金額。

若價格因為任何原因產生變更，購買便無法完成。當您從 EC2 預留執行個體 Marketplace 購買第三方賣家的預留執行個體時，如果有類似於您選擇的產品，但預付價格較低，則會以較低的前期價格向您 AWS 銷售產品。

在您確認購買前，請檢閱您計劃購買之預留執行個體的詳細資訊，並確認所有的參數都正確。在您購買預留執行個體之後 (無論是從預留執行個體 Marketplace 的第三方賣家或從 AWS)，您就無法取消購買。

若要購買和修改預留執行個體，請確保使用者具有適當的許可，例如描述可用區域的能力。如需詳細資訊，請參閱 [the section called “使用預留執行個體” \(API\)](#) 或 [the section called “使用預留執行個體” \(主控台\)](#)。

## 主題

- [選擇平台](#)
- [將您的購買項目放入佇列](#)
- [購買標準版 預留執行個體](#)
- [購買 可轉換預留執行個體](#)
- [從預留執行個體 Marketplace 購買](#)
- [檢視您的 預留執行個體](#)
- [取消佇列購買](#)
- [續約 預留執行個體](#)

## 選擇平台

Amazon EC2 針對預留執行個體支援下列平台：

- Linux/UNIX
- Linux with SQL Server Standard
- Linux with SQL Server Web
- Linux with SQL Server Enterprise
- SUSE Linux
- Red Hat Enterprise Linux
- 帶 HA 的 Red Hat Enterprise Linux
- Windows
- Windows with SQL Server Standard
- Windows with SQL Server Web
- Windows with SQL Server Enterprise

當您購買預留執行個體時，您必須選擇代表您執行個體作業系統之平台的優惠。

## Linux 執行個體

- 針對 SUSE Linux 和 RHEL 分佈，您必須選擇那些特定平台的優惠，例如 SUSE Linux 或 Red Hat Enterprise Linux 平台。

- 針對所有其他的 Linux 分佈 (包含 Ubuntu)，請選擇 Linux/UNIX 平台的優惠。
- 如果您帶著現有的 RHEL 訂閱，您必須選擇 Linux/UNIX 平台的方案，而非 Red Hat Enterprise Linux 平台的方案。

## Windows 執行個體

- 針對 Windows with SQL Standard、Windows with SQL Server Enterprise 和 Windows with SQL Server Web，您必須選擇那些特定平台的優惠。
- 針對所有其他的 Windows 版本，請選擇 Windows 平台的優惠。

### Note

Ubuntu Pro 不可作為預留執行個體使用。與隨需執行個體定價相比，可節省大量成本，建議您搭配使用 Ubuntu Pro 與 Savings Plans。如需詳細資訊，請參閱 [Savings Plans 使用者指南](#)。

### Important

如果您計畫購買預留執行個體，以套用至從 AWS Marketplace AMI 啟動的隨需執行個體，請先檢查 AMI 的 PlatformDetails 欄位。此 PlatformDetails 欄位會指出要購買的預留執行個體項目。AMI 的平台詳細資訊必須與預留執行個體的平台相符，否則預留執行個體將不會套用到隨需執行個體。如需有關如何檢視 AMI 平台詳細資訊的詳細資訊，請參閱 [了解 AMI 帳單資訊](#)。

## 將您的購買項目放入佇列

根據預設，當您購買預留執行個體時，購買會立即進行。或者，您可以將購買放入未來日期和時間的佇列。例如，您可以將購買放入約為現有預留執行個體到期時的佇列。這可協助您確保您的涵蓋範圍不會間斷。

您可以將購買放入地區預留執行個體的佇列，但不是其他賣方的區域預留執行個體或預留執行個體。最早可提前三年將購買放入佇列。在排程日期和時間，購買會依據預設的付款方式進行。付款成功之後，計費優勢便會套用。

您可以在 Amazon EC2 主控台中查看您的佇列購買。佇列購買的狀態為 `queued` (佇列)。在排程時間之前，您可以隨時取消佇列購買。如需詳細資訊，請參閱 [取消佇列購買](#)。



## 購買標準版 預留執行個體

您可以在特定可用區域中購買標準預留執行個體，取得容量保留。或者，您可以放棄容量保留，購買地區標準預留執行個體。

### New console

#### 使用主控台購買標準預留執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)，然後選擇 Purchase 預留執行個體 (購買 預留執行個體)。
3. 針對 Offering class (優惠類別)，選擇 Standard (標準) 以顯示標準 預留執行個體。
4. 若要購買容量保留，請在購買畫面的右上角開啟 Only show offerings that reserve capacity (只顯示預留容量的優惠)。當您切換此設定時，會出現 Availability Zone (可用區域) 欄位。

若要購買區域 預留執行個體，請關閉此設定。當您關閉此設定時，Availability Zone (可用區域) 欄位即會消失。

5. 視需要選取其他組態，然後選取 Search (搜尋)。
6. 在您想要購買的每一個 預留執行個體 中，輸入所需數量，然後選擇 Add to cart (新增到購物車)。

若要從預留執行個體市場購買標準預留執行個體，請在搜尋結果中的 Seller (賣方) 列中尋找 3rd party (第三方)。Term (期間) 資料行會顯示非標準期間。如需詳細資訊，請參閱 [從預留執行個體 Marketplace 購買](#)。

7. 若要查看您選取之預留執行個體的摘要，請選取 View cart (檢視購物車)。
8. 如果 Order on (訂購日期) 為 Now (立即)，則在您選擇 Order all (全部訂購) 之後會立即完成購買。若要將購買加入佇列，請選取 Now (立即) 並選取日期。您可以為購物車中每項合格產品選取不同的日期。購買項目會放入佇列，直到所選日期的 00:00 UTC。
9. 若要完成訂單，請選擇 Order all (全部訂購)。

如果在下訂單時，有類似於您選擇的產品，但價格較低，則會以較低的價格向您 AWS 銷售產品。

10. 選擇 Close (關閉)。

訂單的狀態會列在 State (狀態) 資料行中。當您的訂單完成時，State (狀態) 的值便會從 Payment-pending 變更為 Active。當預留執行個體為 Active 時，即已準備好可供使用。

**Note**

如果狀態顯示為Retired，AWS 可能尚未收到您的付款。

## Old console

### 使用主控台購買標準預留執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)，然後選擇 Purchase 預留執行個體 (購買 預留執行個體)。
3. 針對 Offering Class (優惠類別)，選擇 Standard (標準) 以顯示標準預留執行個體。
4. 若要購買容量保留，請在購買畫面的右上角選擇 Only show offerings that reserve capacity (只顯示預留容量的優惠)。若要購買地區預留執行個體，請讓核取方塊保持在未勾選的狀態。
5. 視需要選取其他組態，然後選取 Search (搜尋)。

若要從預留執行個體市場購買標準預留執行個體，請在搜尋結果中的 Seller (賣方) 列中尋找 3rd Party (第三方)。Term (期間) 資料行會顯示非標準期間。

6. 在您想要購買的每一個 預留執行個體 中，輸入數量，然後選擇 Add to Cart (新增到購物車)。
7. 若要查看您選取之預留執行個體的摘要，請選取 View Cart (檢視購物車)。
8. 如果 Order On (訂購時間) 為 Now (立即)，便會立刻完成購買。若要將購買加入佇列，請選取 Now (立即) 並選取日期。您可以為購物車中每項合格產品選取不同的日期。購買項目會放入佇列，直到所選日期的 00:00 UTC。
9. 若要完成訂單，請選擇 Order (訂購)。

如果在下訂單時，有類似於您選擇的產品，但價格較低，則會以較低的價格向您 AWS 銷售產品。

10. 選擇 Close (關閉)。

訂單的狀態會列在 State (狀態) 資料行中。當您的訂單完成時，State (狀態) 的值便會從 payment-pending 變更為 active。當預留執行個體為 active 時，即已準備好可供使用。

**Note**

如果狀態顯示為 `retired`，AWS 可能尚未收到您的付款。

## 若要使用 AWS CLI

1. 使用 [describe-reserved-instances-offerings](#) 命令搜尋可用的預留執行個體。為 `--offering-class` 參數指定 `standard`，以僅傳回標準預留執行個體。您可以套用其他參數來縮小結果範圍。例如，若您希望購買使用預設租用 `t2.large` 為期僅一年的區域 Linux/UNIX 預留執行個體：

```
aws ec2 describe-reserved-instances-offerings \
 --instance-type t2.large \
 --offering-class standard \
 --product-description "Linux/UNIX" \
 --instance-tenancy default \
 --filters Name=duration,Values=31536000 Name=scope,Values=Region
```

若只尋找預留執行個體 Marketplace 上的預留執行個體，請使用 `marketplace` 篩選條件，並且不要在請求中指定有效期間，因為有效期間可能會短於 1 年或 3 年。

```
aws ec2 describe-reserved-instances-offerings \
 --instance-type t2.large \
 --offering-class standard \
 --product-description "Linux/UNIX" \
 --instance-tenancy default \
 --filters Name=marketplace,Values=true
```

當您找到符合您需求的預留執行個體時，請記下產品 ID。例如：

```
"ReservedInstancesOfferingId": "bec624df-a8cc-4aad-a72f-4f8abc34caf2"
```

2. 使用 [purchase-reserved-instances-offering](#) 命令購買您的預留執行個體。您必須指定您在先前步驟中取得的預留執行個體優惠 ID，並指定保留的執行個體數目。

```
aws ec2 purchase-reserved-instances-offering \
 --reserved-instances-offering-id bec624df-a8cc-4aad-a72f-4f8abc34caf2 \
 --instance-count 1
```

依預設，購買會立刻完成。或者，若要將購買加入佇列，則在先前的呼叫中新增下列參數。

```
--purchase-time "2020-12-01T00:00:00Z"
```

3. 使用 [describe-reserved-instances](#) 命令取得您預留執行個體的狀態。

```
aws ec2 describe-reserved-instances
```

或者，您也可以使用下列 AWS Tools for Windows PowerShell 指令：

- [Get-EC2ReservedInstancesOffering](#)
- [New-EC2ReservedInstance](#)
- [Get-EC2ReservedInstance](#)

購買完成之後，若您已有符合預留執行個體規格的執行中執行個體，即會立即套用計費優勢。您不需要重新啟動您的執行個體。若您沒有適合的執行中執行個體，請啟動執行個體，確保符合您在預留執行個體中指定的條件。如需詳細資訊，請參閱 [使用您的預留執行個體](#)。

如需預留執行個體套用到您執行中執行個體方式的範例，請參閱 [預留執行個體的套用方式](#)。

## 購買可轉換預留執行個體

您可以在特定可用區域中購買可轉換預留執行個體，取得容量保留。或者，您可以放棄容量保留並購買地區可轉換預留執行個體。

### New console

#### 使用主控台購買可轉換預留執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)，然後選擇 Purchase 預留執行個體 (購買預留執行個體)。
3. 針對 Offering class (優惠類別)，選擇 Convertible (可轉換) 以顯示標準可轉換預留執行個體。
4. 若要購買容量保留，請在購買畫面的右上角開啟 Only show offerings that reserve capacity (只顯示預留容量的優惠)。當您切換此設定時，會出現 Availability Zone (可用區域) 欄位。

若要購買區域 預留執行個體，請關閉此設定。當您關閉此設定時，Availability Zone (可用區域) 欄位即會消失。

5. 視需要選取其他組態，然後選取 Search (搜尋)。
6. 在您想要購買的每一個 可轉換預留執行個體 中，輸入數量，然後選擇 Add to cart (新增到購物車)。
7. 若要查看您選取項目的摘要，請選取 View cart (檢視購物車)。
8. 如果 Order on (訂購日期) 為 Now (立即)，則在您選擇 Order all (全部訂購) 之後會立即完成購買。若要將購買加入佇列，請選取 Now (立即) 並選取日期。您可以為購物車中每項合格產品選取不同的日期。購買項目會放入佇列，直到所選日期的 00:00 UTC。
9. 若要完成訂單，請選擇 Order all (全部訂購)。

如果在下訂單時，有類似於您選擇的產品，但價格較低，則會以較低的價格向您 AWS 銷售產品。

10. 選擇 Close (關閉)。

訂單的狀態會列在 State (狀態) 資料行中。當您的訂單完成時，State (狀態) 的值便會從 Payment-pending 變更為 Active。當預留執行個體為 Active 時，即已準備好可供使用。

#### Note

如果狀態顯示為 Retired，AWS 可能尚未收到您的付款。

## Old console

### 使用主控台購買可轉換預留執行個體


1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)，然後選擇 Purchase 預留執行個體 (購買 預留執行個體)。
3. 針對 Offering Class (優惠類別)，選擇 Convertible (可轉換) 以顯示標準可轉換預留執行個體。
4. 若要購買容量保留，請在購買畫面的右上角選擇 Only show offerings that reserve capacity (只顯示預留容量的優惠)。若要購買地區預留執行個體，請讓核取方塊保持在未勾選的狀態。
5. 視需要選取其他組態，然後選取 Search (搜尋)。

6. 在您想要購買的每一個可轉換預留執行個體中，輸入數量，然後選擇 Add to Cart (新增到購物車)。
7. 若要查看您選取項目的摘要，請選取 View Cart (檢視購物車)。
8. 如果 Order On (訂購時間) 為 Now (立即)，便會立刻完成購買。若要將購買加入佇列，請選取 Now (立即) 並選取日期。您可以為購物車中每項合格產品選取不同的日期。購買項目會放入佇列，直到所選日期的 00:00 UTC。
9. 若要完成訂單，請選擇 Order (訂購)。

如果在下訂單時，有類似於您選擇的產品，但價格較低，則會以較低的價格向您 AWS 銷售產品。

10. 選擇 Close (關閉)。

訂單的狀態會列在 State (狀態) 資料行中。當您的訂單完成時，State (狀態) 的值便會從 payment-pending 變更為 active。當預留執行個體為 active 時，即已準備好可供使用。

 Note

如果狀態顯示為retired，AWS 可能尚未收到您的付款。

## 使用購買可轉換預留執行個體 AWS CLI

1. 使用 [describe-reserved-instances-offerings](#) 命令搜尋可用的預留執行個體。為 --offering-class 參數指定 convertible，以僅傳回可轉換預留執行個體。您可以套用額外的參數縮減結果，例如，若您希望購買使用預設租用 t2.large 的區域 Linux/UNIX 預留執行個體：

```
aws ec2 describe-reserved-instances-offerings \
 --instance-type t2.large \
 --offering-class convertible \
 --product-description "Linux/UNIX" \
 --instance-tenancy default \
 --filters Name=scope,Values=Region
```

當您找到符合您需求的預留執行個體時，請記下產品 ID。例如：

```
"ReservedInstancesOfferingId": "bec624df-a8cc-4aad-a72f-4f8abc34caf2"
```

2. 使用 [purchase-reserved-instances-offering](#) 命令購買您的預留執行個體。您必須指定您在先前步驟中取得的預留執行個體優惠 ID，並指定保留的執行個體數目。

```
aws ec2 purchase-reserved-instances-offering \
 --reserved-instances-offering-id bec624df-a8cc-4aad-a72f-4f8abc34caf2 \
 --instance-count 1
```

依預設，購買會立刻完成。或者，若要將購買加入佇列，則在先前的呼叫中新增下列參數。

```
--purchase-time "2020-12-01T00:00:00Z"
```

3. 使用 [describe-reserved-instances](#) 命令取得您預留執行個體的狀態。

```
aws ec2 describe-reserved-instances
```

或者，您也可以使用下列 AWS Tools for Windows PowerShell 指令：

- [Get-EC2ReservedInstancesOffering](#)
- [New-EC2ReservedInstance](#)
- [Get-EC2ReservedInstance](#)

若您已有符合預留執行個體規格的執行中執行個體，即會立即套用計費利益。您不需要重新啟動您的執行個體。若您沒有適合的執行中執行個體，請啟動執行個體，確保符合您在預留執行個體中指定的條件。如需詳細資訊，請參閱 [使用您的預留執行個體](#)。

如需預留執行個體套用到您執行中執行個體方式的範例，請參閱 [預留執行個體的套用方式](#)。

### 從預留執行個體 Marketplace 購買

您可以從第三方賣方購買預留執行個體，他們擁有預留執行個體 Marketplace 不再需要的預留執行個體。您可以使用 Amazon EC2 主控台或命令列工具來執行這項作業。此程序類似於從中購買預留執行個體 AWS。如需詳細資訊，請參閱 [購買標準版預留執行個體](#)。

在預留執行個體 Marketplace 場購買的預留執行個體與直接從下列地點購買的預留執行個體有一些差異 AWS：

- 有效期間 – 您從第三方賣方購買的預留執行個體具有低於完整標準有效期間的剩餘時間。從 AWS 運行一年或三年的完整標準條款。

- 預付價格 – 第三方預留執行個體可使用不同的預付價格出售。使用量或週期性費用與預留執行個體最初購買時所設定的費用相同 AWS。
- 預留執行個體類型 – 只能從預留執行個體市場購買 Amazon EC2 標準預留執行個體。可轉換預留執行個體、Amazon RDS 和 Amazon ElastiCache 預留執行個體無法在預留執行個體 Marketplace 購買。

您的基本資訊會與賣方共享 (例如您的郵遞區號和地區/國家資訊)。

此資訊可讓賣方計算任何需要匯款給政府的必要交易稅 (例如營業稅或增值稅)，並會做為收款報告提供。在極少數情況下，賣家可 AWS 能必須向賣家提供你的電郵地址，以便他們就銷售相關問題 (例如稅務問題) 與你聯絡。

出於類似原因，在買方的購買發票上 AWS 共享賣家的法定實體名稱。若您因為稅務或其他相關原因，需要賣方的額外資訊，請聯絡 [AWS Support](#)。

## 檢視您的 預留執行個體

您可以使用 Amazon EC2 主控台或命令列工具檢視您購買的預留執行個體。

在主控台中檢視您的預留執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)。
3. 您的已排入佇列、作用中和已淘汰 預留執行個體 會在此列出。State (狀態) 資料行會顯示狀態。
4. 若您是預留執行個體 Marketplace 中的賣方，My Listings (我的清單) 標籤會顯示在 [Reserved Instance Marketplace](#) (預留執行個體 Marketplace) 中列出的預留狀態。如需詳細資訊，請參閱 [預留執行個體清單狀態](#)。

使用命令列檢視您的預留執行個體

- [describe-reserved-instances](#) (AWS CLI)
- [Get-EC2ReservedInstance](#)(視窗工具 PowerShell)

## 取消佇列購買

最早可提前三年將購買放入佇列。在排程時間之前，您可以隨時取消佇列購買。



## New console

### 取消佇列購買

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)。
3. 選取一或多個 預留執行個體。
4. 選擇 Actions (動作)、Delete queued Reserved Instances (刪除佇列預留執行個體)。
5. 出現確認提示時，依序選擇 Delete (刪除)、Close (關閉)。

## Old console

### 取消佇列購買

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)。
3. 選取一或多個 預留執行個體。
4. 選擇 Actions (動作)、Delete Queued Reserved Instances (刪除佇列預留執行個體)。
5. 出現確認提示時，選擇 Yes, Delete (是，刪除)。

## 使用命令列取消佇列的購買

- [delete-queued-reserved-instances](#) (AWS CLI)
- [Remove-EC2QueuedReservedInstance](#)(視窗工具 PowerShell)

## 續約 預留執行個體

您可以在預留執行個體排定到期前續約。續約預留執行個體會將具有相同預留執行個體組態的購買排入佇列，直到當前預留執行個體到期為止。

## New console

### 使用已佇列的購買續約預留執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)。

3. 選取要續約的預留執行個體。
4. 選擇 Actions (動作)、Renew Reserved Instances (續約預留執行個體)。
5. 若要完成訂單，請選擇 Order all (全部訂購)，然後選擇 Close (關閉)。

## Old console

使用已佇列的購買續約預留執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)。
3. 選取要續約的預留執行個體。
4. 選擇 Actions (動作)、Renew Reserved Instances (續約預留執行個體)。
5. 若要完成訂單，請選擇 Order (訂購)。

## 在預留執行個體 Marketplace 銷售

預留執行個體 Marketplace 是一個平台，可支援銷售第三方和 AWS 客戶未使用的標準預留執行個體，期限長度和定價選項各有不同。例如，您可能想要在將執行個體移至新 AWS 區域、變更為新執行個體類型、在期限到期前結束專案、業務需要變更時或您有不需要的容量時，銷售預留執行個體。

在預留執行個體 Marketplace 中列出您的預留執行個體後，它們便可供潛在買方尋找。所有預留執行個體都會根據剩餘期間的長度和每小時的價格分組。

若要滿足買方透過 EC2 預留執行個體 Marketplace 購買第三方賣家預留執行個體的要求，請 AWS 先以指定群組中最低的預付價格銷售預留執行個體。然後，以下一個最低的價格 AWS 出售預留執行個體，直到買方的整筆訂單出貨為止。AWS 然後處理交易，並將預留執行個體的擁有權移轉給買方。

直到賣出之前，您都會擁有您的預留執行個體。在銷售之後，您便已放棄容量保留和折扣後的經常性費用。如果您繼續使用執行個體，則會從預留執行個體銷售時開始向您 AWS 收取隨需價格的費用。

若您希望在預留執行個體 Marketplace 中銷售您未使用的預留執行個體，您必須符合特定適用性條件。

關於在預留執行個體 Marketplace 上購買預留執行個體的資訊，請參閱 [從預留執行個體 Marketplace 購買](#)。

## 內容

- [法規與限制](#)

- [註冊為賣家](#)
- [用於收款的銀行帳戶](#)
- [稅務資訊](#)
- [為您的 預留執行個體 定價](#)
- [列出您的 預留執行個體](#)
- [預留執行個體清單狀態](#)
- [清單的生命週期](#)
- [在您的預留執行個體售出後](#)
- [收取款項](#)
- [與買方共享的資訊](#)

## 法規與限制

在您販售未使用的預留前，您必須先在預留執行個體 Marketplace 中註冊為賣方。如需相關資訊，請參閱[註冊為賣家](#)。

以下限制會在販售預留執行個體時適用：

- Amazon EC2 標準地區與區域預留執行個體只能在預留執行個體市場中銷售。
- Amazon EC2 可轉換預留執行個體不能在預留執行個體市場中銷售。
- 其他 AWS 服務的預留執行個體 (例如 Amazon RDS 和 Amazon ElastiCache) 無法在預留執行個體 Marketplace 銷售。
- 標準預留執行個體的剩餘期間至少要一個月。
- 您無法在[預設停用](#)的區域中販售標準版 預留執行個體。
- 預留執行個體 Marketplace 允許的最低價為 0.00 美元。
- 您可以在預留執行個體 Marketplace 中銷售無預留、部分預付或全部預付等執行個體，只要其在您的帳戶中處於活動狀態至少 30 天。此外，若預留執行個體有預付款，則只有在 AWS 收到預付款後才能出售。
- 您無法直接在預留執行個體 Marketplace 中修改您的清單。但是，您可以藉由先取消它，然後再使用新的參數建立另一個清單，來變更您的清單。如需相關資訊，請參閱[為您的 預留執行個體 定價](#)。您也可以將它們列出之前修改您的預留執行個體。如需相關資訊，請參閱「[修改 預留執行個體](#)」。
- AWS 收取您在預留執行個體 Marketplace 銷售的每個標準預留執行個體預付總價的 12% 的服務費。預付價格為賣方向標準預留執行個體收取的價格。

- 當您註冊成為賣方時，您指定的銀行必須具有美國地址。如需詳細資訊，請參閱 AWS Marketplace 賣方指南中的 [付費產品的其他賣方需求](#)。
- Amazon Web Services 印度私人有限公司 (AWS 印度) 客戶無法在預留執行個體 Marketplace 銷售預留執行個體，即使他們擁有美國的銀行帳戶也一樣。如需詳細資訊，請參閱 [AWS 帳戶 和 AWS India 帳戶之間有何差異？](#)

## 註冊為賣家

### Note

只有 AWS 帳戶根使用者 可以註冊一個帳戶作為賣家。

若要在預留執行個體 Marketplace 中進行銷售，您必須先註冊為賣方。在註冊期間，您會提供下列資訊：

- 銀行信息 —AWS 必須擁有您的銀行信息，以便在您出售預訂時支付收集的資金。您指定的銀行必須具有美國地址。如需詳細資訊，請參閱 [用於收款的銀行帳戶](#)。
- 稅務資訊 – 所有賣方都必須完成稅務資訊面試，才能判斷任何必要的稅務報告義務。如需詳細資訊，請參閱 [稅務資訊](#)。

AWS 收到完成的賣家註冊後，您會收到一封確認註冊的電子郵件，並通知您可以開始在預留執行個體 Marketplace 銷售產品。

## 用於收款的銀行帳戶

AWS 必須擁有您的銀行資訊，才能支付銷售預留執行個體時所收集的資金。您指定的銀行必須具有美國地址。如需詳細資訊，請參閱 AWS Marketplace 賣方指南中的 [付費產品的其他賣方需求](#)。

## 註冊預設銀行帳戶以用於接收款項

1. 開啟 [預留執行個體市場賣方註冊](#) 頁面，然後使用您的 AWS 登入資料登入。
2. 在 Manage Bank Account (管理銀行帳戶) 頁面上，提供下列關於接收款項銀行的資訊：
  - 銀行帳戶持有人名稱
  - 路由號碼
  - 帳戶號碼

- 銀行帳戶類型

**Note**

若您使用的是企業銀行帳戶，您會收到提示，請您透過傳真 (1-206-765-3424) 傳送銀行帳戶的資訊。

在註冊之後，提供的銀行帳戶便會設為預設，等待銀行進行驗證。驗證新的銀行帳戶最多可能需要兩週的時間，在此期間您無法收到款項。針對已建立的帳戶，支付款項通常需要兩天的時間才能完成。

#### 變更接收款項的預設銀行帳戶

1. 在[預留執行個體市場賣方註冊](#)頁面上，使用您用來註冊的帳戶登入。
2. 在 Manage Bank Account (管理銀行帳戶) 頁面上，新增銀行帳戶或視需要修改銀行帳戶。

#### 稅務資訊

您販售預留執行個體的行為必須遵守交易式稅務的規範，例如營業稅或增值稅。建議您和您公司的稅務、法律、財務或會計部門確認，以判斷是否適用於交易式的稅務。您必須負責收集和將交易式的稅務傳送給適當的稅務主管機關。

做為賣方註冊程序的一部分，您必須在[賣方註冊入口網站](#)中完成稅務面試。面試會收集您的稅務資訊，並填入 IRS W-9、W-8BEN，或 W-8BEN-E 表單。這份表單會用於判斷是否有任何必要的稅務報告義務。

您做為稅務面試一部分所輸入的稅務資訊可能會有所不同，取決於您是否是以個人或公司運作，以及您或您的公司是否為美國或非美國的個人或實體。當您填寫稅務面試時，請謹記下列幾點：

- 提供的資訊 (包括本主題中的資訊) 並不構成稅務、法律或其他專業建議。AWS 若要了解 IRS 報告需求影響您企業的方式，或是您有其他問題，請聯絡您的稅務、法律或其他專業顧問。
- 若要盡可能有效率的完成 IRS 報告需求，請回答所有問題並輸入所有面試中請求的資訊。
- 檢查您的答案。避免拼字錯誤或輸入不正確的稅務識別號碼。這可能會導致無效的稅務表單。

根據您的稅務面試回應和 IRS 報告閾值，Amazon 可能會提出 1099-K 表單。Amazon 會將您 1099-K 表單的複本在您的稅務帳戶達到閾值層級年份的下一年，於 1 月 31 日或在 1 月 31 日前寄送給您。例如，若您的帳戶在 2018 年達到閾值，則您的 1099-K 表單會在 2019 年 1 月 31 日或之前寄送給您。

如需 IRS 需求和 1099-K 表單的詳細資訊，請參閱 [IRS](#) 網站。

## 為您的 預留執行個體 定價

設定預留執行個體的價格時，請考慮下列事項：

- 預付價格 - 預付價格是您可以為要販售之預留執行個體指定的唯一價格。預付價格是買方購買預留執行個體時所需支付的一次性價格。

由於預留執行個體的值會隨著時間的推移而減少，因此依預設，AWS 可以將價格設定為每月以相等的遞增方式降低。但是，您可以根據販售您保留的時間，設定不同的預付價格。例如，若您的預留執行個體具有九個月的剩餘期間，您可以指定客戶購買具有九個月剩餘期間的該預留執行個體時，您所能接受的價格。您可以為剩餘期間五個月設定另一個價格，以及為剩餘期間一個月設定另一個價格。

預留執行個體 Marketplace 允許的最低價為 0.00 美元。

- 限制 - 銷售預留執行個體的以下限制適用於 AWS 帳戶的生命週期。這不是年度限制。
  - 您可以在預留執行個體賣出最多 50,000 美元。
  - 您最多可以賣 5,000 個預留執行個體

這些限制通常無法增加，但會 case-by-case 根據要求進行評估。如需請求提高限制，請完成[提高服務限制](#)表單。針對限制類型，請選擇 EC2 預留執行個體銷售。

- 無法修改 - 您無法直接修改清單。但是，您可以藉由先取消它，然後再使用新的參數建立另一個清單，來變更您的清單。
- 可以取消 - 只要清單處於 active 狀態，即可隨時取消您的清單。若清單已相符或正在為銷售進行處理，您便無法取消清單。若您清單中的某些執行個體已相符且您取消了清單，則只有剩餘未相符的執行個體會從清單中移除。

## 列出您的 預留執行個體

身為已註冊賣方，您可以選擇銷售一或多個預留執行個體。您可以選擇一次出售全部或部分出售。此外，您可以使用任何執行個體類型、平台和範圍組態，列出預留執行個體。

主控台會決定建議價格。它會檢查符合 預留執行個體 的優惠，並匹配最低價格的產品。否則會根據預留執行個體 剩餘時間的成本來計算建議價格。如果計算的值低於 1.01 USD，則建議價格為 1.01 USD。

若您取消您的清單，但該清單中的一部分已賣出，則取消對已經賣出的部分不會有任何效果。只有清單中尚未賣出的部分才會無法在預留執行個體 Marketplace 中取得。

若要在預留執行個體 Marketplace 中使用 AWS Management Console

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)。
3. 選取要列出的預留執行個體，然後選取 Actions (動作)、Sell 預留執行個體 (販售 預留執行個體)。
4. 在 Configure Your 預留執行個體 Listing (設定您的預留執行個體清單) 頁面上，在相關資料行中設定要販售的執行個體數目，以及剩餘期間的預付價格。選取 Months Remaining (剩餘月數) 資料行，查看您保留的值隨剩餘期間變化的方式。
5. 若您是進階使用者希望自訂定價，您可以為後續月份輸入不同的值。若要返回預設的線性價格下降，請選擇 Reset (重設)。
6. 當您完成設定清單後，請選擇 Continue (繼續)。
7. 在 Confirm Your 預留執行個體 Listing (確認您的預留執行個體清單) 頁面上確認您清單的詳細資訊，若您已感到滿意，請選擇 List Reserved Instance (列出預留執行個體)。

在主控台中檢視您的清單

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)。
3. 選取您已列出的預留執行個體，並選取頁面底部附近的 My Listings (我的清單) 索引標籤。

在預留執行個體 Marketplace 中管理預留執行個體 AWS CLI

1. 透過使用 [describe-reserved-instances](#) 命令來取得您預留執行個體的清單。
2. 記下您希望列出之預留執行個體的 ID，然後呼叫 [create-reserved-instances-listing](#)。您必須指定預留執行個體的 ID、執行個體數目，以及定價排程。
3. 若要檢視您的清單，請使用 [describe-reserved-instances-listings](#) 命令。
4. 若要取消您的清單，請使用 [cancel-reserved-instances-listings](#) 命令。

預留執行個體清單狀態

預留執行個體頁面的 My Listings (我的清單) 標籤上的 Listing State (清單狀態) 會顯示您清單目前的狀態：

Listing State (清單狀態) 顯示的資訊與預留執行個體市場中的清單狀態有關。它與 Reserved Instances (預留執行個體) 中 State (狀態) 資料行顯示的狀態資訊不同。此 State (狀態) 資訊與您的保留有關。

- active (作用中) – 清單可供購買。
- canceled (已取消) - 清單已取消，無法在預留執行個體市場中購買。
- closed (已關閉) – 預留執行個體並未列出。預留執行個體可能會因為清單的銷售已完成而為 closed 狀態。

## 清單的生命週期

當您清單中的所有執行個體都已符合並售出，My Listings (我的清單) 標籤便會顯示 Total instance count (總執行個體計數) 符合 Sold (已售出) 下方列出的計數。此外，您的清單中將不會有任何剩餘 Available (可用) 的執行個體，且其 Status (狀態) 為 closed。

當只銷售部分清單時，會 AWS 淘汰清單中的預留執行個體，並建立與計數中剩餘的預留執行個體數量相等。因此，清單 ID 和其代表的清單 (目前具有較少銷售中的保留) 仍會處於作用中。

任何之後此清單內發生的預留執行個體銷售都會使用相同的方式處理。出售清單中的所有預留執行個體後，將清單 AWS 標示為 closed。

例如，您建立預留執行個體 listing ID 5ec28771-05ff-4b9b-aa31-9e57dexample，其清單計數為 5。

Reserved Instance (預留執行個體) 主控台頁面中的 My Listings (我的清單) 標籤會以此方式顯示清單：

Reserved Instance listing ID 5ec28771-05ff-4b9b-aa31-9e57dexample

- 總保留計數 = 5
- 售出 = 0
- 可用 = 5
- 狀態 = 作用中

買方購買兩個保留，使剩餘可供銷售的保留計數為 3。因為此部分銷售，AWS 會使用計數三建立新的保留，代表剩餘可供銷售的保留。

這是在 My Listings (我的清單) 標籤中您清單的外觀：

Reserved Instance listing ID 5ec28771-05ff-4b9b-aa31-9e57dexample

- 總保留計數 = 5
- 售出 = 2
- 可用 = 3



- 狀態 = 作用中

若您取消您的清單，但該清單中的一部分已賣出，則取消對已經賣出的部分不會有任何效果。只有清單中尚未賣出的部分才會無法在預留執行個體 Marketplace 中取得。

### 在您的預留執行個體售出後

售出預留執行個體時，AWS 會傳送電子郵件通知給您。每天若有任何類型的活動，您都會接收到一封電子郵件通知，擷取一天內所有的活動。您可以在建立或出售刊登物品時，或是 AWS 將資金匯入您的帳戶時進行的活動。

### 在主控台追蹤預留執行個體清單的狀態

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)。
3. 選擇 My Listings (我的清單) 索引標籤。

My Listings (我的清單) 標籤包含 Listing State (清單狀態) 值。它同時也包含期間、清單價格，清單中有多少可用、擱置中、已售出和已取消執行個體解析的相關資訊。

您也可以搭配適當的篩選條件，使用 [describe-reserved-instances-listings](#) 命令來取得您清單的相關資訊。

### 收取款項

一旦 AWS 收到買方的款項，系統就會向已售出預留執行個體的註冊擁有者帳戶電子郵件傳送訊息。

AWS 將自動結算所 (ACH) 電匯發送到您指定的銀行帳戶。通常，這種匯款會在賣出您預留執行個體的一到三天後發生。每天會進行一次匯款。您將收到一封電子郵件，含有釋出基金後的收款報告。請記得，您無法在 AWS 收到來自您銀行的驗證之前接收款項。這可能需要兩週的時間。

您賣出的預留執行個體會繼續在您描述您的預留執行個體時出現。

您可以透過電匯直接存入您的銀行帳戶，為預留執行個體收到現金支付。AWS 收取您在預留執行個體 Marketplace 銷售的每個預留執行個體預付總價的 12% 的服務費。

### 與買方共享的資訊

當您在預留執行個體 Marketplace 銷售產品時，AWS 會根據美國法規在買方聲明上提供貴公司的法定名稱。此外，如果買家 AWS Support 因為發票或其他稅務相關原因需要聯絡你，買家可 AWS 能需要向買家提供你的電郵地址，方便買家直接聯絡你。

基於相似的原因，也會在賣方的收款報告中提供買方的郵遞區號和國家資訊。做為賣方，您可能需要此資訊伴隨任何您匯款給政府的必要交易稅 (例如營業稅和加值稅)。

AWS 無法提供稅務建議，但如果您的稅務專家認為您需要特定的額外資訊，[請聯絡 AWS Support](#)。

## 修改 預留執行個體

當您的需求變更時，您可以修改您的標準或可轉換預留執行個體，並繼續獲得計費利益。您可以修改預留執行個體的屬性，例如可用區域、執行個體大小 (位於相同執行個體系列和層代內) 和範圍。

### Note

您也可以將可轉換預留執行個體與另一個具有不同組態的可轉換預留執行個體交換。如需詳細資訊，請參閱 [交換 可轉換預留執行個體](#)。

您也可以修改所有預留執行個體或子集。您可以將原始預留執行個體分隔為兩個或多個新的預留執行個體。例如，若您在 us-east-1a 中有 10 個執行個體的保留，並決定將 5 個執行個體移動到 us-east-1b，修改請求結果會導致兩個新的保留：其中一個保留針對位於 us-east-1a 中的 5 個執行個體，另外一個保留則針對位於 us-east-1b 的 5 個執行個體。

您也可以將兩個或多個預留執行個體合併為單一預留執行個體。例如，若您有四個 t2.small 預留執行個體，每一個都具有一個執行個體，您可以合併它們來建立一個 t2.large 預留執行個體。如需詳細資訊，請參閱 [支援執行個體大小的修改](#)。

在修改後，預留執行個體的利益僅會套用到符合新參數的執行個體。例如，若您變更保留的可用區域，容量保留和定價利益會自動套用到新可用區域中的執行個體用量。不再符合新參數的執行個體會以隨需費率計費，除非您的帳戶有其他適用的保留。

若您的修改請求成功：

- 修改後的保留會立即生效，定價利益會在修改請求的小時內開始套用到新的執行個體。例如，若您在下午 9:15 成功修改您的保留，定價利益會在下午 9:00 傳送到您新的執行個體。您可以藉由使用 [describe-reserved-instances](#) 命令來取得修改後預留執行個體的生效日期。
- 原始的保留會淘汰。其結束日期便是新保留的開始日期，新保留的結束日期則與原始預留執行個體的結束日期相同。若您修改具有 16 個月剩餘期間的三年保留，其結果產生的修改後保留便是 16 個月的保留，其結束日期與原先的保留相同。
- 修改後的保留列出的固定價格為 0 USD，而非原始保留的固定價格。

- 修改後保留的固定價格不會影響套用到您帳戶的折扣定價方案計算，因為該計算是以原始保留的固定價格為基礎。

若您的修改請求失敗，您的預留執行個體將會維持其原始的組態，並可立即進行另一次的修改請求。

修改無須支付任何費用，您不會收到任何新的帳單或發票。

您可以無限次數的修改您的保留，但在您提交後，您便無法變更或取消擱置中的修改請求。在修改成功完成後，您可以提交另一個修改請求，轉返任何您進行的變更 (若需要的話)。

## 內容

- [修改的要求和限制](#)
- [支援執行個體大小的修改](#)
- [提交修改請求](#)
- [故障診斷修改請求](#)

## 修改的要求和限制

您可以修改這些屬性如下。

| 可修改屬性                  | 支援平台            | 限制及考量                                                                                                                                                     |
|------------------------|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|
| 變更相同區域內的可用區域           | Linux 與 Windows | -                                                                                                                                                         |
| 將範圍從「可用區域」變更為「區域」，反之亦然 | Linux 與 Windows | <p>區域性預留執行個體的範圍限定為可用區域，並在該可用區域中預留容量。若您將範圍從可用區域變更為區域 (換言之，從區域到地區)，您將會喪失容量保留利益。</p> <p>地區性預留執行個體的範圍限定為區域。您的預留執行個體折扣可套用到該區域內任何可用區域中正在執行的執行個體。此外，預留執行個體折扣</p> |

| 可修改屬性                        | 支援平台                                                                                                                                                                                                                                                                                           | 限制及考量                                                                                                                                  |
|------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------|
|                              |                                                                                                                                                                                                                                                                                                | <p>適用於選定執行個體系列中所有大小的執行個體用量。若您將範圍從區域變更為可用區域 (換言之，從地區到區域)，您將會喪失可用區域彈性和執行個體大小彈性 (若適用)。</p> <p>如需詳細資訊，請參閱 <a href="#">預留執行個體的套用方式</a>。</p> |
| <p>在相同執行個體系列和層代內變更執行個體大小</p> | <p>僅限 Linux/UNIX</p> <p>執行個體大小的彈性不適用於其他平台上的預留執行個體，這些平台包括 Linux with SQL Server Standard、Linux with SQL Server Web、Linux with SQL Server Enterprise、Red Hat Enterprise Linux、SUSE Linux、Windows、Windows with SQL Standard、Windows with SQL Server Enterprise 和 Windows with SQL Server Web。</p> | <p>保留必須使用預設租用。某些執行個體系列不受支援，因為沒有其他可用的大小。如需更多資訊，請參閱 <a href="#">支援執行個體大小的修改</a></p>                                                      |

## 要求

Amazon EC2 會在有足夠容量接受您的新組態 (若適用)，以及滿足下列條件時處理您的修改請求：

- 預留執行個體在您購買之前或當時無法修改
- 預留執行個體必須為作用中
- 不能有擱置的修改請求
- 預留執行個體 Marketplace 中未列出預留執行個體。

- 原始保留的執行個體大小使用量和新組態必須相符。如需詳細資訊，請參閱 [支援執行個體大小的修改](#)。
- 原始 預留執行個體 全部為標準版 預留執行個體 或全部為 可轉換預留執行個體，而非每個類型各有一些
- 原始 預留執行個體 如果是標準版 預留執行個體，則必須在同一小時內過期
- 預留執行個體不是 G4、G4ad、G4 DN、G5、G5G、InF1 或 In2 執行個體。

## 支援執行個體大小的修改

若符合下列要求，您可以修改預留執行個體的執行個體大小。

### 要求

- 此平台為 Linux/UNIX。
- 您必須選取相同 [執行個體系列](#) (用字母表示，例如 T) 和 [層代](#) (用數字表示，例如 2) 中的其它執行個體大小。

例如，您可以將預留執行個體從 t2.small 修改為 t2.large，因為它們都位於相同的 T2 系列和層代中。但是您無法將預留執行個體從 T2 修改為 M2，或從 T2 修改為 T3，因為在這兩個範例中，目標執行個體系列和層代與原始預留執行個體不同。

- 您無法修改下列執行個體的預留執行個體的執行個體大小，因為下列每一個都只有一個大小：
  - t1.micro
- 您無法針對下列執行個體系列、層代和屬性組合修改預留執行個體的執行個體大小：
  - G4ad
  - G4dn
  - G5
  - G5g
  - Inf1
  - Inf2
- 原始和新的 預留執行個體 必須具有相同的執行個體大小使用量。

### 內容

- [執行個體大小使用量](#)
- [裸機執行個體的正規化因素](#)

## 執行個體大小使用量

每一個預留執行個體都具有執行個體大小使用量，由執行個體大小的正規化因素和保留中的執行個體數目決定。當您修改預留執行個體中的執行個體大小時，新組態的使用量必須符合原始組態的使用量，否則不會處理修改請求。

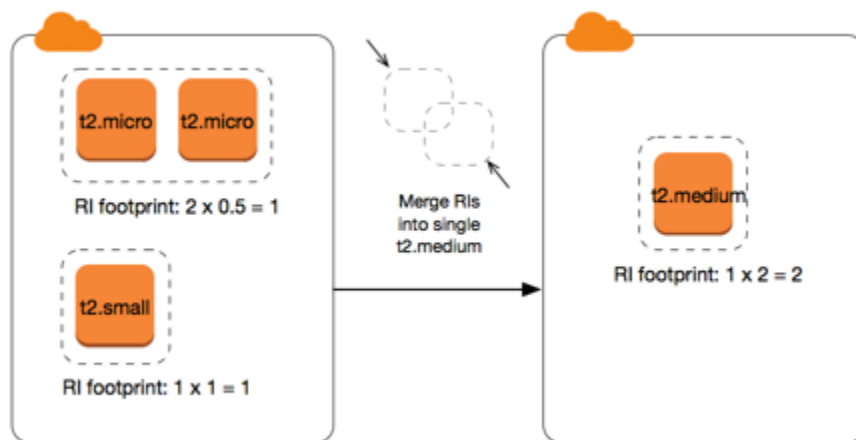
若要計算預留執行個體的執行個體大小使用量，請將執行個體的數目乘上正規化因素。在 Amazon EC2 主控台中，正規化因素以單位進行測量。下表說明執行個體系列中執行個體大小的正規化因素。例如，t2.medium 的正規化因素為 2，因此四個 t2.medium 執行個體的保留具有 8 個單位的使用量。

| 執行個體大小   | 正規化因素 |
|----------|-------|
| nano     | 0.25  |
| micro    | 0.5   |
| small    | 1     |
| medium   | 2     |
| large    | 4     |
| xlarge   | 8     |
| 2xlarge  | 16    |
| 3xlarge  | 24    |
| 4xlarge  | 32    |
| 6xlarge  | 48    |
| 8xlarge  | 64    |
| 9xlarge  | 72    |
| 10xlarge | 80    |
| 12xlarge | 96    |

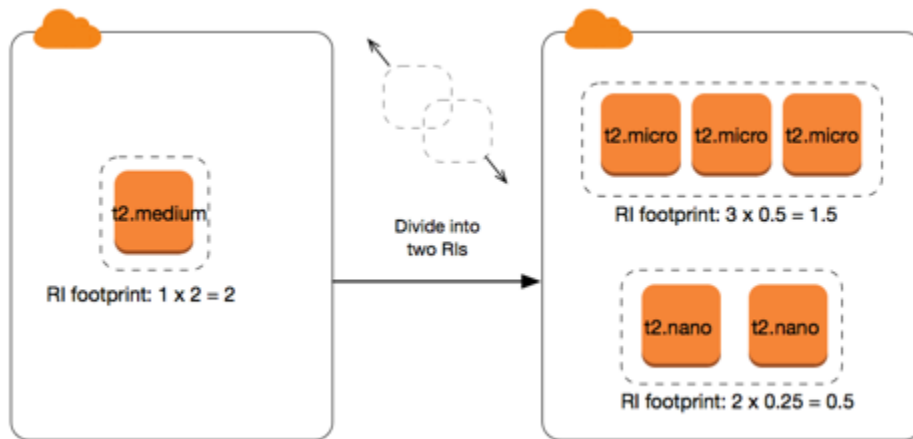
| 執行個體大小    | 正規化因素 |
|-----------|-------|
| 16xlarge  | 128   |
| 18xlarge  | 144   |
| 24xlarge  | 192   |
| 32xlarge  | 256   |
| 48xlarge  | 384   |
| 56xlarge  | 448   |
| 112xlarge | 896   |

您可以將您的保留在相同執行個體系列中配置到不同的執行個體大小，只要您保留的執行個體大小使用量維持不變。例如，您可以將一個 t2.large (1 @ 4 個單位) 執行個體的保留分割為四個 t2.small (4 @ 1 個單位) 執行個體。同樣地，您可以將四個 t2.small 執行個體的保留合併為一個 t2.large 執行個體。不過，您無法將兩個 t2.small 執行個體的保留變更為一個 t2.large 執行個體，因為新保留的使用量 (4 個單位) 大於原始保留的使用量 (2 個單位)。

在下列範例中，您有兩個 t2.micro 執行個體的保留 (1 個單位)，以及一個 t2.small 執行個體的保留 (1 個單位)。如果您將這兩個保留合併為具有一個 t2.medium 執行個體 (2 個單位) 的單一保留，新保留的使用量等於合併保留的使用量。



您也可以修改保留，將其分割為兩個或多個保留。在下列範例中，您有一個具有 t2.medium 執行個體 (2 個單位) 的保留。您可以將保留分割為兩個保留，一個具有兩個 t2.nano 執行個體 (5 個單位)，另一個具有三個 t2.micro 執行個體 (1.5 個單位)。



### 裸機執行個體的正規化因素

您可以使用相同執行個體系列內的其他大小，修改具有 metal 執行個體的保留。同樣地，您可以使用相同執行個體系列內的 metal 大小，修改具有裸機執行個體之外的執行個體的保留。一般而言，裸機執行個體的大小與相同執行個體系列中的最大可用執行個體大小相同。例如，i3.metal 執行個體的大小與 i3.16xlarge 執行個體相同，因此有相同的正規化因素。

下表說明具有裸機執行個體之執行個體系列中，裸機執行個體大小的正規化因素。與其他執行個體大小不同的是，metal 執行個體的正規化因素取決於執行個體系列。

| 執行個體大小                                                                                                                                              | 正規化因素 |
|-----------------------------------------------------------------------------------------------------------------------------------------------------|-------|
| a1.metal                                                                                                                                            | 32    |
| m5zn.metal   x2iezn.metal   z1d.metal                                                                                                               | 96    |
| c6g.metal   c6gd.metal   i3.metal   m6g.metal   m6gd.metal<br>  r6g.metal   r6gd.metal   x2gd.metal                                                 | 128   |
| c5n.metal                                                                                                                                           | 144   |
| c5.metal   c5d.metal   i3en.metal   m5.metal   m5d.metal  <br>m5dn.metal   m5n.metal   r5.metal   r5b.metal   r5d.metal  <br>r5dn.metal   r5n.metal | 192   |



| 執行個體大小                                                                      | 正規化因素 |
|-----------------------------------------------------------------------------|-------|
| c6i.metal   c6id.metal   m6i.metal   m6id.metal   r6d.metal<br>  r6id.metal | 256   |
| u-*.metal                                                                   | 896   |

例如，i3.metal 執行個體的正規化因素為 128。如果您購買 i3.metal 預設租用 Amazon Linux/Unix 預留執行個體，您可以將保留分割如下：

- i3.16xlarge 的大小與 i3.metal 執行個體相同，因此正規化因素為 128 (128/1)。一個 i3.metal 執行個體的保留可以修改為一個 i3.16xlarge 執行個體。
- i3.8xlarge 的大小是 i3.metal 執行個體的一半，因此正規化因素為 64 (128/2)。一個 i3.metal 執行個體的保留可以分割成兩個 i3.8xlarge 執行個體。
- i3.4xlarge 的大小是 i3.metal 執行個體的四分之一，因此正規化因素為 32 (128/4)。一個 i3.metal 執行個體的保留可以分割成四個 i3.4xlarge 執行個體。

## 提交修改請求

在修改預留執行個體之前，請確定您已閱讀適用的[限制](#)。在修改執行個體大小之前，請計算要修改之原始保留項目的總[執行個體大小佔用空間](#)，並確保其與新規劃的執行個體大小總佔用空間相符。

## New console

若要使用修改預留執行個體 AWS Management Console

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在 Reserved Instances (預留執行個體) 頁面上，選取一或多個要修改的預留執行個體，然後選取 Actions (動作)，Modify Reserved Instances (修改預留執行個體)。

### Note

若您的預留執行個體並未處於作用中的狀態或無法修改，Modify 預留執行個體 (修改預留執行個體) 便會停用。

3. 修改表格的第一個項目會顯示選取 預留執行個體 的屬性，以及至少一個位於下方的目標組態。Units (單位) 資料行會顯示總執行個體大小使用量。針對每個要新增的新組態，選擇 Add (新增)。視需要修改每個組態的屬性。
  - Scope (範圍)：選擇組態是否套用到可用區域或是整個區域。
  - Availability Zone (可用區域)：選擇必要的可用區域。不適用於區域預留執行個體。
  - 執行個體類型：選取所需的執行個體類型。結合的組態必須等於原始組態的執行個體大小使用量。
  - Count (計數)：指定執行個體的數目。若要將 預留執行個體 分割為多個組態，請減少計數，選擇 Add (新增)，然後指定額外組態的計數。例如，如果您有一個計數為 10 的組態，您可以將其計數變更為 6，然後新增計數為 4 的組態。此程序會在新的 預留執行個體 啟動之後重試原始的 預留執行個體。
4. 選擇 Continue (繼續)。
5. 若要在完成指定您的目標組態後確認您的修改選擇，請選擇 Submit modifications (提交修改)。
6. 您可以透過查看預留執行個體畫面中的 State (狀態) 資料行，判斷您修改請求的狀態。以下為可能的狀態。
  - 作用中 (待定修改) — 原始預留執行個體的轉移狀態
  - 已淘汰 (待定修改) – 建立新的預留執行個體時，原始預留執行個體的轉移狀態
  - 淘汰 – 已成功修改並取代預留執行個體
  - 作用中 – 下列其中一項：
    - 已成功從修改請求建立新的預留執行個體
    - 修改請求失敗後的原始預留執行個體

## Old console

若要使用修改預留執行個體 AWS Management Console

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在 Reserved Instances (預留執行個體) 頁面上，選取一或多個要修改的預留執行個體，然後選取 Actions (動作)，Modify Reserved Instances (修改預留執行個體)。

**Note**

若您的預留執行個體並未處於作用中的狀態或無法修改，Modify 預留執行個體 (修改預留執行個體) 便會停用。

3. 修改表格的第一個項目會顯示選取預留執行個體的屬性，以及至少一個位於下方的目標組態。Units (單位) 資料行會顯示總執行個體大小使用量。針對每個要新增的新組態，選擇 Add (新增)。視需要針對每個組態修改屬性，然後選擇 Continue (繼續)：
  - Scope (範圍)：選擇組態是否套用到可用區域或是整個區域。
  - Availability Zone (可用區域)：選擇必要的可用區域。不適用於區域預留執行個體。
  - 執行環境類型：選取所需的執行環境類型。結合的組態必須等於原始組態的執行個體大小使用量。
  - Count (計數)：指定執行個體的數目。若要將預留執行個體分割為多個組態，請減少計數，選擇 Add (新增)，然後指定額外組態的計數。例如，如果您有一個計數為 10 的組態，您可以將其計數變更為 6，然後新增計數為 4 的組態。此程序會在新的預留執行個體啟動之後重試原始的預留執行個體。
4. 若要在完成指定您的目標組態後確認您的修改選擇，請選擇 Submit Modifications (提交修改)。
5. 您可以透過查看預留執行個體畫面中的 State (狀態) 資料行，判斷您修改請求的狀態。以下為可能的狀態。
  - 作用中 (待定修改) — 原始預留執行個體的轉移狀態
  - 已淘汰 (待定修改) – 建立新的預留執行個體時，原始預留執行個體的轉移狀態
  - 淘汰 – 已成功修改並取代預留執行個體
  - 作用中 – 下列其中一項：
    - 已成功從修改請求建立新的預留執行個體
    - 修改請求失敗後的原始預留執行個體

### 使用命令列檢視您的預留執行個體

1. 若要修改您的預留執行個體，您可以使用下列其中一個命令：
  - [modify-reserved-instances](#) (AWS CLI)
  - [Edit-EC2ReservedInstance](#) (AWS Tools for Windows PowerShell)
2. 若要取得修改狀態的狀態 (processing、fulfilled 或 failed)，請使用下列其中一個命令：

- [describe-reserved-instances-modifications](#) (AWS CLI)
- [Get-EC2ReservedInstancesModification](#) (AWS Tools for Windows PowerShell)

## 故障診斷修改請求

若您請求的目標組態設定為獨特，您會收到訊息，告知您請求正在處理中。此時，Amazon EC2 僅判斷您修改請求的參數為有效。您的修改請求仍然可能會在處理過程中因為不可用的容量而失敗。

在某些情況下，您可能會收到訊息，表示修改請求未完成或失敗，而非確認。請使用此類訊息中的資訊，做為重新提交另一個修改請求的出發點。確認您已在提交請求前詳閱適用的[限制](#)。

並非所有選取的預留執行個體都可進行修改處理

Amazon EC2 會識別並列出無法修改的預留執行個體。若您收到如這類的訊息，請前往 Amazon EC2 主控台內的 Reserved Instances (預留執行個體) 頁面，檢查預留執行個體的資訊。

## 處理您的修改請求時發生錯誤

您提交一或多個預留執行個體進行修改，但您的請求全部都無法處理。取決於您要修改的保留數目，您可能會收到不同版本的訊息。

Amazon EC2 會顯示無法處理您請求的原因。例如，您可能為一或多個您要修改的預留執行個體子集指定了相同的目標組態 (可用區域和平台的組合)。請嘗試再次提交修改請求，但請確認保留的執行個體詳細資訊確實符合，且每個要修改子集的目標組態都是唯一的。

## 交換可轉換預留執行個體

您可以將一或多個可轉換預留執行個體與另一個具有不同組態的可轉換預留執行個體交換，包括執行個體系列、作業系統和租用。執行交換的次數沒有限制，只要新可轉換預留執行個體的值等於或高於您要交換的可轉換預留執行個體即可。

當您交換您的可轉換預留執行個體時，您目前保留中的執行個體數目會和涵蓋新可轉換預留執行個體組態相同或更高值的執行個體數目交換。Amazon EC2 會計算您在交換之後能接收到的預留執行個體數目。

您無法交換標準版預留執行個體，但您可以修改它們。如需詳細資訊，請參閱[修改預留執行個體](#)。

## 內容

- [交換可轉換預留執行個體的需求](#)

- [計算 可轉換預留執行個體 交換](#)
- [合併 可轉換預留執行個體](#)
- [交換 可轉換預留執行個體 的一部分](#)
- [提交交換請求](#)

### 交換可轉換預留執行個體的需求

若符合下列條件，Amazon EC2 便會處理您的交換請求。您的可轉換預留執行個體必須是：

- 作用中
- 並未擱置先前的修改請求
- 在到期前至少還剩 24 小時

適用的規定如下：

- 可轉換預留執行個體只能與另一個目前由 AWS 提供的可轉換預留執行個體交換。
- 可轉換預留執行個體與特定區域相關聯，該區域在保留的期間內固定不變。您無法和位於不同區域的可轉換預留執行個體交換可轉換預留執行個體。
- 您只能一次為一個可轉換預留執行個體交換一或多個可轉換預留執行個體。
- 若要交換可轉換預留執行個體的一部分，您可以修改它，將它分成兩個或多個保留，然後為新的可轉換預留執行個體交換一或多個保留。如需詳細資訊，請參閱 [交換 可轉換預留執行個體 的一部分](#)。如需修改預留執行個體的詳細資訊，請參閱 [修改 預留執行個體](#)。
- 所有預付可轉換預留執行個體都可和部分預付可轉換預留執行個體交換，反之亦然。

#### Note

若交換所需的總預付款 (校正成本) 低於 0.00 美元，AWS 會自動在可轉換預留執行個體中給予您執行個體的數量，確保校正成本大於或等於 0.00 美元。

#### Note

如果新可轉換預留執行個體的總價值 (前期價格 + 小時價格 \* 剩餘小時數) 小於交換的可轉換預留執行個體的總價值，AWS 會自動為您提供可轉換預留執行個體中的執行個體數量，以確保總值與交換的可轉換預留執行個體的總值相同或高於交換的可轉換預留執行個體。

- 若要從最佳的定價獲得利益，您可以將無預付款可轉換預留執行個體交換為全預付款或部分預付款可轉換預留執行個體。
- 您無法以全預付款及部分預付款可轉換預留執行個體交換無預付款可轉換預留執行個體。
- 只有在新可轉換預留執行個體的每小時價格等於或高於交換可轉換預留執行個體的每小時價格時，您才能以無預付款可轉換預留執行個體交換另一個無預付款可轉換預留執行個體。

#### Note

如果新可轉換預留執行個體的總值 (小時價格 \* 剩餘小時數) 小於交換的可轉換預留執行個體的總價值，則 AWS 會自動為您提供可轉換預留執行個體中的執行個體數量，以確保總值與交換的可轉換預留執行個體相同或高於。

- 若您交換多個具有不同過期日期的可轉換預留執行個體，新可轉換預留執行個體的過期日期將會是距離現在最遠的日期。
- 若您交換單一可轉換預留執行個體，它必須與新可轉換預留執行個體具有相同的期間 (1 年或 3 年)。若您合併多個具有不同期間長度的可轉換預留執行個體，新的可轉換預留執行個體將具有 3 年的期間。如需詳細資訊，請參閱 [合併可轉換預留執行個體](#)。
- Amazon EC2 交換可轉換預留執行個體時，會淘汰相關的預留，並將結束日期轉移到新的預留。交換後，Amazon EC2 會將舊預留的結束日期和新預留的開始日期設定為等於交換日期。例如，若您交換具有 16 個月剩餘期間的三年預留，新預留是 16 個月預留，其結束日期與您交換的可轉換預留執行個體的預留相同。

## 計算可轉換預留執行個體交換

交換可轉換預留執行個體是免費的。但是，您可能需要支付校正成本，即您擁有的原始可轉換預留執行個體與您從交換中獲得新可轉換預留執行個體之間的差異，依比例分配的預付成本。

每個可轉換預留執行個體都具有清單值。此清單值會和您希望取得的可轉換預留執行個體清單值進行比較，以判斷您可以從交換中接收到多少執行個體保留。

例如：您有 1 個清單值為 35 USD 的可轉換預留執行個體，希望和清單值為 10 USD 的執行個體類型交換。

$$\$35/\$10 = 3.5$$

您可以使用三個 10 USD 的可轉換預留執行個體交換您的可轉換預留執行個體。因為您無法購買半個保留，因此您必須購買額外的可轉換預留執行個體才能涵蓋剩餘的部分：

3.5 = 3 whole Convertible Reserved Instances + 1 additional Convertible Reserved Instance

第四個可轉換預留執行個體具有和其他三個可轉換預留執行個體相同的結束日期。若您要交換部分或全預付可轉換預留執行個體，您必須為第四個保留支付校正成本。若您可轉換預留執行個體剩餘的預付成本為 500 USD，則依比例分配，新保留通常會需要花費 600 USD。您將需要支付 100 USD 的費用。

\$600 prorated upfront cost of new reservations - \$500 remaining upfront cost of old reservations = \$100 difference

### 合併可轉換預留執行個體

若您合併兩個或多個可轉換預留執行個體，新可轉換預留執行個體的期間必須和舊可轉換預留執行個體相同，或為舊可轉換預留執行個體中最高的值。新可轉換預留執行個體的過期日期將會是距離現在最遠的日期。

例如，您的帳戶中有下列可轉換預留執行個體：

| Reserved Instance ID | 術語  | 過期日期       |
|----------------------|-----|------------|
| aaaa1111             | 1 年 | 2018-12-31 |
| bbbb2222             | 1 年 | 2018-07-31 |
| cccc3333             | 3 年 | 2018-06-30 |
| dddd4444             | 3 年 | 2019-12-31 |

- 您可以合併 aaaa1111 和 bbbb2222，並以它們交換 1 年的可轉換預留執行個體。您無法以它們交換 3 年的可轉換預留執行個體。新可轉換預留執行個體的過期日期為 2018-12-31。
- 您可以合併 bbbb2222 和 cccc3333，並以它們交換 3 年的可轉換預留執行個體。您無法以它們交換 1 年的可轉換預留執行個體。新可轉換預留執行個體的過期日期為 2018-07-31。
- 您可以合併 cccc3333 和 dddd4444，並以它們交換 3 年的可轉換預留執行個體。您無法以它們交換 1 年的可轉換預留執行個體。新可轉換預留執行個體的過期日期為 2019-12-31。

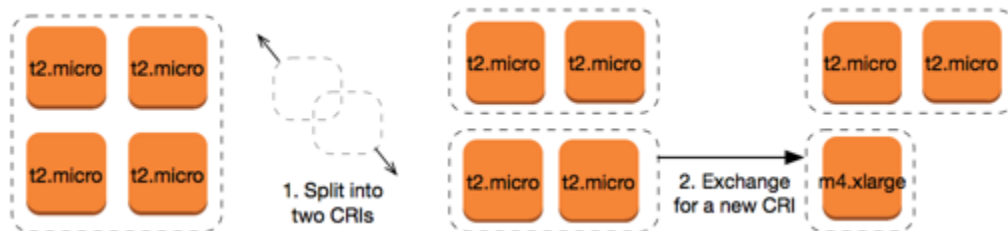
## 交換 可轉換預留執行個體 的一部分

您可以使用修改程序將您的可轉換預留執行個體分割為較小的保留，然後以一或多個新保留交換新的可轉換預留執行個體。下列範例示範執行此作業的方式。

### Example 範例：具有多個執行個體的可轉換預留執行個體

在此範例中，您有一個保留中有四個執行個體的 t2.micro 可轉換預留執行個體。若要以兩個 t2.micro 執行個體交換一個 m4.xlarge 執行個體：

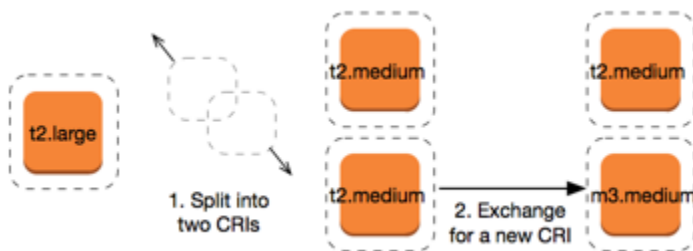
1. 修改 t2.micro 可轉換預留執行個體，方式為將其分割為兩個 t2.micro 可轉換預留執行個體，各自具有兩個執行個體。
2. 以其中一個新的 t2.micro 可轉換預留執行個體交換 m4.xlarge 可轉換預留執行個體。



### Example 範例：具有單一執行個體的可轉換預留執行個體

在此範例中，您有一個 t2.large 可轉換預留執行個體。若要將其變更為較小的 t2.medium 執行個體和 m3.medium 執行個體：

1. 修改 t2.large 可轉換預留執行個體，方式為將其分割為兩個 t2.medium 可轉換預留執行個體。執行個體大小使用量與兩個 t2.large 執行個體相同的單一 t2.medium 執行個體。
2. 以其中一個新的 t2.medium 可轉換預留執行個體交換 m3.medium 可轉換預留執行個體。



如需詳細資訊，請參閱 [支援執行個體大小的修改](#) 及 [提交交換請求](#)。



## 提交交換請求

您可以使用 Amazon EC2 主控台或命令列工具交換您的可轉換預留執行個體。

### 使用主控台交換 可轉換預留執行個體

您可以搜尋可轉換預留執行個體優惠，然後從提供的選取中選取您的新組態。

#### New console

使用 Amazon EC2 主控台交換可轉換預留執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 選取 Reserved Instances (預留執行個體)、選取要交換的可轉換預留執行個體，然後選取 Actions (動作)，Exchange 預留執行個體 (交換預留執行個體)。
3. 選取所需組態的屬性，然後選取 Find offering (尋找優惠)。
4. 選取新的 可轉換預留執行個體。在畫面底部，您可以檢視兌換所收到的 預留執行個體 數量，以及任何額外費用。
5. 當您選取符合您需求的 可轉換預留執行個體 時，請選取 Review (檢閱)。
6. 選擇 Exchange (交換)，然後選擇 Close (關閉)。

#### Old console

使用 Amazon EC2 主控台交換可轉換預留執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 選取 Reserved Instances (預留執行個體)、選取要交換的可轉換預留執行個體，然後選取 Actions (動作)，Exchange 預留執行個體 (交換預留執行個體)。
3. 選取所需組態的屬性，然後選取 Find Offering (尋找優惠)。
4. 選取新的 可轉換預留執行個體。Instance Count (執行個體計數) 資料欄會顯示您可以從交換中接收到的 預留執行個體 數目。當您選取符合您需求的可轉換預留執行個體時，請選取 Exchange (交換)。

交換的預留執行個體會遭到淘汰，新的預留執行個體會顯示在 Amazon EC2 主控台中。此程序需要幾分鐘的時間散佈。

## 使用命令列介面交換 可轉換預留執行個體

若要交換 可轉換預留執行個體，請先尋找符合您需求的新 可轉換預留執行個體：

- [describe-reserved-instances-offerings](#) (AWS CLI)
- [Get-EC2ReservedInstancesOffering](#)(視窗工具 PowerShell)

取得交換的報價，其中包含您可以從交換中取得的預留執行個體數目，以及交換的校正成本：

- [get-reserved-instances-exchange-quote](#) (AWS CLI)
- [截取 C2-ReservedInstances ExchangeQuote](#) (視窗專用工具) PowerShell

最後，執行交換：

- [accept-reserved-instances-exchange-quote](#) (AWS CLI)
- [Confirm-EC2ReservedInstancesExchangeQuote](#)(視窗工具 PowerShell)

## 預留執行個體配額

您可以每月購買新的預留執行個體。您每月可購買的新預留執行個體數量取決於您的每月配額，如下所示：

| 配額說明                         | 預設配額         |
|------------------------------|--------------|
| 新的 <a href="#">區域</a> 預留執行個體 | 每月每區域 20 個   |
| 新的 <a href="#">地區</a> 預留執行個體 | 每月每可用區域 20 個 |

例如，在具有 3 個可用區域的區域中，預設配額為每月 80 個新的預留執行個體 (計算方式如下)：

- 該區域的 20 個區域預留執行個體
- 加上 60 個地區預留執行個體 (三個可用區域各 20 個)

running狀態中的執行個體會計入您的配額中。位

於pending、stopping、stopped和hibernated狀態的執行個體不會計入您的配額中。

## 檢視您已購買的預留執行個體數

您購買的預留執行個體數量由 Instance count (執行個體計數) 欄位 (主控台) 或 InstanceCount 參數 (AWS CLI) 來指示。當您購買新的預留執行個體時，會根據執行個體總數來測量配額。例如，如果您購買執行個體數量為 10 的單一預留執行個體組態，則該購買所佔配額將為 10，而非 1。

可以使用 Amazon EC2 或 AWS CLI 來檢視您已購買了多少個預留執行個體。

### Console

若要檢視您已購買的預留執行個體數

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)。
3. 從資料表中選取預留執行個體組態，然後勾選 Instance count (執行個體計數) 欄位。

在下列螢幕擷取畫面中，所選行表示 t3.micro 執行個體類型的單一預留執行個體組態。資料表檢視中的 Instance count (執行個體計數) 欄和詳細資訊檢視中的 Instance count (執行個體計數) 欄位指示此組態有 10 個預留執行個體。

EC2 > Reserved Instances

Reserved Instances (32) Info

Filter by attributes or search by keyword

| Instance ty...                               | Scope  | Availabilit... | Instance count | Start                              | Expires                            | Offering cl... |
|----------------------------------------------|--------|----------------|----------------|------------------------------------|------------------------------------|----------------|
| <input checked="" type="checkbox"/> t3.micro | Region | -              | 10             | August 27, 2022, 15:29 (UTC+2:00)  | August 27, 2023, 15:29 (UTC+2:00)  | Standard       |
| <input type="checkbox"/> t3.micro            | Region | -              | 4              | November 8, 2021, 14:19 (UTC+2:00) | November 8, 2022, 14:19 (UTC+2:00) | Standard       |

1 Reserved Instance selected

Details My Listings

Reserved Instance ID: 2fbf16dd-98b6-4a3a-955f-83f87790f04b Info

|                                            |                                     |                                              |                            |
|--------------------------------------------|-------------------------------------|----------------------------------------------|----------------------------|
| Instance type<br>t3.micro                  | Scope<br>Region                     | Instance count<br>10                         | Availability Zone<br>-     |
| Start<br>August 27, 2022, 15:29 (UTC+2:00) | Platform<br>Linux/UNIX              | Expires<br>August 27, 2023, 15:29 (UTC+2:00) | Term<br>1 year             |
| Payment option<br>All upfront              | Time left<br>around 50 weeks 6 days | Upfront price<br>\$59.00                     | Offering class<br>Standard |
| Usage price<br>\$0.00                      | State<br>Active                     | Hourly charges<br>\$0.00                     | Tenancy<br>Default         |

### AWS CLI

若要檢視您已購買的預留執行個體數

使用 [describe-reserved-instances](#) CLI 命令並指定預留執行個體組態的 ID。

```
aws ec2 describe-reserved-instances \
 --reserved-instances-ids 2fbf16dd-98b6-4a3a-955f-83f87790f04b \
 --output table
```

輸出範例 - InstanceCount 欄位指示此組態有 10 個預留執行個體。

```

| DescribeReservedInstances |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|| ReservedInstances ||
|+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
	CurrencyCode	USD	
	Duration	31536000	
	End	2023-08-27T13:29:44+00:00	
	FixedPrice	59.0	
	InstanceCount	10	
	InstanceTenancy	default	
	InstanceType	t3.micro	
	OfferingClass	standard	
	OfferingType	All Upfront	
	ProductDescription	Linux/UNIX	
	ReservedInstancesId	2fbf16dd-98b6-4a3a-955f-83f87790f04b	
	Scope	Region	
	Start	2022-08-27T13:29:45.938000+00:00	
	State	active	
	UsagePrice	0.0	
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+			
		RecurringCharges	
	+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+		
		Amount	0.0
		Frequency	Hourly
	+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+		

```

## 考量事項

區域預留執行個體會將折扣套用至執行中隨需執行個體。預設隨需執行個體限制為 20。您無法透過購買區域預留執行個體來超出執行中隨需執行個體限制。例如，如果您已有 20 個執行中隨需執行個體，而且您購買 20 個區域預留執行個體，則 20 個區域預留執行個體是用來將折扣套用至 20 個執行中隨需

執行個體。如果您購買更多的區域預留執行個體，將無法啟動更多的執行個體，因為您已達到隨需執行個體限制。

在購買地區預留執行個體之前，請確認您的隨需執行個體限制符合或超過您想要擁有的地區預留執行個體數。如有必要，請務必先請求調高您的隨需執行個體限制，再購買更多地區預留執行個體。

區域性預留執行個體 (針對特定可用區域購買的預留執行個體) 會提供容量保留以及折扣。您可以透過購買區域性預留執行個體來超出執行中隨需執行個體限制。例如，如果您已有 20 個執行中隨需執行個體，而且購買 20 個區域性預留執行個體，則您可以另外啟動 20 個符合區域性預留執行個體之規格的隨需執行個體，如此總共有 40 個執行中執行個體。

### 檢視預留執行個體配額並請求增加配額

Amazon EC2 主控台提供了配額資訊。您還可以請求增加您的配額。如需詳細資訊，請參閱 [檢視您目前的配額](#) 及 [請求增加](#)。

## Spot 執行個體

Spot 執行個體是使用備用 EC2 容量的執行個體，可在低於隨需價格時使用它。競價型執行個體可讓您以相當優惠的折扣，來請求未使用的 EC2 執行個體，進而大幅降低 Amazon EC2 成本。Spot 執行個體的每小時價格稱為 Spot 價格。各可用區域中各執行個體類型的 Spot 價格由 Amazon EC2 決定，並會根據競價型執行個體的長期供需情況逐漸調整。只要有容量可用，您的 Spot 執行個體便會執行。

如能彈性決定應用程式執行的時機以及應用程式是否可以中斷，則競價型執行個體會是一項具成本效益的選擇。例如，競價型執行個體非常適合用於資料分析、批次處理任務、背景處理和選擇性的任務。如需詳細資訊，請參閱 [Amazon EC2 競價型執行個體](#)。

如需 EC2 執行個體不同購買選項的比較，請參閱 [執行個體購買選項](#)。

### 主題

- [概念](#)
- [如何開始](#)
- [相關服務](#)
- [定價與節省](#)

### 概念

在開始使用 Spot 執行個體之前，您應該先熟悉以下概念：

- Spot 容量集區 – 一組具有相同執行個體類型 (例如 m5.large) 和可用區域的未使用 EC2 執行個體。
- Spot 價格 – Spot 執行個體目前的每小時價格。
- Spot 執行個體請求 – 請求 Spot 執行個體。當容量可用時，Amazon EC2 會履行您的請求。Spot 執行個體請求為一次性或持久性。與請求相關聯的 Spot 執行個體中斷之後，Amazon EC2 會自動重新提交持續的 Spot 執行個體請求。
- EC2 執行個體重新平衡建議 - Amazon EC2 發出執行個體重新平衡建議訊號，以通知您 Spot 執行個體的中斷風險升高。此訊號讓您有機會在現有或新的 Spot 執行個體上主動重新平衡工作負載，而無需等待兩分鐘的 Spot 執行個體中斷通知。
- Spot 執行個體中斷 – 當 Amazon EC2 需要取回容量，Amazon EC2 會終止、停止 Spot 執行個體或將其休眠。Amazon EC2 會提供 Spot 執行個體中斷通知，在執行個體中斷前會向執行個體發出兩分鐘的警告。

## 競價型執行個體與隨需執行個體之間的主要差異

下表列出了 Spot 執行個體與[隨需執行個體](#)之間的主要差異。

|        | Spot Instances                                                                                  | On-Demand Instances                 |
|--------|-------------------------------------------------------------------------------------------------|-------------------------------------|
| 啟動時間   | 只有當 Spot 執行個體請求正在作用中，而且有可用的容量時，才能立即啟動。                                                          | 只有在進行手動啟動請求，而且有可用的容量時，才能立即啟動。       |
| 可用容量   | 如果沒有可用的容量，則 Spot 執行個體請求會繼續自動發出啟動請求，直到有可用的容量出現。                                                  | 如果在您發出啟動請求時沒有可用的容量，會出現容量不足錯誤 (ICE)。 |
| 每小時價格  | Spot 執行個體的每小時價格會隨長期的供給和需求而有所不同。                                                                 | 隨需執行個體的每小時價格為靜態定價。                  |
| 重新平衡建議 | 執行個體的中斷風險升高時，Amazon EC2 為執行中的 Spot 執行個體發出的訊號。                                                   | 您可以決定中斷 (停止、休眠或終止) 隨需執行個體的時間點。      |
| 執行個體中斷 | 您可以停止和啟動 Amazon EBS 支援的 Spot 執行個體。此外，如果不再有可用的容量，Amazon EC2 還可以 <a href="#">中斷</a> 個別 Spot 執行個體。 | 您可以決定中斷 (停止、休眠或終止) 隨需執行個體的時間點。      |

## 如何開始

您首先必須進行設定以使用 Amazon EC2。在啟動 競價型執行個體之前，先獲得啟動隨需執行個體的經驗，也會很有幫助。

### Spot 基本概念

- [競價型執行個體的運作方式](#)

## 使用 競價型執行個體

- [建立 Spot 執行個體請求](#)
- [取得請求狀態資訊](#)
- [Spot 執行個體中斷](#)

## 相關服務

您可以使用 Amazon EC2 來直接佈建 競價型執行個體。您也可以使用 AWS 中的其他服務來佈建 Spot 執行個體。如需詳細資訊，請參閱下列文件。

### Amazon EC2 Auto Scaling 和 競價型執行個體

您可以建立啟動範本或組態，以便 Amazon EC2 Auto Scaling 可以啟動 Spot 執行個體。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南 中 [請求 競價型執行個體 取得容錯和彈性應用程式與具有多種執行個體類型和購買選項的 Auto Scaling 群組](#)。

### Amazon EMR 和 競價型執行個體

某些情境下，在 Amazon EMR 叢集中執行 競價型執行個體會很有幫助。如需詳細資訊，請參閱 Amazon EMR 管理指南中的 [競價型執行個體](#) 和 [應使用 競價型執行個體的時機](#)。

### AWS CloudFormation 模板

AWS CloudFormation 可讓您使用 JSON 格式的範本建立和管理 AWS 資源集合。如需詳細資訊，請參閱 [EC2 Spot 執行個體更新-Auto Scaling 和 CloudFormation 整合](#)。

### AWS SDK for Java

您可以使用 Java 程式設計語言來管理您的 競價型執行個體。如需詳細資訊，請參閱 [教學課程：Amazon EC2 競價型執行個體](#) 和 [教學課程：進階 Amazon EC2 Spot 請求管理](#)。

### AWS SDK for .NET

您可以使用 .NET 程式設計語言環境來管理您的 競價型執行個體。如需詳細資訊，請參閱 [教學課程：Amazon EC2 競價型執行個體](#)。

## 定價與節省

您會為 競價型執行個體支付 Spot 價格，該價格由 Amazon EC2 決定，並會根據 競價型執行個體的長期供需情況逐漸調整。在您終止 Spot 執行個體前、不再有可用的容量前，或是 Amazon EC2 Auto Scaling 群組在 [縮減](#) 時終止 Spot 執行個體前，您的 Spot 執行個體會持續執行。



如果您或 Amazon EC2 中斷執行中的 Spot 執行個體，則會按使用秒數或整整一小時向您收費，也可能免費，根據使用的作業系統和中斷 Spot 執行個體的人而定。如需詳細資訊，請參閱 [針對已中斷的競價型執行個體收費](#)。

Savings Plans 不涵蓋 Spot 執行個體。如果您有 Savings Plan，除了使用 Spot 執行個體所節省的成本之外，它不會提供額外的節省成本。此外，您在 Spot 執行個體上的支出不適用於 Compute Savings Plans 中的承諾。

## 檢視價格

若要檢視每個執行個體類型目前 (每 AWS 區域 五分鐘更新一次) 的最低競價型價格，請參閱 [Amazon EC2 競價型執行個體定價](#) 頁面。

若要檢視過去三個月的 Spot 價格歷史，請使用 Amazon EC2 主控台或 [describe-spot-price-history](#) 命令 (AWS CLI)。如需詳細資訊，請參閱 [Spot 執行個體定價歷史記錄](#)。

我們獨立地將可用區域映射到每個代碼 AWS 帳戶。如此，不同帳戶之間的相同可用區域代碼 (例如 us-west-2a)，就會得到不同的結果。

## 檢視節省

您可以檢視針對單一 [Spot 機群](#) 或所有 Spot 執行個體使用 Spot 執行個體所節省的成本。您可以檢視前一個小時或前三天的節省，也可以檢視每 vCPU 小時和每記憶體 (GiB) 小時的平均成本。節省是估計值，因為不含用量的帳單調整，所以會與實際的節省有所出入。如需檢視節省資訊的詳細資訊，請參閱 [購買 競價型執行個體省下的成本](#)。

## 檢視帳單

您的帳單會提供您服務用量的詳細資訊。如需詳細資訊，請參閱 AWS Billing 使用者指南中的 [檢視您的帳單](#)。

## EC2 最佳實務

Amazon EC2 競價型執行個體是備用 EC2 運算容量 AWS 雲端，相較於隨需價格，您可以節省高達 90% 的折扣。隨需執行個體與競價型執行個體之間唯一的差別在於，當 Amazon EC2 需要取回容量時，競價型執行個體可以由 Amazon EC2 利用兩分鐘的通知來中斷。

競價型執行個體建議用於無狀態、容錯、靈活的應用程式。例如，競價型執行個體適用於大數據、容器化工作負載、CI/CD、無狀態 Web 伺服器、高效能運算 (HPC) 和轉譯工作負載。

執行時，競價型執行個體與隨需執行個體完全相同。但是，Spot 並無法保證您可以保持執行中的執行個體有足夠時間來完成工作負載。Spot 也無法保證您可以立即取得所尋找執行個體的可用性，也無

法保證您一律可以取得所請求的彙總容量。此外，Spot 執行個體中斷和容量可能會隨著時間而改變，因為 Spot 執行個體可用性會因供給和需求而有所不同，而過去的效能並不能保證未來的結果。

競價型執行個體 不適合執行個體節點之間不靈活、可設定狀態、錯誤不容忍或緊密結合的工作負載。對於無法容忍整個目標容量無法完全可用的偶爾期間的工作負載，我們不建議使用 Spot 執行個體。儘管遵循 Spot 最佳實務可靈活調整執行個體類型，而可用區域可提供高可用性的最佳機會，但無法保證容量可用，因為隨需執行個體的需求激增可能會中斷 Spot 執行個體上的工作負載。

我們強烈建議將 Spot 執行個體用於這些工作負載，或嘗試容錯移轉到隨需執行個體以處理中斷或無法使用的期間。容錯移轉到隨需執行個體可能會無意中導致其他 Spot 執行個體的中斷。此外，如果執行個體類型和可用區域組合的 Spot 執行個體遭到中斷，您可能很難獲得具有相同組合的隨需執行個體。

無論您是經驗豐富的 Spot 使用者或 Spot 執行個體的新使用者，如果您目前遇到 Spot 執行個體中斷或可用性問題，建議您遵循這些最佳做法，以獲得使用 Spot 服務的最佳體驗。

## Spot 最佳實務

- [針對中斷準備個別執行個體](#)
- [對執行個體類型和可用區域具有彈性](#)
- [使用 EC2 Auto Scaling 群組或 EC2 Fleet 來管理彙總容量](#)
- [使用價格和容量最佳化分配策略](#)
- [使用整合式 AWS 服務來管理 Spot 執行個體](#)
- [使用哪種 Spot 請求方法最好？](#)

## 針對中斷準備個別執行個體

從容處理 Spot 執行個體中斷的最佳方法，就是建立容錯的應用程式架構。若要達到此目的，您可以利用 EC2 執行個體重新平衡建議和 Spot 執行個體中斷通知。

EC2 執行個體重新平衡推薦是一種訊號，可在 Spot 執行個體的中斷風險升高時通知您。該訊號使您有機會在兩分鐘 Spot 執行個體中斷通知之前主動管理 Spot 執行個體。您可以決定將工作負載重新平衡至未處於提高之中斷風險的新的或現有 競價型執行個體。我們已經使用 Auto Scaling 群組和 EC2 Fleet 中的容量重新平衡功能，讓您輕鬆地使用此訊號。

Spot 執行個體中斷通知是在 Amazon EC2 中斷 Spot 執行個體的兩分鐘之前發出的警告。如果您的工作負載具有「時間彈性」，當執行個體中斷時，您可以將它們設定為停止或休眠，而非終止。Amazon EC2 會在中斷時自動停止或休眠 Spot 執行個體，並在我們有可用容量時自動繼續執行個體。

我們建議您在 [Amazon EventBridge](#) 中建立擷取重新平衡建議和中斷通知的規則，然後觸發檢查點以檢查工作負載的進度，或妥善處理中斷。如需詳細資訊，請參閱 [監控重新平衡建議訊號](#)。如需如何建立和使用事件規則的詳細範例，請參閱 [利用 Amazon EC2 Spot 執行個體中斷通知](#)。

如需詳細資訊，請參閱 [EC2 執行個體重新平衡建議](#) 及 [Spot 執行個體中斷](#)。

對執行個體類型和可用區域具有彈性

Spot 容量集區是一組未使用的 EC2 執行個體，具有相同執行個體類型 (例如 m5.large) 和可用區域 (例如 us-east-1a)。您應該對於請求的執行個體類型，以及可在其中部署工作負載的可用區域具有彈性。這讓 Spot 有更好的機會找到並配置您所需的運算容量。例如，不要只要求 c5.large 是否願意使用來自 c4、m5 和 m4 系列的 large。

根據您的特定需求，您可以評估哪些執行個體類型可以具彈性，以滿足您的運算需求。如果工作負載可以垂直擴展，則應在請求中包含較大的執行個體類型 (更多 vCPU 和記憶體)。如果您只能水平擴展，則應納入舊一代執行個體類型，因為隨需客戶的需求較少。

一個很好的經驗法則是針對每個工作負載的至少 10 個執行個體類型上具有彈性。此外，請確定所有可用區域均已設定為在 VPC 中使用，並針對您的工作負載選取。

使用 EC2 Auto Scaling 群組或 EC2 Fleet 來管理彙總容量

Spot 可讓您以彙總容量 (包括 vCPU、記憶體、儲存或網路輸送量單位) 的方式來思考，而不是以個別執行個體的方式來思考。Auto Scaling 群組和 EC2 Fleet 可讓您啟動和維護目標容量，並自動請求資源以取代中斷或手動終止的任何資源。設定 Auto Scaling 群組或 EC2 Fleet 時，只需要根據應用程式需求指定執行個體類型和目標容量。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南 中的 [Auto Scaling 群組](#) 和本使用者指南中的 [建立 EC2 Fleet](#)。

使用價格和容量最佳化分配策略

Auto Scaling 群組中的分配策略可協助您佈建目標容量，而無需手動尋找具有備用容量的 Spot 容量集區。建議使用 price-capacity-optimized 策略，因為此策略會自動從最可用且價格最低的 Spot 容量集區佈建執行個體。您也可以利用 EC2 Fleet 中的 price-capacity-optimized 配置策略。由於您的 Spot 執行個體容量來自具有最佳容量的集區，因此可降低回收 Spot 執行個體的可能性。如需有關分配策略的詳細資訊，請參閱《Amazon EC2 Auto Scaling 使用者指南》中的 [Spot 執行個體](#) 和本使用者指南中的 [當工作負載具有較高的中斷成本時](#)。

使用整合式 AWS 服務來管理 Spot 執行個體

其他 AWS 服務與 Spot 整合，可降低整體運算成本，而無需管理個別執行個體或叢集。我們建議您針對適用的工作負載考慮下列解決方案：Amazon EMR、Amazon 彈性容器服務 AWS Batch、Amazon

Elastic Kubernetes Service SageMaker、AWS Elastic Beanstalk Amazon 和 Amazon。GameLift 若要深入了解搭配這些服務的 Spot 最佳做法，請參閱 [Amazon EC2 競價型執行個體 研討會網站](#)。

使用哪種 Spot 請求方法最好？

使用下表確定在請求 Spot 執行個體時使用哪個 API。

| API                                    | 何時使用？                                                                                                                                        | 使用案例                                                                                                                                                               | 我應該使用此 API 嗎？                  |
|----------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------|
| <a href="#">CreateAutoScalingGroup</a> | <ul style="list-style-type: none"> <li>您需要具有單一組態或混合組態的多個執行個體。</li> <li>您希望透過可設定的 API 自動執行生命週期管理。</li> </ul>                                  | 建立 Auto Scaling 群組，管理執行個體的生命週期，同時維護所需的執行個體數量。支援在指定的最小和最大限制之間水平擴展 (新增更多執行個體)。                                                                                       | 是                              |
| <a href="#">CreateFleet</a>            | <ul style="list-style-type: none"> <li>您需要具有單一組態或混合組態的多個執行個體。</li> <li>您希望自行管理您的執行個體生命週期。</li> <li>如果您不需要自動擴展，建議您使用 instant 類型機群。</li> </ul> | 在單一請求中建立隨需執行個體和 Spot 執行個體的機群，並具有隨執行個體類型、AMI、可用區域或子網而異的多個啟動規格。Spot 執行個體分配策略預設為每單位 lowest-price，但您可以將其變更為 price-capacity-optimized、capacity-optimized 或 diversified。 | 是 – 在 instant 模式中 (如果您不需要自動擴展) |
| <a href="#">RunInstances</a>           | <ul style="list-style-type: none"> <li>您已經在使用 RunInstances API</li> </ul>                                                                    |                                                                                                                                                                    |                                |

| API                                  | 何時使用？                                                                                                                                                                                                           | 使用案例                                         | 我應該使用此 API 嗎？                                |
|--------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------|----------------------------------------------|
|                                      | <p>啟動隨需執行個體，而且只想變更單一參數來變更為啟動 Spot 執行個體。</p> <ul style="list-style-type: none"> <li>• 您不需要具有不同執行個體類型的多個執行個體。</li> </ul>                                                                                          | <p>使用 AMI 和一個執行個體類型，啟動指定數量的執行個體。</p>         | <p>否 — 因為 RunInstances 不允許在單一要求中混合執行個體類型</p> |
| <p><a href="#">RequestSpot艦隊</a></p> | <ul style="list-style-type: none"> <li>• 我們強烈建議使用 RequestSpotFleet API，因為它是沒有計劃投資的舊版 API。</li> <li>• 如果您想要管理執行個體生命週期，請使用 CreateFleet API。</li> <li>• 如果您不想管理執行個體生命週期，請使用 CreateAutoScalingGroup API。</li> </ul> | <p>不要使用。RequestSpotFleet 是沒有計劃投資的傳統 API。</p> | <p>否</p>                                     |

| API                            | 何時使用？                                                                                                  | 使用案例                                      | 我應該使用此 API 嗎？ |
|--------------------------------|--------------------------------------------------------------------------------------------------------|-------------------------------------------|---------------|
| <a href="#">RequestSpot 實例</a> | <ul style="list-style-type: none"> <li>我們強烈建議使用 RequestSpotInstances API，因為它是沒有計劃投資的舊版 API。</li> </ul> | 不要使用。RequestSpotInstances 是沒有計劃投資的傳統 API。 | 否             |

## 競價型執行個體的運作方式

若要啟動 Spot 執行個體，您可以建立 Spot 執行個體請求，或 Amazon EC2 代表您建立 Spot 執行個體請求。Spot 執行個體請求完成後，會啟動 Spot 執行個體。

您可以使用數個不同的服務來啟動 Spot 執行個體。如需詳細資訊，請參閱 [Amazon EC2 Spot 執行個體入門](#)。在本使用者指南中，我們描述了使用 EC2 啟動 Spot 執行個體的以下方法：

- 您可以使用 Amazon EC2 主控台中的 [啟動執行個體精靈或執行個體命令](#) 來建立競價型執行個體 AWS CLI 請求。如需詳細資訊，請參閱 [建立 Spot 執行個體請求](#)。
- 您可以建立 EC2 Fleet，在其中指定所需數量的 Spot 執行個體。Amazon EC2 為 EC2 Fleet 中指定的每個 Spot 執行個體建立 Spot 執行個體請求。如需詳細資訊，請參閱 [建立 EC2 Fleet](#)。
- 您可以建立 Spot Fleet 請求，在其中指定所需數量的 Spot 執行個體。Amazon EC2 為 Spot Fleet 請求中指定的每個 Spot 執行個體建立 Spot 執行個體請求。如需詳細資訊，請參閱 [建立 Spot Fleet 請求](#)。

如果有可用容量，您的 Spot 執行個體就會啟動。

您的 Spot 執行個體會執行直到您停止或終止它，或者直到 Amazon EC2 中斷它 (稱為 Spot 執行個體中斷)。

使用 Spot 執行個體時，您必須對中斷做好準備。當 Spot 執行個體的需求增加，或是 Spot 執行個體的供給減少時，Amazon EC2 可以中斷您的 Spot 執行個體。當 Amazon EC2 中斷 Spot 執行個體時，會提供 Spot 執行個體中斷通知，在 Amazon EC2 中斷執行個體前，向該執行個體發出兩分鐘的警告。您無法為競價型執行個體啟用終止保護。如需詳細資訊，請參閱 [Spot 執行個體中斷](#)。

您可以停止、啟動、重新啟動或終止 Amazon EBS 支援的執行個體。Spot 服務可以在 Spot 執行個體中斷時停止、終止或休眠它。

## 內容

- [在啟動群組中啟動 競價型執行個體](#)
- [在可用區域群組中啟動 競價型執行個體](#)
- [在 VPC 中啟動 競價型執行個體](#)

### 在啟動群組中啟動 競價型執行個體

在 Spot 執行個體請求中指定啟動群組，告知 Amazon EC2 啟動一組 Spot 執行個體，但前提它能夠啟動其中的所有執行個體。此外，如果 Spot 服務必須終止啟動群組內的其中一個執行個體，則必須終止群組中所有的執行個體。不過，如果您終止啟動群組內的一個或多個執行個體，Amazon EC2 不會終止該啟動群組中其餘的執行個體。

雖然此選項很實用，但加上此限制可能會降低完成 Spot 執行個體請求的機率，並提高終止 Spot 執行個體的機率。例如，您的啟動群組包含多個可用區域的執行個體。如果其中一個可用區域的容量減少，且不再提供使用，則 Amazon EC2 會終止該啟動群組的所有執行個體。

如果建立另一個成功的 Spot 執行個體請求，它指定與先前的成功請求相同的 (現有的) 啟動群組，則新的執行個體會加入該啟動群組。之後，如果這個啟動群組中的執行個體遭到終止，則該啟動群組中的所有執行個體都會終止，這也包括由第一個和第二個請求啟動的執行個體。

### 在可用區域群組中啟動 競價型執行個體

在您的 Spot 執行個體請求中指定可用區域群組，以告知 Amazon EC2 在同一個 Availability Zone (可用區域) 中啟動一組 Spot 執行個體。Amazon EC2 不需同時中斷可用區域群組中的所有執行個體。如果 Amazon EC2 必須中斷可用區域群組中的其中一個執行個體，則其他的執行個體仍會繼續執行。

雖然此選項很實用，但加上這項限制可能會降低完成 Spot 執行個體請求的機率。

如果指定了可用區域群組，但未在 Spot 執行個體請求中指定可用區域，則結果取決於您指定的網路。

## 預設 VPC

Amazon EC2 會使用指定子網適用的可用區域。如未指定子網，它會替您選取一個可用區域及其預設子網，但不一定是最低價的區域。如果您刪除可用區域適用的預設子網，則必須指定不同的子網。

## 非預設 VPC

Amazon EC2 會使用指定子網適用的可用區域。

在 VPC 中啟動 競價型執行個體

您可以用為 競價型執行個體指定子網的同一種方法，來為 隨需執行個體指定子網。

- [預設 VPC] 如果希望在指定的低價可用區域中啟動 Spot 執行個體，您必須在 Spot 執行個體請求中指定對應的子網。如果未指定子網，Amazon EC2 會替您選取一個，而這個子網的可用區域可能不具有最低的 Spot 價格。
- [非預設 VPC] 您必須指定 Spot 執行個體適用的子網。

## Spot 執行個體定價歷史記錄

Spot 執行個體的價格由 Amazon EC2 制定，然後根據 Spot 執行個體容量的長期供需趨勢逐漸調整。

滿足您的 Spot 請求後，您會以目前的 Spot 價格啟動 Spot 執行個體，而不會超過隨需價格。您可以檢視過去 90 天的 Spot 價格歷史，根據執行個體類型、作業系統和可用區域來進行篩選。

檢視目前的 Spot 價格

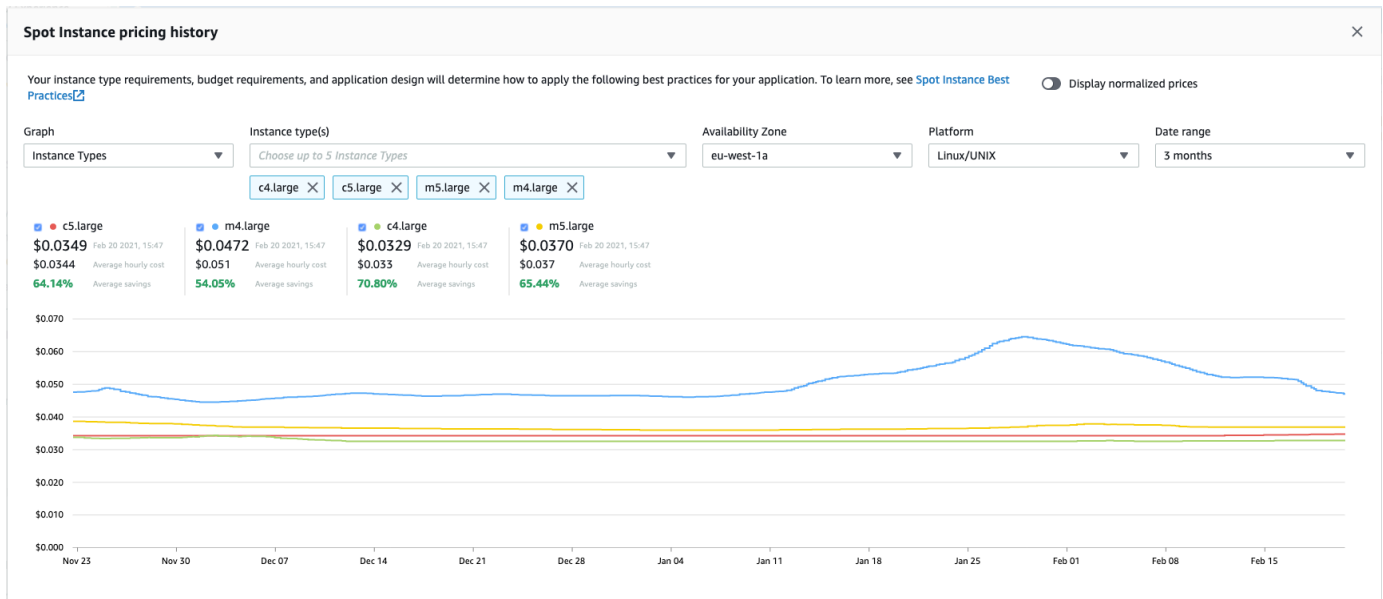
如需目前的 Spot 執行個體價格，請參閱 [Amazon EC2 Spot 執行個體定價](#)。

使用主控台檢視現貨價格歷史記錄

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選擇定價歷史。
4. 在 Graph (圖表) 中，選擇依 Availability Zones (可用區域) 或 Instance Types (執行個體類型) 來比較價格歷史記錄。
  - 若您選擇 Availability Zones (可用區域)，則選擇 Instance type (執行個體類型)、作業系統 (Platform (平台))，以及要檢視價格歷史記錄的 Date range (日期範圍)。
  - 若您選擇 Instance Types (執行個體類型)，則選擇最多五種 Instance type(s) (執行個體類型)、Availability Zone (可用區域)、作業系統 (Platform (平台))，以及要檢視價格歷史記錄的 Date range (日期範圍)。

下列螢幕擷取畫面顯示不同執行個體類型的價格比較。





- 將您的指標懸停至圖表上方，可顯示所選取日期範圍中特定時間點的價格。價格顯示在圖表上方的資訊區塊中。顯示於頂列的價格會顯示指定日期的價格。顯示於第二列的價格會顯示所選日期範圍的平均價格。
- 若要顯示每個 vCPU 的價格，請開啟 Display normalized prices (顯示標準化價格)。若要顯示該執行個體類型的價格，請關閉 Display normalized prices (顯示標準化價格)。

## 使用命令列來檢視 Spot 價格歷史

您可以使用下列其中一個命令。如需詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [describe-spot-price-history](#) (AWS CLI)
- [Get-EC2SpotPriceHistory](#) (AWS Tools for Windows PowerShell)

## 購買 競價型執行個體省下的成本

您可檢視每個叢集之 競價型執行個體或所有執行中 競價型執行個體的用量和節省資訊。每個叢集的用量和節省資訊包括叢集啟動及終止的所有執行個體。您可檢視前一小時或前三天的資訊。

Savings (節省) 區段中的下列螢幕擷取畫面顯示 Spot 用量和 Spot Fleet 的節省資訊。

**Spot usage and savings**

|                |            |                |                 |                            |                                |
|----------------|------------|----------------|-----------------|----------------------------|--------------------------------|
| <b>4</b>       | <b>266</b> | <b>700</b>     | <b>\$9.55</b>   | <b>\$2.99</b>              | <b>69%</b>                     |
| Spot Instances | vCPU-hours | Mem(GiB)-hours | On-Demand total | Spot total                 | Savings                        |
|                |            |                |                 | <b>\$0.0112</b>            | <b>\$0.0043</b>                |
|                |            |                |                 | Average cost per vCPU-hour | Average cost per mem(GiB)-hour |

**Details**

|               |                |                    |              |             |
|---------------|----------------|--------------------|--------------|-------------|
| t3.medium (1) | 2 vCPU hours   | 4 mem(GiB)-hours   | \$0.01 total | 70% savings |
| m4.large (1)  | 144 vCPU hours | 576 mem(GiB)-hours | \$2.52 total | 68% savings |
| t2.micro (2)  | 120 vCPU hours | 120 mem(GiB)-hours | \$0.46 total | 70% savings |

您可以檢視下列用量和節省資訊：

- Spot 執行個體— Spot Fleet 啟動和終止的 Spot 執行個體數量。檢視節省摘要時，數字代表所有正在執行的競價型執行個體。
- vCPU-hours (vCPU 時數) – 所選時段內所有競價型執行個體使用的 vCPU 時數。
- Mem(GiB)-hours (記憶體 (GiB) 時數) – 所選時段內所有競價型執行個體使用的 GiB 時數。
- On-Demand total (隨需總額) – 如果在所選時段內啟動這些執行個體做為隨需執行個體使用，所要支付的總金額。
- Spot total (Spot 總額) – 所選時段要支付的總金額。
- Savings (節省) – 不支付隨需價格的節省百分比。
- Average cost per vCPU-hour (vCPU 每小時的平均成本) – 所有競價型執行個體在選取時段內使用 vCPU 的每小時平均成本，計算方式如下： $\text{vCPU 每小時的平均成本} = \text{Spot 總額} / \text{vCPU 時數}$ 。
- 每 Mem 的平均成本 (GiB)-小時 — 所選時間範圍內所有競價型執行個體使用的平均小時成本，計算方式如下： $\text{每 Mem (GiB) 小時的平均每小時成本} = \text{Spot 總計} / \text{Mem (GiB)-小時}$ 。GiBs
- Details (詳細資訊) 表 – 構成 Spot Fleet 的不同執行個體類型 (每種執行個體類型的執行個體數量所佔百分比)。檢視節省摘要時，這些項目構成您所有正在執行的競價型執行個體。

您只能使用 Amazon EC2 主控台檢視節省資訊。

## 使用主控台檢視 Spot 叢集的節省資訊

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上，選擇 Spot Requests (Spot 請求)。
3. 選取 Spot Fleet 請求的 ID 並捲動至 Savings (節省) 區段。

或者，選取 Spot Fleet 請求 ID 旁邊的核取方塊，然後選取 Savings (節省) 索引標籤。

4. 根據預設，頁面會顯示前三天的用量和節省資訊。您可選擇 last hour (前一小時) 或 last three days (前三天)。若為啟動不到一小時的 Spot Fleets，頁面顯示一小時的預估節省。

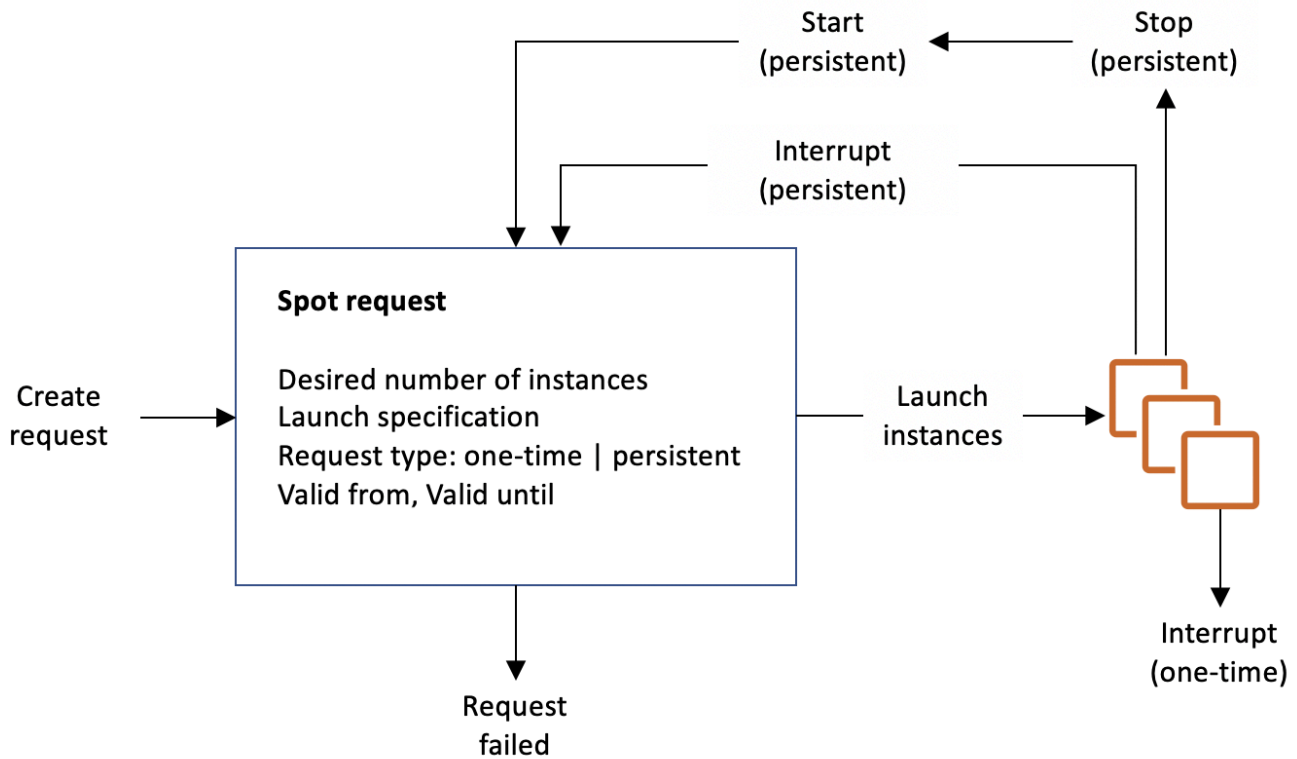
## 使用主控台檢視所有執行中 Spot 執行個體的節省資訊

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格上，選擇 Spot Requests (Spot 請求)。
3. 選擇 Savings Summary (節省摘要)。

## 使用 競價型執行個體

若要使用 Spot 執行個體，您可以建立 Spot 執行個體請求，其中包含所需的執行個體數量、執行個體類型，以及 Availability Zone (可用區域)。如果容量可用，Amazon EC2 會即刻履行您的請求。否則，Amazon EC2 會等到您的請求可以完成，或是您取消請求。

下圖顯示了 Spot 執行個體請求的運作方式。請注意，請求類型 (一次性或持久性) 會決定在 Amazon EC2 中斷 Spot 執行個體時或者如果您停止 Spot 執行個體，是否會再次開啟請求。如果是持久性請求，則會在 Spot 執行個體中斷之後再次開啟。如果是持久性請求，而且您停止 Spot 執行個體，只會在您啟動 Spot 執行個體後開啟請求。



## 目錄

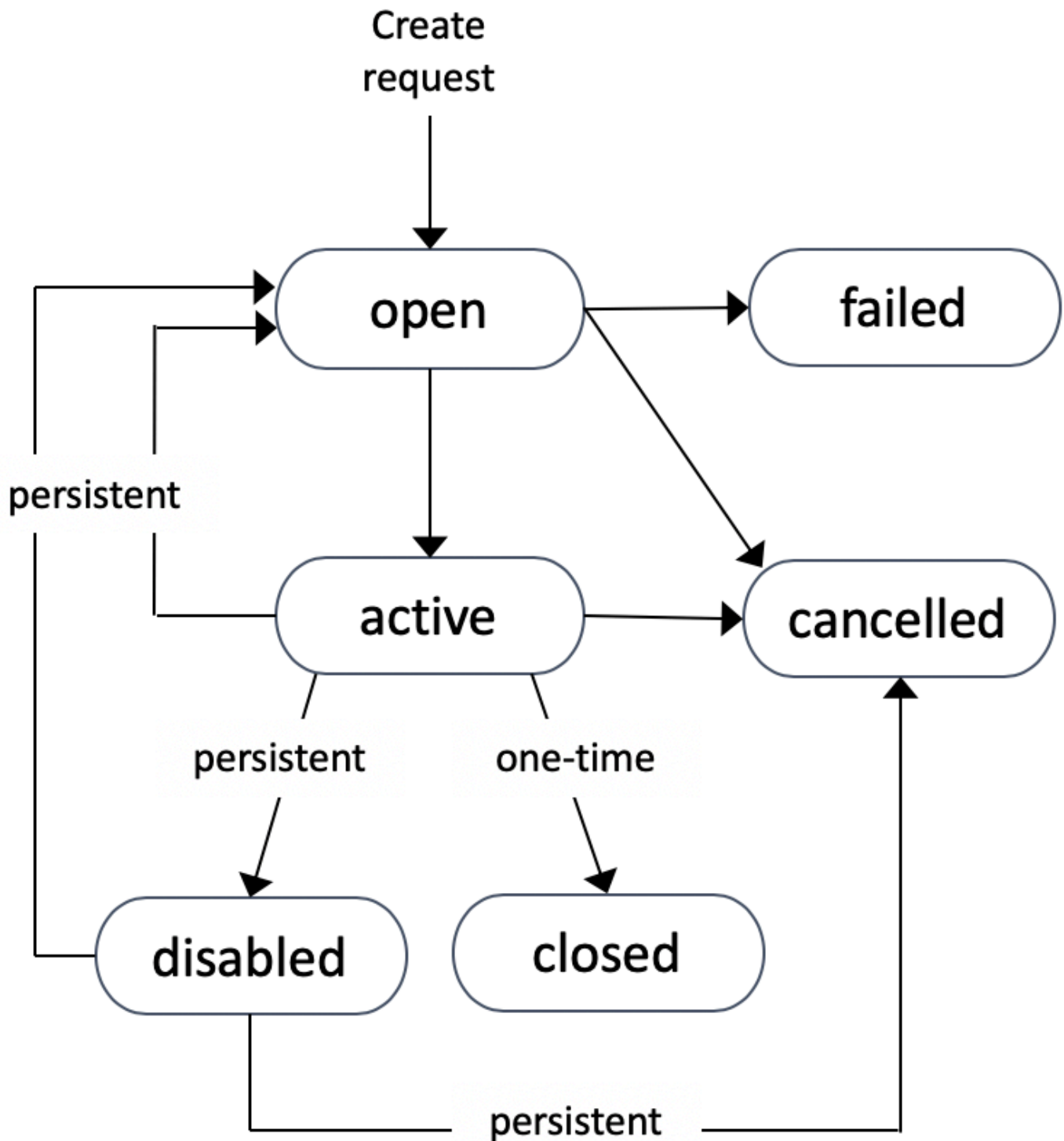
- [Spot 執行個體請求狀態](#)
- [為 競價型執行個體 指定租用](#)
- [Spot 執行個體請求的服務連結角色](#)
- [建立 Spot 執行個體請求](#)
- [尋找您的 Spot 執行個](#)
- [標記 Spot 執行個體請求](#)
- [若要取消 Spot 執行個體請求](#)
- [停止 Spot 執行個體](#)
- [啟動 Spot 執行個體](#)
- [終止 Spot 執行個體](#)
- [Spot 執行個體請求啟動規格範例](#)

## Spot 執行個體請求狀態

Spot 執行個體請求可以是下列任一種狀態：

- open – 請求正等待完成。
- active – 請求已完成，並具有關聯的 Spot 執行個體。
- failed – 請求具有一個或多個無效的參數。
- closed – Spot 執行個體已中斷或終止。
- disabled – 您已停止 Spot 執行個體。
- cancelled – 您已取消請求，或請求已過期。

下圖顯示了請求狀態之間的轉換。請注意，轉換取決於請求的類型 (一次性或持久性)。



在 Amazon EC2 啟動 Spot 執行個體、請求過期、或者您取消請求之前，一次性 Spot 執行個體請求會維持有效運作。如果沒有容量可用，則 Spot 執行個體會終止，而且 Spot 執行個體請求也會結束。

即使請求已完成，持久性的 Spot 執行個體請求在過期或是您取消它之前，都會繼續維持有效運作。如果沒有容量可用，則 Spot 執行個體會中斷。在執行個體中斷後，當容量可再次提供使用時，會啟動停止或恢復休眠的 Spot 執行個體。如果有容量可用，您可以停止 Spot 執行個體並將其重新啟動。如果 Spot 執行個體已終止 (不論 Spot 執行個體處於已停止或執行中狀態)，則 Spot 執行個體請求都會再次開啟，而 Amazon EC2 會啟動新的 Spot 執行個體。如需詳細資訊，請參閱 [停止 Spot 執行個體](#)、[啟動 Spot 執行個體](#) 及 [終止 Spot 執行個體](#)。

您可以透過狀態，追蹤 Spot 執行個體請求的狀態和已啟動的 Spot 執行個體的状态。如需詳細資訊，請參閱 [Spot 請求狀態](#)。

## 為 競價型執行個體 指定租用

您可以在單一租戶的硬體上執行 Spot 執行個體。專用 Spot 執行個體與屬於其他 AWS 帳戶的執行個體實體隔離。如需詳細資訊，請參閱 [專用執行個體](#) 和 [Amazon EC2 專用執行個體](#) 的產品頁面。

若要執行專用 Spot 執行個體，請進行下列其中一項動作：

- 在建立 Spot 執行個體請求時指定 dedicated 的租用。如需詳細資訊，請參閱 [建立 Spot 執行個體請求](#)。
- 使用 dedicated 的執行個體租用，在 VPC 中請求 Spot 執行個體。如需詳細資訊，請參閱 [建立具有專用預留執行個體租用的 VPC](#)。如果使用 default 的執行個體租用，在 VPC 中請求 Spot 執行個體，就不能使用 dedicated 的租用來請求它。

除了 T 執行個體之外，所有執行個體群都支援專用 競價型執行個體。對於每個支援的執行個體系列而言，只有最大型的執行個體大小或硬體大小可支援專用 競價型執行個體。

## Spot 執行個體請求的服務連結角色

Amazon EC2 使用許可的服務連結角色，它需要代表您呼叫其他 AWS 服務。服務連結角色是直接連結至 AWS 服務的唯一 IAM 角色類型。服務連結角色提供將權限委派給 AWS 服務的安全方式，因為只有連結的服務可以擔任服務連結角色。如需詳細資訊，請參閱 [IAM 使用者指南](#) 中的使用服務連結角色。

Amazon EC2 使用名為 AWSServiceRoleForEC2Spot 的服務連結角色代表您啟動和管理 Spot 執行個體。

## AWSServiceRoleForEC2Spot 授予的許可

Amazon EC2 用 AWSServiceRoleForEC2Spot 於完成以下操作：

- ec2:DescribeInstances - 描述 Spot 執行個體

- `ec2:StopInstances` - 停止 Spot 執行個體
- `ec2:StartInstances` - 啟動 Spot 執行個體

## 建立服務連結角色

在大部分情況下，您不需要手動建立服務連結角色。當您第一次使用主控台請求競價型執行個體時，Amazon EC2 會建立 `AWSServiceRoleForEC2Spot` 服務連結角色。

如果您在 2017 年 10 月之前有一個有效的競價型執行個體請求，則當 Amazon EC2 開始支援此服務連結角色時，Amazon EC2 會在您的 AWS 帳戶中建立該 `AWSServiceRoleForEC2Spot` 角色。如需詳細資訊，請參閱 [IAM 使用者指南](#) 中的 在我的帳戶中出現新的角色。

如果您使用 AWS CLI 或 API 請求競價型執行個體，則必須首先確保此角色存在。

### 使用主控台建立 `AWSServiceRoleForEC2Spot`

1. 在以下網址開啟 IAM 主控台：<https://console.aws.amazon.com/iam/>。
2. 在導覽窗格中，選擇角色。
3. 選擇 Create Role (建立角色)。
4. 在 Select type of trusted entity (選取信任實體類型) 頁面上，選取 EC2、選取 EC2 - Spot Instances (EC2 - Spot 執行個體)，然後選取 Next: Permissions (下一步：許可)。
5. 在下一頁選擇 Next: Review (下一步：檢閱)。
6. 在檢閱頁面上選擇建立角色。

### 若要 `AWSServiceRoleForEC2Spot` 使用建立 AWS CLI

使用如下所示的 [create-service-linked-role](#) 命令。

```
aws iam create-service-linked-role --aws-service-name spot.amazonaws.com
```

如果您不再需要使用 Spot 執行個體，建議您刪除該 `AWSServiceRoleForEC2Spot` 角色。從您的帳戶中刪除此角色後，如果您請求 競價型執行個體，Amazon EC2 會再次建立此角色。

### 授予客戶受管金鑰存取權，以便與加密的 AMI 和 EBS 快照搭配使用

如果您為競價型執行個體指定加密的 [AMI](#) 或加密的 Amazon EBS 快照，並使用客戶受管金鑰進行加密，則必須授與該 `AWSServiceRoleForEC2Spot` 角色使用客戶受管金鑰的權限，以便 Amazon EC2 可



以代表您啟動 Spot 執行個體。若要執行此動作，您必須將授予新增至客戶受管金鑰，如下列程序所示。

提供許可時，授與為金鑰政策的替代方案。如需詳細資訊，請參閱 [開發人員指南](#) 中的 [使用授予 AWS KMS](#) 和在 AWS Key Management Service 中使用金鑰政策。

若要授予 AWSServiceRoleForEC2Spot 角色許可以使用客戶受管金鑰

- 使用 [create-grant](#) 命令可將授權新增至客戶管理的金鑰，並指定授與執行授權之作業的主體 (AWSServiceRoleForEC2Spot 服務連結角色)。客戶受管金鑰由 key-id 參數和客戶受管金鑰的 ARN 指定。主體是由 grantee-principal 參數和 AWSServiceRoleForEC2Spot 服務連結角色的 ARN 所指定。

```
aws kms create-grant \
 --region us-east-1 \
 --key-id arn:aws:kms:us-east-1:444455556666:key/1234abcd-12ab-34cd-56ef-1234567890ab \
 --grantee-principal arn:aws:iam::111122223333:role/aws-service-role/
spot.amazonaws.com/AWSServiceRoleForEC2Spot \
 --operations "Decrypt" "Encrypt" "GenerateDataKey"
"GenerateDataKeyWithoutPlaintext" "CreateGrant" "DescribeKey" "ReEncryptFrom"
"ReEncryptTo"
```

## 建立 Spot 執行個體請求

您可以使用 Amazon EC2 主控台中的 [啟動執行個體精靈](#) 或 [執行個體 AWS CLI 命令來請求競價型](#) 執行個體，方法與啟動隨需執行個體相同。僅出於下列原因，建議使用此方法：

- 您已經在使用 [啟動執行個體精靈](#) 或 [run-instances](#) 命令來啟動隨需執行個體，而且您只想透過變更單一參數來變更為啟動 Spot 執行個體。
- 您不需要具有不同執行個體類型的多個執行個體。

通常不建議在啟動 Spot 執行個體時使用此方法，因為您無法指定多個執行個體類型，並且無法在同一請求中同時啟動 Spot 執行個體和隨需執行個體。如需有關啟動 Spot 執行個體的慣用方法，其中包括啟動含有使用多個執行個體類型的 Spot 執行個體和隨需執行個體的機群，請參閱 [使用哪種 Spot 請求方法最好？](#)

如果您一次請求多個 Spot 執行個體，Amazon EC2 會建立不同的 Spot 執行個體請求，如此您就可以單獨追蹤每個請求的狀態。如需追蹤 Spot 執行個體請求的詳細資訊，請參閱 [Spot 請求狀態](#)。

## New console

### 使用啟動執行個體精靈建立 Spot 執行個體請求

步驟 1–9 與您用於啟動隨需執行個體的步驟相同。在步驟 10 中，您可以設定 Spot 執行個體請求。

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在螢幕上方的導覽列上，選取「區域」。
3. 從 Amazon EC2 主控台儀表板選擇 Launch Instance (啟動執行個體)。
4. (選用) 在 Name and tags (名稱和標籤) 下，您可以命名執行個體，並標記 Spot 執行個體請求、執行個體、磁碟區和彈性圖形。如需標籤的相關資訊，請參閱 [標記您的 Amazon EC2 資源](#)。

- a. 對於 Name (名稱)，輸入執行個體的描述性名稱。

執行個體名稱是一個標籤，其中鍵是 Name (名稱)，而值是您指定的名稱。如果您未指定名稱，則可以透過其 ID 來標識執行個體，該 ID 將在您啟動執行個體時自動產生。

- b. 若要標記 Spot 執行個體請求、執行個體、磁碟區和彈性圖形，請選擇 Add additional tags (新增其他標籤)。選取 Add tag (新增標籤)，然後輸入鍵和值，然後選取要標記的資源類型。為每個要新增的其他標籤重新選擇 Add tag (新增標籤)。
5. 在 Application and OS Images (Amazon Machine Image) (應用程式和作業系統映像 (Amazon Machine Image)) 下，選取執行個體的作業系統，然後選取 AMI。如需詳細資訊，請參閱 [應用程式和作業系統映像 \(Amazon Machine Image\)](#)。
  6. 在 Instance type (執行個體類型) 下，選取符合您執行個體硬體組態和大小要求的執行個體類型。如需詳細資訊，請參閱 [執行個體類型](#)。
  7. 在 Key pair (login) (金鑰對 (登入))，選擇現有的金鑰對，或選擇 Create new key pair (建立新的金鑰對) 以建立新的金鑰對。如需詳細資訊，請參閱 [Amazon EC2 金鑰配對](#) 和 [Amazon EC2 執行個體](#)。

#### Important

如果您選擇 Proceed without key pair (Not recommended) (繼續而不使用金鑰對 (不建議)) 選項，您將無法連線到執行個體，除非您選擇已設定為允許使用者透過其他方式登入的 AMI。

8. 在 Network settings (網路設定) 下，使用預設設定，或視需要選擇 Edit (編輯) 以便設定網路設定。


安全群組構成網路設定的一部分，並定義執行個體的防火牆規則。這些規則會指定應交付至您執行個體的傳入網路流量。

如需詳細資訊，請參閱 [Network settings \(網路設定\)](#)。

9. 您選取的 AMI 包含一或多個儲存體磁碟區，包含根設備磁碟區。在 Configure storage (設定儲存) 下，您可以透過選擇 Add new volume (新增新磁碟區)，指定要連接到執行個體的額外磁碟區。如需詳細資訊，請參閱 [設定儲存](#)。
10. 在 Advanced details (進階詳細資訊)，請設定 Spot 執行個體請求，如下所示：
  - a. 在 Purchasing option (購買選項)，請選取 Request Spot Instances (請求 Spot 執行個體) 核取方塊。
  - b. 您可以保留 Spot 執行個體請求的預設組態，也可以選擇右側的 Customize (自訂) 為您的 Spot 執行個體請求指定自訂設定。

當您選擇 Customize (自訂) 時，會出現下列欄位。

- i. Maximum price (最高價格)：您可以採用 Spot 價格 (以隨需價格為上限) 請求 Spot 執行個體，也可以指定您願意支付的最高金額。

 Warning

如果您指定最高價格，您的執行個體將會比選擇無最高價時更頻繁地遭到中斷。

- No maximum price (無最高價格)：您的 Spot 執行個體將以目前的 Spot 價格啟動。價格絕不會超過隨需價格。(建議)
- Set your maximum price (per instance/hour) (設定您的最高價格 (每個執行個體/小時))：您可以指定您願意支付的最高金額。
  - 如果您指定的最高價小於目前的 Spot 價格，則不會啟動 Spot 執行個體。
  - 如果您指定的最高價格大於目前的 Spot 價格，Spot 執行個體將會啟動並按目前的 Spot 價格收費。Spot 執行個體執行後，如果 Spot 價格超過您的最高價格，Amazon EC2 將中斷您的 Spot 執行個體。
  - 無論您指定的最高價格如何，都將始終按當前 Spot 價格向您收取費用。

若要回顧 Spot 價格的趨勢，請參閱 [Spot 執行個體定價歷史記錄](#)。

- ii. Request type (請求類型)：您選擇的 Spot 執行個體請求類型會決定 Spot 執行個體中斷時會發生什麼狀況。
  - One-time (一次性)：Amazon EC2 會為您的 Spot 執行個體發出一次性請求。如果您的 Spot 執行個體中斷，即不會重新提交請求。
  - Persistent request (持久性請求)：Amazon EC2 會為您的 Spot 執行個體發出持久性請求。如果您的 Spot 執行個體中斷，系統會重新提交請求以補充中斷的 Spot 執行個體。

若未指定數值，則預設值為一次性請求。

- iii. Valid to (有效至)：持久性 Spot 執行個體請求的到期日期。

一次性請求不支援此欄位。一次性請求在請求中的所有執行個體均啟動，或是您取消請求之前，都會繼續維持有效運作。

- No request expiry date (無請求過期日期)：請求會維持在有效狀態，直到您將其取消為止。
  - Set your request expiry date (設定您的請求過期日期)：持久性請求在您指定的日期之前會維持在有效狀態，或者直到您將其取消為止。
- iv. Interruption behavior (中斷行為)：您選擇的行為會決定 Spot 執行個體中斷時會發生什麼狀況。
    - 針對持久性請求，有效值為 Stop (停止) 和 Hibernate (休眠)。當執行個體停止時，將收取 EBS 磁碟區儲存費用。

**Note**


Spot 執行個體現在使用與隨需執行個體相同的休眠功能。若要啟用休眠功能，您可以在此選擇休眠，也可以從停止 - 休眠行為欄位 (顯示於啟動執行個體精靈下半部) 中選擇啟用。如需休眠必要條件，請參閱[亞馬遜 EC2 執行個體休眠的先決條件](#)。

- 針對一次性請求，有效值僅有 Terminate (終止)。

若未指定值，則預設值為 Terminate (終止)，這對於持久性 Spot 執行個體請求無效。如果您保留預設值，並嘗試啟動持久性 Spot 執行個體請求，則會出現錯誤。

如需詳細資訊，請參閱 [Spot 執行個體中斷的行為](#)。

11. 在 Summary (摘要) 面板上，針對 Number of instances (執行個體的數量)，輸入要啟動的執行個體數量。

 Note

Amazon EC2 為每個 Spot 執行個體建立一個單獨的請求。


12. 在 Summary (摘要) 面板上，檢閱執行個體的詳細資訊，並進行任何必要的變更。提交 Spot 執行個體請求後，您就無法變更請求的參數。您可以直接導覽至啟動執行個體精靈中的某個區段，方法是在 Summary (摘要) 面板中選擇其連結。如需詳細資訊，請參閱 [Summary](#)。
13. 當您準備好啟動執行個體時，請選擇 Launch instance (啟動執行個體)。

如果執行個體無法啟動或狀態立即進入 terminated 而非 running，請參閱 [針對執行個體啟動問題進行故障診斷](#)。

## Old console


使用啟動執行個體精靈建立 Spot 執行個體請求

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在螢幕上方的導覽列上，選取「區域」。
3. 從 Amazon EC2 主控台儀表板選擇 Launch Instance (啟動執行個體)。
4. 在選擇 Amazon Machine Image (AMI) 頁面上，選擇 AMI，如下所示：如需詳細資訊，請參閱 [步驟 1：選擇 Amazon Machine Image \(AMI\)](#)。
5. 在 Choose an Instance Type (選取執行個體類型) 頁面上，選取要啟動的執行個體硬體組態和大小，然後選取下一步：設定執行個體詳細資訊。如需詳細資訊，請參閱 [步驟 2：選擇執行個體類型](#)。
6. 在設定執行個體詳細資訊頁面上，設定 Spot 執行個體請求，如下所示：
  - Number of instances (執行個體數目)：輸入要啟動的執行個體數目。

 Note

Amazon EC2 為每個 Spot 執行個體建立一個單獨的請求。

- (選用) 若要協助確認您維持足以處理您應用程式的正確執行個體數目，您可以選擇 Launch into Auto Scaling Group (啟動到 Auto Scaling 群組) 以建立啟動組態及 Auto Scaling 群組。Auto Scaling 會根據您的規格擴展群組中的執行個體數目。如需詳細資訊，請參閱 [Amazon EC2 Auto Scaling 使用者指南](#)。
- Purchasing option (購買選項)：選擇 Request Spot instances (請求 Spot 執行個體) 啟動 Spot 執行個體。當您選擇此選項時，會出現下列欄位。
- 目前價格：針對您選取的執行個體類型，顯示每個可用區域中的目前 Spot 價格。
- (選擇性) 最高價格：您可以將欄位留空，也可以指定您願意支付的最高金額。

 Warning

如果您指定最高價格，執行個體將會比將欄位保留空白時更頻繁地遭到中斷。

- 如果您指定的最高價小於 Spot 價格，則不會啟動 Spot 執行個體。
- 如果您指定的最高價格大於目前的 Spot 價格，Spot 執行個體將會啟動並按目前的 Spot 價格收費。Spot 執行個體執行後，如果 Spot 價格超過您的最高價格，Amazon EC2 將中斷您的 Spot 執行個體。
- 無論您指定的最高價格如何，都將始終按當前 Spot 價格向您收取費用。
- 如果您將欄位保留空白，則需按目前的 Spot 價格支付費用。
- 持久性請求：如果 Spot 執行個體中斷，請選擇 Persistent request (持久性請求) 以重新提交 Spot 執行個體請求。
- 中斷行為：根據預設，Spot 服務會在中斷時終止 Spot 執行個體。如果您選擇 Persistent request (持久性請求)，則可以指定 Spot 服務在中斷 Spot 執行個體時停止或休眠它。如需詳細資訊，請參閱 [Spot 執行個體中斷的行為](#)。
- (選用) 請求有效期：選擇 Edit (編輯) 以指定 Spot 執行個體請求到期的時間。

如需有關設定 Spot 執行個體的詳細資訊，請參閱 [步驟 3：設定執行個體詳細資訊](#)。

7. 您選取的 AMI 包含一或多個儲存體磁碟區，包含根設備磁碟區。在 Add Storage (新增儲存體) 頁面上，您可以透過選擇 Add New Volume (新增新磁碟區)，指定要連接到執行個體的額外磁碟區。如需詳細資訊，請參閱 [步驟 4：新增儲存體](#)。
8. 在 Add Tags (新增標籤) 頁面上，透過提供鍵和值的組合，指定 [標籤](#)。如需詳細資訊，請參閱 [步驟 5：新增標籤](#)。

9. 在 Configure Security Group (設定安全群組) 頁面上，使用安全群組定義您執行個體的防火牆規則。這些規則會指定應交付至您執行個體的傳入網路流量。所有其他流量都會遭到忽略。(如需安全群組的詳細資訊，請參閱[適用於 Amazon EC2 執行個體的亞馬遜 EC2 安全群組](#)。) 選取或建立安全群組，如下所示，然後選取檢閱和啟動。如需詳細資訊，請參閱 [步驟 6：設定安全群組](#)。
10. 在 Review Instance Launch (檢閱執行個體啟動) 頁面上，檢查您執行個體的詳細資訊，並透過選擇適當的 Edit (編輯) 連結進行任何必要的變更。當您就緒後，選擇 Launch (啟動)。如需詳細資訊，請參閱 [步驟 7：檢閱執行個體啟動並選取金鑰對](#)。
11. 在選取現有金鑰對或建立新的金鑰對對話方塊中，您可以選取現有的金鑰對或建立新的金鑰對。例如，選取選取現有的金鑰對，然後選取您在設定時建立的金鑰對。如需詳細資訊，請參閱 [Amazon EC2 金鑰配對](#)和 [Amazon EC2 執行個體](#)。

**⚠ Important**

若您選擇繼續而不使用金鑰對選項，您將無法連線到執行個體，除非您選擇已設定為允許使用者透過其他方式登入的 AMI。

12. 若要啟動執行個體，請選取確認核取方塊，然後選取啟動執行個體。

如果執行個體無法啟動或狀態立即進入 terminated 而非 running，請參閱[針對執行個體啟動問題進行故障診斷](#)。

## AWS CLI

若要使用 [run-instances](#) 建立 Spot 執行個體請求

使用 [run-instances](#) 命令，並在 `--instance-market-options` 參數中指定 Spot 執行個體選項。


```
aws ec2 run-instances \
 --image-id ami-0abcdef1234567890 \
 --instance-type t2.micro \
 --count 5 \
 --subnet-id subnet-08fc749671b2d077c \
 --key-name MyKeyPair \
 --security-group-ids sg-0b0384b66d7d692f9 \
 --instance-market-options file://spot-options.json
```

以下是在 JSON 檔案中指定 `--instance-market-options` 的資料結構。您也可以指定 `ValidUntil` 和 `InstanceInterruptionBehavior`。如果您沒有在資料結構中指定欄位，則會使用預設值。

以下範例會建立 `persistent` 請求。

```
{
 "MarketType": "spot",
 "SpotOptions": {
 "SpotInstanceType": "persistent"
 }
}
```

若要使用 [request-spot-instances](#) 建立 Spot 執行個體請求

 Note

我們強烈反對使用 [request-spot-instances](#) 命令來請求 Spot 執行個體，因為這是一個沒有計劃投資的舊式 API。如需詳細資訊，請參閱[使用哪種 Spot 請求方法最好？](#)

使用 [request-spot-instances](#) 命令來建立一次性請求。

```
aws ec2 request-spot-instances \
 --instance-count 5 \
 --type "one-time" \
 --launch-specification file://specification.json
```

使用 [request-spot-instances](#) 命令來建立一個永久的請求。

```
aws ec2 request-spot-instances \
 --instance-count 5 \
 --type "persistent" \
 --launch-specification file://specification.json
```

例如，啟動規格檔案以搭配這些命令使用，請參閱[Spot 執行個體請求啟動規格範例](#)。如果您從 Spot 請求主控台下載啟動規格檔案，則必須改用 [request-spot-fleet](#) 命令 (Spot 請求主控台使用 Spot 機群指定 Spot 執行個體請求)。



## 尋找您的 Spot 執行個體

當有容量可用時，Amazon EC2 會啟動 Spot 執行個體。Spot 執行個體會執行，直到它中斷或您自己終止它。

Spot 執行個體會與隨需執行個體一起出現在主控台的「執行個體」頁面中。請使用下列程序來尋找您的 Spot 執行個體。

### Console

使用主控台尋找您的 Spot 執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 若要尋找所有競價型執行個體，請在搜尋窗格中選擇「執行個體生命週期 = 定點」。
4. 若要確認例證是否為 Spot 例證，請選取例證，選擇「詳細資料」頁籤，然後檢查「生命週期」的值。Spot 執行個體的值為spot，隨需執行個體的值為normal。

### AWS CLI

若要使用 AWS CLI

使用[描述實例](#)命令搭配選項。--filters

```
aws ec2 describe-instances \
 --filters "Name=instance-lifecycle,Values=spot"
```

判斷執行個體是否為競價型執行個體

使用[描述執行個體](#)指令，使用--query選項來檢查生命週期值。

```
aws ec2 describe-instances \
 --instance-ids i-0123a456700123456 \
 --query "Reservations[*].Instances[*].InstanceLifecycle" \
 --output text
```

如果輸出為spot，則執行個體為 Spot 執行個體。如果沒有輸出，則執行個體為隨需執行個體。

使用下列程序尋找從特定競價型執行個體或 Spot 叢集請求啟動的 Spot 執行個體。

## Console

使用主控台尋找要求的 Spot 執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot 請求。此清單包含 Spot 執行個體請求和 Spot 叢集請求。
3. 如果已滿足競價型執行個體請求，容量就是競價型執行個體的 ID。對於 Spot Fleet，容量表示已完成所請求容量的程度。若要檢視 Spot Fleet 中的執行個體 ID，請選取展開箭頭，或選取該機群並選取 Instances (執行個體)。
4. 對於競價型叢集，容量會指出要求的容量已滿足多少。若要檢視 Spot 叢集中執行個體的 ID，請選擇要開啟其詳細資料頁面的叢集 ID，然後找到執行個體窗格。

## AWS CLI

若要使用 AWS CLI

使用 [描述-點-執行個體-請求命令搭配](#) 選項。--query

```
aws ec2 describe-spot-instance-requests \
 --query "SpotInstanceRequests[*].{ID:InstanceId}"
```

下列為範例輸出：

```
[
 {
 "ID": "i-1234567890abcdef0"
 },
 {
 "ID": "i-0598c7d356eba48d7"
 }
]
```

## 標記 Spot 執行個體請求

為了協助分類和管理您的 Spot 執行個體請求，您可以使用自訂中繼資料來標記它們。您可以在建立 Spot 執行個體請求時或之後將標記指派給請求。您可以使用 Amazon EC2 主控台或命令列工具來指派標籤。

當您標記 Spot 執行個體請求時，系統不會自動標記 Spot 執行個體請求所啟動的執行個體和磁碟區。您需要明確標記 Spot 執行個體請求啟動的執行個體和磁碟區。您可以在啟動期間或之後將標籤指派給 Spot 執行個體和磁碟區。

如需標籤運作方式的詳細資訊，請參閱[標記您的 Amazon EC2 資源](#)。

## 目錄

- [必要條件](#)
- [標記新的 Spot 執行個體請求](#)
- [標記現有的 Spot 執行個體請求](#)
- [檢視 Spot 執行個體請求標籤](#)

## 必要條件

授予使用者標記資源的許可。如需有關 IAM 政策和政策範例的詳細資訊，請參閱[範例：標籤資源](#)。

您建立的 IAM 政策取決於建立 Spot 執行個體請求時使用的方法。

- 如果您使用啟動執行個體精靈或 `run-instances` 來請求 競價型執行個體，請參閱[To grant a user the permission to tag resources when using the launch instance wizard or run-instances](#)。
- 如果您使用 `request-spot-instances` 命令請求 Spot 執行個體，請參閱[To grant a user the permission to tag resources when using request-spot-instances](#)。

若要授予使用者許可，以便在使用啟動執行個體精靈或 `run-instances` 時標記資源

建立包含下列項目的 IAM 政策：

- `ec2:RunInstances` 動作。這會授予使用者啟動執行個體的許可。
- 對於 `Resource`，請指定 `spot-instances-request`。這可讓使用者建立 Spot 執行個體請求，它們會請求 Spot 執行個體。
- `ec2:CreateTags` 動作。這會授予使用者建立標籤的許可。
- 對於 `Resource`，請指定 `*`。這可讓使用者標記執行個體啟動期間建立的所有資源。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
```

```
 "Sid": "AllowLaunchInstances",
 "Effect": "Allow",
 "Action": [
 "ec2:RunInstances"
],
 "Resource": [
 "arn:aws:ec2:us-east-1:image/*",
 "arn:aws:ec2:us-east-1:*:subnet/*",
 "arn:aws:ec2:us-east-1:*:network-interface/*",
 "arn:aws:ec2:us-east-1:*:security-group/*",
 "arn:aws:ec2:us-east-1:*:key-pair/*",
 "arn:aws:ec2:us-east-1:*:volume/*",
 "arn:aws:ec2:us-east-1:*:instance/*",
 "arn:aws:ec2:us-east-1:*:spot-instances-request/*"
]
},
{
 "Sid": "TagSpotInstanceRequests",
 "Effect": "Allow",
 "Action": "ec2:CreateTags",
 "Resource": "*"
}
]
```

當您使用該 `RunInstances` 動作建立競價型執行個體請求並在建立時標記競價型執行個體請求時，您需要瞭解 Amazon EC2 如何在 IAM 政策中評估該 `spot-instances-request` 資源的 `RunInstances` 陳述式中評估資源，如下所示：

- 如果您在建立時未標記競價型執行個體請求，Amazon EC2 不會評估 `RunInstances` 陳述式中的 `spot-instances-request` 資源。
- 如果您在建立時標記競價型執行個體請求，Amazon EC2 會評估 `RunInstances` 陳述式中的 `spot-instances-request` 資源。

因此，對於 `spot-instances-request` 資源，下列規則適用於 IAM 政策：

- 如果您使 `RunInstances` 用建立競價型執行個體請求，且不打算在建立時標記競價型執行個體請求，則不需要明確允許 `spot-instances-request` 資源；呼叫將會成功。
- 如果您使 `RunInstances` 用建立競價型執行個體請求並打算在建立時標記競價型執行個體請求，則必須在 `RunInstances allow` 陳述式中包含 `spot-instances-request` 資源，否則呼叫將會失敗。

- 如果您使用 RunInstances 用建立競價型執行個體請求並打算在建立時標記競價型執行個體請求，則必須在 CreateTags allow 陳述式中指定 spot-instances-request 資源或包含 \* 萬用字元，否則呼叫將失敗。

例如，IAM 政策 (包括 Spot 執行個體請求不支援的政策)，請參閱 [使用 競價型執行個體](#)。

若要授予使用者許可，以便在使用 request-spot-instances 時標記資源

建立包含下列項目的 IAM 政策：

- ec2:RequestSpotInstances 動作。這會授予使用者建立 Spot 執行個體請求的許可。
- ec2:CreateTags 動作。這會授予使用者建立標籤的許可。
- 對於 Resource，請指定 spot-instances-request。這允許使用者只標記 Spot 執行個體請求。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "TagSpotInstanceRequest",
 "Effect": "Allow",
 "Action": [
 "ec2:RequestSpotInstances",
 "ec2:CreateTags"
],
 "Resource": "arn:aws:ec2:us-east-1:111122223333:spot-instances-request/*"
 }
]
}
```

## 標記新的 Spot 執行個體請求

### Console

若要使用主控台標記新的 Spot 執行個體請求

1. 遵循 [建立 Spot 執行個體請求](#) 程序。
2. 若要新增標記，請在新增標記頁面上選擇新增標記，然後輸入標記的金鑰和值。為每個額外的標籤選擇新增其他標籤。

對於每個標籤，您可以用相同的標籤來標記 Spot 執行個體請求、Spot 執行個體以及磁碟區。若要標記所有這三個項目，請確保已選取執行個體、磁碟區和 Spot 執行個體請求。若只要標記一個或兩個，請確定已選取要標記的資源，並清除其他資源。

3. 完成建立 Spot 執行個體請求的必要欄位，然後選擇 Launch (啟動)。如需詳細資訊，請參閱 [建立 Spot 執行個體請求](#)。

## AWS CLI

若要使用標記新的 Spot 執行個體請求 AWS CLI

若要在建立 Spot 執行個體請求時進行標記，請依照下列方式設定 Spot 執行個體請求組態：

- 使用 `--tag-specification` 參數指定 Spot 執行個體請求的標籤。
- 對於 `ResourceType`，請指定 `spot-instances-request`。如果您指定另一個值，Spot 執行個體請求將失敗。
- 對於 `Tags`，指定金鑰值對。您可以指定多個金鑰值對。

在下面的例子中，Spot 執行個體請求標記了兩個標籤：Key=Environment and Value=Production 和 Key=Cost-Center and Value=123。

```
aws ec2 request-spot-instances \
 --instance-count 5 \
 --type "one-time" \
 --launch-specification file://specification.json \
 --tag-specification 'ResourceType=spot-instances-
request,Tags=[{Key=Environment,Value=Production},{Key=Cost-Center,Value=123}]'
```

## 標記現有的 Spot 執行個體請求

### Console

若要使用主控台標記現有的 Spot 執行個體請求

建立 Spot 執行個體請求後，您可以使用主控台為 Spot 執行個體請求新增標籤。

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot 請求。

3. 選取您的 Spot 執行個體請求。
4. 選擇標籤索引標籤，然後選擇建立標籤。

若要使用主控台標記現有的 Spot 執行個體

在您的 Spot 執行個體請求啟動 Spot 執行個體之後，您可以使用主控台將標籤新增到執行個體。如需詳細資訊，請參閱 [新增和刪除個別資源上的標籤](#)。

## AWS CLI

若要使用標記現有的 Spot 執行個體請求或 Spot 執行個體 AWS CLI

使用 [建立標籤](#) 命令來標記現有資源。在下面的例子中，現有的 Spot 執行個體請求和 Spot 執行個體會使用 Key=purpose and Value=test 進行標記。

```
aws ec2 create-tags \
 --resources sir-08b93456 i-1234567890abcdef0 \
 --tags Key=purpose,Value=test
```

## 檢視 Spot 執行個體請求標籤

### Console

若要使用主控台檢視 Spot 執行個體請求

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取您的 Spot 執行個體請求並選取 Tags (標籤) 標籤。

## AWS CLI

若要描述 Spot 執行個體請求標籤

您可以透過描述 Spot 執行個體請求來檢視 Spot 執行個體請求的標籤。使用 [describe-spot-instance-requests](#) 命令來檢視指定 Spot 執行個體請求的組態，其中包括為請求指定的任何標籤。

```
aws ec2 describe-spot-instance-requests \
 --spot-instance-request-ids sir-EXAMPLE1 \
 --query "SpotInstanceRequests[*].Tags"
```

下列為範例輸出。

```
[
 [
 {
 "Key": "Environment",
 "Value": "Production"
 },
 {
 "Key": "Department",
 "Value": "101"
 }
]
]
```

### 若要取消 Spot 執行個體請求

如果您不再需要 Spot 執行個體請求，可以取消它。您只能取消狀態為 open、active 或 disabled 的 Spot 執行個體請求。

- 當請求尚未處理完成，而且尚未啟動任何執行個體時，Spot 執行個體請求的狀態為 open。
- 當請求已處理完成，而且因此啟動了 Spot 執行個體時，Spot 執行個體請求的狀態為 active。
- 當您停止 Spot 執行個體時，您的 Spot 執行個體請求的狀態為 disabled。

如果您的 Spot 執行個體請求的狀態為 active，並且擁有相關聯的執行中 Spot 執行個體，則取消請求並不會終止執行個體。如需有關終止 Spot 執行個體的詳細資訊，請參閱[終止 Spot 執行個體](#)。

### Console

#### 使用主控台取消 Spot 執行個體請求

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取競價型執行個體請求。
4. 選擇動作、取消請求。
5. (選擇性) 如果您已完成與競價型執行個體的關聯，您可以終止它們。在取消 Spot 請求對話方塊中，選取終止執行個體，然後選取確認。



## AWS CLI

若要取消 Spot 執行個體請求 AWS CLI

使用 [cancel-spot-instance-requests](#) 命令來取消指定的 Spot 執行個體請求。

```
aws ec2 cancel-spot-instance-requests --spot-instance-request-ids sir-08b93456
```

## 停止 Spot 執行個體

如果您現在不需要您的 競價型執行個體，但想稍後重新啟動它們，而不會遺失 Amazon EBS 磁碟區中保留的資料，您可以停止它們。停止 Spot 執行個體的步驟與停止隨需執行個體的步驟類似。

### Note

停止 Spot 執行個體時，您可以修改其某些執行個體屬性，但無法修改執行個體類型。我們不會對已停止的 Spot 執行個體計費，也不會收取數據傳輸費，但是會收取任何 Amazon EBS 磁碟區的儲存費用。

## 限制

- 如果從 persistent Spot 執行個體請求中啟動 Spot 執行個體，則只能停止 Spot 執行個體。
- 如果已取消關聯的 Spot 執行個體請求，則無法停止 Spot 執行個體。當 Spot 執行個體請求被取消時，您只能終止 Spot 執行個體。
- 如果 Spot 執行個體屬於某個機群或啟動群組或可用區域群組，則無法停止它。

## Console

使用主控台停止 Spot 執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取定點例證。如果您未儲存競價型執行個體的執行個體 ID，請參閱 [the section called “尋找您的 Spot 執行個體”](#)。
4. 選擇 Instance state (執行個體狀態)、Stop instance (停止執行個體)。

5. 出現確認提示時，請選擇 Stop (停止)。

## AWS CLI

若要使用停止定點例證 AWS CLI

使用停止[執行個體](#)指令手動停止 Spot 執行個體。

```
aws ec2 stop-instances --instance-ids i-1234567890abcdef0
```

## 啟動 Spot 執行個體

您可以啟動之前已停止的 Spot 執行個體。

### 必要條件

只有在下列情況下，才能啟動 Spot 執行個體：

- 您手動停止了 Spot 執行個體。
- Spot 執行個體是 EBS 支援的執行個體。
- Spot 執行個體容量可用。
- Spot 價格低於您的最高價格。

### 限制

- 如果 Spot 執行個體屬於某個機群或啟動群組或可用區域群組，則無法啟動它。

啟動 Spot 執行個體的步驟與啟動隨需執行個體的步驟類似。

## Console

使用主控台啟動 Spot 執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取定點例證。如果您未儲存競價型執行個體的執行個體 ID，請參閱[the section called “尋找您的 Spot 執行個”](#)。

4. 選擇 Instance state (執行個體狀態)、Start instance (啟動執行個體)。

## AWS CLI

若要啟動定點例證 AWS CLI

使用[啟動執行個體](#)指令手動啟動 Spot 執行個體。

```
aws ec2 start-instances --instance-ids i-1234567890abcdef0
```

## 終止 Spot 執行個體

如果終止持久性 Spot 執行個體請求啟動的執行中或已停止的 Spot 執行個體，Spot 執行個體請求會轉變為 open 狀態，以便啟動新的 Spot 執行個體。若要確保沒有啟動新的 Spot 執行個體，您必須先取消 Spot 執行個體請求。

如果您取消具有執行中 Spot 執行個體的 active Spot 執行個體請求，則執行中的 Spot 執行個體不會自動終止；您必須手動終止 Spot 執行個體。

如果您取消具有已停止的 Spot 執行個體的 disabled Spot 執行個體請求，則 Amazon EC2 Spot 服務會自動終止已停止的 Spot 執行個體。取消 Spot 執行個體請求和 Spot 服務終止 Spot 執行個體之間可能有短暫延遲。

如需詳細資訊，請參閱 [若要取消 Spot 執行個體請求](#)。

## Console

若要使用主控台手動終止 Spot 執行個體

1. 終止執行個體前，請檢查您的 Amazon EBS 磁碟區不會在終止時被刪除，並確認已將執行個體存放區磁碟區上的所需資料複製到持久性儲存 (例如 Amazon EBS 或 Amazon S3)，藉此確認您不會遺失任何資料。
2. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
3. 在導覽窗格中，選擇執行個體。
4. 選取定點例證。如果您未儲存競價型執行個體的執行個體 ID，請參閱 [the section called “尋找您的 Spot 執行個”](#)。
5. 選擇 Instance state (執行個體狀態)、Terminate instance (終止執行個體)。
6. 出現確認提示時，請選擇 Terminate (終止)。

## AWS CLI

使用手動終止 Spot 執行個體 AWS CLI

使用[終止執行個體](#)命令手動終止 Spot 執行個體。

```
aws ec2 terminate-instances --instance-ids i-1234567890abcdef0 i-0598c7d356eba48d7
```

## Spot 執行個體請求啟動規格範例

下列範例顯示您可搭配 [request-spot-instances](#) 命令使用的啟動組態，以建立 Spot 執行個體請求。如需詳細資訊，請參閱 [建立 Spot 執行個體請求](#)。

### Important

我們強烈反對使用 [request-spot-instances](#) 命令來請求 Spot 執行個體，因為這是一個沒有計劃投資的舊式 API。如需詳細資訊，請參閱 [使用哪種 Spot 請求方法最好？](#)

## 範例

- [範例 1：啟動 競價型執行個體](#)
- [範例 2：在指定的可用區域中啟動 競價型執行個體](#)
- [範例 3：在指定的子網中啟動 競價型執行個體](#)
- [範例 4：啟動專用 Spot 執行個體](#)

### 範例 1：啟動 競價型執行個體

下列範例不包括可用區域或子網。Amazon EC2 為您選取可用區域。Amazon EC2 會在所選可用區域的預設子網中啟動執行個體。

```
{
 "ImageId": "ami-0abcdef1234567890",
 "KeyName": "my-key-pair",
 "SecurityGroupIds": ["sg-1a2b3c4d5e6f7g8h9"],
 "InstanceType": "m5.medium",
 "IamInstanceProfile": {
 "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
 }
}
```

```
}
```

### 範例 2：在指定的可用區域中啟動 競價型執行個體

下列範例包括可用區域。Amazon EC2 會在指定可用區域的預設子網中啟動執行個體。

```
{
 "ImageId": "ami-0abcdef1234567890",
 "KeyName": "my-key-pair",
 "SecurityGroupIds": ["sg-1a2b3c4d5e6f7g8h9"],
 "InstanceType": "m5.medium",
 "Placement": {
 "AvailabilityZone": "us-west-2a"
 },
 "IamInstanceProfile": {
 "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
 }
}
```

### 範例 3：在指定的子網中啟動 競價型執行個體

下列範例包含子網。Amazon EC2 會在指定子網中啟動執行個體。如果 VPC 為非預設 VPC，則執行個體預設不會接收公有 IPv4 地址。

```
{
 "ImageId": "ami-0abcdef1234567890",
 "SecurityGroupIds": ["sg-1a2b3c4d5e6f7g8h9"],
 "InstanceType": "m5.medium",
 "SubnetId": "subnet-1a2b3c4d",
 "IamInstanceProfile": {
 "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
 }
}
```

若要指派公有 IPv4 位址給非預設 VPC 中的執行個體，請依下列範例所示來指定 `AssociatePublicIpAddress` 欄位。指定網路介面時，您必須填入網路介面的子網 ID 和安全群組 ID，而非使用前例程式碼區塊所示的 `SubnetId` 和 `SecurityGroupIds` 欄位。

```
{
 "ImageId": "ami-0abcdef1234567890",
 "KeyName": "my-key-pair",
```

```
"InstanceType": "m5.medium",
"NetworkInterfaces": [
 {
 "DeviceIndex": 0,
 "SubnetId": "subnet-1a2b3c4d5e6f7g8h9",
 "Groups": ["sg-1a2b3c4d5e6f7g8h9"],
 "AssociatePublicIpAddress": true
 }
],
"IamInstanceProfile": {
 "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
}
}
```

#### 範例 4：啟動專用 Spot 執行個體

下列範例請求具有dedicated租用的 Spot 執行個體。專用 Spot 執行個體必須在 VPC 中啟動。

```
{
 "ImageId": "ami-0abcdef1234567890",
 "KeyName": "my-key-pair",
 "SecurityGroupIds": ["sg-1a2b3c4d5e6f7g8h9"],
 "InstanceType": "c5.8xlarge",
 "SubnetId": "subnet-1a2b3c4d5e6f7g8h9",
 "Placement": {
 "Tenancy": "dedicated"
 }
}
```

## Spot 請求狀態

為協助您追蹤 Spot 執行個體請求並規劃 Spot 執行個體的使用，請使用 Amazon EC2 提供的請求狀態。例如，請求狀態可提供 Spot 請求尚未完成的理由，或列出阻礙完成 Spot 請求的限制。

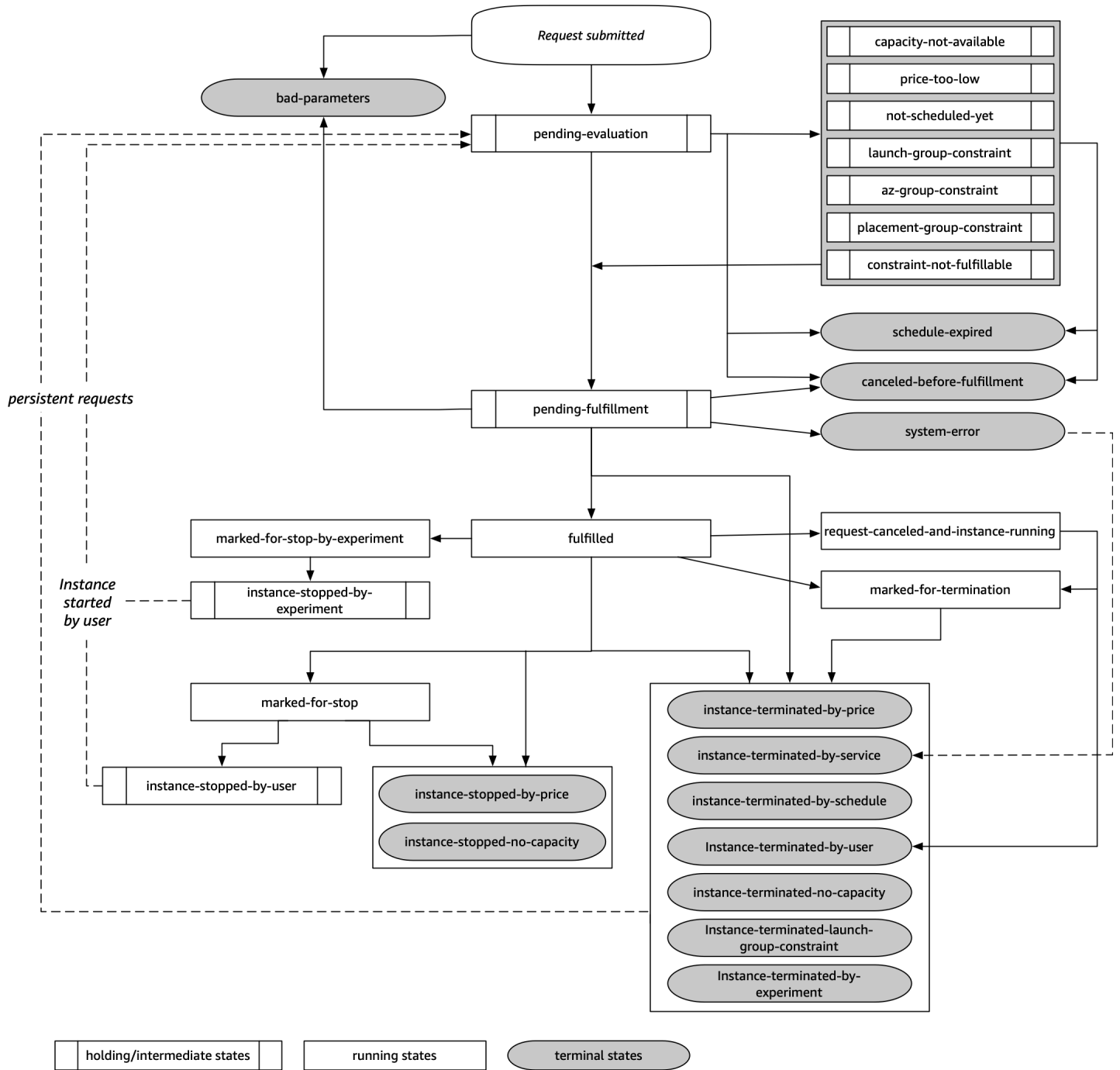
在程序的每個步驟中— (此程序也稱為 Spot 請求生命週期)—，特定的事件會決定相繼請求的狀態。

### 內容

- [Spot 請求的生命週期](#)
- [取得請求狀態資訊](#)
- [Spot 請求狀態碼](#)
- [EC2 Spot 執行個體請求履行事件](#)

### Spot 請求的生命週期

下圖顯示 Spot 請求在整個生命週期的期間中可遵循的路徑 (從提交到終止)。每個步驟皆以節點表示，而每個節點的狀態碼則描述了 Spot 請求和 Spot 執行個體的状态。



### 等待評估

除非有一個或多個請求參數無效 (`pending-evaluation`)，否則當您建立 Spot 執行個體請求後，它就會立即進入 `bad-parameters` 狀態。

| 狀態碼                | 請求狀態   | 執行個體狀態 |
|--------------------|--------|--------|
| pending-evaluation | open   | 不適用    |
| bad-parameters     | closed | 不適用    |

## 保留

如果有一個或多個請求限制為有效但尚未滿足，或是沒有足夠的容量，則請求會進入保留狀態，以等待限制條件滿足。請求選項會影響請求完成的可能性。例如，如果沒有容量，則請求會持續處於保留狀態，直到有可用的容量為止。如果指定了可用區域群組，則請求會持續處於保留狀態，直到可用區域的限制條件滿足。

如果其中一個可用區域中斷，其他可用區域內的 Spot 執行個體請求可用的備用 EC2 容量也可能會受到影響。

| 狀態碼                        | 請求狀態 | 執行個體狀態 |
|----------------------------|------|--------|
| capacity-not-available     | open | 不適用    |
| price-too-low              | open | 不適用    |
| not-scheduled-yet          | open | 不適用    |
| launch-group-constraint    | open | 不適用    |
| az-group-constraint        | open | 不適用    |
| placement-group-constraint | open | 不適用    |
| constraint-not-fulfillable | open | 不適用    |



## 等待評估/實現-終止

如果所建立的請求只在特定期間內有效，而且在請求進入待完成階段之前，此期間就會到期，則此 Spot 執行個體請求可以進入 terminal 狀態。如果您取消請求或系統發生錯誤，也會發生此狀況。

| 狀態碼                                    | 請求狀態      | 執行個體狀態 |
|----------------------------------------|-----------|--------|
| schedule-expired                       | cancelled | 不適用    |
| cancel-before-fulfillment <sup>1</sup> | cancelled | 不適用    |
| bad-parameters                         | failed    | 不適用    |
| system-error                           | closed    | 不適用    |

<sup>1</sup> 若您取消請求。

## 等待實現

當您所指定的限制條件 (如果有的話) 已滿足，則 Spot 請求會進入 pending-fulfillment 狀態。

此時，Amazon EC2 會準備佈建您所請求的執行個體。如果程序在此時停止，可能是因為使用者已在啟動 Spot 執行個體前取消程序。也可能是因為發生未預期的系統錯誤。

| 狀態碼                 | 請求狀態 | 執行個體狀態 |
|---------------------|------|--------|
| pending-fulfillment | open | 不適用    |

## 已實現

當滿足 Spot 執行個體的所有規格時，Spot 請求即已完成。Amazon EC2 會啟動 Spot 執行個體，這可能需要幾分鐘。如果 Spot 執行個體已進入休眠或是在中斷時停止，它將會繼續處於此狀態，直到請求可以再次完成或取消。

| 狀態碼       | 請求狀態   | 執行個體狀態            |
|-----------|--------|-------------------|
| fulfilled | active | pending → running |
| fulfilled | active | stopped → running |

如果您停止 Spot 執行個體，則 Spot 請求會進入 `marked-for-stop` 或 `instance-stopped-by-user` 狀態，直到 Spot 執行個體可以再次啟動或取消請求為止。

| 狀態碼                                   | 請求狀態                               | 執行個體狀態   |
|---------------------------------------|------------------------------------|----------|
| marked-for-stop                       | active                             | stopping |
| instance-stopped-by-user <sup>1</sup> | disabled 或是 cancelled <sup>2</sup> | stopped  |

<sup>1</sup> 如果您停止執行個體或從執行個體執行 `shutdown` 指令，Spot 執行個體就會進入 `instance-stopped-by-user` 狀態。停止執行個體之後，您可以重新啟動它。重新啟動時，Spot 執行個體請求返回 `pending-evaluation` 狀態，然後在符合限制時 Amazon EC2 會啟動新的 Spot 執行個體。

<sup>2</sup> 如果您停止 Spot 執行個體但不取消請求，Spot 請求狀態則為 `disabled`。如果 Spot 執行個體已停止且請求到期，請求狀態則為 `cancelled`。

### 已實現-終止

只要有該執行個體類型可用的容量，且您未終止執行個體，您的 Spot 執行個體就會持續運作。如果 Amazon EC2 必須終止 Spot 執行個體，則 Spot 請求會進入終止狀態。如果取消 Spot 請求或終止 競價型執行個體，請求也會進入終止狀態。

| 狀態碼                                   | 請求狀態      | 執行個體狀態  |
|---------------------------------------|-----------|---------|
| request-canceled-and-instance-running | cancelled | running |
| marked-for-stop                       | active    | running |

| 狀態碼                                         | 請求狀態                             | 執行個體狀態     |
|---------------------------------------------|----------------------------------|------------|
| marked-for-termination                      | active                           | running    |
| instance-stopped-by-price                   | disabled                         | stopped    |
| instance-stopped-by-user                    | disabled                         | stopped    |
| instance-stopped-no-capacity                | disabled                         | stopped    |
| instance-terminated-by-price                | closed ( 一次性 ) , open ( 持續性 )    | terminated |
| instance-terminated-by-schedule             | closed                           | terminated |
| instance-terminated-by-service              | cancelled                        | terminated |
| instance-terminated-by-user                 | closed 或是 cancelled <sup>1</sup> | terminated |
| instance-terminated-no-capacity             | closed ( 一次性 ) , open ( 持續性 )    | running †  |
| instance-terminated-no-capacity             | closed ( 一次性 ) , open ( 持續性 )    | terminated |
| instance-terminated-launch-group-constraint | closed ( 一次性 ) , open ( 持續性 )    | terminated |

<sup>1</sup> 如果您終止執行個體但未取消請求，則請求狀態為 closed。如果終止執行個體而且取消請求，則請求狀態為 cancelled。即使您在取消 Spot 執行個體的請求之前就先終止它，在 Amazon EC2 偵測到

Spot 執行個體已終止前，可能會有一段延遲時間。在此種情況中，請求的狀態可以是 `cancelled` 或 `closed`。

† 當 Amazon EC2 中斷 Spot 執行個體 (若其需要回收容量) 和將執行個體設定為中斷時終止，狀態會立即設定為 `instance-terminated-no-capacity` (其未設定為 `marked-for-termination`)。但是，執行個體會保留在 `running` 狀態 2 分鐘，反映執行個體收到 Spot 執行個體中斷通知的 2 分鐘時段。2 分鐘後，執行個體狀態設定為 `terminated`。

## 中斷實驗

您可以使用 AWS Fault Injection Service 來啟動 Spot 執行個體中斷，以便測試 Spot 執行個體上的應用程式回應方式。如果 AWS FIS 停止 Spot 執行個體，您的 Spot 請求會進入 `marked-for-stop-by-experiment` 狀態，然後進入 `instance-stopped-by-experiment` 狀態。如果 AWS FIS 終止競價型執行個體，則您的 Spot 請求會進入該 `instance-terminated-by-experiment` 狀態。如需詳細資訊，請參閱 [the section called “啟動中斷”](#)。

| 狀態碼                                            | 請求狀態                  | 執行個體狀態                  |
|------------------------------------------------|-----------------------|-------------------------|
| <code>marked-for-stop-by-experiment</code>     | <code>active</code>   | <code>running</code>    |
| <code>instance-stopped-by-experiment</code>    | <code>disabled</code> | <code>stopped</code>    |
| <code>instance-terminated-by-experiment</code> | <code>closed</code>   | <code>terminated</code> |

## 永久性請求

當 Spot 執行個體終止時 (由您或 Amazon EC2 終止)，如果 Spot 請求為持久性請求，則會回到 `pending-evaluation` 狀態，之後 Amazon EC2 就可以在滿足限制條件時啟動新的 Spot 執行個體。

## 取得請求狀態資訊

您可以使用 AWS Management Console 或命令列工具取得要求狀態資訊。

### 使用主控台取得要求狀態資訊

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。

2. 在瀏覽窗格中，選取 Spot 請求，然後選取 Spot 請求。
3. 若要檢查狀態，請在描述索引標籤上檢查狀態欄位。

使用命令列來取得請求狀態資訊

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [describe-spot-instance-requests](#) (AWS CLI)
- [Get-EC2SpotInstanceRequest](#) (AWS Tools for Windows PowerShell)

Spot 請求狀態碼

Spot 請求狀態資訊包含狀態碼、更新時間和狀態訊息。綜合這些資訊，可協助您決定 Spot 請求的處理方式。

下列是 Spot 請求的狀態碼：

`az-group-constraint`

Amazon EC2 無法在同一個可用區域中，啟動您請求的所有執行個體。

`bad-parameters`

Spot 請求的一個或多個參數無效 (例如，指定的 AMI 不存在)。狀態訊息會顯示哪個參數無效。

`canceled-before-fulfillment`

Spot 請求在完成之前已遭到使用者取消。

`capacity-not-available`

您所請求的執行個體沒有足夠的可用容量。

`constraint-not-fulfillable`

無法完成 Spot 請求，因為有一個或多個限制條件無效 (例如，可用區域不存在)。狀態訊息會顯示哪個限制條件無效。

`fulfilled`

Spot 請求是 active，並且 Amazon EC2 正在啟動您的 競價型執行個體。

`instance-stopped-by-price`

您的執行個體已停止，因為 Spot 價格超過您的最高價格。

## instance-stopped-by-user

您的執行個體已停止，因為使用者已停止該執行個體或從該執行個體執行關機命令。

## instance-stopped-no-capacity

由於 EC2 容量管理需求，您的執行個體已停止。

## instance-terminated-by-price

您的執行個體已終止，因為 Spot 價格超過您的最高價格。如果請求為永久性，程序會重新開始執行，因此請求會處於等待評估的狀態。

## instance-terminated-by-schedule

您的 Spot 執行個體已在排程持續時間結束時終止。

## instance-terminated-by-service

執行個體會從停止狀態終止。

## instance-terminated-by-user 或 spot-instance-terminated-by-user

您終止了已完成的 Spot 執行個體，因此請求狀態為 closed (除非為持久性請求)，而執行個體狀態為 terminated。

## instance-terminated-launch-group-constraint

啟動群組中的一個或多個執行個體已終止，因此不再滿足啟動群組的限制條件。

## instance-terminated-no-capacity

由於標準容量管理程序，您的執行個體已終止。

## launch-group-constraint

Amazon EC2 無法同時啟動您請求的所有執行個體。啟動群組中的所有執行個體會同時啟動和終止。

## limit-exceeded

已超過 EBS 磁碟區數量或磁碟區儲存空間總數量的限制。如需有關這些限制和如何申請提高限制的詳細資訊，請參閱《Amazon Web Services 一般參考》中的 [Amazon EBS 限制](#)。

## marked-for-stop

Spot 執行個體已標記停止。

## marked-for-termination

Spot 執行個體已標記終止。

## not-scheduled-yet

在尚未達到排程日期之前，將不會評估 Spot 請求。

## pending-evaluation

在您發出 Spot 執行個體請求後，在系統評估請求的參數時，該請求會進入 pending-evaluation 狀態。

## pending-fulfillment

Amazon EC2 正在嘗試佈建 競價型執行個體。

## placement-group-constraint

尚無法完成 Spot 請求，因為目前無法將 Spot 執行個體加入置放群組。

## price-too-low

尚無法完成 Spot 請求，因為最高價格低於 Spot 價格。在這種情況中，不會啟動任何執行個體，而請求的狀態仍為 open。

## request-canceled-and-instance-running

您在 競價型執行個體仍在執行時取消了 Spot 請求。請求已 cancelled，但執行個體仍處於 running 狀態。

## schedule-expired

Spot 請求已過期，因為未在指定日期之前完成。

## system-error

出現未預期的系統錯誤。如果這是一個經常性的問題，請聯繫 AWS Support 尋求幫助。

## EC2 Spot 執行個體請求履行事件

完成競價型執行個體請求後，Amazon EC2 會將 EC2 競價型執行個體請求履行事件傳送給 Amazon EventBridge。您可以建立規則以在發生此事件時採取動作，例如叫用 Lambda 函數或通知 Amazon SNS 主題。

以下是此事件的範例資料。

```
{
 "version": "0",
```

```
"id": "01234567-1234-0123-1234-012345678901",
"detail-type": "EC2 Spot Instance Request Fulfillment",
"source": "aws.ec2",
"account": "123456789012",
"time": "yyyy-mm-ddThh:mm:ssZ",
"region": "us-east-2",
"resources": ["arn:aws:ec2:us-east-2:123456789012:instance/i-1234567890abcdef0"],
"detail": {
 "spot-instance-request-id": "sir-1a2b3c4d",
 "instance-id": "i-1234567890abcdef0"
}
}
```

如需詳細資訊，請參閱 [Amazon EventBridge 使用者指南](#)。

## EC2 執行個體重新平衡建議

EC2 執行個體重新平衡推薦是一種訊號，可在 Spot 執行個體的中斷風險升高時通知您。該訊號能夠比 [兩分鐘 Spot 執行個體中斷通知](#) 更快到達，讓您有機會主動管理 Spot 執行個體。您可以決定將工作負載重新平衡至未處於提高之中斷風險的新的或現有 競價型執行個體。

Amazon EC2 並不總是能夠在兩分鐘 Spot 執行個體中斷通知之前傳送重新平衡建議訊號。因此，重新平衡建議訊號可以與兩分鐘的中斷通知一起到達。

重新平衡建議可作為 EventBridge 事件和 Spot 執行個體上 [執行個體中繼資料](#) 中的項目提供。盡可能發出事件。

### Note

僅針對在 2020 年 11 月 5 日 00:00 UTC 之後推出的 競價型執行個體 支援重新平衡建議。

## 主題

- [您可以採取的重新平衡動作](#)
- [監控重新平衡建議訊號](#)
- [使用重新平衡建議訊號的服務](#)

## 您可以採取的重新平衡動作

以下是您可以採取的一些可能的重新平衡動作：



## 順利關機

當您收到 Spot 執行個體的重新平衡建議訊號時，您可以啟動執行個體關機程序，其中可能包括確保在程序停止之前完成程序。例如，您可以將系統或應用程式日誌上傳到 Amazon Simple Storage Service (Amazon S3)、關閉 Amazon SQS 工作者，或者您可以從網域名稱系統 (DNS) 完成取消註冊。您也可以將工作儲存在外部儲存體中，稍後再繼續執行。

## 防止排程新工作

當您收到 Spot 執行個體的重新平衡建議訊號時，您可以防止在執行個體上排程新工作，同時繼續使用執行個體，直到排定的工作完成為止。

## 主動啟動新的替代執行個體

您可以設定 Auto Scaling 群組、EC2 Fleet 或 Spot Fleet，在發出重新平衡建議訊號時自動啟動替代 Spot 執行個體。如需詳細資訊，請參閱《Amazon EC2 Auto Scaling 使用者指南》中的[使用容量重新平衡處理 Amazon EC2 Spot 中斷](#)，以及本使用者指南中的 EC2 機群的[容量重新平衡](#)和 Spot 機群的[容量重新平衡](#)。

## 監控重新平衡建議訊號

您可以監視重新平衡建議訊號，以便在發出重新平衡建議訊號時，您可以採取在上一部分中指定的動作。重新平衡建議信號可作為傳送至 Amazon EventBridge (以前稱為 Amazon E CloudWatch vents) 的事件提供，以及作為競價型執行個體上的執行個體中繼資料。

監控重新平衡建議訊號：

- [使用 Amazon EventBridge](#)
- [使用執行個體中繼資料](#)

## 使用 Amazon EventBridge

當針對競價型執行個體發出重新平衡建議訊號時，訊號的事件會傳送至 Amazon。EventBridge 如果 EventBridge 偵測到符合規則中定義的模式的事件模式，則會 EventBridge 叫用規則中指定的一個或多個目標。

以下是重新平衡建議訊號的範例事件。

```
{
 "version": "0",
 "id": "12345678-1234-1234-1234-123456789012",
```

```
"detail-type": "EC2 Instance Rebalance Recommendation",
"source": "aws.ec2",
"account": "123456789012",
"time": "yyyy-mm-ddThh:mm:ssZ",
"region": "us-east-2",
"resources": ["arn:aws:ec2:us-east-2:123456789012:instance/i-1234567890abcdef0"],
"detail": {
 "instance-id": "i-1234567890abcdef0"
}
}
```

下列欄位會形成規則中定義的事件模式：

```
"detail-type": "EC2 Instance Rebalance Recommendation"
```

識別事件是重新平衡建議事件

```
"source": "aws.ec2"
```

識別該事件是來自 Amazon EC2

## 建立 EventBridge 規則

您可以撰寫 EventBridge 規則，並在事件模式符合規則時自動執行哪些動作。

下列範例會建立 EventBridge 規則，以便在每次 Amazon EC2 發出重新平衡建議訊號時傳送電子郵件、文字訊息或行動推播通知。將該信號作為 EC2 Instance Rebalance Recommendation 事件發出，這觸發了規則定義的動作。

在建立 EventBridge 規則之前，您必須為電子郵件、文字訊息或行動推播通知建立 Amazon SNS 主題。

若要建立重新平衡建議事件的 EventBridge 規則

1. 在以下位置打開 Amazon EventBridge 控制台 <https://console.aws.amazon.com/events/>。
2. 選擇建立規則。
3. 針對 Define rule detail (定義規則詳細資訊) 執行下列動作：
  - a. 輸入規則的 Name (名稱)，或者輸入描述。

在同一個區域和同一個事件匯流排上，規則不能與另一個規則同名。

- b. 針對 Event bus (事件匯流排) 選擇 default (預設值)。當您帳戶中的 AWS 服務產生事件時，一律會前往您帳戶的預設事件匯流排。
  - c. 針對 Rule type (規則類型) 選擇 Rule with an event pattern (具有事件模式的規則)。
  - d. 選擇下一步。
4. 針對 Build event pattern (建置事件模式) 執行下列動作：
- a. 對於事件來源，請選擇AWS 事件或 EventBridge合作夥伴事件。
  - b. 針對此範例的 Event pattern (事件模式)，您需指定下列事件模式，以便與 EC2 Instance Rebalance Recommendation 事件相符，然後選擇 Save (儲存)。

```
{
 "source": ["aws.ec2"],
 "detail-type": ["EC2 Instance Rebalance Recommendation"]
}
```

若要新增事件模式，您可以透過選擇 Event pattern form (事件模式表單) 使用範本，或者透過選擇 Custom pattern (JSON editor) (自定義模式 (JSON 編輯器)) 指定自有模式，如下：

- i. 若要使用範本建立事件模式，請執行下列動作：
    - A. 選擇 Event pattern form (事件模式表單)。
    - B. 在 Event source (事件來源) 欄位中，選擇 AWS services (服務)。
    - C. 針對 AWS Service (服務)，請選擇 EC2 Spot Fleet (EC2 Spot 機群)。
    - D. 針對 Event type (事件類型)，選擇 EC2 Instance Rebalance Recommendation (EC2 執行個體重新平衡建議)。
    - E. 若要自訂範本，請選擇 Edit pattern (編輯模式) 並進行變更以與範例事件模式相符。
  - ii. (替代) 若要指定自訂事件模式，請執行下列動作：
    - A. 選擇 Custom pattern (JSON editor) (自訂模式 (JSON 編輯器))。
    - B. 在 Event pattern (事件模式) 方塊中，為此範例新增事件模式。
  - c. 選擇下一步。
5. 針對 Select target(s) (選取目標) 執行下列動作：
- a. 在 Target types (目標類型) 欄位中，選擇 AWS service (服務)。
  - b. 針對 Select a target (選取目標) 選取 SNS topic (SNS 主題)，以便在事件發生時傳送電子郵件、簡訊或行動裝置推送通知。

- c. 針對 Topic (主題)，請選擇現有的主題。您首先需要使用 Amazon SNS 主控台建立 Amazon SNS 主題。如需詳細資訊，請參閱 [Amazon 簡單通知服務開發人員指南中的使用 Amazon SNS 進行 application-to-person \(A2P\) 簡訊](#)。
  - d. (選用) 在 Additional settings (其他設定) 下，您可以選擇性地設定其他設定。如需詳細資訊，請參閱 [Amazon EventBridge 使用者指南中的建立對事件做出反應的 Amazon EventBridge 規則](#) (步驟 16)。
  - e. 選擇下一步。
6. (選用) 針對 Tags (標籤)，您可以選擇性地將一或多個標籤指派給您的規則，然後選擇 Next (下一步)。
  7. 針對 Review and create (檢閱和建立) 執行下列動作：
    - a. 檢閱規則的詳細資訊，然後視需求進行修改。
    - b. 選擇建立規則。

有關更多信息，請參閱 [Amazon EventBridge 用戶指南中的 Amazon EventBridge 規則和 Amazon EventBridge 事件模式](#)

## 使用執行個體中繼資料

執行個體中繼資料類別 `events/recommendations/rebalance` 在為 Spot 執行個體發出重新平衡建議訊號時提供大約的時間 (UTC)。

我們建議您每 5 秒檢查重新平衡建議訊號，這樣您就不會錯過對重新平衡建議採取行動的機會。

如果 Spot 執行個體收到重新平衡建議，則發出訊號的時間會出現在執行個體中繼資料中。您可以擷取如下所示的發出訊號時間。

針對您的作業系統使用此指令。

### Linux

#### IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/events/recommendations/rebalance
```

## IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/events/recommendations/rebalance
```

## Windows

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/events/recommendations/rebalance
```

以下是範例輸出，它表示為 Spot 執行個體發出重新平衡建議訊號的時間 (UTC)。

```
{"noticeTime": "2020-10-27T08:22:00Z"}
```

如果尚未針對執行個體發出訊號，則 `events/recommendations/rebalance` 不存在，並且您會在嘗試擷取它時收到 HTTP 404 錯誤。

### 使用重新平衡建議訊號的服務

Amazon EC2 Auto Scaling、EC2 Fleet 和 Spot Fleet 使用重新平衡建議信號，讓您在執行中的執行個體收到兩分鐘 Spot 執行個體中斷通知之前，藉由使用新的 Spot 執行個體主動擴增您的機群，讓您輕鬆維持工作負載可用性。您可以讓這些服務主動地監控並回應影響 競價型執行個體 的可用性的變更 如需詳細資訊，請參閱下列內容：

- 《Amazon EC2 Auto Scaling 使用者指南》中的 [使用容量重新平衡處理 Amazon EC2 Spot 中斷](#)
- 在本使用者指南的 EC2 Fleet 主題中的 [容量重新平衡](#)
- 在本使用者指南的 Spot Fleet 主題中的 [容量重新平衡](#)

## Spot 執行個體中斷

您可以在備用 EC2 容量上啟動 競價型執行個體以獲得大幅折扣，做為在 Amazon EC2 需要取回容量時歸還的交換。當 Amazon EC2 回收 Spot 執行個體時，我們將此事件稱為 Spot 執行個體中斷。

當 Amazon EC2 中斷 Spot 執行個體時，它會終止、停止或休眠執行個體，具體取決於您在建立 Spot 請求時指定的功能。

對 競價型執行個體的需求可能會隨時產生極大的變化，而取決於有多少可用的未使用 EC2 執行個體，競價型執行個體的可用性也可能會有顯著的變化。Spot 執行個體隨時都有可能中斷。

EC2 Fleet 或 Spot Fleet 中指定的隨需執行個體不能中斷。

## 目錄

- [Spot 執行個體中斷的原因](#)
- [Spot 執行個體中斷的行為](#)
- [停止中斷的 競價型執行個體](#)
- [休眠中斷的 競價型執行個體](#)
- [終止已中斷的 Spot 執行個體](#)
- [為 Spot 執行個體中斷做好準備](#)
- [啟動 Spot 執行個體中斷](#)
- [Spot 執行個體中斷通知](#)
- [尋找中斷的 競價型執行個體](#)
- [確定 Amazon EC2 是否已終止 Spot 執行個體](#)
- [針對已中斷的 競價型執行個體收費](#)

## Spot 執行個體中斷的原因

下列是 Amazon EC2 會中斷 競價型執行個體的可能原因：

### 容量

Amazon EC2 可以在需要取回時中斷 Spot 執行個體。EC2 取回您的執行個體主要用於重新規劃容量，但也可能基於其他原因，例如主機維護或硬體解除委任。

### 價格

Spot 價格高於您的最高價格。

您可以在 Spot 請求中指定最高價格。然而，如果您指定了最高價格，執行個體將比未指定最高價時更頻繁受到中斷。

### 限制條件

如果 Spot 請求包含限制條件，例如啟動群組或可用區域群組，則不再滿足限制條件時，Spot 執行個體會以群組的形式終止。

您可以在 [Spot Instance Advisor](#) (Spot 執行個體顧問) 中查看執行個體類型的歷史中斷率。

## Spot 執行個體中斷的行為

您可以指定 Amazon EC2 應在中斷 Spot 執行個體時執行下列其中一項動作：

- [停止中斷的 競價型執行個體](#)
- [休眠中斷的 競價型執行個體](#)
- [終止已中斷的 Spot 執行個體](#) (這是預設行為)

### 指定中斷行為

您可以在建立 Spot 請求時指定中斷行為。若未指定中斷行為，預設是 Amazon EC2 在 Spot 執行個體中斷時將其終止。

您指定中斷行為的方式會根據您請求 競價型執行個體的方式而有所不同。

- 如果您使用 [啟動執行個體精靈](#) 請求 Spot 執行個體，可以按照下列方法指定中斷行為：在啟動執行個體精靈中，展開進階詳細資訊，然後選取請求 Spot 執行個體核取方塊。請選擇 Customize (自訂)。從中斷行為中，選擇中斷行為。如果中斷行為是休眠，您也可以為停止 - 休眠行為選擇啟用。
- 如果您使用 [run-instances](#) CLI 請求 Spot 執行個體，您可以按照下列方法指定中斷行為：在請求組態 (--instance-market-options) 中，針對 InstanceInterruptionBehavior，指定中斷行為。如果中斷行為是 hibernate，您也可以使用 --hibernation-options Configured=true 參數啟用休眠。
- 如果您在 [啟動範本](#) 中設定 競價型執行個體，您可以按照下列方法指定中斷行為：在啟動範本中，展開進階詳細資訊，然後選取 請求 競價型執行個體 核取方塊。選擇自訂，然後從中斷行為中選擇中斷行為。
- 如果您使用 [Spot 主控台](#) 請求 競價型執行個體，您可以按照下列方法指定中斷行為：選取維護目標容量核取方塊，然後從中斷行為中選取中斷行為。
- 如果您在使用 [create-fleet](#) CLI 時在啟動組態中設定 Spot 執行個體，您可以按照下列方法指定中斷行為：針對 InstanceInterruptionBehavior，指定中斷行為。
- 如果您在使用 [request-spot-fleet](#) CLI 時在啟動組態中設定 Spot 執行個體，您可以按照下列方法指定中斷行為：針對 InstanceInterruptionBehavior，指定中斷行為。
- 如果您使用 [request-spot-instances](#) CLI 設定 競價型執行個體，您可以按照下列方法指定中斷行為：針對 --instance-interruption-behavior，指定中斷行為。

**Note**

我們強烈反對使用 [request-spot-fleet](#) 和 [request-spot-instances](#) 命令來請求 Spot 執行個體，因為這是一個沒有計劃投資的舊式 API。如需更多資訊，請參閱 [使用哪種 Spot 請求方法最好？](#)

## 停止中斷的 競價型執行個體

您可以指定 Amazon EC2 在您的 Spot 執行個體中斷時將其停止。如需詳細資訊，請參閱 [指定中斷行為](#)。

### 考量事項

- 只有 Amazon EC2 才能重新啟動中斷的已停止 Spot 執行個體。
- 針對由 persistent Spot 執行個體請求啟動的 Spot 執行個體：當相同可用區域中的容量可用，且執行個體類型與已停止執行個體相同時，Amazon EC2 便會重新啟動已停止的執行個體。
- 對於由 EC2 Fleet 或 maintain 類型的 Spot Fleet 啟動的 Spot 執行個體：在 Spot 執行個體中斷後，Amazon EC2 會啟動替代執行個體來維持目標容量。Amazon EC2 會根據指定的分配策略 (lowestPrice、diversified 或 InstancePoolsToUseCount) 找出最佳 Spot 容量集區，它不會優先採用具有先前已停用之執行個體的集區。稍後，如果分配策略產生的集區包含先前已停用的執行個體，則 Amazon EC2 會重新啟動已停止的執行個體來符合目標容量。

例如，考量具有 lowestPrice 配置策略的 Spot Fleet。最初啟動時，c3.large 集區符合啟動規格的 lowestPrice 條件。稍後，當 c3.large 執行個體中斷時，則 Amazon EC2 會停止執行個體，並從另一個符合 lowestPrice 策略的集區補充容量。此時，集區剛好是 c4.large 集區，且 Amazon EC2 會啟動 c4.large 執行個體來符合目標容量。同樣地，Spot Fleet 下次可以移至 c5.large 集區。在其中每一個轉換中，Amazon EC2 不會優先採用具有先前已停用之執行個體的集區，而是僅根據指定的配置策略設定優先順序。lowestPrice 策略可以回到具有先前已停用之執行個體的集區。例如，如果執行個體中斷於 c5.large 集區，且 lowestPrice 策略讓它回到 c3.large 或 c4.large 集區，則先前已停用的執行個體會重新啟動，來實現目標容量。

- 停止 Spot 執行個體時，您可以修改其某些執行個體屬性，但無法修改執行個體類型。如果卸載或刪除了 EBS 磁碟區，當 Spot 執行個體啟動時，不會連接 EBS 磁碟區。如果卸載了根磁碟區，而 Amazon EC2 試著啟動 Spot 執行個體，則執行個體會啟動失敗，而且 Amazon EC2 會終止已停止的執行個體。
- 您可以在 Spot 執行個體停止時將其終止。
- 如果取消 Spot 執行個體請求、EC2 Fleet 或 Spot Fleet，Amazon EC2 會終止已停止的任何相關 Spot 執行個體。



- 當終止的 Spot 執行個體停止時，您只需支付保留 EBS 磁碟區的費用。使用 EC2 Fleet 和 Spot Fleet 時，如果有許多已停止的執行個體，可能會超出您帳戶的 EBS 磁碟區數量限制。如需在 Spot 執行個體中斷時計費方式的詳細資訊，請參閱 [針對已中斷的 競價型執行個體收費](#)。
- 確定您熟悉停止執行個體的含義。如需停止執行個體時會發生狀況的資訊，請參閱 [重新啟動、停止、休眠和終止之間的差異](#)。

## 必要條件

若要停止已終止的執行個體，必須具備下列必要條件：

### Spot 請求類型

Spot 執行個體請求類型 - 必須為 persistent。您無法在 Spot 執行個體請求中指定啟動群組。

EC2 Fleet 或 Spot Fleet 請求類型 - 必須為 maintain。

### 根磁碟區類型

必須是 EBS 磁碟區，而非執行個體存放區磁碟區。

## 休眠中斷的 競價型執行個體

您可以指定 Amazon EC2 在您的 Spot 執行個體中斷時將其休眠。如需詳細資訊，請參閱 [休眠您的 Amazon EC2 實例](#)。

Amazon EC2 現在提供與隨需執行個體目前使用相同的 Spot 執行個體休眠體驗。支援更廣泛，現在針對 Spot 執行個體休眠支援下列內容：

- [更多支援的 AMI](#)
- [更多支援的執行個體系列](#)
- [使用者啟動的休眠](#)

## 終止已中斷的 Spot 執行個體

當 Amazon EC2 中斷 Spot 執行個體時，預設情況下會終止執行個體，除非您指定了其他中斷行為，例如停止或休眠。如需詳細資訊，請參閱 [指定中斷行為](#)。

## 為 Spot 執行個體中斷做好準備

對競價型執行個體的需求可能會隨時產生極大的變化，而取決於有多少可用的未使用 EC2 執行個體，競價型執行個體的可用性也可能會有顯著的變化。Spot 執行個體隨時都有可能中斷。因此，您必須確定應用程式已經為 Spot 執行個體中斷做好準備。

建議您遵循這些最佳實務，以便為 Spot 執行個體中斷做好準備。

- 使用 Auto Scaling 群組建立 Spot 請求。如果您的 Spot 執行個體中斷，Auto Scaling 群組將自動啟動替換執行個體。如需詳細資訊，請參閱《Amazon EC2 Auto Scaling 使用者指南》中的[具備多個執行個體類型及購買選項的 Auto Scaling 群組](#)。
- 使用包含所需軟體組態的 Amazon Machine Image (AMI)，來確保執行個體在請求完成時能夠立即開始運作。您也可以利用使用者資料，在啟動時執行命令。
- 在停止或終止執行個體時，執行個體儲存體磁碟區上的資料會遺失。將執行個體儲存體磁碟區上的所有重要資料備份到更持久的儲存體，例如 Amazon S3、Amazon EBS 或 Amazon DynamoDB。
- 請將重要的資料定期存放於當 Spot 執行個體終止時不會受到影響的位置。例如，您可以使用 Amazon S3、Amazon EBS 或 DynamoDB。
- 將工作分割為較小的任務 (使用 Grid、Hadoop 或採用佇列的架構) 或使用檢查點，如此您就可以經常儲存工作。
- Amazon EC2 會在執行個體處在提高的中斷風險時對 Spot 執行個體發出重新平衡建議訊號。您可以依賴重新平衡建議，主動管理 Spot 執行個體中斷，而不必等待兩分鐘的 Spot 執行個體中斷通知。如需詳細資訊，請參閱[EC2 執行個體重新平衡建議](#)。
- 使用兩分鐘的 Spot 執行個體中斷通知來監控 Spot 執行個體的狀態。如需詳細資訊，請參閱[Spot 執行個體中斷通知](#)。
- 雖然我們努力地盡快提供這些警示，但您的 Spot 執行個體還是有可能在警示發布之前中斷。測試您的應用程式，確保它能夠從容地處理非預期的執行個體中斷，即使您正在監控重新平衡建議訊號和中斷通知。您可以執行使用隨需執行個體的應用程式，然後自行終止隨需執行個體，從而完成此任務。
- 執行受控故障注入實驗，AWS Fault Injection Service 以測試 Spot 執行個體中斷時應用程式的回應方式。如需詳細資訊，請參閱 AWS Fault Injection Service 使用者指南中[教學課程：使用 AWS FIS 測試 Spot 執行個體中斷](#)。

## 啟動 Spot 執行個體中斷

您可以在 Amazon EC2 主控台中選取 Spot 執行個體請求或 Spot 機群請求並啟動 Spot 執行個體中斷，以便您可以測試 Spot 執行個體上的應用程式如何處理中斷。當您啟動 Spot 執行個體中斷時，Amazon EC2 會通知您 Spot 執行個體將在兩分鐘後中斷，接著在兩分鐘後，執行個體會中斷。

執行 Spot 執行個體中斷的基礎服務為 AWS Fault Injection Service (AWS FIS)。如需有關的資訊 AWS FIS，請參閱 [AWS Fault Injection Service](#)。

#### Note

中斷行為是 terminate、stop 以及 hibernate。如果將中斷行為設定為 hibernate，當您啟動 Spot 執行個體中斷時，休眠程序將立即開始。

AWS 區域 除亞太區域 (雅加達)、亞太區域 (大阪)、中國 (北京)、中國 (寧夏) 和中東 (阿拉伯聯合大公國) 以外的所有項目都支援啟動 Spot 執行個體中斷。

#### 主題

- [啟動 Spot 執行個體中斷](#)
- [確認 Spot 執行個體中斷](#)
- [配額](#)

#### 啟動 Spot 執行個體中斷

您可以使用 EC2 主控台快速啟動 Spot 執行個體中斷。當您選取 Spot 執行個體請求時，您可以啟動一個 Spot 執行個體的中斷。當您選取 Spot 機群請求時，您可以一次啟動多個 Spot 執行個體的中斷。

如需測試 Spot 執行個體中斷的更進階實驗，您可以使用 AWS FIS 主控台建立自己的實驗。

使用 EC2 主控台啟動 Spot 執行個體請求中的一個 Spot 執行個體的中斷

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot Requests (Spot 請求)。
3. 選取 Spot 執行個體請求，然後選取 Actions (動作)、Initiate interruption (啟動中斷)。您無法選取多個 Spot 執行個體請求來啟動中斷。
4. 在 Initiate Spot Instance interruption (啟動 Spot 執行個體中斷) 對話方塊，於 Service access (服務存取) 下，使用預設角色或選擇現有角色。若要選取現有角色，請選取使用現有的服務角色，然後對於 IAM 角色，選取要使用的角色。
5. 當您準備好要啟動 Spot 執行個體中斷時，請選擇 Initiate interruption (啟動中斷)。

使用 EC2 主控台啟動 Spot 機群請求中的一個或多個 Spot 執行個體的中斷

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot Requests (Spot 請求)。
3. 選取 Spot 機群請求，然後選取動作、啟動中斷。您無法選取多個 Spot 機群請求來啟動中斷。
4. 在指定 Spot 執行個體的數目對話方塊中，對於要中斷之執行個體的數目，輸入要中斷之 Spot 執行個體的數目，然後選擇確認。

#### Note

此數量不得超過叢集中的 Spot 執行個體數量，或您每個實驗 AWS FIS 可能中斷的 Spot 執行個體數量 [配額](#)。

5. 在 Initiate Spot Instance interruption (啟動 Spot 執行個體中斷) 對話方塊，於 Service access (服務存取) 下，使用預設角色或選擇現有角色。若要選取現有角色，請選取使用現有的服務角色，然後對於 IAM 角色，選取要使用的角色。
6. 當您準備好要啟動 Spot 執行個體中斷時，請選擇 Initiate interruption (啟動中斷)。

使用 AWS FIS 主控台建立更進階的實驗以測試 Spot 執行個體中斷

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot Requests (Spot 請求)。
3. 選擇 Actions (動作)，Create advanced experiments (建立進階實驗)。

AWS FIS 主控台隨即開啟。如需詳細資訊，請參閱《AWS Fault Injection Service 使用者指南》中的 [教學課程：使用 AWS FIS 測試 Spot 執行個體中斷](#)。

確認 Spot 執行個體中斷

在您啟動中斷後，會發生下列情況：

- Spot 執行個體會收到 [執行個體重新平衡建議](#)。
- [Spot 執行個體中斷通知](#) 會在執行個體中 AWS FIS 斷前兩分鐘發出。
- 兩分鐘後，系統會中斷 Spot 執行個體。
- 由停止的 Spot 執行個體會 AWS FIS 保持停止狀態，直到您重新啟動為止。

## 確認執行個體在您啟動中斷後是否已中斷

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 從導覽窗格中，在單獨的瀏覽器索引標籤或視窗中開啟 Spot Requests (Spot 請求) 和 Instances (執行個體)。
3. 對於 Spot 請求，選取 Spot 執行個體請求或 Spot 機群請求。起始狀態為 fulfilled。執行個體中斷後，根據中斷行為，狀態會有如下變更：
  - terminate – 狀態會變成 instance-terminated-by-experiment。
  - stop – 狀態會變成 marked-for-stop-by-experiment，然後再變成 instance-stopped-by-experiment。
4. 對於 Instances (執行個體)，選取 Spot 執行個體。起始狀態為 Running。收到 Spot 執行個體中斷通知後兩分鐘，根據中斷行為，狀態會有如下變更：
  - stop – 狀態會變成 Stopping，然後再變成 Stopped。
  - terminate – 狀態會變成 Shutting-down，然後再變成 Terminated。

## 配額

對於每 AWS 帳戶 個實驗 AWS FIS 可中斷的 Spot 執行個體數量，您的預設配額如下。

| 名稱                                | 預設         | 可調整 | 描述                                                                               |
|-----------------------------------|------------|-----|----------------------------------------------------------------------------------|
| aws SpotInstances 的目標：ec2：發送點實例中斷 | 每個受支援的區域：5 | 是   | 當您使用每個實驗使用標籤識別目標時，aw: ec2: send-spot-instance-interruptions 可以鎖定的 Spot 執行個體數目上限。 |

您可以要求增加配額。如需詳細資訊，請參閱《Service Quotas 使用者指南》中的[請求提高配額](#)。

若要檢視的所有配額 AWS FIS，請開啟「[Service Quotas](#)」主控台。在導覽窗格中，選擇 AWS 服務，然後選取 AWS Fault Injection Service。您也可以在此 AWS Fault Injection Service 使用者指南中檢視 [AWS Fault Injection Service](#) 的所有配額。

## Spot 執行個體中斷通知

Spot 執行個體中斷通知是在 Amazon EC2 停止或終止 Spot 執行個體的兩分鐘之前發出的警告。如果將休眠指定為中斷時的行為，您會收到中斷通知，但不會提前兩分鐘收到警告，因為休眠程序會立刻開始執行。

從容處理 Spot 執行個體中斷的最佳方法，就是建立容錯的應用程式架構。若要達成此目的，您可以利用 Spot 執行個體中斷通知。建議您每隔 5 秒檢查這些中斷通知。

中斷通知可作為 EventBridge 事件和 Spot 執行個體上[執行個體中繼資料](#)中的項目提供。中斷通知的原則是盡可能發出。

### EC2 Spot Instance interruption notice

當 Amazon EC2 要中斷您的 Spot 執行個體時，它會在實際中斷前兩分鐘發出一個事件 (休眠除外，它會收到中斷通知，但不會提前兩分鐘，因為休眠會立即開始)。Amazon 可以檢測到此事件 EventBridge。如需有關 EventBridge 事件的詳細資訊，請參閱 [Amazon EventBridge 使用者指南](#)。如需如何建立和使用事件規則的詳細範例，請參閱[利用 Amazon EC2 Spot 執行個體中斷通知](#)。

下列是 Spot 執行個體中斷事件的範例。instance-action 可能的值為 hibernate、stop 或 terminate。

```
{
 "version": "0",
 "id": "12345678-1234-1234-1234-123456789012",
 "detail-type": "EC2 Spot Instance Interruption Warning",
 "source": "aws.ec2",
 "account": "123456789012",
 "time": "yyyy-mm-ddThh:mm:ssZ",
 "region": "us-east-2",
 "resources": ["arn:aws:ec2:us-east-2a:instance/i-1234567890abcdef0"],
 "detail": {
 "instance-id": "i-1234567890abcdef0",
 "instance-action": "action"
 }
}
```

#### Note

Spot 執行個體中斷事件的 ARN 格式為 `arn:aws:ec2:availability-zone:instance/instance-id`。此格式不同於 [EC2 資源 ARN 格式](#)。

## instance-action

如果 Spot 執行個體已標記為由 Amazon EC2 停止或終止，則[執行個體中繼資料](#)中會出現 instance-action 項目。否則不會存在。您可以 instance-action 使用執行個體中繼資料服務版本 2 (IMDSv2) 擷取，如下所示。

針對您的作業系統使用此指令。

### Linux

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/spot/instance-action
```

### Windows

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/meta-data/spot/instance-action
```

instance-action 項目會指定動作和動作發生的約略時間 (UTC)。

下列範例輸出指出此執行個體將會停止的時間點。

```
{"action": "stop", "time": "2017-09-18T08:22:00Z"}
```

下列範例輸出指出此執行個體將會終止的時間點。

```
{"action": "terminate", "time": "2017-09-18T08:22:00Z"}
```

如果 Amazon EC2 未準備停止或終止執行個體，或是您自行終止了執行個體，則執行個體中繼資料中不會出現 instance-action，而且您會在嘗試擷取它時收到 HTTP 404 錯誤。

## termination-time

保留此項目是為了回溯相容性；您應改用 instance-action。

如果您的競價型執行個體被 Amazon EC2 標記為終止 (可能是因為中斷行為設定為的競價型執行個體中斷 terminate，或是因為永久 Spot 執行個體請求取消)，則該 termination-time 項目會出現在[執行個體中繼資料](#)中。否則不會存在。您可以 termination-time 使用 ImDSv2 擷取，如下所示。

針對您的作業系統使用此指令。

## Linux

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"`
[ec2-user ~]$ if curl -H "X-aws-ec2-metadata-token: $TOKEN" -s http://169.254.169.254/latest/meta-data/spot/termination-time | grep -q .*T.*Z; then echo
 termination_scheduled; fi
```

## Windows

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{ "X-aws-ec2-metadata-token-ttl-seconds" = "21600" } -Method PUT -Uri http://169.254.169.254/latest/meta-data/spot/termination-time
```

此 `termination-time` 項目會指定執行個體接收關機訊號時的大約時間 (以 UTC 為單位)。下列為範例輸出。

```
2015-01-05T18:02:00Z
```

如果 Amazon EC2 並未準備終止執行個體 (可能是因為沒有競價型執行個體中斷或是因為中斷行為設定為 `stop` 或 `hibernate`)，或者如果您自行終止 Spot 執行個體，則該 `termination-time` 項目可能不存在於執行個體中繼資料中 (因此您會收到 HTTP 404 錯誤) 或包含非時間值的值。

如果 Amazon EC2 終止執行個體的動作失敗，則請求狀態會設定為 `fulfilled`。`termination-time` 值仍會存在於執行個體中繼資料中，包含原來的約略時間 (現在這個時間已經成為過去)。

### 尋找中斷的 競價型執行個體

在主控台中，執行個體窗格會顯示所有執行個體，包括 競價型執行個體。Spot 執行個體的執行個體生命週期為 `spot`。Spot 執行個體的執行個體狀態為 `stopped` 或 `terminated`，視您設定的中斷行為而定。若為已休眠的 Spot 執行個體，執行個體狀態為 `stopped`。

### 使用主控台尋找中斷的 Spot 執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 套用下列篩選條件：執行個體生命週期=`spot`。



4. 根據您設定的中斷行為，套用執行個體狀態=已停止或執行個體狀態=已終止篩選條件。
5. 對於每個 Spot 執行個體，在詳細資訊索引標籤的執行個體詳細資訊下，找到狀態轉換訊息。下列程式碼指出 Spot 執行個體已中斷。
  - `Server.SpotInstanceShutdown`
  - `Server.SpotInstanceTermination`
6. 如需中斷原因的其他詳細資訊，請檢查 Spot 要求狀態碼。如需詳細資訊，請參閱 [the section called "Spot 請求狀態"](#)。

## 若要使用 AWS CLI

您可以使用 [describe-instances](#) 命令搭配 `--filters` 參數來列出中斷的 競價型執行個體。若只要在輸出中列出執行個體 ID，請包含 `--query` 參數。

如果執行個體中斷行為是終止 Spot 執行個體，請使用下列命令：

```
aws ec2 describe-instances \
 --filters Name=instance-lifecycle,Values=spot Name=instance-state-
 name,Values=terminated Name=state-reason-code,Values=Server.SpotInstanceTermination \
 --query "Reservations[*].Instances[*].InstanceId"
```

如果執行個體中斷行為是停止 Spot 執行個體，請使用下列命令：

```
aws ec2 describe-instances \
 --filters Name=instance-lifecycle,Values=spot Name=instance-state-
 name,Values=stopped Name=state-reason-code,Values=Server.SpotInstanceShutdown \
 --query "Reservations[*].Instances[*].InstanceId"
```

## 確定 Amazon EC2 是否已終止 Spot 執行個體

如果競價型執行個體終止，您可以使用 CloudTrail 來查看 Amazon EC2 是否終止競價型執行個體。在 AWS CloudTrail 中，事件名稱 `BidEvictedEvent` 指出 Amazon EC2 已終止 Spot 執行個體。

## 若要檢視中的 BidEvictedEvent 事件 CloudTrail

1. [請在以下位置開啟 CloudTrail 主控台。](https://console.aws.amazon.com/cloudtrail/) <https://console.aws.amazon.com/cloudtrail/>
2. 在導覽窗格中，選擇 Event history (事件歷史記錄)。
3. 在篩選器下拉式清單中，選擇 [事件名稱]，然後在右側的篩選欄位中輸入 [BidEvicted事件]。

4. 在結果清單中選擇「BidEvicted事件」以檢視其詳細資訊。在 Event record (事件記錄) 下，您可以找到執行個體 ID。

如需使用的更多資訊 CloudTrail，請參閱[使用記錄 Amazon EC2 API 呼叫 AWS CloudTrail](#)。

### 針對已中斷的 競價型執行個體收費

當 Spot 執行個體中斷時，您得支付執行個體和 EBS 磁碟區用量的費用，您可能需要支付其他費用，收費如下。

### 執行個體用量

| 誰中斷 Spot 執行個體              | 作業系統               | 在第一個小時中斷            | 在第一個小時之後的任何小時中斷          |
|----------------------------|--------------------|---------------------|--------------------------|
| 如果您停止或終止 Spot 執行個體         | 視窗和工作系統 (不包括 SUSE) | 按使用秒數收費             | 按使用秒數收費                  |
|                            | SUSE               | 按整整一小時收費，即使只使用半小時亦同 | 按使用的完整時數收費，中斷半小時以整整一小時收費 |
| 如果 Amazon EC2 中斷 Spot 執行個體 | 視窗和工作系統 (不包括 SUSE) | 免費                  | 按使用秒數收費                  |
|                            | SUSE               | 免費                  | 按使用的完整時數收費，但中斷半小時免費      |

### EBS 磁碟區用量

當中斷的 Spot 執行個體停止時，您只需支付保留 EBS 磁碟區的費用。

使用 EC2 Fleet 和 Spot Fleet 時，如果有許多已停止的執行個體，可能會超出您帳戶的 EBS 磁碟區數量限制。

### 其他變更

如果執行中的競價型執行個體需要支付其他服務的費用，例如資料傳輸、彈性 IP 地址或使用其他 AWS 受管理服務，則需按使用量向您收費。這與中斷 Spot 執行個體的對象和時間無關。即使

Amazon EC2 在第一個小時中斷您的 Spot 執行個體，您也不必因此支付 Spot 執行個體用量費用，但您可能需要支付其他費用。

如需其他費用的詳細資訊，請參閱 [Amazon EC2 隨需定價](#)。

## Spot 配置分數

Spot 放置分數功能可以根據您的競價型容量需求推薦區 AWS 域或可用區域。Spot 容量會波動，而且您無法確定您始終都能取得所需的容量。Spot 配置分數表示 Spot 請求在區域或可用區域中成功的可能性。

### Note

Spot 配置分數不會就可用容量或中斷風險提供任何保證。Spot 配置分數僅作為建議提供。

## 優勢

您可以針對下列情況使用 Spot 配置分數功能：

- 視需要在不同區域中重新放置並擴展 Spot 運算容量，以回應目前區域中增加的容量需求或降低的可用容量。
- 識別執行單一可用區域工作負載的最佳可用區域。
- 模擬未來的 Spot 容量需求，以便您可以挑選最佳區域來擴充 Spot 型工作負載。
- 尋找執行個體類型的最佳組合以滿足您的 Spot 容量需求。

## 主題

- [成本](#)
- [Spot 配置分數的運作方式](#)
- [限制](#)
- [所需的 IAM 許可](#)
- [計算 Spot 配置分數](#)
- [範例組態](#)

## 成本

使用 Spot 配置分數功能無需額外費用。

## Spot 配置分數的運作方式

當您使用 Spot 配置分數功能時，首先要針對 Spot 執行個體指定運算需求，然後 Amazon EC2 會傳回 Spot 請求可能成功的前 10 個區域或可用區域。每個區域或可用區域的分數範圍從 1 到 10，其中 10 表示您的前 Spot 請求很可能成功，1 表示您的 Spot 請求不可能成功。

若要使用 Spot 配置分數功能，請遵循下列步驟：

- [步驟 1：指定您的 Spot 需求](#)
- [步驟 2：篩選 Spot 配置分數回應](#)
- [步驟 3：檢閱建議](#)
- [步驟 4：使用建議](#)

### 步驟 1：指定您的 Spot 需求

首先，您可以指定所需的目標 Spot 容量和運算需求，如下所示：

#### 1. 指定目標 Spot 容量，並選擇性指定目標容量單位。

您可以根據執行個體或 vCPU 的數量或 MiB 中的記憶體容量來指定所需的目標 Spot 容量。若要以 vCPU 數目或記憶體數量指定目標容量，您必須將目標容量單位指定為 `vcpu` 或 `memory-mib`。否則，其會預設為執行個體的數量。

根據 vCPU 數目或記憶體數量來指定目標容量，您可以在計算總容量時使用這些單位。例如，如果想要混合使用不同大小的執行個體，您可以將目標容量指定為 vCPU 總數。Spot 配置分數功能接著會透過其 vCPU 數目來考量請求中的每個執行個體類型，並在加總目標容量時計算 vCPU 總數，而不是執行個體總數。

例如，假設您指定的目標容量總計為 30 個 vCPU，而執行個體類型清單包含 `c5.xlarge` (4 個 vCPU)、`m5.2xlarge` (8 個 vCPU) 和 `r5.large` (2 個 vCPU)。若要達到總數 30 個的 vCPU，您可以取得 2 `c5.xlarge` (2\*4 個 vCPU)、2 `m5.2xlarge` (2\*8 個 vCPU) 和 3 `r5.large` (3\*2 個 vCPUs) 的混合。

#### 2. 指定執行個體類型或執行個體屬性。

您可以指定要使用的執行個體類型，也可以指定運算需求所需的執行個體屬性，然後讓 Amazon EC2 識別具有這些屬性的執行個體類型。這就是所謂的屬性型執行個體類型選項。

您無法在相同的 Spot 配置分數請求中同時指定執行個體類型和執行個體屬性。

如果指定執行個體類型，則您必須至少指定三種不同的執行個體類型，否則 Amazon EC2 將傳回低 Spot 配置分數。同樣地，如果您指定執行個體屬性，它們必須解析為至少三種不同的執行個體類型。

如需指定 Spot 需求的不同方式範例，請參閱 [範例組態](#)。

### 步驟 2：篩選 Spot 配置分數回應

Amazon EC2 會計算每個區域或可用區域的 Spot 配置分數，並傳回 Spot 請求可能成功的前 10 個區域或前 10 個可用區域。預設值是傳回得分區域的清單。如果您計劃將所有 Spot 容量啟動至單一可用區域，請求得分可用區域的清單很有用。

您可以指定區域篩選條件來縮小將在回應中傳回的區域。

您可以結合區域篩選條件和得分可用區域的請求。如此一來，得分可用區域僅限於您已篩選的區域。若要找出區域中得分最高的可用區域，請僅指定該區域，然後回應將傳回該區域中所有可用區域的得分清單。

### 步驟 3：檢閱建議

每個區域或可用區域的 Spot 配置分數是根據目標容量、執行個體類型的組成、歷史和目前 Spot 使用趨勢，以及請求的時間來計算。由於 Spot 容量持續波動，因此在不同時間計算時，相同的 Spot 配置分數請求可能會產生不同的分數。

區域和可用區域的分數範圍是從 1 到 10。分數 10 表示您的 Spot 請求很有可能 (但不保證) 成功。分數 1 表示您的 Spot 請求完全不可能成功。對於不同的區域或可用區域，可能會傳回相同的分數。

如果傳回低分數，您可以編輯運算需求並重新計算分數。您也可以在一天中的不同時間，針對相同的運算需求，請求 Spot 配置分數建議。

### 步驟 4：使用建議

僅在 Spot 請求的組態與 Spot 配置分數組態 (目標容量、目標容量單位，以及執行個體類型或執行個體屬性) 完全相同，且該請求設定為使用 `capacity-optimized` 配置策略時，Spot 配置分數才相關。否則，取得可用 Spot 容量的可能性將不符合分數。

雖然 Spot 配置分數充當指導方針，而且沒有任何分數保證您的 Spot 請求會完全或部分實現，但您可以使用下列資訊來取得最佳結果：

- 使用相同組態 – 僅在 Auto Scaling 群組、EC2 Fleet 或 Spot Fleet 中的 Spot 請求組態 (目標容量、目標容量單位，以及執行個體類型或執行個體屬性) 與您為了取得 Spot 配置分數而輸入的組態相同時，Spot 配置分數才相關。

如果您在 Spot 配置分數請求中使用了屬性型執行個體類型選取，則可以使用屬性型執行個體類型選取，來設定 Auto Scaling 群組、EC2 Fleet 或 Spot Fleet。如需詳細資訊，請參閱[建立對所使用之執行個體類型具有一組需求的 Auto Scaling 群組](#)、[EC2 Fleet 的屬性型執行個體類型選擇](#)和 [Spot Fleet 的屬性型執行個體類型選擇](#)。

#### Note

如果您根據 vCPU 數目或記憶體數量指定目標容量，並在 Spot 配置分數組態中指定了執行個體類型，請注意，您目前無法在 Auto Scaling 群組、EC2 Fleet 或 Spot Fleet 中建立此組態。反之，您必須使用 `WeightedCapacity` 參數，手動設定執行個體加權。

- 使用 **capacity-optimized** 分配策略 – 任何分數都假設您的機群請求將設定為使用所有的可用區域 (用於跨區域請求容量) 或單一可用區域 (如果在某個可用區域請求容量) 和 `capacity-optimized` Spot 分配策略，以便您可以成功請求 Spot 容量。如果您使用其他配置策略 (例如 `lowest-price`)，取得可用 Spot 容量的可能性將不符合分數。
- 立即對分數採取行動 – Spot 配置分數建議會反映請求時的可用 Spot 容量，並且由於 Spot 容量波動，因此在不同時間計算時，相同的組態可能會產生不同的分數。雖然分數 10 表示您的 Spot 容量請求很可能 (但不保證) 成功，但為了取得最佳結果，我們建議您立即對分數採取行動。我們也建議您在每次嘗試容量請求時取得全新的分數。

## 限制

- 目標容量限制 – 您的 Spot 配置分數目標容量限制是基於最近的 Spot 用量，同時考量潛在的用量成長。如果沒有最近的 Spot 用量，我們會為您提供符合 Spot 請求限制的低預設限制。
- 請求組態限制 – 如果我們偵測到未與 Spot 配置分數功能之預期用途相關聯的模式，可以在 24 小時期間內限制新請求組態的數目。如果達到限制，您可以重試已使用的請求組態，但無法指定新的請求組態，直到下一個 24 小時期間。
- 執行個體類型數目下限 – 如果指定執行個體類型，則您必須至少指定三種不同的執行個體類型，否則 Amazon EC2 將傳回低 Spot 配置分數。同樣地，如果您指定執行個體屬性，它們必須解析為至少三種不同的執行個體類型。如果執行個體類型具有不同的名稱，則會將其視為不同。例如，`m5.8xlarge`、`m5a.8xlarge` 和 `m5.12xlarge` 都會視為不同。

## 所需的 IAM 許可

根據預設，IAM 身分 (使用者、角色或群組) 沒有使用 Spot 配置分數功能的許可。若要允許 IAM 身分使用 Spot 配置分數功能，您必須建立 IAM 政策，授予使用 `ec2:GetSpotPlacementScores` EC2 API 動作的許可。接著將政策連接至需要此許可的 IAM 身分。

以下為一個範例 IAM 政策，該政策授予使用 `ec2:GetSpotPlacementScores` EC2 API 動作的許可。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "ec2:GetSpotPlacementScores",
 "Resource": "*"
 }
]
}
```

如需編輯 IAM 政策的相關資訊，請參閱《IAM 使用者指南》[https://docs.aws.amazon.com/IAM/latest/UserGuide/access\\_policies\\_manage-edit.html](https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies_manage-edit.html)中的編輯 IAM 政策。

若要提供存取權，請新增權限至您的使用者、群組或角色：

- 使用者和群組位於 AWS IAM Identity Center：

建立權限合集。請按照 AWS IAM Identity Center 使用者指南 中的 [建立權限合集](#) 說明進行操作。

- 透過身分提供者在 IAM 中管理的使用者：

建立聯合身分的角色。請按照 IAM 使用者指南 的 [為第三方身分提供者 \(聯合\) 建立角色](#) 中的指示進行操作。

- IAM 使用者：

- 建立您的使用者可擔任的角色。請按照 IAM 使用者指南 的 [為 IAM 使用者建立角色](#) 中的指示進行操作。

- (不建議) 將政策直接附加至使用者，或將使用者新增至使用者群組。請遵循 IAM 使用者指南的 [新增許可到使用者 \(主控台\)](#) 中的指示。



## 計算 Spot 配置分數

您可以使用 Amazon EC2 主控台或 AWS CLI，計算 Spot 配置分數。

### 主題

- [指定執行個體屬性來計算 Spot 配置分數 \(主控台\)](#)
- [指定執行個體類型來計算 Spot 配置分數 \(主控台\)](#)
- [計算 Spot 配置分數 \(AWS CLI\)](#)

### 指定執行個體屬性來計算 Spot 配置分數 (主控台)

#### 執行個體屬性來計算 Spot 配置分數

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選擇 Spot placement score (Spot 配置分數)。
4. 選擇 Enter requirements (輸入需求)。
5. 對於 Target capacity (目標容量)，根據 instances (執行個體) 或 vCPUs 的數目，或 memory (MiB) (記憶體 (MiB)) 的數量輸入您所需的容量。
6. 對於 Instance type requirements (執行個體類型需求)，若要指定您的運算需求，並讓 Amazon EC2 識別符合這些要求的最佳執行個體類型，請選擇 Specify instance attributes that match your compute requirements (指定符合運算需求的執行個體屬性)。
7. 對於 vCPUs，輸入所需的 vCPU 數量下限和上限。若要指定無限制，請選取 No minimum (無下限)、No maximum (無上限)，或兩者。
8. 對於 Memory (GiB) (記憶體 (GiB))，輸入所需記憶體數量的下限和上限。若要指定無限制，請選取 No minimum (無下限)、No maximum (無上限)，或兩者。
9. 對於 CPU architecture (CPU 架構)，選取所需的執行個體架構。
10. (選用) 對於 Additional instance attributes (其他執行個體屬性)，您可以選擇性地指定一或多個屬性，以更詳細地表達您的運算需求。每個額外屬性都會將進一步的限制新增至您的請求。您可以省略額外屬性；省略時，就會使用預設值。如需每個屬性及其預設值的描述，請參閱 Amazon EC2 命令列參考中的 [get-spot-placement-scores](#)。
11. (選用) 若要檢視具有所指定屬性的執行個體類型，請展開 Preview matching instance types (預覽相符的執行個體類型)。若要排除執行個體類型，使其不在配置評估中使用，請選取執行個體，然後選取 Exclude selected instance types (排除選取的執行個體類型)。
12. 選擇 Load placement scores (載入配置分數)，然後檢閱結果。

13. (選用) 若要顯示特定區域的 Spot 配置分數，請針對 Regions to evaluate (要評估的區域)，選取要評估的區域，然後選取 Calculate placement scores (計算配置分數)。
14. (選用) 若要顯示所顯示區域中可用區域的 Spot 配置分數，請選取 Provide placement scores per Availability Zone (提供每個可用區域的配置分數) 核取方塊。如果您想要將所有 Spot 容量啟動至單一可用區域，得分可用區域的清單很有用。
15. (選用) 若要編輯您的運算需求並取得新的配置分數，請選擇 Edit (編輯)、進行必要的調整，然後選擇 Calculate placement scores (計算配置分數)。

指定執行個體類型來計算 Spot 配置分數 (主控台)

指定執行個體類型來計算 Spot 配置分數

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選擇 Spot placement score (Spot 配置分數)。
4. 選擇 Enter requirements (輸入需求)。
5. 對於 Target capacity (目標容量)，根據 instances (執行個體) 或 vCPUs 的數目，或 memory (MiB) (記憶體 (MiB)) 的數量輸入您所需的容量。
6. 對於 Instance type requirements (執行個體類型需求)，若要指定要使用的執行個體類型，請選取 Manually select instance types (手動選取執行個體類型)。
7. 選取 Select instance types (選取執行個體類型)、選取要使用的執行個體類型，然後選取 Select (選取)。若要快速尋找執行個體類型，您可以使用篩選條件列，依不同屬性篩選執行個體類型。
8. 選擇 Load placement scores (載入配置分數)，然後檢閱結果。
9. (選用) 若要顯示特定區域的 Spot 配置分數，請針對 Regions to evaluate (要評估的區域)，選取要評估的區域，然後選取 Calculate placement scores (計算配置分數)。
10. (選用) 若要顯示所顯示區域中可用區域的 Spot 配置分數，請選取 Provide placement scores per Availability Zone (提供每個可用區域的配置分數) 核取方塊。如果您想要將所有 Spot 容量啟動至單一可用區域，得分可用區域的清單很有用。
11. (選用) 若要編輯執行個體類型的清單並取得新的配置分數，請選擇 Edit (編輯)、進行必要的調整，然後選擇 Calculate placement scores (計算配置分數)。

## 計算 Spot 配置分數 (AWS CLI)

### 計算 Spot 配置分數

1. (選用) 若要產生可以針對 Spot 配置分數組態指定的所有可能參數，請使用 [get-spot-placement-scores](#) 命令和 `--generate-cli-skeleton` 參數。

```
aws ec2 get-spot-placement-scores \
 --region us-east-1 \
 --generate-cli-skeleton
```

### 預期的輸出結果

```
{
 "InstanceTypes": [
 ""
],
 "TargetCapacity": 0,
 "TargetCapacityUnitType": "vcpu",
 "SingleAvailabilityZone": true,
 "RegionNames": [
 ""
],
 "InstanceRequirementsWithMetadata": {
 "ArchitectureTypes": [
 "x86_64_mac"
],
 "VirtualizationTypes": [
 "hvm"
],
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 0,
 "Max": 0
 },
 "MemoryMiB": {
 "Min": 0,
 "Max": 0
 },
 "CpuManufacturers": [
 "amd"
],
 },
 },
}
```

```
"MemoryGiBPerVCpu": {
 "Min": 0.0,
 "Max": 0.0
},
"ExcludedInstanceTypes": [
 ""
],
"InstanceGenerations": [
 "previous"
],
"SpotMaxPricePercentageOverLowestPrice": 0,
"OnDemandMaxPricePercentageOverLowestPrice": 0,
"BareMetal": "excluded",
"BurstablePerformance": "excluded",
"RequireHibernateSupport": true,
"NetworkInterfaceCount": {
 "Min": 0,
 "Max": 0
},
"LocalStorage": "included",
"LocalStorageTypes": [
 "hdd"
],
"TotalLocalStorageGB": {
 "Min": 0.0,
 "Max": 0.0
},
"BaselineEbsBandwidthMbps": {
 "Min": 0,
 "Max": 0
},
"AcceleratorTypes": [
 "fpga"
],
"AcceleratorCount": {
 "Min": 0,
 "Max": 0
},
"AcceleratorManufacturers": [
 "amd"
],
"AcceleratorNames": [
 "vu9p"
],
```

```
 "AcceleratorTotalMemoryMiB": {
 "Min": 0,
 "Max": 0
 }
 },
 "DryRun": true,
 "MaxResults": 0,
 "NextToken": ""
}
```

2. 使用上一個步驟的輸出來建立 JSON 組態檔，並將其設定如下：

- a. 對於 TargetCapacity，根據執行個體或 vCPU 的數目，或記憶體 (MiB) 的數量輸入您所需的 Spot 容量。
- b. 對於 TargetCapacityUnitType，輸入目標容量的單位。如果省略此參數，則其會預設為 units。

有效值：units (其會轉換為執行個體的數量) | vcpu | memory-mib

- c. 對於 SingleAvailabilityZone，指定 true，以取得傳回得分可用區域清單的回應。如果您想要將所有 Spot 容量啟動至單一可用區域，得分可用區域的清單很有用。如果省略此參數，則其會預設為 false，並且回應會傳回得分區域的清單。
- d. (選用) 對於 RegionNames，請指定要用作篩選條件的區域。您必須指定區域代碼，例如 us-east-1。

搭配區域篩選條件，回應只會傳回您指定的區域。如果您針對 SingleAvailabilityZone 指定了 true，則回應只會傳回所指定區域中的可用區域。

- e. 您可以包含 InstanceTypes 或 InstanceRequirements，但兩者不能同時位於相同的組態中。

在 JSON 組態中指定下列其中一個項目：

- 若要指定執行個體類型的清單，請在 InstanceTypes 參數中指定執行個體類型。指定至少三種不同的執行個體類型。如果您只指定一個或兩個執行個體類型，Spot 配置分數會傳回低分數。如需執行個體類型的清單，請參閱 [Amazon EC2 執行個體類型](#)。
- 若要指定執行個體屬性，以便 Amazon EC2 可以識別符合這些屬性的執行個體類型，請指定位於 InstanceRequirements 結構中的屬性。

您必須提供 VCpuCount、MemoryMiB 和 CpuManufacturers 的值。您可以省略其他屬性；省略時，就會使用預設值。如需每個屬性及其預設值的描述，請參閱 Amazon EC2 命令列參考中的 [get-spot-placement-scores](#)。

如需組態範例，請參閱 [範例組態](#)。

- 若要取得您在 JSON 檔案中指定之需求的 Spot 配置分數，請使用 [get-spot-placement-scores](#) 命令，並使用 `--cli-input-json` 參數指定 JSON 檔案的名稱和路徑。

```
aws ec2 get-spot-placement-scores \
 --region us-east-1 \
 --cli-input-json file://file_name.json
```

範例輸出：如果 `SingleAvailabilityZone` 設定為 `false` 或將其省略 (如果省略，其會預設為 `false`)，則會傳回區域的得分清單

```
"SpotPlacementScores": [
 {
 "Region": "us-east-1",
 "Score": 7
 },
 {
 "Region": "us-west-1",
 "Score": 5
 },
 ...
```

範例輸出：如果 `SingleAvailabilityZone` 設定為 `true`，則會傳回可用區域的得分清單

```
"SpotPlacementScores": [
 {
 "Region": "us-east-1",
 "AvailabilityZoneId": "use1-az1"
 "Score": 8
 },
 {
 "Region": "us-east-1",
 "AvailabilityZoneId": "usw2-az3"
 "Score": 6
 },
```

...

## 範例組態

使用時 AWS CLI，您可以使用下列範例組態。

## 範例組態

- [範例：指定執行個體類型和目標容量](#)
- [範例：根據記憶體指定執行個體類型和目標容量](#)
- [範例：為屬性型執行個體類型選取指定屬性](#)
- [範例：為屬性型執行個體類型選取指定屬性，並傳回可用區域的得分清單](#)

### 範例：指定執行個體類型和目標容量

下列範例組態指定三種不同的執行個體類型，以及有 500 個 Spot 執行個體的目標 Spot 容量。

```
{
 "InstanceTypes": [
 "m5.4xlarge",
 "r5.2xlarge",
 "m4.4xlarge"
],
 "TargetCapacity": 500
}
```

### 範例：根據記憶體指定執行個體類型和目標容量

下列範例組態指定了三種不同的執行個體類型，以及有 500,000 MiB 記憶體的目標 Spot 容量，其中要啟動的 Spot 執行個體數目必須提供總計 500,000 MiB 的記憶體。

```
{
 "InstanceTypes": [
 "m5.4xlarge",
 "r5.2xlarge",
 "m4.4xlarge"
],
 "TargetCapacity": 500000,
 "TargetCapacityUnitType": "memory-mib"
}
```

## 範例：為屬性型執行個體類型選取指定屬性

下列範例組態是針對屬性型執行個體類型選取設定的，其後是範例組態的文字說明。

```
{
 "TargetCapacity": 5000,
 "TargetCapacityUnitType": "vcpu",
 "InstanceRequirementsWithMetadata": {
 "ArchitectureTypes": ["arm64"],
 "VirtualizationTypes": ["hvm"],
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 1,
 "Max": 12
 },
 "MemoryMiB": {
 "Min": 512
 }
 }
 }
}
```

### InstanceRequirementsWithMetadata

若要使用屬性型執行個體類型選取，您必須在組態中包含 InstanceRequirementsWithMetadata 結構，並為 Spot 執行個體指定所需的屬性。

在上述範例中，指定了下列必要的執行個體屬性：

- ArchitectureTypes – 執行個體類型的架構類型必須是 arm64。
- VirtualizationTypes – 執行個體類型的虛擬化類型必須是 hvm。
- VCpuCount – 執行個體類型必須具有最少 1 個，最多 12 個的 vCPU。
- MemoryMiB – 執行個體類型必須具有最少 512 MiB 的記憶體。省略 Max 參數，表示沒有最大限制。

請注意，有幾個您可以指定的其他選用屬性。如需屬性清單，請參閱 Amazon EC2 命令列參考中的 [get-spot-placement-scores](#)。

### TargetCapacityUnitType



`TargetCapacityUnitType` 參數會指定目標容量的單位。在此範例中，目標容量為 5000，而目標容量單位類型為 `vcpu`，它們一起指定了所需目標容量 (5000 個 vCPU)，其中要啟動的 Spot 執行個體數目必須提供總計 5000 個的 vCPU。

範例：為屬性型執行個體類型選取指定屬性，並傳回可用區域的得分清單

下列範例組態是針對屬性型執行個體類型選取而設定的。透過指定 `"SingleAvailabilityZone": true`，回應將傳回得分可用區域的清單。

```
{
 "TargetCapacity": 1000,
 "TargetCapacityUnitType": "vcpu",
 "SingleAvailabilityZone": true,
 "InstanceRequirementsWithMetadata": {
 "ArchitectureTypes": ["arm64"],
 "VirtualizationTypes": ["hvm"],
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 1,
 "Max": 12
 },
 "MemoryMiB": {
 "Min": 512
 }
 }
 }
}
```

## Spot 執行個體資料摘要

為協助您了解 Spot 執行個體的費用，Amazon EC2 提供了資料摘要來說明您的 Spot 執行個體使用狀況和定價。當您訂閱資料摘要時，此資料摘要會傳送到您所指定的 Amazon S3 儲存貯體。

資料摘要檔案通常會每隔一小時送達您的儲存貯體，通常會在單一資料檔案中記載每個小時的使用狀況。這些檔案在傳送到您的儲存貯體之前，會先經過壓縮 (gzip)。如果有特定某小時的使用狀況，產生了大型檔案，則 Amazon EC2 可以寫入多個檔案 (例如，當該小時的檔案內容在壓縮前已超過 50 MB 時)。

**Note**

每個執行個體只能建立一個 Spot 執行個體資料饋送 AWS 帳戶。如果在某個小時內沒有執行 Spot 執行個體，您不會收到那個小時的資料摘要檔案。

除中國 (北京)、中國 (寧夏)、AWS GovCloud (美國) 和 [預設停用的 AWS 區域以外的所有區域](#) 均支援 Spot 執行個體資料饋送。

## 目錄

- [資料摘要檔案名稱與格式](#)
- [Amazon S3 儲存貯體需求](#)
- [訂閱 Spot 執行個體資料摘要](#)
- [描述 Spot 執行個體資料摘要](#)
- [檢視資料饋送中的資料](#)
- [刪除 Spot 執行個體資料摘要](#)

## 資料摘要檔案名稱與格式

Spot 執行個體資料摘要檔案名稱使用下列格式 (日期和小時採用 UTC 時間)：

```
bucket-name.s3.amazonaws.com/optional-prefix/aws-account-id.YYYY-MM-DD-HH.n.unique-id.gz
```

例如，如果儲存貯體的名稱為 **my-bucket-name**、前綴字首為 **my-prefix**，則檔案名稱會類似於下列名稱：

```
my-bucket-name.s3.amazonaws.com/my-prefix/111122223333.2023-12-09-07.001.b959dbc6.gz
```

如需儲存貯體名稱的詳細資訊，請參閱《Amazon S3 使用者指南》中的 [儲存貯體命名規則](#)。

Spot 執行個體資料摘要檔案以製表符 (tab) 分隔。資料檔案中的每一行皆對應至一個執行個體小時，而且包含下表中所列出的欄位。

| 欄位 | 描述 |
|----|----|
|----|----|

| 欄位          | 描述                                                                                                                            |
|-------------|-------------------------------------------------------------------------------------------------------------------------------|
| Timestamp   | 時間戳記，用來判定使用此執行個體的收費價格。                                                                                                        |
| UsageType   | 計費的使用類型和執行個體類型。對於 m1.small 競價型執行個體，此欄位設定為 SpotUsage。如果是其他所有的執行個體類型，此欄位會設定為 SpotUsage: {instance-type}。例如：SpotUsage:c1.medium。 |
| Operation   | 收費的產品。若是 Linux 競價型執行個體，此欄位會設定為 RunInstances。若是 Windows 競價型執行個體，此欄位會設定為 RunInstances:0002。Spot 的使用會根據可用區域分組。                   |
| InstanceID  | 產生此執行個體用量的 Spot 執行個體的 ID。                                                                                                     |
| MyBidID     | 產生此執行個體用量的 Spot 執行個體請求的 ID。                                                                                                   |
| MyMaxPrice  | 針對此 Spot 請求所指定的最高價格。                                                                                                          |
| MarketPrice | Timestamp 欄位中所指定時間點當時的 Spot 價格。                                                                                               |
| Charge      | 針對此執行個體用量的計費價格。                                                                                                               |
| Version     | 資料饋送版本。可用的版本為 1.0。                                                                                                            |

## Amazon S3 儲存貯體需求

訂閱資料摘要時，您必須指定用來存放資料摘要檔案的 Amazon S3 儲存貯體。

在選擇資料摘要的 Amazon S3 儲存貯體之前，請考量下列事項：

- 您必須具有儲存貯體的 FULL\_CONTROL 許可。如果您是儲存貯體擁有者，預設會具有此項權限。否則，值區擁有者必須授予您 AWS 帳戶此權限。
- 當您訂閱資料饋送時，系統會使用這些權限來更新值區 ACL，以授與資 AWS 料饋送帳戶 FULL\_CONTROL 權限。資 AWS 料饋送帳戶會將資料饋送檔案寫入值區。如果您的帳戶沒有所需

的權限，則資料摘要檔案無法寫入儲存貯體。如需詳細資訊，請參閱 [Amazon 日誌使用者指南中的傳送至 Amazon S3](#) 的 CloudWatch 日誌。

#### Note

如果您更新 ACL 並移除資 AWS 料饋送帳戶的權限，則無法將資料饋送檔案寫入值區。您必須重新訂閱資料摘要，以接收資料摘要檔案。

- 每個資料摘要檔案皆擁有自己的 ACL (和儲存貯體的 ACL 不同)。儲存貯體擁有者具有對資料檔案的 FULL\_CONTROL 權限。資 AWS 料饋送帳戶具有讀取和寫入權限。
- 如果您為儲存貯體套用已停用的 ACL，請新增儲存貯體政策，以便允許擁有完整控制權的使用者寫入儲存貯體。如需詳細資訊，請參閱 [檢閱和更新儲存貯體政策](#)。
- 如果您刪除資料饋送訂閱，Amazon EC2 不會移除儲存貯體或資料檔案上 AWS 資料饋送帳戶的讀取和寫入許可。您必須自行移除這些權限。
- 如果您使用伺服器端加密和存放在 AWS Key Management Service (SSE-KMS) 中的金鑰來加密 Amazon S3 儲存貯體，則必須使用客戶受管金 AWS KMS 鑰。如需詳細資訊，請參閱 [Amazon S3 儲存貯體伺服器端加密](#) 的 Amazon CloudWatch 日誌使用者指南。

#### Note

對於競價型執行個體資料饋送，產生 S3 檔案的資源不再是 Amazon CloudWatch 日誌。因此，您必須從 S3 儲存貯體許可政策和 KMS 政策移除 `aws:SourceArn` 區段。

## 訂閱 Spot 執行個體資料摘要

若要訂閱資料饋送，請使用 [create-spot-datafeed-subscription](#) 命令。

```
aws ec2 create-spot-datafeed-subscription \
 --bucket my-bucket-name \
 [--prefix my-prefix]
```

## 範例輸出

```
{
 "SpotDatafeedSubscription": {
 "OwnerId": "111122223333",
 "Bucket": "my-bucket-name",
```

```
 "Prefix": "my-prefix",
 "State": "Active"
 }
}
```

## 描述 Spot 執行個體資料摘要

若要描述您的資料饋送訂閱，請使用 [describe-spot-datafeed-subscription](#) 命令。

```
aws ec2 describe-spot-datafeed-subscription
```

## 範例輸出

```
{
 "SpotDatafeedSubscription": {
 "OwnerId": "123456789012",
 "Prefix": "spotdata",
 "Bucket": "my-s3-bucket",
 "State": "Active"
 }
}
```

## 檢視資料饋送中的資料

在中 AWS Management Console，開啟 AWS CloudShell。使用下列 [s3 sync](#) 命令，從資料饋送的 S3 儲存貯體取得 .gz 檔案，並將這些檔案存放於您指定的資料夾。

```
aws s3 sync s3://my-s3-bucket ./data-feed
```

若要顯示 .gz 檔案的內容，請變更至您存放 S3 儲存貯體內容的資料夾。

```
cd data-feed
```

使用 ls 命令檢視檔案的名稱。使用具有檔案名稱的 zcat 命令，顯示壓縮檔案的內容。下列是範例命令。

```
zcat 111122223333.2023-12-09-07.001.b959dbc6.gz
```

下列為範例輸出。

```
#Version: 1.0
#Fields: Timestamp UsageType Operation InstanceID MyBidID MyMaxPrice MarketPrice Charge
Version
2023-12-09 07:13:47 UTC USE2-SpotUsage:c7a.medium RunInstances:SV050
i-0c3e0c0b046e050df sir-pwq6nmfp 0.0510000000 USD 0.0142000000 USD
0.0142000000 USD 1
```

## 刪除 Spot 執行個體資料摘要

若要刪除資料饋送，請使用 [delete-spot-datafeed-subscription](#) 命令。

```
aws ec2 delete-spot-datafeed-subscription
```

## Spot 執行個體配額

每個區域每個 AWS 帳戶的執行中 Spot 執行個體請求和待定 Spot 執行個體請求有數量配額。待定 Spot 執行個體請求完成後，請求就不會再計入配額中，因為執行中的執行個體會計入配額中。

Spot 執行個體配額是根據執行中的 Spot 執行個體正在使用或將使用的虛擬中央處理單元 (vCPU) 的數量來管理的，直到完成開放的 Spot 執行個體請求。如果您終止 Spot 執行個體但未取消 Spot 執行個體請求，請求會計入您的 Spot 執行個體 vCPU 配額，直到 Amazon EC2 偵測到 Spot 執行個體終止並關閉請求。

我們提供以下 Spot 執行個體配額類型：

- 所有 DL Spot 執行個體請求
- 所有 F Spot 執行個體請求
- 所有 G 和 VT Spot 執行個體請求
- 所有 Inf Spot 執行個體請求
- 所有 P Spot 執行個體請求
- 所有標準 (A, C, D, H, I, M, R, T, Z) Spot 執行個體請求
- 所有 Trn Spot 執行個體請求
- 所有 X Spot 執行個體請求

每個配額類型可指定一個或多個執行個體系列的 vCPU 數量上限。如需不同執行個體系列、產生和大小的相關資訊，請參閱 [Amazon EC2 執行個體類型](#)。

您可以啟動任意執行個體類型的組合，這些類型組合可滿足不斷變化的應用程式需求。例如，在 256 個 vCPU 的標準 Spot 執行個體請求配額下，您可以啟動 32 個 m5.2xlarge Spot 執行個體 (32 x 8 個 vCPU) 或 16 個 c5.4xlarge Spot 執行個體 (16 x 16 個 vCPU)。

## 任務

- [監控 Spot 執行個體配額和用量](#)
- [請求提高配額](#)

## 監控 Spot 執行個體配額和用量

您可以使用下列方式檢視和管理您的 Spot 執行個體配額：

- Service Quotas 主控台中的 Amazon EC2 [服務配額頁面](#)
- [取得服務配額 AWS CLI](#)

如需詳細資訊，請參閱 [Service Quotas 使用指南中的 Amazon EC2 服務配額和檢視服務配額](#)。

透過 Amazon CloudWatch 指標整合，您可以根據自己的配額監控 EC2 使用情況。您也可以設定警示，在接近配額時提出警告。如需詳細資訊，請參閱 [Service Quotas 使用指南中的 Service Quotas 和 Amazon CloudWatch 警示](#)，以。

## 請求提高配額

即使 Amazon EC2 根據使用量自動提高 Spot 執行個體配額，您也可以在必要時要求提高配額。例如，如果您打算啟動的 Spot 執行個體超過目前配額允許的範圍，您可以請求提高配額。如果您送出 Spot 執行個體請求並收到錯誤訊息 Max spot instance count exceeded，您可以要求增加請求的配額。若要請求增加配額，可以使用 [Amazon EC2 服務配額](#) 中所述的 Service Quotas 主控台。

## 爆量效能執行個體

T 執行個體類型是 [爆量效能執行個體](#)。如果您使用爆量效能執行個體啟動 Spot 執行個體，而且您計劃立即使用爆量效能 Spot 執行個體，且在短時間內沒有閒置時間來累積 CPU 額度，建議您以 [標準模式](#) 將其啟用，以避免支付更高的成本。如果您在 [無限制模式](#) 下啟動高載效能 競價型執行個體，而且 CPU 立即高載，則您將耗費高載的剩餘額度。如果您短暫使用執行個體，則您的執行個體沒有時間產生 CPU 額度來償還剩餘額度，而且當終止執行個體時，會針對剩餘額度向您收費。

只有在執行個體執行夠長時間，可為爆量產生 CPU 額度時，無限制模式才適合用於高載效能 競價型執行個體。否則，支付剩餘額度會使爆量效能 競價型執行個體比使用其他執行個體更為昂貴。如需詳細資訊，請參閱 [使用無限制模式與固定 CPU 的時機](#)。

T2 執行個體在**標準模式**中設定時，可取得**啟動額度**。T2 執行個體是唯一可獲得啟動額度的爆量效能執行個體。啟動額度旨在透過提供足夠的運算資源來設定執行個體，創造 T2 執行個體具有生產力的初始啟動體驗。不允許重複啟動 T2 執行個體來獲得新的啟動額度。如果您需要持續的 CPU，可以賺取額度 (藉由在某些期間閒置)、為 T2 競價型執行個體使用**無限制模式**，或將執行個體類型與專屬 CPU 搭配使用。

## Dedicated Hosts

Amazon EC2 專用主機是完全專供您使用的實體伺服器。您可以選擇與其他 AWS 帳戶共用執行個體容量。如需詳細資訊，請參閱[使用共用的 專用執行個體](#)。

專用主機提供執行個體放置的可見度和控制權，並支援主機相似性。這表示您可以在特定主機上啟動和執行執行個體，而且可以確保執行個體僅在特定主機上執行。如需詳細資訊，請參閱[了解自動配置和親和性](#)。

專用主機提供全面的自攜授權 (BYOL) 支援。根據您的授權條款，它們可讓您使用現有的每個插槽、每個核心或虛擬機器的軟體授權，包括 Windows Server、SQL Server、SUSE Linux 企業伺服器、RHEL 或其他繫結至虛擬機器、通訊端或實體核心的軟體授權。

如果您要求執行個體在專用硬體上執行，但不需要掌握或控制執行個體放置，而且不需要使用每個插槽或每核心的軟體授權，則可以考慮改用專用執行個體。專用執行個體和專用主機都可用於在專用實體伺服器上啟動 Amazon EC2 執行個體。專用執行個體和專用執行個體上的執行個體之間沒有效能、安全或實體差異。但是，它們之間存在一些關鍵差異。下表特別列出了專用執行個體與專用執行個體之間的主要差異：

|                   | Dedicated Host            | Dedicated Instance |
|-------------------|---------------------------|--------------------|
| 專屬實體伺服器           | 實體伺服器具有完全專供您使用的執行個體容量。    | 專用於單一客戶帳戶的實體伺服器。   |
| 實例容量共享            | 可與其他帳戶共用執行個體容量。           | 不支援                |
| 計費                | 按每台主機計費                   | 按每個執行個體計費          |
| 通訊端、核心和主機 ID 的可見度 | 提供通訊端與實體核心數目的可見度          | 沒有可見度              |
| 主機和執行個體親和性        | 可讓您隨時間一致地將執行個體部署到相同的實體伺服器 | 不支援                |



|                       | Dedicated Host                       | Dedicated Instance |
|-----------------------|--------------------------------------|--------------------|
| 訂定執行個體配置目標            | 對於執行個體在實體伺服器上的配置方式，提供額外可見性和控制權       | 不支援                |
| 自動執行個體復原功能            | 支援。如需詳細資訊，請參閱 <a href="#">主機復原</a> 。 | 支援                 |
| 使用自有授權 (BYOL)         | 支援                                   | 部分支援 *             |
| Capacity Reservations | 不支援                                  | 支援                 |

\* 透過軟體保證而實現的授權行動性的 Microsoft SQL Server 和 Windows 虛擬桌面存取 (VDA) 授權可以與專用預留執行個體一起使用。

如需專用預留執行個體的詳細資訊，請參閱 [專用執行個體](#)。

## 目錄

- [執行個體容量組態](#)
- [使用自有授權](#)
- [定價和計費](#)
- [專用執行個體上的爆量效能 T3 執行個體](#)
- [專用執行個體 限制](#)
- [使用 專用執行個體](#)
- [使用共用的 專用執行個體](#)
- [專用主機 AWS Outposts](#)
- [主機復原](#)
- [主機維護](#)
- [追蹤組態變更](#)

## 執行個體容量組態

專用執行個體支援不同組態 (實體核心、通訊端和 VCPU)，能讓您執行不同系列和大小的執行個體。

在帳戶中配置專用執行個體時，您可以選擇能支援單一執行個體類型的組態，或相同執行個體系列中多種執行個體類型的組態。您可在主機上執行的執行個體數量將依您選擇的組態而定。

## 目錄

- [單一執行個體類型支援](#)
- [多種執行個體類型支援](#)

### 單一執行個體類型支援

您可以配置僅支援一種執行個體類型的專用執行個體。若使用此組態，您在專用執行個體上啟動的每個執行個體，其執行個體類型皆必須與配置主機時指定的執行個體類型相同。

例如，您可以配置僅支援 m5.4xlarge 執行個體類型的主機。在此情況下，您僅能執行該主機上的 m5.4xlarge 執行個體。

您可在主機上啟動的執行個體數量，將依該主機提供的實體核心數量以及指定執行個體類型使用的核心數目而定。例如，如果您為 m5.4xlarge 執行個體配置主機，且該主機提供 48 個實體核心，則每個 m5.4xlarge 執行個體會消耗 8 個實體核心。這表示您最多可在該主機上啟動 6 個執行個體 (48 個實體核心 / 每個執行個體 8 個核心 = 6 個執行個體)。

### 多種執行個體類型支援

您可以配置能支援相同執行個體系列中多種執行個體類型的專用執行個體。此舉可讓您在相同主機上執行不同類型的執行個體，只要這些執行個體位於相同執行個體系列中，且主機具有足夠的執行個體容量即可。

例如，您可以配置能支援 R5 執行個體系列中不同執行個體類型的主機。在此情況下，您可以在該主機上啟動任意 R5 執行個體類型的組合，例如 r5.large、r5.xlarge、r5.2xlarge 以及 r5.4xlarge，實際情況依主機的實體核心容量而定。

下列執行個體系列可支援具有多種執行個體類型支援的專用執行個體：

- 一般用途：A1、M5、M5n、M6i 和 T3
- 運算優化：C5、C5n 和 C6i
- 記憶體優化：R5、R5n 和 R6i

您可在主機上執行的執行個體數量，將依該主機提供的實體核心數量以及在該主機上執行的每種執行個體類型所消耗的核心數目而定。例如，如果配置一個會提供 48 個實體核心的 R5 主機，在執行兩個 r5.2xlarge 執行個體 (4 個核心 x 2 個執行個體) 以及三個 r5.4xlarge 執行個體 (8 個核心 x 3

個執行個體) 時，這些執行個體總共會消耗 32 個核心，而且只要不超過剩餘的 16 個核心，您就可執行任意組合的 R5 執行個體。

但是，對於每個執行個體系列，每種執行個體大小都有限制可執行的執行個體數目。例如，R5 專用執行個體最多可支援 2 個 r5.8xlarge 執行個體，使用 32 個實體核心。在此情況下，可使用其他較小的 R5 執行個體來將主機填滿至核心容量。如需了解每個執行個體系列支援的執行個體大小數量，請參閱[專用執行個體組態表格](#)。

下表所示為執行個體類型組合範例：

| 執行個體系列 | 範例執行個體大小組合                                                                                                                                                                                |
|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| R5     | <ul style="list-style-type: none"> <li>範例 1 : 4 x r5.4xlarge + 4 x r5.2xlarge</li> <li>範例 2 : 1 x r5.12xlarge + 1 x r5.4xlarge + 1 x r5.2xlarge + 5 x r5.xlarge + 2 x r5.large</li> </ul> |
| C5     | <ul style="list-style-type: none"> <li>範例 1 : 1 x c5.9xlarge + 2 x c5.4xlarge + 1 x c5.xlarge</li> <li>範例 2 : 4 x c5.4xlarge + 1 x c5.xlarge + 2 x c5.large</li> </ul>                    |
| M5     | <ul style="list-style-type: none"> <li>範例 1 : 4 x m5.4xlarge + 4 x m5.2xlarge</li> <li>範例 2 : 1 x m5.12xlarge + 1 x m5.4xlarge + 1 x m5.2xlarge + 5 x m5.xlarge + 2 x m5.large</li> </ul> |

## 考量事項

在使用支援多種執行個體類型的專用執行個體時，請注意下列各項：

- 若為 N 類型專用執行個體 (例如 C5n、M5n 和 R5n)，您無法以組合方式使用更小型的執行個體 (2xlarge 和更小型) 及更大型的執行個體 (4xlarge 和更大型，包括 metal)。若您要在 N 類型專用執行個體上同時用更小型和更大型執行個體，則必須為更小型和更大型的執行個體配置不同主機。

- 建議您先啟動較大類型的執行個體，然後視需要以較小類型的執行個體填滿剩餘的執行個體容量。

## 使用自有授權

專用執行個體可讓您使用現有的每個通訊端、每個核心或每個 VM 軟體授權。當您使用自有授權時，需負責管理自己的授權。不過，Amazon EC2 具有的功能可協助您保有授權合規，例如執行個體親和性和鎖定的放置。

若要在 Amazon EC2 中使用自有的磁碟區授權機器映像，需遵循以下一般步驟。

1. 確定授權條款 (控管機器映像的使用辦法) 允許您在虛擬化雲端環境中使用機器映像。如需 Microsoft 授權的詳細資訊，請參閱 [Amazon Web Services](#) 和 [Microsoft 授權](#)。
2. 確定您可以在 Amazon EC2 中使用機器映像後，請使用 VM Import/Export 將其匯入。如需如何匯入機器映像的資訊，請參閱 [VM Import/Export 使用者指南](#)。
3. 匯入機器映像之後，您就可以透過該映像在中專執行個體上啟動執行個體。
4. 依據作業系統而定，當您執行這些執行個體時，您可能必須為自己的 KMS 伺服器 (例如 Windows Server 或 Windows SQL Server) 啟用執行個體。您無法為 Amazon Windows KMS 伺服器啟用匯入的 Windows AMI。

### Note

若要追蹤影像的使用方式 AWS，請在中啟用主持人錄製 AWS Config。您可以使用 AWS Config 將組態變更記錄到專用主機，並使用輸出作為授權報告的資料來源。如需詳細資訊，請參閱 [追蹤組態變更](#)。

## 定價和計費

專用主機 定價會根據付款選項有所差異。

### 付款選項

- [隨需專用執行個體](#)
- [Dedicated Host Reservations](#)
- [Savings Plans](#)
- [在專用執行個體的 Windows Server 定價](#)

## 隨需專用執行個體

當您將專用執行個體配置到帳戶時，會自動啟用隨需計費。

專用執行個體的隨需價格依執行個體系列和區域而異。不論您選擇在專用執行個體中啟動的執行個體數量或大小為何，都會以每秒 (最低 60 秒) 費率計費。如需隨需定價的詳細資訊，請參閱 [Amazon EC2 專用執行個體 隨需定價](#)。

您可以隨時釋出隨需專用執行個體以停止累積計費。如需如何釋出專用執行個體的資訊，請參閱 [發行版本 專用執行個體](#)。

## Dedicated Host Reservations

專用執行個體預留提供相當於執行隨需專用執行個體的計費折扣。保留有三種付款選項可用：

- 無預付— 無預付保留可提供您一段期間的專用執行個體用量折扣，且不需要預付款。提供一年期和三年期。只有部分執行個體系列支援三年期無預付保留。
- 部分預付— 必須預先支付其中一部分的保留，期間內剩餘的小時數則會以折扣後的費率計費。提供一年期和三年期。
- 全額預付— 提供最低的有效價位。提供一年期和三年期，並預付整期的成本，日後不需另外付費。

您的帳戶中必須具備作用中專用執行個體，才能購買保留。每個保留可以涵蓋單一可用區域中支援相同執行個體系列的一或多部主機。保留會套用至主機上的執行個體系列，而不是執行個體大小。若您有三個具有不同大小執行個體的專用執行個體 (m4.xlarge、m4.medium 和 m4.large)，您可以將單一 m4 保留與所有那些專用執行個體建立關聯。保留的執行個體系列和可用區域必須符合您想與其建立關聯之專用執行個體的執行個體系列和可用區域。

當保留與專用執行個體建立關聯時，直到保留的期間結束前都不能釋出專用執行個體。

如需保留定價的詳細資訊，請參閱 [Amazon EC2 專用執行個體定價](#)。

## Savings Plans

Savings Plans 是個彈性定價模式，提供 隨需執行個體 大幅節省成本。透過 Savings Plans，您可以對一致的使用量設定合約，每小時以美金計價，綁約一年或三年。這可提供您彈性使用 專用執行個體 以達到更佳的需求，且持續的節省金費，而不是與特定的 專用執行個體 綁約。如需詳細資訊，請參閱 [AWS Savings Plans 使用者指南](#)。

**Note**

Savings Plans 不支援

u-6tb1.metal、u-9tb1.metal、u-12tb1.metal、u-18tb1.metal 及 u-24tb1.metal 專用執行個體。

## 在專用執行個體的 Windows Server 定價

根據 Microsoft 授權條款，您可以將現有的 Windows Server 和 SQL Server 授權帶到 專用執行個體。如果您選取自帶授權，則軟體使用不收取額外費用。

此外，您也可以使用 Amazon 提供的 Windows Server AMI，在 專用執行個體 上執行最新版本的 Windows Server。對於現有 SQL Server 授權有資格執行 專用執行個體，但需要 Windows Server 來執行 SQL Server 工作負載時，這種情況很常見。Amazon 提供的 Windows 伺服器 AMI 僅支援目前一代的執行個體類型。如需詳細資訊，請參閱 [Amazon EC2 專用執行個體定價](#)。

## 專用執行個體上的爆量效能 T3 執行個體

專用執行個體支援爆量效能 T3 執行個體。T3 執行個體可讓您在專用硬體上使用合格的 BYOL 授權軟體，可說是符合成本效益的方式。T3 執行個體的 vCPU 佔用空間較小，合併工作負載時所需的主機較少，而且每個核心授權都可以發揮最大的使用率。

T3 專用執行個體最適合執行 BYOL 軟體，而 CPU 使用率為低度至中等。這包括符合資格的依通訊端數、依核心數或依虛擬機器數的軟體授權，例如 Windows Server、Windows 桌面、SQL Server、SUSE Enterprise Linux Server、Red Hat Enterprise Linux 及 Oracle 資料庫。例如，適用於 T3 專用執行個體的工作負載包括中小型資料庫、虛擬桌面、開發及測試環境、程式碼儲存庫和產品原型等。T3 專用執行個體不建議用於 CPU 持續使用率高的工作負載，或同時遇到相關 CPU 高載問題的工作負載。

專用執行個體上的 T3 執行個體使用的額度模型是與共用租用硬體上的 T3 執行個體相同。但是，其僅支援 standard 額度模型；並不支援 unlimited 額度模型。在 standard 模式中，專用執行個體上的 T3 執行個體使用與共用租用硬體上爆量效能執行個體相同的方式獲得、花費以及累積額度。它們提供基準 CPU 效能，並可大幅超越基準水準。若要讓效能突增至超過基準，執行個體會用掉在 CPU 額度餘額中所累積的額度。當累積的額度耗盡時，CPU 使用率會降低至基準水準。如需 standard 模式的詳細資訊，請參閱 [標準爆量效能執行個體如何運作](#)。

T3 專用執行個體支援 Amazon EC2 專用執行個體提供的所有功能，包括在單一主機、主機資源群組和 BYOL 上的多種執行個體規模。

## 支援的 T3 執行個體大小和組態

T3 專用執行個體執行一般用途高載 T3 執行個體，透過提供基準 CPU 效能及在需要時高載執行更高的效能，來分享主機的 CPU 資源。如此一來，擁有 48 個核心的 T3 專用執行個體，每台主機最多可支援 192 個執行個體。為了有效利用主機的資源並提供最佳執行個體效能，Amazon EC2 執行個體置放演算法會自動計算可支援主機上啟動的執行個體數量和執行個體大小這兩者的組合。

T3 專用執行個體支援同一主機上的多個執行個體類型。所有 T3 執行個體大小都受專用執行個體的支援。您可執行各種不同的 T3 執行個體組合，只要不超出主機的 CPU 上限即可。

下表列出支援的執行個體類型、摘要說明每個執行個體類型的效能，並指明可啟動之每種大小的執行個體數目上限。

| 執行個體類型     | vCPU | 記憶體 (GiB) | 每個 vCPU 的基準 CPU 使用率 | 網路高載頻寬 (Gbps) | Amazon EBS 高載頻寬 (Mbps) | 每個專用執行個體的執行個體數目上限 |
|------------|------|-----------|---------------------|---------------|------------------------|-------------------|
| t3.nano    | 2    | 0.5       | 5%                  | 5             | 最高 2,085               | 192               |
| t3.micro   | 2    | 1         | 10%                 | 5             | 最高 2,085               | 192               |
| t3.small   | 2    | 2         | 20%                 | 5             | 最高 2,085               | 192               |
| t3.medium  | 2    | 4         | 20%                 | 5             | 最高 2,085               | 192               |
| t3.large   | 2    | 8         | 30%                 | 5             | 2,780                  | 96                |
| t3.xlarge  | 4    | 16        | 40%                 | 5             | 2,780                  | 48                |
| t3.2xlarge | 8    | 32        | 40%                 | 5             | 2,780                  | 24                |

## 監控 T3 專用執行個體的 CPU 使用率

您可以使用 `DedicatedHostCPUUtilization` Amazon CloudWatch 指標監控專用主機的 vCPU 使用率。EC2 命名空間和 `Per-Host-Metrics` 維度均可使用此指標。如需詳細資訊，請參閱 [專用執行個體指標](#)。

## 專用執行個體 限制

在您配置專用執行個體之前，請記下以下的法規與限制：

- 若要在 專用執行個體 上執行 RHEL、SUSE Linux 和 SQL Server，您必須使用自有的 AMI。由提供 AWS 或可用的 RHEL、SUSE Linux 和 SQL 伺服器 AMI AWS Marketplace 無法與專用主機搭配使用。如需如何建立自己的 AMI 的詳細資訊，請參閱 [使用自有授權](#)。

此限制不適用於配置給記憶體密集型執行個體

(`u-6tb1.metal`、`u-9tb1.metal`、`u-12tb1.metal`、`u-18tb1.metal` 和 `u-24tb1.metal`) 的主機。由提供 AWS 或提供的 RHEL 和 SUSE Linux AMI AWS Marketplace 可用於這些主機。

- 每個區域每個 AWS 帳戶每個執行個體系列的執行中專用執行個體有數量限制。配額僅適用於執行中的執行個體。如果執行個體處於待定、停止中或已停止狀態，則不會計入您的配額。若要檢視帳戶配額，或請求提升配額，可使用 [Service Quotas 主控台](#)。
- 執行於專用執行個體上的執行個體僅可在 VPC 中啟動。
- 僅在使用指定主機資源群組的啟動範本時，才支援 Auto Scaling 群組。如需詳細資訊，請參閱《Amazon EC2 Auto Scaling 使用者指南》中的 [使用進階設定建立啟動範本](#)。
- 不支援 Amazon RDS 執行個體。
- AWS 免費用量層不適用於專用主機。
- 執行個體的配置控制是指針對專用執行個體中所啟動的執行個體進行管理。您無法在置放群組中啟動專用執行個體。
- 如果您為虛擬執行個體類型分配主機，則在分配主機之後，無法將執行個體類型修改為 `.metal` 執行個體類型。例如，如果您為 `m5.large` 執行個體類型分配主機，則無法將執行個體類型修改為 `m5.metal`。

同樣，如果您為 `.metal` 執行個體類型分配主機，則在分配主機之後，無法將執行個體類型修改為虛擬執行個體類型。例如，如果您為 `m5.metal` 執行個體類型分配主機，則無法將執行個體類型修改為 `m5.large`。

## 使用 專用執行個體

若要使用專用執行個體，您必須先配置帳戶中要使用的主機。接著，您可以指定執行個體的「主機」租用，以啟動主機上的執行個體。您必須為執行個體選取要在哪個特定主機上啟動，或者，您可以允許執



行個體在已啟用自動配置且符合其執行個體類型的任何主機上啟動。停止並重新啟動執行個體時，「主機親和性」設定可決定要在相同或不同的主機上重新啟動。

如果您不再需要隨需主機，可以停止在主機上執行的執行個體，並指示它們在另一個主機上啟動，接著再釋出主機。

專用主機也與 AWS License Manager. 您可以使用 License Manager 建立主機資源群組，這是做為單一實體管理的專用執行個體集合。建立主機資源群組時，您可以指定專用執行個體的主機管理偏好設定，例如自動配置和自動發佈。這讓您在無需手動配置和管理這些主機的情況下，在專用執行個體中啟動執行個體。如需詳細資訊，請參閱《AWS License Manager 使用者指南》中的 [主機資源群組](#)。

## 內容

- [配置 專用執行個體](#)
- [在 專用執行個體 中啟動執行個體](#)
- [將執行個體啟動至主機資源群組](#)
- [了解自動配置和親和性](#)
- [修改 專用執行個體 自動配置](#)
- [修改支援的執行個體類型](#)
- [修改執行個體租用和親和性](#)
- [檢視 專用執行個體](#)
- [標記 專用執行個體](#)
- [監控 專用執行個體](#)
- [發行版本 專用執行個體](#)
- [購買 專用執行個體預留](#)
- [檢視 專用執行個體 保留](#)
- [標記 專用執行個體預留](#)

## 配置 專用執行個體

若要開始使用 專用執行個體，您可以使用 Amazon EC2 主控台或命令列工具將 專用執行個體 配置到帳戶中。配置 專用執行個體 之後，會立即將 專用執行個體 容量提供給您的帳戶，而且您可以開始在 專用執行個體 上啟動執行個體。

在帳戶中配置專用執行個體時，您可以選擇能支援單一執行個體類型的組態，或相同執行個體系列中多種執行個體類型的組態。您可在主機上執行的執行個體數量將依您選擇的組態而定。如需更多資訊，請參閱[執行個體容量組態](#)。

## Console

### 配置專用執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Dedicated Hosts (專用執行個體)，然後選擇 Allocate 專用執行個體 (配置專用執行個體)。
3. 對於 Instance family (執行個體系列)，選擇適用於 專用執行個體 的執行個體系列。
4. 指定 專用執行個體 是否支援所選執行個體系列內的多種執行個體容量，或是僅支援特定執行個體類型。執行下列操作。
  - 若要將 專用執行個體 設定為支援所選執行個體系列內的多個執行類型，請針對 Support multiple instance types (支援多個執行個體類型) 選取 Enable (啟用)。啟用此功能，您就能在 專用執行個體 上啟動來自相同執行個體系列的不同執行個體容量。例如，如果您選擇 m5 執行個體系列且選擇了此選項，您就能在 專用執行個體 上啟動 m5.xlarge 及 m5.4xlarge。
  - 若要將 專用執行個體 設定為支援所選執行個體系列內的單一執行特定類型，請清除 Support multiple instance types (支援多個執行個體類型)，然後針對 Instance type (執行個體類型) 選取要支援的執行個體類型。這可讓您在 專用執行個體 上啟動單一執行個體類型。例如，如果您選擇此選項並指定 m5.4xlarge 作為支援的執行個體類型，則您僅能在 專用執行個體 上啟動 m5.4xlarge。
5. 對於 Availability Zone (可用區域)，選擇要配置 專用執行個體 的可用區域。
6. 若要允許 專用執行個體 接受符合其執行個體類型的未鎖定執行個體啟動，對於 Instance auto-placement (執行個體自動放置)，請選擇 Enable (啟用)。如需自動配置的詳細資訊，請參閱 [了解自動配置和親和性](#)。
7. 若要啟用 專用執行個體 的主機復原，對於 Host recovery (主機復原)，請選擇 Enable (啟用)。如需詳細資訊，請參閱 [主機復原](#)。
8. 對於 Quantity (數量)，輸入要配置的 專用執行個體 數量。
9. (選用) 選擇新增標籤，然後輸入標籤金鑰和標籤值。
10. 選擇 Allocate (配置)。

## AWS CLI

### 配置專用執行個體

使用[分配主機命令](#) AWS CLI。下列命令會配置一個專用執行個體，其可支援 us-east-1a 可用區域中來自 m5 執行個體系列的多個執行個體類型。主機也已啟用主機復原功能，而且已停用自動配置功能。

```
aws ec2 allocate-hosts --instance-family "m5" --availability-zone "us-east-1a" --auto-placement "off" --host-recovery "on" --quantity 1
```

以下命令會配置一個專用執行個體，其可支援在 eu-west-1a 可用區域中啟動之未訂定目標的 m4.large 執行個體、啟用主機復原，並套用含 purpose 金鑰和 production 值的標籤。

```
aws ec2 allocate-hosts --instance-type "m4.large" --availability-zone "eu-west-1a" --auto-placement "on" --host-recovery "on" --quantity 1 --tag-specifications 'ResourceType=dedicated-host,Tags=[{Key=purpose,Value=production}]'
```

## PowerShell

### 配置專用執行個體

使用指[New-EC2Host](#) AWS Tools for Windows PowerShell 令。下列命令會配置一個專用執行個體，其可支援 us-east-1a 可用區域中來自 m5 執行個體系列的多個執行個體類型。主機也已啟用主機復原功能，而且已停用自動配置功能。

```
PS C:\> New-EC2Host -InstanceFamily m5 -AvailabilityZone us-east-1a -AutoPlacement Off -HostRecovery On -Quantity 1
```

以下命令會配置一個專用執行個體，其可支援在 eu-west-1a 可用區域中啟動之未訂定目標的 m4.large 執行個體、啟用主機復原，並套用含 purpose 金鑰和 production 值的標籤。

TagSpecification 參數 (在建立專用執行個體時用來對其套用標籤) 需要使用物件來指定要套用標籤的資源類型、標籤鍵和標籤值。下列命令會建立所需的物件。

```
PS C:\> $tag = @{ Key="purpose"; Value="production" }
PS C:\> $tagspec = new-object Amazon.EC2.Model.TagSpecification
PS C:\> $tagspec.ResourceType = "dedicated-host"
PS C:\> $tagspec.Tags.Add($tag)
```

以下命令會配置專用執行個體，並套用 \$tagspec 物件指定的標籤。

```
PS C:\> New-EC2Host -InstanceType m4.large -AvailabilityZone eu-west-1a -AutoPlacement On -HostRecovery On -Quantity 1 -TagSpecification $tagspec
```

## 在專用執行個體中啟動執行個體

配置好專用執行個體之後，您就可以在其中啟動執行個體。如果您不具備作用中的專用執行個體，且其可用容量不足以啟動您的執行個體類型，您就無法啟動含 host 租用的執行個體。

### Tip

對於支援多種執行個體規模的專用執行個體，建議您先啟動較大的執行個體，然後視需要以較小的執行個體填滿剩餘的執行個體容量。

在您啟動執行個體之前，請記下以下的限制。如需詳細資訊，請參閱 [專用執行個體限制](#)。

您可以使用下列方法，將執行個體啟動至專用執行個體。

### Console

在專用執行個體頁面於特定專用執行個體啟動執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 專用執行個體。
3. 在 Dedicated Hosts (專用執行個體) 頁面上，選取一個主機，然後選取 Actions (動作)、Launch Instance(s) onto Host (在主機啟動執行個體)。
4. 在 Application and OS Images (應用程式和 OS 映像) 區段中，從清單中選取 AMI。

### Note

由 Amazon EC2 提供的 Windows、SUSE 和 RHEL AMI 無法搭配專用執行個體使用。

5. 在 Instance type (執行個體類型) 區段中，選取要啟動的執行個體類型。


### Note

如果專用執行個體僅支援單一執行個體類型，則預設會選取支援的執行個體類型，而且不可變更。

如果專用執行個體支援多個執行個體類型，您必須根據專用執行個體的可用執行個體容量，選取支援執行個體系列內的執行個體類型。建議您先啟動較大的執行個體，然後視需要以較小的執行個體填滿剩餘的執行個體容量。

6. 在 Key pair (金鑰對) 區段中，選取要與執行個體建立關聯的金鑰對。
7. 在 Advanced details (進階詳細資訊) 區段，在 Tenancy affinity (租用親和性) 中，執行下列其中一個操作：
  - 選取 Off (關閉) – 在指定主機上啟動的執行個體，不保證停止後會在相同的專用執行個體上重新啟動。
  - 選取專用執行個體 ID – 執行個體在停止後始終會在此特定主機上重新啟動。

如需親和性的詳細資訊，請參閱[了解自動配置和親和性](#)。


 Note

Tenancy (租用) 和 Host (主機) 選項會依據您選取的主機預先設定。

8. 視需要設定其餘的執行個體選項。如需詳細資訊，請參閱[使用定義的參數啟動執行個體](#)。
9. 選擇啟動執行個體。

使用啟動執行個體精靈，在專用執行個體中啟動執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)、Launch Instance (啟動執行個體)。
3. 在 Application and OS Images (應用程式和 OS 映像) 區段中，從清單中選取 AMI。

 Note

由 Amazon EC2 提供的 Windows、SUSE 和 RHEL AMI 無法搭配專用執行個體使用。

4. 在 Instance type (執行個體類型) 區段中，選取要啟動的執行個體類型。
5. 在 Key pair (金鑰對) 區段中，選取要與執行個體建立關聯的金鑰對。
6. 在 Advanced details (進階詳細資料) 區段中，執行下列操作：
  - a. 對於 Tenancy (租用)，選取 Dedicated Host (專用執行個體)。
  - b. 對於 Target host by (目標主機依據) 中，選取 Host ID (主機 ID)。
  - c. 對於 Target host ID (目標主機 ID)，選取要在其上啟動執行個體的主機。
  - d. 對於 Tenancy affinity (租用親和性)，執行下列其中一個操作：

- 選取 Off (關閉) – 在指定主機上啟動的執行個體，不保證停止後會在相同的專用執行個體上重新啟動。
- 選取專用執行個體 ID – 執行個體在停止後始終會在此特定主機上重新啟動。

如需親和性的詳細資訊，請參閱[了解自動配置和親和性](#)。

7. 視需要設定其餘的執行個體選項。如需詳細資訊，請參閱[使用定義的參數啟動執行個體](#)。
8. 選擇啟動執行個體。

## AWS CLI

將執行個體啟動至專用執行個體

使用[執行個體](#) AWS CLI 命令，並在 Placement 要求參數中指定執行個體相似性、租用和主機。

## PowerShell

將執行個體啟動至專用執行個體

使用命[New-EC2Instance](#) AWS Tools for Windows PowerShell 令並在 Placement request 參數中指定執行個體相似性、租用和主機。

## 將執行個體啟動至主機資源群組

當您將執行個體啟動至具有可用執行個體容量之專用執行個體的主機資源群組時，Amazon EC2 會將執行個體啟動到該主機上。如果主機資源群組沒有具有可用執行個體容量的主機，則 Amazon EC2 會自動在主機資源群組中配置新主機，然後將執行個體啟動到該主機上。如需詳細資訊，請參閱《AWS License Manager 使用者指南》中的[主機資源群組](#)。

## 需求與限制

- 您必須將以核心或通訊端為基礎的授權組態與 AMI 建立關聯。
- 您無法將 Amazon EC2 提供的 SQL Server、SUSE 或 RHEL AMI 與專用執行個體搭配使用。
- 您無法透過選擇主機 ID 鎖定特定主機為目標，而且在將執行個體啟動至主機資源群組時，也無法啟用執行個體親和性。

您可以使用下列方法，將執行個體啟動至主機資源群組。

## Console

### 將執行個體啟動至主機資源群組

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)、Launch Instance (啟動執行個體)。
3. 在 Application and OS Images (應用程式和 OS 映像) 區段中，從清單中選取 AMI。

#### Note

由 Amazon EC2 提供的 Windows、SUSE 和 RHEL AMI 無法搭配專用執行個體使用。

4. 在 Instance type (執行個體類型) 區段中，選取要啟動的執行個體類型。
5. 在 Key pair (金鑰對) 區段中，選取要與執行個體建立關聯的金鑰對。
6. 在 Advanced details (進階詳細資料) 區段中，執行下列操作：
  - a. 對於 Tenancy (租用)，選取 Dedicated Host (專用執行個體)。
  - b. 對於 Target host by (目標主機依據)，選取 Host resource group (主機資源群組)。
  - c. 對於 Tenancy host resource group (租用主機資源群組)，選取要在其中啟動執行個體的主機資源群組。
  - d. 對於 Tenancy affinity (租用親和性)，執行下列其中一個操作：
    - 選取 Off (關閉) – 在指定主機上啟動的執行個體，不保證停止後會在相同的專用執行個體上重新啟動。
    - 選取專用執行個體 ID – 執行個體在停止後始終會在此特定主機上重新啟動。

如需親和性的詳細資訊，請參閱 [了解自動配置和親和性](#)。

7. 視需要設定其餘的執行個體選項。如需詳細資訊，請參閱 [使用定義的參數啟動執行個體](#)。
8. 選擇啟動執行個體。

## AWS CLI

### 將執行個體啟動至主機資源群組

使用 [執行個體](#) AWS CLI 命令，並在 Placement 要求參數中省略租用選項，並指定主機資源群組 ARN。

## PowerShell

將執行個體啟動至主機資源群組

使用命 `New-EC2Instance` AWS Tools for Windows PowerShell 令，並在 Placement 要求參數中省略「租用」選項，並指定主機資源群組 ARN。

### 了解自動配置和親和性

您可以在執行個體層級和主機層級上進行 專用執行個體 配置控制。

#### 自動配置

您會在主機層級設定自動配置。這可讓您管理執行個體要在特定主機上啟動，或在任何具有相符組態的可用主機上啟動。

當專用執行個體的自動配置為停用時，它僅會接受主機租用執行個體的啟動 (指定其唯一主機 ID)。這是新專用執行個體的預設設定。

當專用執行個體的自動配置為啟用時，它會接受符合其執行個體類型組態之任何未訂定目標的執行個體啟動。

在啟動執行個體時，您必須設定其租用。若在專用執行個體上啟動執行個體而未提供特定 HostId，該執行個體就會在啟用自動配置且符合其執行個體類型的任何專用執行個體上啟動。

#### 主機親和性

您必須在執行個體層級設定主機親和性。它會建立執行個體和專用執行個體之間的啟動關係。

當親和性設為 Host 時，先前在特定主機上啟動的執行個體若曾停止，執行個體一律會在相同的主機上重新啟動。這適用於有訂定與未訂定目標的啟動。

當親和性設為 Default 且您停止並重新啟動執行個體時，它會在任何可用的主機上重新啟動。不過，它會嘗試在先前執行的專用執行個體上啟動 (以最佳做法為主)。

#### 修改 專用執行個體 自動配置

將專用主機配置到 AWS 帳戶後，您可以使用下列其中一種方法修改專用主機的自動放置設定。

## Console

修改專用執行個體的自動配置

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。



2. 在導覽窗格中，選擇 專用執行個體。
3. 選取主機並選取 Actions (動作)、Modify host (修改主機)。
4. 對於 Instance auto-placement (執行個體自動配置)，選擇 Enable (啟用) 以啟用自動配置，或清除 Enable (啟用) 以停用自動配置。如需詳細資訊，請參閱 [了解自動配置和親和性](#)。
5. 選擇 Save (儲存)。

## AWS CLI

### 修改專用執行個體的自動配置

使用「修[改主](#) AWS CLI 機」指令。下列範例會針對指定的專用執行個體啟用自動配置。

```
aws ec2 modify-hosts --auto-placement on --host-ids h-012a3456b7890cdef
```

## PowerShell

### 修改專用執行個體的自動配置

使用指[Edit-EC2Host](#) AWS Tools for Windows PowerShell 令。下列範例會針對指定的專用執行個體啟用自動配置。

```
PS C:\> Edit-EC2Host --AutoPlacement 1 --HostId h-012a3456b7890cdef
```

## 修改支援的執行個體類型

下列執行個體系列可支援同一專用執行個體上的多個執行個體類型：C5、M5、R5、C5n、R5n、M5n 以及 T3。其他執行個體族群僅支援同一執行個體類型 專用執行個體。

您可以使用以下方法配置一個專用執行個體。

您可以修改 專用執行個體 來變更其支援的執行個體類型。如果其目前支援單一執行個體類型，您可以進行修改，以支援該執行個體系統內的多個執行個體類型。同樣地，如果其目前支援多個執行個體類型，您可以進行修改，僅支援特定執行個體類型。

若要修改 專用執行個體 以支援多個執行個體類型，您必須先停止主機上所有執行的執行個體。此修改需要約 10 分鐘的時間完成。進行修改時，專用執行個體 會轉移為 pending 狀態。當執行個體處於 pending 狀態時，您無法開始已停止的執行個體，或是啟動 專用執行個體 上的新執行個體。

若要將支援多個執行個體類型的 專用執行個體 修改為僅支援單一執行個體類型，主機必須沒有執行中的執行個體，或是執行中的執行個體必須是您希望主機支援的執行個體類型。例如，若要將支援 m5 執

行個體系列內多個執行個體類型的主機修改為僅支援 m5.large 執行個體，專用執行個體 必須沒有執行中的執行個體，或是其上必須只有 m5.large 個執行個體在執行中。

如果您為虛擬執行個體類型分配主機，則在分配主機之後，無法將執行個體類型修改為 .metal 執行個體類型。例如，如果您為 m5.large 執行個體類型分配主機，則無法將執行個體類型修改為 m5.metal。同樣，如果您為 .metal 執行個體類型分配主機，則在分配主機之後，無法將執行個體類型修改為虛擬執行個體類型。例如，如果您為 m5.metal 執行個體類型分配主機，則無法將執行個體類型修改為 m5.large。

您可以使用下列其中一種方法，來修改支援的執行個體類型。

## Console

### 修改專用執行個體支援的執行個體類型

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Dedicated Host (專用執行個體)。
3. 選取要修改的專用執行個體，然後選取 Actions (動作)、Modify host (修改主機)。
4. 根據專用執行個體的目前組態，執行下列其中一個動作：
  - 如果專用執行個體目前支援特定執行個體類型，則不會啟用 Support multiple instance types (支援多個執行個體類型)，且 Instance type (執行個體類型) 會列出支援的執行個體類型。若要將主機修改為支援目前執行個體系列中的多個類型，請針對 Support multiple instance types (支援多個執行個體類型)，選取 Enable (啟用)。

您必須先停止主機上所有執行的執行個體，然後再將之修改為支援多個執行個體類型。

- 如果專用執行個體目前支援執行個體系列內的多個執行個體類型，請針對 Support multiple instance types (支援多個執行個體類型) 選取 Enabled (已啟用)。若要將主機修改為支援特定執行個體類型，請針對 Support multiple instance types (支援多個執行個體類型) 清除 Enable (啟用)，然後針對 Instance type (執行個體類型) 選取要支援的特定執行個體類型。

您無法變更 專用執行個體 支援的執行個體系列。

5. 選擇 Save (儲存)。

## AWS CLI

### 修改專用執行個體支援的執行個體類型

使用「[修改主](#) AWS CLI 機」指令。

下列命令會修改 專用執行個體 以支援 m5 執行個體系列內的多個執行個體類型。

```
aws ec2 modify-hosts --instance-family m5 --host-ids h-012a3456b7890cdef
```

下列命令會修改 專用執行個體，以便僅支援 m5.xlarge 執行個體。

```
aws ec2 modify-hosts --instance-type m5.xlarge --instance-family --host-ids h-012a3456b7890cdef
```

## PowerShell

修改專用執行個體支援的執行個體類型

使用指 [Edit-EC2Host](#) AWS Tools for Windows PowerShell 令。

下列命令會修改 專用執行個體 以支援 m5 執行個體系列內的多個執行個體類型。

```
PS C:\> Edit-EC2Host --InstanceFamily m5 --HostId h-012a3456b7890cdef
```

下列命令會修改 專用執行個體，以便僅支援 m5.xlarge 執行個體。

```
PS C:\> Edit-EC2Host --InstanceType m5.xlarge --HostId h-012a3456b7890cdef
```

## 修改執行個體租用和親和性

啟動執行個體之後，就可以變更執行個體的租用。您也可以修改執行個體的親和性以鎖定特定主機，或允許在帳戶中具有相符屬性的任何可用專用執行個體上啟動。若要修改執行個體租用或親和性，執行個體必須處於 stopped 狀態。

執行個體的作業系統詳細資訊 (以及是否已安裝 SQL Server) 會影響支援的轉換。如需執行個體可用租用轉換路徑的詳細資訊，請參閱《License Manager 使用指南》中的 [租用轉換](#)。

### Note

若使用 T3 執行個體，您必須在專用執行個體上啟動執行個體，才能使用 host 的租用。使用 T3 執行個體時，您無法將租用從 host 變更為 dedicated 或 default。嘗試進行上述任一不受支援的租用變更時，會顯示 InvalidRequest 錯誤代碼。

您可以使用下列方法來修改執行個體的租用和親和性。

## Console

### 修改執行個體租用或親和性

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 選取 Instances (執行個體)，再選取要修改的執行個體。
3. 選擇 Instance state (執行個體狀態)、Stop (停止)。
4. 在已選取執行個體的情況下，選擇動作、執行個體設定、修改執行個體的配置。
5. 在 [修改執行個體放置] 頁面上，設定下列項目：
  - Tenancy (租用)— 選擇下列其中一項：
    - 執行專用硬體執行個體 – 以專用預留執行個體形式啟動執行個體。如需詳細資訊，請參閱 [專用執行個體](#)。
    - 在專用執行個體上啟動執行個體 – 在可設定親和性的專用執行個體上啟動執行個體。
  - Affinity (親和性)— 選擇下列其中一項：
    - 此執行個體可在我的任一主機上執行 – 執行個體會在您帳戶中任何支援其執行個體類型的可用專用執行個體上啟動。
    - 此執行個體僅能在所選的主機上執行 – 執行個體僅能在為 Target Host (目標主機) 選取的專用執行個體上執行。
  - Target Host (目標主機)— 選取執行個體必須在其上執行的專用執行個體。如果沒有列出任何目標主機，您的帳戶中可能沒有可用且相容的專用執行個體。

如需詳細資訊，請參閱 [了解自動配置和親和性](#)。

6. 選擇 Save (儲存)。

## AWS CLI

### 修改執行個體租用或親和性

使用「[修改](#)-AWS CLI 執行個體-放置」指令。以下範例會將指定執行個體的親和性從 default 變更為 host，並指定執行個體要與哪個專用執行個體保有親和性。

```
aws ec2 modify-instance-placement --instance-id i-1234567890abcdef0 --affinity host
--tenancy host --host-id h-012a3456b7890cdef
```

## PowerShell

### 修改執行個體租用或親和性

使用指 [Edit-EC2InstancePlacement](#) AWS Tools for Windows PowerShell 令。以下範例會將指定執行個體的親和性從 default 變更為 host，並指定執行個體要與哪個專用執行個體保有親和性。

```
PS C:\> Edit-EC2InstancePlacement -InstanceId i-1234567890abcdef0 -Affinity host -
Tenancy host -HostId h-012a3456b7890cdef
```

## 檢視 專用執行個體

您可以使用下列方法，檢視專用執行個體和其中個別執行個體的詳細資訊。

### Console

#### 檢視專用執行個體的詳細資訊

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 專用執行個體。
3. 在 專用執行個體 頁面上，選取一個主機。
4. 如需主機的相關資訊，請選擇 Details (詳細資訊)。

Available vCPU (可用的 vCPU) 代表 專用執行個體 上的可用 vCPU 適用於新的執行個體啟動。例如，支援 c5 執行個體系列內多個執行個體類型的 專用執行個體，而且其上沒有執行個體執行中，則有 72 個可用的 vCPU。這代表您可以在 專用執行個體 上啟動不同組合的執行個體類型，以使用 72 個可用的 vCPU。

如需在主機上執行之執行個體的相關資訊，請選擇 Running instances (執行中的執行個體)。

### AWS CLI

#### 檢視專用執行個體的容量

使用 [描述主機命令](#) AWS CLI。

下列範例使用 [describe-hosts](#) (AWS CLI) 命令以檢視專用執行個體的可用執行個體容量，其支援 c5 執行個體系列內多個執行個體類型。專用執行個體 已經有兩個 c5.4xlarge 執行個體，而且其上有四個 c5.2xlarge 執行個體執行中。

```
aws ec2 describe-hosts --host-id h-012a3456b7890cdef
```

```
"AvailableInstanceCapacity": [
 { "AvailableCapacity": 2,
 "InstanceType": "c5.xlarge",
 "TotalCapacity": 18 },
 { "AvailableCapacity": 4,
 "InstanceType": "c5.large",
 "TotalCapacity": 36 }
],
"AvailableVCpus": 8
```

## PowerShell

檢視專用執行個體的執行個體容量

使用指 [Get-EC2Host](#) AWS Tools for Windows PowerShell 令。

```
PS C:\> Get-EC2Host -HostId h-012a3456b7890cdef
```

## 標記 專用執行個體

您可以將自訂標籤指派給現有的專用執行個體，並以不同的方式將其分類，例如：依用途、擁有者或環境。這可協助您根據所指派的自訂標籤，快速找到特定專用執行個體。專用執行個體標籤也可用於成本配置追蹤。

您也可以在建立期間將標籤套用至專用執行個體。如需詳細資訊，請參閱 [配置 專用執行個體](#)。

您可以使用下列方式來標記專用執行個體。

## Console

### 標記專用執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 專用執行個體。
3. 選取要標記的專用執行個體，然後選取 Actions (動作)、Manage tags (管理標籤)。
4. 在 Manage tags (管理標籤) 畫面中，選擇 Add tag (新增標籤)，然後指定標籤的金鑰和值。
5. (選用) 選擇 Add tag (新增標籤) 以新增其他標籤至專用執行個體。

## 6. 選擇 Save changes (儲存變更)。

### AWS CLI

#### 標記專用執行個體

使用「[建立標籤](#)」指令 AWS CLI。

下列命令會以 Owner=TeamA 標記指定的專用執行個體。

```
aws ec2 create-tags --resources h-abc12345678909876 --tags Key=Owner,Value=TeamA
```

### PowerShell

#### 標記專用執行個體

使用指[New-EC2Tag](#) AWS Tools for Windows PowerShell 令。

New-EC2Tag 命令需要 Tag 物件，這會指定要用於專用執行個體標籤的金鑰對和值對。下列命令會建立名為 Tag 的 \$tag 物件，各自包含 Owner 和 TeamA 的金鑰對和值對。

```
PS C:\> $tag = New-Object Amazon.EC2.Model.Tag
PS C:\> $tag.Key = "Owner"
PS C:\> $tag.Value = "TeamA"
```

下列命令會以 \$tag 物件標記指定的專用執行個體。

```
PS C:\> New-EC2Tag -Resource h-abc12345678909876 -Tag $tag
```

### 監控 專用執行個體

Amazon EC2 會持續監控您的 專用執行個體 的狀態。更新會在 Amazon EC2 主控台上傳達。您可以使用下列方法，檢視專用執行個體的相關資訊。

### Console

#### 檢視專用執行個體的狀態

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 專用執行個體。

3. 在清單中找出專用執行個體，並檢閱 State (狀態) 欄的值。

## AWS CLI

檢視專用執行個體的狀態

使用[描述主機](#) AWS CLI 指令，然後檢閱回應元素中的statehostSet性質。

```
aws ec2 describe-hosts --host-id h-012a3456b7890cdef
```

## PowerShell

檢視專用執行個體的狀態

使用指[Get-EC2Host](#) AWS Tools for Windows PowerShell 令，然後檢閱hostSet回應元素中的state性質。

```
PS C:\> Get-EC2Host -HostId h-012a3456b7890cdef
```

下表說明可能的專用執行個體狀態。

| 州                 | Description                                                                            |
|-------------------|----------------------------------------------------------------------------------------|
| available         | AWS 沒有偵測到專用主機的問題。沒有排定的維護或修復。執行個體可在此專用執行個體上啟動。                                          |
| released          | 專用執行個體已釋出。主機 ID 已不再使用。已釋出的主機即無法重複使用。                                                   |
| under-assessment  | AWS 正在探索專用主機的可能問題。如果必須採取行動，系統會透過 AWS Management Console 或電子郵件通知您。執行個體無法在此狀態的專用執行個體上啟動。 |
| pending           | 專用執行個體 無法用於新的執行個體啟動。 <a href="#">已經修改為支援多個執行個體類型</a> ，或是 <a href="#">主機復原</a> 正在進行中。   |
| permanent-failure | 已偵測到無法復原的失敗。您會透過執行個體和電子郵件收到移出通知。執行個體可繼續運作。如果您停止或終止具有此狀態的專用主機上的所有                       |



| 州                          | Description                                                 |
|----------------------------|-------------------------------------------------------------|
|                            | 執行個體，則會 AWS 淘汰主機。AWS 不會重新啟動處於此狀態的執行個體。執行個體無法在此狀態下於專用執行個體啟動。 |
| released-permanent-failure | AWS 永久釋出失敗且不再具有執行中執行個體的專用主機。專用執行個體 ID 將無法再使用。               |

## 發行版本 專用執行個體

您必須先停止在專用執行個體上執行的任何執行個體，才能釋出主機。您可以將這些執行個體遷移到帳戶中的其他專用執行個體，以便繼續使用。這些步驟僅適用於隨需專用執行個體。

您可以使用下列方法來釋出專用執行個體。

### Console

#### 釋出專用執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 專用執行個體。
3. 在 專用執行個體 (專用執行個體) 頁面中，選取要釋出的專用執行個體。
4. 選擇 Actions (動作)、Release host (釋出主機)。
5. 若要確認，請選擇 Release (釋出)。

### AWS CLI

#### 釋出專用執行個體

使用「[發布 AWS CLI 主機](#)」命令。

```
aws ec2 release-hosts --host-ids h-012a3456b7890cdef
```

### PowerShell

#### 釋出專用執行個體

使用指[Remove-EC2Hosts](#) AWS Tools for Windows PowerShell 令。

```
PS C:\> Remove-EC2Hosts -HostId h-012a3456b7890cdef
```

在您釋出專用執行個體之後，即無法再次使用相同的主機或主機 ID，也不會再以隨需計費費率向您收費。專用執行個體的狀態會變更為 released，且您將無法在該主機上啟動任何執行個體。

#### Note

如果您最近曾釋出專用執行個體，其可能需要一些時間才能停止計入限制。在這段期間，您可能會在嘗試配置新的專用執行個體時遇到 LimitExceeded 錯誤。若是這種情況，請嘗試在幾分鐘後再次配置新的主機。

停止的執行個體仍可供使用，並會列在 Instances (執行個體) 頁面中。這些執行個體仍保有其 host 租用設定。

## 購買 專用執行個體預留

您可以使用下列方法來購買保留：

### Console

#### 購買保留

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 選擇 專用執行個體、專用執行個體預留、Purchase 專用執行個體預留 (購買 專用執行個體預留)。
3. 在「尋找提供項目」畫面上，執行下列動作：
  - a. 針對執行個體系列，選取要為其購買專用主機保留項目之專用主機的執行個體系列。
  - b. 對於付款選項，選擇並配置您偏好的付款選項。
4. 選擇下一步。
5. 選取要與專用主機保留區產生關聯的專用主機，然後選擇下一步。
6. (選擇性) 將標記指派給專用主機保留區。
7. 檢查您的訂單並選擇「購買」。

## AWS CLI

### 購買保留

1. 使用 [\[描述-主機-保留-提供項目\]](#) AWS CLI 指令列出符合您需求的可用訂購項目。下列範例列出的方案可支援 m4 執行個體系列中的執行個體，並具有一年期限。

#### Note

此期限以秒為單位指定。一年期限包括 31,536,000 秒，三年期限則為 94,608,000 秒。

```
aws ec2 describe-host-reservation-offerings --filter Name=instance-family,Values=m4 --max-duration 31536000
```

此命令會傳回符合您條件的方案清單。請記下要購買方案的 offeringId。

2. 使用 [購買-主機預訂](#) AWS CLI 指令購買提供項目，並提供上一個步驟中 offeringId 所述的項目。下列範例會購買指定的保留區，並將其與 AWS 帳戶中已配置的特定專用主機產生關聯，並套具有金鑰 purpose 且值為的標記 production。

```
aws ec2 purchase-host-reservation --offering-id hro-03f707bf363b6b324 --host-id-set h-013abcd2a00cbd123 --tag-specifications 'ResourceType=host-reservation,Tags={Key=purpose,Value=production}'
```

## PowerShell

### 購買保留

1. 使用指 [Get-EC2HostReservationOffering](#) AWS Tools for Windows PowerShell 令列出符合您需求的可用供應項目。下列範例列出的方案可支援 m4 執行個體系列中的執行個體，並具有一年期限。

#### Note

此期限以秒為單位指定。一年期限包括 31,536,000 秒，三年期限則為 94,608,000 秒。

```
PS C:\> $filter = @{Name="instance-family"; Value="m4"}
```

```
PS C:\> Get-EC2HostReservationOffering -filter $filter -MaxDuration 31536000
```

此命令會傳回符合您條件的方案清單。請記下要購買方案的 offeringId。

2. 使用指 [New-EC2HostReservation](#) AWS Tools for Windows PowerShell 令購買訂購項目，並提供上一個步驟中 offeringId 所述的項目。下列範例會購買指定的保留區，並將其與 AWS 帳戶中已配置的特定專用主機產生關聯。

```
PS C:\> New-EC2HostReservation -OfferingId hro-03f707bf363b6b324 -
HostIdSet h-013abcd2a00cbd123
```

## 檢視 專用執行個體 保留

您可以檢視與您的保留相關聯之專用執行個體的相關資訊，包括：

- 保留的期限
- 付款選項
- 開始和結束日期

您可以使用下列方法，檢視專用執行個體保留的詳細資訊。

## Console

檢視專用執行個體保留的詳細資訊

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 專用執行個體。
3. 在 專用執行個體 (專用執行個體) 頁面中，選取 專用執行個體 Reservations (專用執行個體保留)，然後從提供的清單中選取保留。
4. 選擇 Details (詳細資訊) 以取得保留的資訊。
5. 選擇 Hosts (主機) 以取得與保留相關聯的專用執行個體資訊。

## AWS CLI

檢視專用執行個體保留的詳細資訊

使用 [描述-主機-保留命令](#) AWS CLI 。

```
aws ec2 describe-host-reservations
```

## PowerShell

檢視專用執行個體保留的詳細資訊

使用指 [Get-EC2HostReservation](#) AWS Tools for Windows PowerShell 令。

```
PS C:\> Get-EC2HostReservation
```

## 標記 專用執行個體預留

您可以將自訂標籤指派給專用執行個體預留，以不同的方式將其分類，例如：依用途、擁有者或環境。這可協助您根據所指派的自訂標籤，快速找到特定專用執行個體預留。

您僅可使用命令列工具標記專用執行個體預留。

## AWS CLI

標記專用執行個體預留

使用「[建立標籤](#)」指令 AWS CLI 。

```
aws ec2 create-tags --resources hr-1234563a4ffc669ae --tags Key=Owner,Value=TeamA
```

## PowerShell

標記專用執行個體預留

使用指 [New-EC2Tag](#) AWS Tools for Windows PowerShell 令。

New-EC2Tag 命令需要 Tag 參數，這指定要用於專用執行個體預留標籤的索引鍵/值對。下列命令會建立 Tag 參數。

```
PS C:\> $tag = New-Object Amazon.EC2.Model.Tag
PS C:\> $tag.Key = "Owner"
```

```
PS C:\> $tag.Value = "TeamA"
```

```
PS C:\> New-EC2Tag -Resource hr-1234563a4ffc669ae -Tag $tag
```

## 使用共用的 專用執行個體

專用主機共用可讓專用主機擁有者與其他 AWS 帳戶或 AWS 組織內共用其專用主機。這可讓您集中建立和管理專用主機，以及跨多個 AWS 帳戶或 AWS 組織內共用專用主機。

在此模型中，擁有專用主機 (擁有者) 的 AWS 帳戶會與其他 AWS 帳戶 (消費者) 共用該帳戶。消費者可以在與他們共用的專用執行個體中啟動執行個體，就好像他們在自己的帳戶中配置的專用執行個體中啟動執行個體一樣。擁有者負責管理專用執行個體和他們在其中啟動的執行個體。擁有者無法修改消費者在共用的專用執行個體上啟動的執行個體。消費者負責管理在分享給他們的專用執行個體中啟動的執行個體。消費者無法檢視或修改其他消費者或專用執行個體擁有者所擁有的執行個體，也無法修改與他們共用的專用執行個體。

專用執行個體擁有者可以將專用執行個體分享給下列對象：

- AWS 組織內部或外部的特定 AWS 帳戶
- 組織內的組 AWS 織單位
- 它的整個 AWS 組織

## 目錄

- [共用 專用執行個體 的先決條件](#)
- [共用 專用執行個體 的限制](#)
- [相關服務](#)
- [跨可用區域共用](#)
- [共用 專用執行個體](#)
- [取消共享已共用的 專用執行個體](#)
- [識別共用的 專用執行個體](#)
- [檢視在共用 專用執行個體 上執行的執行個體](#)
- [共用的 專用執行個體 許可](#)
- [計費和計量](#)
- [專用執行個體 限制](#)

## • [主機復原和 專用執行個體 共用](#)

### 共用 專用執行個體 的先決條件

- 若要共用專用主機，您必須在 AWS 帳戶中擁有該主機。您無法將已分享給您的專用執行個體再分享出去。
- 若要與您的組織或 AWS 組織中的組織單位共用專用主機，您 AWS 必須啟用與共用 AWS Organizations。如需詳細資訊，請參閱 AWS RAM 使用者指南中的[透過 AWS Organizations 啟用共用](#)。

### 共用 專用執行個體 的限制

您無法共用已針對下列執行個體類型配置的 專用執行個體：u-6tb1.metal、u-9tb1.metal、u-12tb1.metal、u-18tb1.metal 和 u-24tb1.metal。

### 相關服務

#### AWS Resource Access Manager

專用主機共用與 AWS Resource Access Manager (AWS RAM) 整合。AWS RAM 是一項服務，可讓您與任何 AWS 帳戶或透過其他帳戶共用 AWS 資源 AWS Organizations。您可以透過建立資源共用 AWS RAM 來共用您所擁有的資源。資源共享指定要共用的資源，以及共用它們的消費者。消費者可以是個人 AWS 帳戶，也可以是組織單位或整個組織 AWS Organizations。

若要取得有關的更多資訊 AWS RAM，請參閱[AWS RAM 使用者指南](#)。

### 跨可用區域共用

為確保資源分配至區域中的所有可用區域，可用區域會獨立對應至各個帳戶的名稱。這可能導致帳戶之間的可用區域命名出現差異。例如，您 AWS 帳戶的可 us-east-1a 用區域可能與其他 AWS 帳戶 us-east-1a 的位置不同。

若要基於您的帳戶來識別專用執行個體的相對位置，您必須使用可用區域 ID (AZ ID)。可用區域 ID 是所有 AWS 帳戶中可用區域的唯一且一致的識別碼。例如，use1-az1 是 us-east-1 區域的可用區域 ID，它在每一個 AWS 帳戶的位置都相同。

### 檢視您帳戶中可用區域的可用區域 ID

1. [請在以下位置開啟 AWS RAM 主控台](https://console.aws.amazon.com/ram)。 <https://console.aws.amazon.com/ram>

2. 畫面右側的 Your AZ ID (您的 AZ ID) 面板中會顯示目前區域的可用區域 ID。

## 共用 專用執行個體

當擁有者分享專用執行個體，它可讓消費者在主機上啟動執行個體。在共用主機上可用容量所允許下，消費者可以在當中啟動任意數量的執行個體。

### Important

請注意，您應負責確保您擁有適當的授權，可共用您的 專用執行個體 上的任何 BYOL 授權。

如果您在啟用自動配置時共用專用執行個體，請記住下列事項，因為這可能會導致意外的專用執行個體使用量：

- 如果消費者使用專用執行個體租用啟動執行個體，但他們的帳戶中擁有的專用執行個體上沒有容量，則該執行個體會自動啟動到共用的專用執行個體。

若要分享專用執行個體，您必須將它新增至資源共享。資源共用是一 AWS RAM 種可讓您跨 AWS 帳號共用資源的資源。資源共享指定要分享的資源，以及共用它們的消費者。您可以將專用執行個體新增至現有的資源，或將其新增至新的資源共享。

如果您是組織的一員，AWS Organizations 且已啟用組織內的共用功能，則組織中的取用者會自動獲得共用專用主機的存取權。否則，消費者會收到加入資源共享的邀請，並且在接受邀請後便能存取共用的專用執行個體。

### Note

分享專用執行個體後，消費者可能需要幾分鐘的時間才能存取它。

您可以使用下列其中一種方法，來共用您擁有的專用執行個體。

## Amazon EC2 console

如何使用 Amazon EC2 主控台共用您擁有的專用執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 專用執行個體。



3. 選擇專用執行個體來共用，然後選擇 Actions (動作) > Share host (共用主機)。
4. 選取資源共享以新增專用執行個體，然後選取 Share host (共用主機)。

可能需要經過幾分鐘，消費者才能存取共用的主機。

## AWS RAM console

共用您使用主控台擁有的專用 AWS RAM 主機

請參閱《AWS RAM 使用者指南》中的[建立資源共享](#)。

## AWS CLI

使用共用您擁有的專用主機 AWS CLI

使用 [create-resource-share](#) 命令。

## 取消共享已共用的 專用執行個體

專用執行個體擁有者可以隨時取消共享已共用的專用執行個體。當您取消共享已共用的專用執行個體時，請注意以下規則：

- 與其共用專用執行個體的消費者無法再於其上啟動新的執行個體。
- 取消共用時在專用執行個體上執行中的消費者所擁有的執行個體會繼續執行，但排定進行[淘汰](#)。消費者會收到執行個體的淘汰通知，而且他們有兩週的時間可對通知採取行動。但是，如果在淘汰通知期間內與消費者重新共用專用執行個體，則會取消執行個體淘汰。

若要取消共享您擁有的已共用專用執行個體，您必須從資源共享中移除它。您可以使用下列其中一種方法來執行這項作業。

## Amazon EC2 console

如何使用 Amazon EC2 主控台來取消共享您擁有的已共用專用執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 專用執行個體。
3. 選擇要取消共享的專用執行個體，然後選擇 Sharing (共用) 索引標籤。
4. Sharing (共用) 索引標籤列出已新增專用執行個體的資源共享。選取要從中移除專用執行個體的資源共享，然後選取 Remove host from resource share (從資源共享中移除主機)。

## AWS RAM console

使用 AWS RAM 主控台來取消共用您擁有的已共用專用執行個體

請參閱 AWS RAM 使用者指南中的[更新資源共享](#)。

## Command line

若要取消共用您所擁有的共用專用主機 AWS CLI

使用 [disassociate-resource-share](#) 命令。

## 識別共用的 專用執行個體

擁有者和消費者可以使用下列其中一種方法，來識別共用的專用執行個體。

### Amazon EC2 console

使用 Amazon EC2 主控台來識別共用的專用執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 專用執行個體。畫面會列出您擁有的專用執行個體和已分享給您的專用執行個體。Owner (擁有者) 資料欄會顯示專用執行個體擁有者的 AWS 帳戶 ID。

## Command line

使用識別共用專用主機 AWS CLI

使用 [describe-hosts](#) 命令。此命令會傳回您擁有的專用執行個體且和已分享給您的專用執行個體。

## 檢視在共用 專用執行個體 上執行的執行個體

擁有者和消費者可以隨時使用下列其中一種方法，來檢視在共用的專用執行個體上執行的執行個體。

### Amazon EC2 console

使用 Amazon EC2 主控台檢視在共用的專用執行個體上執行的執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 專用執行個體。

3. 選取要檢視其執行個體的專用執行個體，並選取 Instances (執行個體)。此標籤會列出主機上執行的執行個體。擁有者會看到主機上執行的所有執行個體，包括消費者啟動的執行個體。消費者只會看到他們在主機上啟動的執行個體。Owner (擁有者) 欄會顯示啟動執行個體之帳戶的 AWS 帳戶 ID。

## Command line

使用 AWS CLI 檢視在共用的專用執行個體上執行的執行個體

使用 [describe-hosts](#) 命令。該命令會傳回每個專用執行個體上執行的執行個體。擁有者會看到主機上執行的所有執行個體。消費者只會看到他們在共用主機上啟動的執行個體。InstanceOwnerId 會顯示執行個體擁有者的 AWS 帳戶 ID。

## 共用的 專用執行個體 許可

### 擁有者的許可

擁有者負責管理他們共用的專用執行個體以及在其上啟動的執行個體。擁有者可以檢視在共用的專用執行個體上執行的所有執行個體，包括消費者啟動的執行個體。不過，擁有者無法對消費者啟動的執行個體採取任何動作。

### 消費者的許可

消費者負責管理他們在共用的專用執行個體上啟動的執行個體。消費者完全無法修改共用的專用執行個體，也無法檢視或修改其他消費者或專用執行個體擁有者所擁有的執行個體。

## 計費和計量

共用專用執行個體無須額外收費。

擁有者會按他們共用的專用執行個體計費。對於消費者在共用的專用執行個體上啟動的執行個體則不計費。

專用執行個體預留會繼續為共用的專用執行個體提供帳單折扣。只有專用執行個體擁有者可以為他們擁有的共用專用執行個體購買專用執行個體預留。

## 專用執行個體 限制

共用的專用執行個體只會計入擁有者的專用執行個體限制。消費者的專用執行個體限制不受已與他們共用的專用執行個體所影響。同樣地，消費者在共用的專用執行個體上啟動的執行個體不會計入他們的執行個體限制。

## 主機復原和 專用執行個體 共用

主機復原可復原專用執行個體擁有者和其所共用之消費者啟動的執行個體。替換專用執行個體會配置到擁有者的帳戶。它會新增至與原始專用執行個體相同的資源共享，並與相同的消費者共用。

如需詳細資訊，請參閱 [主機復原](#)。

## 專用主機 AWS Outposts

AWS Outposts 是一項完全受控的服務，可將 AWS 基礎架構、服務、API 和工具延伸至您的場所。透過提供 AWS 受管理基礎結構的本機存取權，AWS Outposts 可讓您使用與 AWS 區域相同的程式設計介面在內部部署建置和執行應用程式，同時使用本機運算和儲存資源來降低延遲和本機資料處理需求。

Outpost 是部署在客戶站點的 AWS 計算和儲存容量集區。AWS 作為 AWS 區域的一部分來操作、監控和管理此容量。

您可以在帳戶中擁有的 Outpost 上配置專用執行個體。這讓您可以更輕鬆地將需要專用實體伺服器的現有軟體授權和工作負載帶至 AWS Outposts。您也可以鎖定 Outpost 上的特定硬體資產，協助將工作負載之間的延遲降到最低。

專用執行個體可讓您在 Amazon EC2 上使用符合條件的軟體授權，進而獲得使用自有授權的靈活性和成本效益。綁定至虛擬機器、通訊端或實體核心的其他軟體授權也可以在專用執行個體上使用，但需遵守其授權條款。雖然 Outpost 一直是符合 BYOL 工作負載條件的單一租用戶環境，但專用執行個體允許您將所需的授權限制為單個主機，而不是整個 Outpost 部署。

此外，在 Outpost 上使用專用執行個體可為您在執行個體類型部署方面提供更大的靈活性，並對執行個體放置進行更精細的控制。您可以針對特定主機進行執行個體啟動，並使用主機親和性來確保執行個體始終在該主機上執行，您也可以使用自動放置，在具有相符組態和可用容量的任何可用主機上啟動執行個體。

### 內容

- [必要條件](#)
- [支援的功能](#)
- [考量事項](#)
- [在 Outpost 上配置和使用專用執行個體](#)

### 必要條件

您的站點必須安裝 Outpost。如需詳細資訊，請參閱 AWS Outposts 使用者指南中的 [建立 Outpost 並訂購 Outpost 容量](#)。

## 支援的功能

- 支援以下執行個體系列：C5、M5、R5、C5d、M5d、R5d、G4dn 和 i3en。
- Outpost 上的專用執行個體可設定為支援多種執行個體規模。下列執行個體系列可支援多個執行個體類型：C5、M5、R5、C5d、M5d 和 R5d。如需詳細資訊，請參閱 [執行個體容量組態](#)。
- Outpost 上的專用執行個體支援自動放置和鎖定執行個體啟動。如需詳細資訊，請參閱 [了解自動配置和親和性](#)。
- Outpost 上的專用執行個體支援主機親和性。如需詳細資訊，請參閱 [了解自動配置和親和性](#)。
- Outposts 上的專用主機支持與 AWS RAM 共享。如需詳細資訊，請參閱 [使用共用的專用執行個體](#)。

## 考量事項

- Outpost 上不支援專用執行個體保留功能。
- Outposts 不支援主機資源群組和 AWS License Manager 主機資源群組。
- Outpost 上的專用執行個體不支援爆量 T3 執行個體。
- Outpost 上的專用執行個體不支援主機復原。
- 在 Outposts 上使用專用主機租用的執行個體不支援簡化的自動復原。

## 在 Outpost 上配置和使用專用執行個體

您在 Outpost 上配置和使用專用執行個體的方式與在 AWS 區域中使用專用執行個體的方式相同。

## 必要條件

在 Outpost 上建立子網。如需詳細資訊，請參閱 AWS Outposts 使用者指南中的 [建立子網](#)。

若要在 Outpost 中配置專用執行個體，請使用下列其中一種方法：

### AWS Outposts console

1. 開啟主 AWS Outposts 控制台，網址為 <https://console.aws.amazon.com/outposts/>。
2. 在導覽窗格中，選擇 Outposts。選取 Outpost，然後選取 Actions (動作)、Allocate Dedicated Host (配置專用執行個體)。
3. 視需設定專用執行個體。如需詳細資訊，請參閱 [配置專用執行個體](#)。

**Note**

Availability Zone (可用區域) 和 Outpost ARN 應使用選定 Outpost 的可用區域和 ARN 預先填充。

4. 選擇 Allocate (配置)。

## Amazon EC2 console

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Dedicated Hosts (專用執行個體)，然後選擇 Allocate Dedicated Host (配置專用執行個體)。
3. 對於 Availability Zone (可用區域)，選取與 Outpost 關聯的可用區域。
4. 對於 Outpost ARN，輸入 Outpost 的 ARN。
5. 若要鎖定 Outpost 上的特定硬體資產，請選擇 啟用 鎖定 Outpost 上的特定硬體資產。針對要鎖定的每項硬體資產，請選擇 新增資產 ID，然後輸入硬體資產的 ID。

**Note**

您為 數量 指定的值必須等於所指定的資產 ID 的數量。例如，如果您指定 3 個資產 ID，則數量也必須為 3。

6. 視需進行其餘的專用執行個體設定。如需詳細資訊，請參閱 [配置 專用執行個體](#)。
7. 選擇 Allocate (配置)。

## AWS CLI

使用 [分配主機命令](#) AWS CLI。對於 `--availability-zone`，指定與 Outpost 關聯的可用區域。對於 `--outpost-arn`，指定 Outpost 的 ARN。您也可以針對 `--asset-ids` 選擇指定要鎖定的 Outpost 硬體資產的 ID。

```
aws ec2 allocate-hosts --availability-zone "us-east-1a" --outpost-arn
"arn:aws:outposts:us-east-1a:111122223333:outpost/op-4fe3dc21baEXAMPLE" --asset-
ids asset_id --instance-family "m5" --auto-placement "off" --quantity 1
```

## 將執行個體啟動至 Outpost 上的專用執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 專用執行個體。選取您在上一步驟中配置的專用執行個體，然後選取 Actions (動作)、Launch instance onto host (在主機上啟動執行個體)。
3. 根據需要設定執行個體，然後啟動執行個體。如需詳細資訊，請參閱 [在 專用執行個體 中啟動執行個體](#)。

## 主機復原

如果在您的專用執行個體上偵測到特定問題狀況，專用執行個體自動復原會自動將您的執行個體重新啟動到新的替代主機。若有關於系統電源或網路連線事件的非預期專用執行個體失敗，主機復原可減少手動介入的需要，並降低操作負擔。其他專用執行個體問題將需要手動介入才能從中復原。

### 目錄

- [主機復原基本概念](#)
- [支援的執行個體類型](#)
- [設定主機復原](#)
- [主機復原狀態](#)
- [手動復原不支援的執行個體](#)
- [相關服務](#)
- [定價](#)

### 主機復原基本概念

專用執行個體和主機資源群組復原程序會使用主機層級運作狀態檢查，來存取專用執行個體可用性，以及偵測基礎系統失敗。專用執行個體失敗的類型決定專用執行個體自動復原是否可行。導致主機層級運作狀態檢查失敗的問題範例包括：

- 網路連線中斷
- 系統電力中斷
- 實體主機上的硬體或軟體問題

**⚠ Important**

主機排定為淘汰時，不會發生專用執行個體自動復原。

**專用執行個體自動復原**

在專用主機上偵測到系統電源或網路連線故障時，會啟動專用主機 auto 動復原，Amazon EC2 會在與原始專用主機相同的可用區域中自動配置替換專用主機。替代的 專用執行個體 會收到新的主機 ID，但保留與原始 專用執行個體 相同的屬性，包括：

- 可用區域
- 執行個體類型
- 標籤
- 自動配置設定
- 保留

當配置替代的專用執行個體時，會將執行個體復原至替代的專用執行個體。復原的執行個體會保留與原始執行個體相同的屬性，包括：

- 執行個體 ID
- 私有 IP 地址
- 彈性 IP 地址
- EBS 磁碟區連接
- 所有執行個體中繼資料

此外，與 AWS License Manager 的內置集成可以自動跟踪和管理許可證。

**ℹ Note**

AWS 僅在可使用 License Manager 的地區支援 AWS License Manager 整合。

如果執行個體與受損的 專用執行個體 具有主機親和性關係，復原的執行個體會與替代的 專用執行個體 建立主機親和性。



當所有執行個體都已復原至替代的 專用執行個體 上時，受損的 專用執行個體 會被釋出，並且替代的 專用執行個體 會變得可供使用。

啟動主機復原時，會透過電子郵件和 AWS Health Dashboard 事件通知 AWS 帳戶擁有者。在主機復原已成功完成時，會傳送第二個通知。

如果您使用 AWS License Manager 來追蹤授權，則 AWS License Manager 會根據授權組態限制為替換的專用主機配置新的授權。如果授權組態具有因主機復原而違反的硬性限制，則不允許復原程序，且會透過 Amazon SNS 通知 (如果已為 AWS License Manager 設定通知設定) 通知您主機復原失敗。如果授權組態具有將因為主機復原而違反的軟性限制，將會允許繼續復原，並且將透過 Amazon SNS 通知來通知您相關的限制違反。如需詳細資訊，請參閱 AWS License Manager 使用者指南中的 [使用授權組態](#) 和 [License Manager 中的設定](#)。

### 沒有專用執行個體自動復原的案例

主機排定為淘汰時，不會發生專用執行個體自動復原。您將在 Amazon CloudWatch 事件中收到退休通知 AWS Health Dashboard，AWS 帳戶擁有者電子郵件地址會收到有關專用主機故障的訊息。請在指定的時間段內遵循淘汰通知中所述的補救步驟，以手動復原淘汰主機中的執行個體。

已停止的執行個體不會復原到替代的 專用執行個體 上。如果您嘗試啟動以受損的 專用執行個體 為目標的停止的執行個體，該執行個體啟動會失敗。建議您將已停止的執行個體修改為以不同的 專用執行個體 為目標，或是在具有相符組態並且已啟用自動配置的任何可用的 專用執行個體 上啟動。

具有執行個體儲存體的執行個體不會復原到替代的專用執行個體上。作為補救措施，會將受損的 專用執行個體 標示為供淘汰，並且您會在主機復原完成之後收到淘汰通知。請在指定的時間期間內遵循淘汰通知中所述的補救步驟，以手動復原受損的 專用執行個體 上其餘的執行個體。

### 支援的執行個體類型

下列執行個體系列支援主機復原：A1、C3、C4、C5、C5n、C6a、C6g、C6i、資訊

1、G3、G5G、M4、M5、M5n、M5zn、M6a、M6g、M6i、P2、M4、R5B、R54、R5B、N、R6 克、R6 I、T3、X1、X1e、X2iezn、u-6 坦克 1、u-9 泰必 1、U-十二 TB1、U-十八泰比 1、和 U-24 泰比 1。

若要復原不受支援的執行個體，請參閱 [手動復原不支援的執行個體](#)。

#### Note

支援的金屬執行個體類型的專用執行個體自動復原將需要比非金屬執行個體類型更長的時間來偵測和復原。

## 設定主機復原

您可以在專用主機分配時或使用 Amazon EC2 主控台或 AWS Command Line Interface (CLI) 分配後設定主機復原。

### 目錄

- [啟用主機復原](#)
- [停用主機復原](#)
- [檢視主機復原組態](#)

## 啟用主機復原

您可以在 專用執行個體 配置時或在配置之後啟用主機復原。

如需在 專用執行個體 配置時啟用主機復原的詳細資訊，請參閱 [配置 專用執行個體](#)。

### 使用主控台在配置之後啟用主機復原

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 專用執行個體。
3. 選取要為其啟用主機復原的 專用執行個體，然後選取 Actions (動作)、Modify Host Recovery (修改主機復原)。
4. 對於 Host recovery (主機復原)，選擇 Enable (啟用)，然後選擇 Save (儲存)。

### 使用配置後啟用主機復原 AWS CLI

使用 [modify-hosts](#) 命令，並指定 `host-recovery` 參數。

```
$ aws ec2 modify-hosts --host-recovery on --host-ids h-012a3456b7890cdef
```

## 停用主機復原

您可以在 專用執行個體 配置之後隨時停用主機復原。

### 使用主控台在配置之後停用主機復原

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 專用執行個體。

3. 選取要為其停用主機復原的 專用執行個體，然後選取 Actions (動作)、Modify Host Recovery (修改主機復原)。
4. 對於 Host recovery (主機復原)，選擇 Disable (停用)，然後選擇 Save (儲存)。

### 使用配置後停用主機復原 AWS CLI

使用 [modify-hosts](#) 命令，並指定 `host-recovery` 參數。

```
$ aws ec2 modify-hosts --host-recovery off --host-ids h-012a3456b7890cdef
```

### 檢視主機復原組態

您可以隨時檢視 專用執行個體 的主機復原組態。

使用主控台檢視 專用執行個體 的主機復原組態

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 專用執行個體。
3. 選取 專用執行個體，並在 Description (描述) 索引標籤中，檢閱 Host Recovery (主機復原) 欄位。

使用 AWS CLI 檢視專用執行個體的主機復原組態

使用 [describe-hosts](#) 命令。

```
$ aws ec2 describe-hosts --host-ids h-012a3456b7890cdef
```

HostRecovery 回應元素指出主機復原是否已啟用或已停用。

### 主機復原狀態

偵測到專用執行個體失敗時，受損的專用執行個體會進入 `under-assessment` 狀態，並且所有執行個體會進入 `impaired` 狀態。當受損的 專用執行個體 處於 `under-assessment` 狀態時，您無法對其啟動執行個體。

配置替代的 專用執行個體 之後，它會進入 `pending` 狀態。在主機復原程序完成之前，它會保持在此狀態。當替代的 專用執行個體 處於 `pending` 狀態時，您無法對其啟動執行個體。在復原程序期間，替代的 專用執行個體 上復原的執行個體會維持為 `impaired` 狀態。

在主機復原完成之後，替代的專用執行個體會進入 available 狀態，而復原的執行個體會回到 running 狀態。當替代的專用執行個體進入 available 狀態時，您可以對其啟動執行個體。原始受損的專用執行個體會永久釋出並且進入 released-permanent-failure 狀態。

如果受損的專用執行個體具有不支援主機復原的執行個體，例如具備執行個體存放區磁碟區的執行個體，則不會釋出專用執行個體。而是會將它標記為供淘汰，並進入 permanent-failure 狀態。

### 手動復原不支援的執行個體

主機復原不支援復原使用執行個體存放區磁碟區的執行個體。遵循以下的指示來手動復原您無法自動復原的任何執行個體。

#### Warning

位在執行個體存放區磁碟區的資料都會在執行個體停止、休眠或終止時遺失。這包含連接到根設備為 EBS 磁碟區之執行個體的執行個體存放區磁碟區。若要保護執行個體存放區磁碟區的資料，請在執行個體停止或終止之前，將資料備份至持久性儲存。

### 手動復原 EBS 後端執行個體

對於無法自動復原的 EBS 後端執行個體，建議您手動停止並啟動執行個體，以將它們復原至新的專用執行個體。如需停止您執行個體的詳細資訊，以及在停止時您執行個體組態中發生的變更，請參閱[停止並啟動 Amazon EC2 執行個體](#)。

### 手動復原執行個體存放區後端執行個體

對於無法自動復原的執行個體存放區後端執行個體，建議您執行下列：

1. 從您的最近的 AMI 的新專用執行個體上啟動替代的執行個體。
2. 將所有必要資料遷移至替代的執行個體。
3. 終止受損的專用執行個體上的原始執行個體。

### 相關服務

專用執行個體已與下列服務整合：

- AWS License Manager — 追蹤 Amazon EC2 專用主機上的授權 (僅在提供 AWS License Manager 的區域支援)。如需詳細資訊，請參閱 [AWS License Manager 使用者指南](#)。

## 定價

使用主機復原沒有額外的費用，但適用一般的 專用執行個體 費用。如需詳細資訊，請參閱 [Amazon EC2 專用執行個體定價](#)。

一旦初始化主機復原，將不再就受損的 專用執行個體 向您收費。替代專用執行個體的計費方式只會在進入 available 狀態之後開始。

如果受損的 專用執行個體 是使用隨需費率計費，則替代的 專用執行個體 也會使用隨需費率計費。如果受損的 專用執行個體 有作用中 專用執行個體預留，則會傳輸到替代的 專用執行個體。

## 主機維護

透過主機維護，降級專用主機上的 Amazon EC2 執行個體會在排定的維護事件期間，在替代的專用主機上自動重新啟動。這有助於減少應用程式停機時間，並將無差異化的繁重維護工作卸載到 AWS。也會針對規劃和例行的 Amazon EC2 維護執行主機維護。

透過 Amazon EC2 主控台進行的所有新專用執行個體配置均支援主機維護。對於您 AWS 帳戶 或透過 [AllocateHosts](#) API 配置的任何新專用主機中的任何專用主機，您可以為支援的專用主機設定主機維護。如需詳細資訊，請參閱 [the section called “設定主機維護”](#)。

## 目錄

- [主機維護基礎概念](#)
- [主機維護與主機復原](#)
- [支援的執行個體類型](#)
- [專用執行個體上的執行個體](#)
- [設定主機維護](#)
- [維護事件](#)
- [主機維護狀態](#)
- [相關服務](#)
- [定價](#)

## 主機維護基礎概念

在專用執行個體上偵測到降級時，會配置新的專用執行個體。降級可能是由於基礎硬體降級或偵測到某些有問題的情況而引起。降級的專用主機上的執行個體會排程在替換的專用主機上自動重新開機。

替代的專用執行個體會收到新的主機 ID，但保留與原始專用執行個體相同的屬性。這些屬性包含下列項目。

- 自動配置設定
- 可用區域
- 保留
- 主機親和性
- 主機維護設定
- 主機復原設定
- 執行個體類型
- 標籤

所有受支援的專用主 AWS 區域 機均可進行主機維護。如需不支援主機維護之專用執行個體的詳細資訊，請參閱 [the section called “限制”](#)。

在所有執行個體重新啟動到新的專用執行個體或停止後，會釋放降級的專用執行個體。您可以在排定的維護事件發生之前存取降級之專用執行個體上的執行個體，但不支援在降級的專用執行個體上啟動執行個體。

在排定的維護事件發生之前，您可以使用取代的專用主機在主機上啟動新執行個體。不過，替換主機上的某些執行個體容量會保留給必須從降級主機移轉的執行個體使用。您無法在此保留容量中啟動新執行個體。如需詳細資訊，請參閱 [the section called “專用執行個體上的執行個體”](#)。

## 限制

- Local Zones 和 AWS Wavelength AWS 區不支援主機維護。AWS Outposts
- 對於已在主機資源群組中的主機，無法開啟或關閉主機維護。新增至主機資源群組的主機會保留其主機維護設定。如需詳細資訊，請參閱[主機資源群組](#)。
- 只有特定執行個體類型才支援主機維護。如需詳細資訊，請參閱 [the section called “支援的執行個體類型”](#)。

## 主機維護與主機復原

下表顯示主機復原與主機維護之間的主要差異。

|        | 主機復原             | 主機維護              |
|--------|------------------|-------------------|
| 可存取性   | 無法連接             | 可連接               |
| State  | under-assessment | permanent-failure |
| 動作     | 立即復原             | 已排定維護             |
| 排程彈性   | 無法重新排程           | 可重新排程             |
| 主機資源群組 | 支援               | 不支援               |

如需有關主機復原的詳細資訊，請參閱[主機復原](#)。

### 支援的執行個體類型

下列執行個體系列支援主機維護：

- 一般用途：A1 | M4 | M5 | M5a | M5n | M5zn | M6a | M6g | M6i | M6in | M7a | M7g | M7i | T3
- 運算最佳化：C4 | C5 | C5a | C5n | C6a | C6g | C6gn | C6i | C6in | C7g | C7gn | C7i
- 記憶體最佳化：R4 | R5 | R5a | R5b | R5n | R6a | R6g | R6i | R6in | R7a | R7g | R7iz | u-12tb1 | u-18tb1 | u-24tb1 | u-3tb1 | u-6tb1 | u-9tb1 | X2iezn
- 加速運算：G3 | G5g | Inf1 | P2 | P3

### 專用執行個體上的執行個體

對於將從降級的主機自動遷移的執行個體，Amazon EC2 會在替換主機上自動保留容量。Amazon EC2 不會為無法自動遷移的執行個體 (例如具有執行個體存放根磁碟區的執行個體) 保留替代主機上的容量。保留容量無法用於啟動新執行個體。

#### Note

Amazon EC2 主控台會將保留容量顯示為已使用容量。可能看起來執行個體正在降級的主機和取代主機上執行。不過，執行個體只會在效能降級的主機上繼續執行，直到它們停止或移轉至取代主機上的保留容量為止。

如果您在降級的主機上手動停止可自動移轉的執行個體，則替換主機上為該執行個體保留的容量將會釋出並可供使用。

在排定的維護事件期間，降級主機上的執行個體會重新開機，並移轉至替換專用主機上的保留容量。遷移的執行個體會保留與降級主機上的相同屬性，包括下列項目。

- Amazon EBS 磁碟區連接
- 彈性 IP 地址
- 執行個體 ID
- 執行個體中繼資料
- 私有 IP 地址

您可以在排程維護事件啟動之前的任何時間點，在降級的主機上停止和啟動執行個體。這樣做會將您的執行個體重新啟動到另一台主機，而您的執行個體將不會進行排程維護。您必須將執行個體的主機相似性更新至將要重新啟動執行個體的新主機。如果您在維護事件起始之前停止降級主機上的所有執行個體，降級的主機會被釋放，並取消維護事件。如需詳細資訊，請參閱 [停止並啟動 Amazon EC2 執行個體](#)。

#### Note

在停止和啟動執行個體時，不會保留任何本機儲存體磁碟區上的資料。

執行個體儲存體磁碟區為根裝置的執行個體會指定的終止日期後終止。執行個體終止時，會刪除執行個體儲存體磁碟區上的任何資料。終止的執行個體會永久刪除，且無法再次啟動。對於執行個體儲存體磁碟區為根裝置的執行個體，建議使用最新的 Amazon Machine Image 在不同的專用執行個體上啟動替換執行個體，並在指定的終止日期之前將所有可用資料遷移到替換執行個體。如需詳細資訊，請參閱 [針對執行個體淘汰採取的動作](#)。

無法重新啟動的執行個體會指定日期之後自動停止。您可以在不同的主機上重新啟動這些執行個體。使用 Amazon EBS 磁碟區做為根裝置的執行個體在新主機上啟動後，會繼續使用相同的 Amazon EBS 磁碟區。

透過在 <https://console.aws.amazon.com/ec2/> 中重新排程執行個體重新啟動的開始時間，可設定執行個體重新啟動的順序。



## 設定主機維護

您可以透過 AWS Management Console 或，為所有受支援的專用主機設定主機維護 AWS CLI。如需詳細資訊，請參閱下表。

### AWS Management Console

若要使用啟用專用主機的主機維護 AWS Management Console。

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 專用執行個體。
3. 詳情「專用執行個體」 > 「動作」 > 「修改主機」。
4. 在主機維護欄位中選取開啟。

使用 AWS Management Console 停用專用執行個體的主機維護。

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 專用執行個體。
3. 詳情「專用執行個體」 > 「動作」 > 「修改主機」。
4. 在主機維護欄位中選取關閉。

使用 AWS Management Console 檢視專用執行個體的主機維護組態。

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 專用執行個體。
3. 選取「專用執行個體」，並在描述索引標籤中，檢閱主機維護欄位。

### AWS CLI

使用 AWS CLI，在配置期間啟用或停用專用執行個體的主機維護。

使用 [allocate-hosts](#) 命令。

#### 啟用

```
aws ec2 allocate-hosts --region us-east-1 --quantity 1 --instance-type m3.large --availability-zone us-east-1b --host-maintenance on
```

## 停用

```
aws ec2 allocate-hosts --region us-east-1 --quantity 1 --instance-type m3.large --availability-zone us-east-1b --host-maintenance off
```

使用 AWS CLI 啟用或停用現有專用執行個體的主機維護。

使用 [modify-hosts](#) 命令。

## 啟用

```
aws ec2 modify-hosts --region us-east-1 --host-maintenance on --host-ids h-0d123456bbf78910d
```

## 停用

```
aws ec2 modify-hosts --region us-east-1 --host-maintenance off --host-ids h-0d123456bbf78910d
```

使用 AWS CLI 檢視專用執行個體的主機維護組態。

使用 [describe-hosts](#) 命令。

```
aws ec2 describe-hosts --region us-east-1 --host-ids h-0d123456bbf78910d
```

### Note

如果停用主機維護，則會收到電子郵件通知，以移出降級主機，並在 28 天內將執行個體手動遷移到其他主機。如果您有專用執行個體預留，則會配置替代主機。28 天後，在降級主機上執行的執行個體將終止，並自動釋放主機。

## 維護事件

偵測到降級時，會在 14 天後排程維護事件，以便在新的專用執行個體上重新啟動執行個體。您會收到電子郵件通知，其中提供有關降級的主機、排程的維護事件和維護時段的詳細資訊。如需詳細資訊，請參閱[檢視排程事件](#)。

可以將維護事件重新排程為排程事件日期後七天內的任何一天。如需有關重新排程的詳細資訊，請參閱[對已排定事件進行重新排程](#)。

維護事件通常需要幾分鐘來完成。在少數發生失敗事件的情況下，您會收到電子郵件通知，以便在指定時間範圍內移出降級主機上的執行個體。

## 主機維護狀態

偵測到降級時，專用執行個體會設定為 `permanent-failure` 狀態。您無法在狀態為 `permanent-failure` 的專用執行個體上啟動執行個體。維護事件完成後，會釋放降級主機並進入 `released`、`permanent-failure` 狀態。

在偵測到專用主機上的降級情況並排定維護事件之前，主機維護會自動在您的帳戶中配置替換的專用主機。此取代主機會保持 `pending` 狀態，直到排定維護事件為止。排定維護事件之後，取代的專用主機會移至該 `available` 狀態。

在排定的維護事件發生之前，您可以使用取代的專用主機在主機上啟動新執行個體。不過，替換主機上的某些執行個體容量會保留給必須從降級主機移轉的執行個體使用。您無法在此保留容量中啟動新執行個體。如需詳細資訊，請參閱 [the section called “專用執行個體上的執行個體”](#)。

## 相關服務

專用主機與 Lic AWS ense Manager 整合 — 追蹤 Amazon EC2 專用主機上的授權 (僅在提供 L AWS icense Manager 的區域支援)。如需詳細資訊，請參閱 [AWS License Manager 使用者指南](#)。

您的 AWS 帳戶 新專用主機必須有足夠的授權。當主機在排定的維護事件完成後釋放時，會釋放與降級主機相關聯的授權。

## 定價

使用主機維護沒有額外的費用，但通常收取專用執行個體費用。如需詳細資訊，請參閱 [Amazon EC2 專用執行個體定價](#)。

啟動主機維護之後，將不再為降級專用執行個體付費。替代專用執行個體的計費方式只會在進入 `available` 狀態之後開始。

如果降級的專用主機是使用隨需費率計費，則更換的專用主機也會以隨需費率計費。如果降級的專用執行個體有作用中專用執行個體預留，則會傳輸到新專用執行個體。

## 追蹤組態變更

您可以用 AWS Config 來記錄專用主機的組態變更，以及在其上啟動、停止或終止的執行處理。接著，您就可以將 AWS Config 擷取的資訊做為授權回報的資料來源。

AWS Config 個別記錄專用主機和執行處理的組態資訊，並透過關係配對此資訊。有下列三種回報條件：

- AWS Config 錄製狀態 — 開啟時，AWS Config 正在記錄一或多個 AWS 資源類型，其中可包括專用主機和專用執行個體。為了擷取授權回報所需的資訊，請確認主機和執行個體的下列欄位都有記錄到。
- 主機記錄狀態— 為 Enabled (啟用) 時，會記錄專用執行個體的組態資訊。
- 執行個體記錄狀態 – 為 Enabled (啟用) 時，會記錄專用預留執行個體的組態資訊。

如果上述三個條件中有任一個停用，Edit Config Recording (編輯組態記錄) 按鈕的圖示就會是紅色。若要獲得這項工具的完整效益，請確認所有三個記錄方法均為啟用。所有三個項目均啟用時，圖示即為綠色。若要編輯設定，請選擇 Edit Config Recording (編輯組態記錄)。系統會將您導向至主 AWS Config 控制台中的 [設定 AWS Config] 頁面，您可以在此頁面為主機、執行個體 AWS Config 和其他支援的資源類型設定並開始錄製。如需詳細資訊，請參閱[AWS Config 開發人員指南中的 AWS Config 使用主控台](#)進行設定。

#### Note

AWS Config 在發現資源後記錄您的資源，這可能需要幾分鐘的時間。

AWS Config 開始記錄主機和執行個體的組態變更後，您可以取得已配置或釋放的任何主機，以及您已啟動、停止或終止的任何執行個體的組態歷史記錄。例如，您可以在任何時間點於專用執行個體的組態歷史記錄中，查詢該主機上啟動了多少執行個體，以及該主機上的通訊端與核心數目。對於任何這些執行個體，您也可以查詢其 Amazon Machine Image (AMI) 的 ID。您可以使用這些資訊來回報您自有伺服器固定的軟體授權 (依據每個通訊端或每個核心)。

您可以透過以下任何方式來檢視組態歷史記錄：

- 通過使用控 AWS Config 制台。對於每個記錄的資源，您可以查看時間軸頁面，其中提供組態詳細資訊的歷史記錄。若要查看此頁面，請選擇專用執行個體頁面中的 Config Timeline (組態時間軸) 欄內的灰色圖示。如需詳細資訊，請參閱[AWS Config 開發人員指南中的檢視 AWS Config 主控台](#)中的[組態詳細資料](#)。
- 通過運行 AWS CLI 命令。首先，您可以使用 `list-discovered-resources` 命令，取得所有主機和執行個體的清單。然後，您可以使用 `get-resource-config-history` 命令，取得在特定時間間隔內之主機和執行個體的組態詳細資訊。如需詳細資訊，請參閱[AWS Config 開發人員指南中的使用 CLI 檢視組態詳細資訊](#)。
- 在您的應用程式中使用 AWS Config API。首先，您可以使用「資[ListDiscovered源](#)」動作來取得所有主機和執行個體的清單。然後，您可以使用此[GetResourceConfigHistory](#)動作取得特定時間間隔內主機或執行處理的組態詳細資訊。

例如，若要從中取得所有專用主機的清單 AWS Config，請執行如下的 CLI 命令。

```
aws configservice list-discovered-resources --resource-type AWS::EC2::Host
```

若要從中取得專用主機的組態歷程記錄 AWS Config，請執行如下所示的 CLI 命令。

```
aws configservice get-resource-config-history --resource-type AWS::EC2::Instance --resource-id i-1234567890abcdef0
```

使用主控台管理 AWS Config 設定

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在 專用執行個體 頁面上，選擇 Edit Config Recording (編輯組態記錄)。
3. 在 AWS Config 主控台中，依照提供的步驟開啟錄製功能。如需詳細資訊，請參閱 [AWS Config 使用主控台進行設定](#)。

如需詳細資訊，請參閱在 [AWS Config 主控台中檢視組態詳細資料](#)。

AWS Config 使用命令列或 API 啟動

- AWS CLI：在 AWS Config 開發人員指南中 [檢視組態詳細資料 \(AWS CLI\)](#)。
- Amazon EC2 API：[GetResourceConfigHistory](#)。

## 專用執行個體

依預設，EC2 執行個體在共用的租用硬體上執行。這表示多個 AWS 帳戶可能會共用相同的實體硬體。

專用執行個體是在單一 AWS 帳戶專用的硬體上執行的 EC2 執行個體。這表示專用執行個體在主機硬體層級與屬於其他執行個體的執行個體實際隔離 AWS 帳戶，即使這些帳戶連結到單一付款人帳戶也一樣。不過，專用執行個體可能會與非專用執行個體的其他執行 AWS 帳戶 個體共用硬體。

專用執行個體不提供執行個體放置的可見性或控制權，也不支援主機相似性。如果您停止並啟動專用執行個體，則該執行個體可能無法在相同主機上執行。同樣地，您無法鎖定要啟動或執行執行個體的特定主機。此外，專用執行個體對自攜授權 (BYOL) 提供有限的支援。

如果您需要檢視和控制執行個體放置，以及更全面的 BYOL 支援，請考慮改用專用主機。專用執行個體和專用主機都可以在專用實體伺服器上啟動 Amazon EC2 執行個體。專用執行個體和專用執行個體

上的執行個體之間沒有效能、安全或實體差異。但是，它們之間存在一些關鍵差異。下表特別列出了專用執行個體與專用執行個體之間的主要差異：

|                       | Dedicated Host                       | Dedicated Instance |
|-----------------------|--------------------------------------|--------------------|
| 專屬實體伺服器               | 實體伺服器具有完全專供您使用的執行個體容量。               | 專用於單一客戶帳戶的實體伺服器。   |
| 實例容量共享                | 可與其他帳戶共用執行個體容量。                      | 不支援                |
| 計費                    | 按每台主機計費                              | 按每個執行個體計費          |
| 通訊端、核心和主機 ID 的可見度     | 提供通訊端與實體核心數目的可見度                     | 沒有可見度              |
| 主機和執行個體親和性            | 可讓您隨時間一致地將執行個體部署到相同的實體伺服器            | 不支援                |
| 訂定執行個體配置目標            | 對於執行個體在實體伺服器上的配置方式，提供額外可見性和控制權       | 不支援                |
| 自動執行個體復原功能            | 支援。如需詳細資訊，請參閱 <a href="#">主機復原</a> 。 | 支援                 |
| 使用自有授權 (BYOL)         | 支援                                   | 部分支援 *             |
| Capacity Reservations | 不支援                                  | 支援                 |

\* 透過軟體保證而實現的授權行動性的 Microsoft SQL Server 和 Windows 虛擬桌面存取 (VDA) 授權可以與專用預留執行個體一起使用。

如需專用預留執行個體的詳細資訊，請參閱 [Dedicated Hosts](#)。

## 主題

- [專用執行個體 基本概念](#)
- [支援的功能](#)

- [專用執行個體 限制](#)
- [專用執行個體定價](#)
- [使用 專用執行個體](#)

## 專用執行個體 基本概念

VPC 的租用可以是 default 或 dedicated。依預設，您的 VPC 具有 default 租用，並且在 default 租用 VPC 中啟動的執行個體具有 default 租用。可以執行下列作業來啟動專用執行個體：

- 建立具有 dedicated 租用的 VPC，以便 VPC 中的所有執行個體作為專用執行個體執行。如需詳細資訊，請參閱 [建立具有專用預留執行個體租用的 VPC](#)。
- 建立具有 default 租用的 VPC，並為執行個體手動指定 dedicated 的租用以作為專用執行個體執行。如需詳細資訊，請參閱 [在 VPC 中啟動 專用執行個體](#)。

## 支援的功能

專用執行個體支援下列功能和 AWS 服務整合：

### 主題

- [預留執行個體](#)
- [自動調整規模](#)
- [自動復原](#)
- [專用的 競價型執行個體](#)
- [爆量效能執行個體](#)

## 預留執行個體

若要保留專用執行個體的容量，您可以購買專用預留執行個體或容量保留。如需詳細資訊，請參閱 [Reserved Instances](#) 及 [On-Demand Capacity Reservations](#)。

當您購買專用的預留執行個體時，您是以更為低廉的費用購買在 VPC 中啟動專用執行個體的容量；唯有以專用租用啟動執行個體時才適用此使用費價格優惠。當您購買搭配預設租用的預留執行個體時，它僅適用於搭配 default 租用的運作中執行個體，不適用於搭配 dedicated 租用的運作中執行個體。

購買後，您就無法使用修改程序變更預留執行個體的租用。但您可以交換可轉換預留執行個體和具有不同租用的新可轉換預留執行個體。

## 自動調整規模

您可以使用 Amazon EC2 Auto Scaling 來啟動專用執行個體。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南 中的 [在 VPC 中啟動 Auto Scaling 執行個體](#)。

## 自動復原

如果專用執行個體因基礎硬體故障或需要 AWS 參與修復的問題而受損，您可以為專用執行個體設定自動復原。如需詳細資訊，請參閱 [執行個體恢復](#)。

## 專用的 競價型執行個體

您可以在建立 Spot 執行個體請求時，藉由指定 dedicated 的租用來執行專用 Spot 執行個體。如需詳細資訊，請參閱 [為 競價型執行個體 指定租用](#)。

## 爆量效能執行個體

您可以將在專用租用硬體上執行的好處運用於 [the section called “爆量效能執行個體”](#)。T3 專用執行個體預設會在無限模式下啟動，提供基準水準的 CPU 效能，並可在您的工作負載要求較高 CPU 水準時大幅提升 CPU 效能。T3 基準效能與大幅提升效能的能力，取決於 CPU 的額度。由於 T3 執行個體類型的高載本質，我們建議您監視 T3 執行個體如何使用專用硬體的 CPU 資源以獲得最佳效能。T3 專用執行個體適用於具有各種工作負載且顯示隨機 CPU 行為的客戶，但理想情況下，CPU 平均用量達到或低於基準用量。如需詳細資訊，請參閱 [the section called “重要概念”](#)。

Amazon EC2 備有系統來識別和更正效能變化。不過，如果您啟動多個具有相關 CPU 使用模式的 T3 專用執行個體，仍有可能遇到短期變化。對於這些更高要求或相關的工作負載，我們建議使用 M5 或 M5a 專用執行個體，而不是 T3 專用執行個體。

## 專用執行個體 限制

使用專用執行個體時，請謹記以下事項：

- 執行個體租用設定為 dedicated 的 VPC 不支援某些 AWS 服務或其功能。請參閱各自服務的文件以確認是否有任何限制。
- 有些執行個體類型無法在執行個體租用設定為 dedicated 的 VPC 中啟用。如需受支援之執行個體類型的詳細資訊，請參閱 [Amazon EC2 專用執行個體](#)。
- 當您啟動 Amazon EBS 支援的專用執行個體時，EBS 磁碟區不會在單一租用戶硬體上執行。



## 專用執行個體定價

專用執行個體定價與隨需執行個體定價不同。如需詳細資訊，請參閱 [Amazon EC2 專用執行個體產品頁面](#)。

## 使用 專用執行個體

您可建立執行個體租用設為 dedicated 的 VPC，以確保所有在此 VPC 內啟動的執行個體都是專用執行個體。或者，您可在啟動期間指定執行個體的租用。

### 主題

- [建立具有專用預留執行個體租用的 VPC](#)
- [在 VPC 中啟動 專用執行個體](#)
- [顯示租用資訊](#)
- [變更執行個體租用](#)
- [變更 VPC 租用](#)

### 建立具有專用預留執行個體租用的 VPC

當您建立 VPC 時，您可選擇指定其執行個體租用。如果您在具有執行個體 dedicated 租用的 VPC 中啟動執行個體，則執行個體將會一律以硬體上之專用預留執行個體 (專供您使用) 的形式執行。

如需有關建立 VPC 和選擇租用選項的詳細資訊，請參閱《Amazon VPC 使用者指南》中的 [建立 VPC](#)。


### 在 VPC 中啟動 專用執行個體

您可使用 Amazon EC2 啟動執行個體精靈啟動專用執行個體。

### Console

使用主控台在預設租用 VPC 中啟動專用執行個體

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)、Launch Instance (啟動執行個體)。
3. 在 Application and OS Images (應用程式和 OS 映像) 區段中，從清單中選取 AMI。
4. 在 Instance type (執行個體類型) 區段中，選取要啟動的執行個體類型。

 Note

請務必選擇支援為專用執行個體的執行個體類型。如需詳細資訊，請參閱 [Amazon EC2 專用執行個體](#)。

5. 在 Key pair (金鑰對) 區段中，選取要與執行個體建立關聯的金鑰對。
6. 在 Advanced details (進階詳細資訊) 區段，對於 Tenancy (租用)，選取 Dedicated (專用)。
7. 視需要設定其餘的執行個體選項。如需詳細資訊，請參閱 [使用定義的參數啟動執行個體](#)。
8. 選擇啟動執行個體。

## Command line

使用命令列在啟動期間設定執行個體的租用選項

- [run-instances](#) (AWS CLI)
- [New-EC2Instance](#) (AWS Tools for Windows PowerShell)

如需啟動具有 host 租用之執行個體的詳細資訊，請參閱 [在 專用執行個體 中啟動執行個體](#)。

## 顯示租用資訊

## Console

使用主控台顯示您 VPC 的租用資訊


1. 在 <https://console.aws.amazon.com/vpc/> 開啟 Amazon VPC 主控台。
2. 在導覽窗格中，選擇 Your VPCs (您的 VPC)。
3. 在 Tenancy (租用) 欄中檢查您 VPC 的執行個體租用。
4. 如果未顯示「租賃」欄，請選擇右上角的「設定」



開啟「租賃」，然後選擇「確認」。

使用主控台顯示您執行個體的租用資訊

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。

2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 在 Tenancy (租用) 欄中檢查您執行個體的租用。
4. 如果未顯示「租賃」欄，請執行下列其中一項操作：
  - 選擇右上角的設定  
(),  
開啟「租賃」，然後選擇「確認」。
  - 選取執行個體。在靠近頁面底部的 Details (詳細資訊) 索引標籤中，在 Host and placement group (主機和置放群組) 下，查看 Tenancy (租用) 的值。

## Command line

使用命令列描述您 VPC 的租用

- [describe-vpcs](#) (AWS CLI)
- [Get-EC2Vpc](#) (AWS Tools for Windows PowerShell)

使用命令列描述您執行個體的租用

- [describe-instances](#) (AWS CLI)
- [Get-EC2Instance](#) (AWS Tools for Windows PowerShell)

使用命令列描述預留執行個體的租用值

- [describe-reserved-instances](#) (AWS CLI)
- [Get-EC2ReservedInstance](#) (AWS Tools for Windows PowerShell)

使用命令列描述預留執行個體方案的租用值

- [describe-reserved-instances-offerings](#) (AWS CLI)
- [Get-EC2ReservedInstancesOffering](#) (AWS Tools for Windows PowerShell)

## 變更執行個體租用

可在啟動後變更已停止的執行個體的租用。您所做的變更會在執行個體下次開始時生效。

執行個體的作業系統詳細資訊 (以及是否已安裝 SQL Server) 會影響支援的轉換。如需執行個體可用租用轉換路徑的詳細資訊，請參閱《License Manager 使用指南》中的[租用轉換](#)。

#### Note

若使用 T3 執行個體，您必須在專用執行個體上啟動執行個體，才能使用 host 的租用。您無法將租用從 host 變更為 dedicated 或 default。嘗試進行上述任一不受支援的租用變更時，會顯示 InvalidRequest 錯誤代碼。

## Console

使用主控台變更執行個體的租用

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取您的執行個體。
3. 選擇 Instance state (執行個體狀態)，Stop instance (停止執行個體)，Stop (停止)。
4. 選擇 Actions (動作)、Instance settings (執行個體設定)、Modify instance placement (修改執行個體的配置)。
5. 如果是 Tenancy (租用)，請選擇是否要在專用硬體還是 專用執行個體 上執行您的執行個體。選擇 Save (儲存)。

## Command line

使用命令列修改執行個體的租用值

- [modify-instance-placement](#) (AWS CLI)
- [Edit-EC2InstancePlacement](#) (AWS Tools for Windows PowerShell)

## 變更 VPC 租用

您可在建立 VPC 的執行個體租用之後，將它從 dedicated 變更為 default。修改 VPC 的執行個體租用不影響 VPC 中任何現有執行個體的租用。下次在 VPC 中啟動執行個體時，除非您在啟動期間另行指定，否則它會有 default 租用。

**Note**

您無法在建立之後，將 VPC 的執行個體租用從 default 變更為 dedicated。

您只能使用 AWS 開發套件或 Amazon EC2 API 來 AWS CLI 修改 VPC 的執行個體租用。

**Command line**

若要使用修改 VPC 的執行個體租用屬性 AWS CLI

使用 [modify-vpc-tenancy](#) 命令並指定 VPC ID 和執行個體租用值。唯一支援的值為 default。

```
aws ec2 modify-vpc-tenancy --vpc-id vpc-1a2b3c4d --instance-tenancy default
```

## Capacity Reservations

容量保留可讓您在特定的可用區域中，為 Amazon EC2 執行個體保留運算容量。容量保留有兩種類型，用於不同的使用案例。

**容量保留類型**

- On-Demand Capacity Reservations
- ML 的容量區塊

以下是隨需容量保留的一些常見使用案例：

- 擴展事件 – 在關鍵業務事件之前建立隨需容量保留，以確保您可以在需要時進行擴展。
- 法規要求和災難復原 – 使用隨需容量保留來滿足法規要求，以達到高可用性，並在不同可用區域或區域中保留容量以進行災難復原。

以下是 ML 容量區塊的一些常見使用案例：

- 機器學習 (ML) 模型訓練和微調 – 不間斷存取保留的 GPU 執行個體，以完成 ML 模型訓練和微調。
- ML 實驗和原型 – 進行實驗並建置短時間需要 GPU 執行個體的原型。

**使用隨需容量保留的時機**

如果您有嚴格的容量需求，且正在執行需要容量保證的關鍵業務工作負載，請使用隨需容量保留。使用隨需容量保留，您可以確保只要需要時，即可存取 Amazon EC2 容量。

## 使用 ML 容量區塊的時機

如果您需要確保可在未來的設定時段內不間斷存取 GPU 執行個體，請使用 ML 容量區塊。容量區塊適合訓練和微調 ML 模型、短期實驗執行，以及處理未來暫時提高的推論需求。使用容量區塊，您可以確保在特定日期可存取 GPU 資源以執行 ML 工作負載。

## On-Demand Capacity Reservations

隨需容量預留可讓您在特定的可用區域中，為 Amazon EC2 執行個體預訂任何期限的運算容量。容量保留可降低在容量限制時無法取得隨需容量的風險。如果您有嚴格的容量需求，並且正在執行需要特定層級長期或短期容量保證的業務關鍵工作負載，建議您建立容量保留，以確保只要在需要時，即可有 Amazon EC2 容量存取權。

您可以在任何時間建立容量保留，無須綁約一年或三年，且容量可立即生效。一旦帳戶佈建了容量保留，容量就可立即使用並開始計費。當您不再需要容量保證時，請取消容量保留以釋出容量並停止產生費用。您也可以使用 Savings Plans 和區域預留執行個體提供的帳單折扣來降低容量保留的成本。

建立容量預留時，您可以指定：

- 要預留容量的可用區域
- 要預留容量的執行個體數
- 執行個體屬性，包括執行個體類型、平台、可用區域和租用

容量預留只能用於符合屬性的執行個體。根據預設，它們會自動供符合屬性的執行中執行個體使用。如果執行中的執行個體沒有一個符合容量預留的屬性，在您啟動有相符屬性的執行個體之前，它都保持不用。

## 目錄

- [容量預留、預留執行個體和 Savings Plans 之間的差異](#)
- [支援的平台](#)
- [配額](#)
- [限制](#)
- [容量預留定價和帳單](#)
- [使用容量預留](#)

- [使用 容量預留 群組](#)
- [集群放置群組中的容量保留](#)
- [Local Zones 中的容量保留](#)
- [在 Wavelength 區域中的容量保留](#)
- [容量保留 AWS Outposts](#)
- [使用共用的 容量預留](#)
- [容量預留機群](#)
- [監控容量保留](#)

容量預留、預留執行個體和 Savings Plans 之間的差異

下表反白容量預留、預留執行個體和 Savings Plans 之間的主要差異：

|        | Capacity Reservations | 區域預留執行個體                | 地區預留執行個體              | Savings Plans |
|--------|-----------------------|-------------------------|-----------------------|---------------|
| 期間     | 不綁約。可視需要建立和取消。        | 需要綁約一年或三年               |                       |               |
| 容量優勢   | 在特定的可用區域中預留的容量。       |                         | 無預留容量。                |               |
| 帳單折扣   | 無帳單折扣。†               | 提供計費折扣。                 |                       |               |
| 執行個體限制 | 適用每個區域的隨需執行個體限制。      | 每個可用區域預設為 20。您可以請求提高限制。 | 每個區域預設為 20。您可以請求提高限制。 | 沒有限制。         |

† 您可以將容量預留與 Savings Plans 或區域預留執行個體結合，以獲得折扣。

如需詳細資訊，請參閱下列內容：

- [Reserved Instances](#)
- [Savings Plans 使用者指南](#)

## 支援的平台

您必須使用正確的平台來建立容量預留，以確保它與您的執行個體正確相符。容量預留支援下列平台：

- Linux/UNIX
- Linux with SQL Server Standard
- Linux with SQL Server Web
- Linux with SQL Server Enterprise
- SUSE Linux
- Red Hat Enterprise Linux
- RHEL with SQL Server Standard
- RHEL with SQL Server Enterprise
- RHEL with SQL Server Web
- RHEL with HA
- RHEL with HA and SQL Server Standard
- RHEL with HA and SQL Server Enterprise
- Ubuntu Pro
- Windows
- 具有 SQL Server 的 Windows
- Windows with SQL Server Web
- Windows with SQL Server Standard
- Windows with SQL Server Enterprise

當您購買 容量預留 時，您必須指定代表您執行個體作業系統的平台。

- 針對 SUSE Linux 和 RHEL 發行版本 (不包括 BYOL)，您必須選擇特定平台。例如 SUSE Linux 或 Red Hat Enterprise Linux 平台。
- 針對所有其他的 Linux 發行版本 (包括 Ubuntu)，請選擇 Linux/UNIX 平台。
- 如果您使用現有的 RHEL 訂閱 (BYOL)，則必須選擇 Linux/UNIX 平台。
- 針對 Windows with SQL Standard、Windows with SQL Server Enterprise 和 Windows with SQL Server Web，您必須選擇特定平台。



- 針對所有其他 Windows 版本 (不包括未支援的 BYOL)，請選擇 Windows 平台。

## 配額

您可以預留容量的執行個體數目是以您帳戶的隨需執行個體配額為基礎。您可以預留容量的執行個體數目，為該配額所允許的數目減去已在執行的執行個體數目。

配額僅適用於執行中的執行個體。如果執行個體處於待定中、停止中、已停止或已休眠狀態，則不會計入配額中。

## 限制

在您建立容量預留之前，請記下以下的限制與法規。

- 作用中和未使用的 容量預留 都計入 隨需執行個體 限制。
- 容量保留不能從一個 AWS 帳戶轉移到另一個帳戶。不過，您可以與其他 AWS 帳戶共用容量保留。如需詳細資訊，請參閱 [使用共用的 容量預留](#)。
- 區域 預留執行個體 帳單折扣不適用於 容量預留。
- 集群放置群組中無法建立容量保留。不支援分散和分區置放群組。
- 容量預留 不能與 專用執行個體 搭配使用。容量預留不能與專用執行個體搭配使用。
- [Windows 執行個體] 容量保留不能與自攜授權 (BYOL) 搭配使用。
- 容量預留 不確保休眠執行個體可以在嘗試啟動後恢復。

## 容量預留定價和帳單

### 主題

- [定價](#)
- [帳單](#)
- [帳單折扣](#)
- [檢視您的帳單](#)

## 定價

無論您是否以預留容量執行執行個體，都需要支付按隨需費率計算的容量保留費用。如果您不使用保留，此項目在 Amazon EC2 帳單中會顯示為未使用的保留。當您執行符合保留屬性的執行個體時，您只需要支付執行個體的費用，不必支付保留費用。沒有前期或額外費用。

例如，如果您建立了 20 個 m4.large Linux 執行個體的 容量預留，在同一可用區域中執行了 15 個 m4.large Linux 執行個體，則您需要支付 15 個作用中執行個體和保留之 5 個未使用執行個體的費用。

Savings Plans 和區域預留執行個體的帳單折扣適用於容量預留。如需詳細資訊，請參閱 [帳單折扣](#)。

如需詳細資訊，請參閱 [Amazon EC2 定價](#)。

## 帳單

一旦帳戶中佈建了容量，容量保留就會立即開始計費，並在帳戶容量保留保持佈建狀態時持續計費。

容量預留按秒計費。這表示不按小時計費。例如，如果您帳戶的容量保留作用了 24 小時又 15 分鐘，您需要支付 24.25 小時的保留費用。

以下範例示範容量預留的計費方式。您建立了一個 m4.large Linux 執行個體的 容量預留，其隨需費率為小時 0.10 USD。在此範例中，該帳戶的容量保留佈建了 5 個小時。第一個小時未使用 容量預留，所以未使用的這個小時按 m4.large 執行個體類型的標準隨需費率計費。在第二至第五個小時，m4.large 執行個體佔用了 容量預留。在這段期間，容量預留 不產生任何費用，但帳戶需要支付 m4.large 執行個體佔用它的費用。第六個小時取消了 容量預留，m4.large 執行個體在預留容量外正常執行。這個小時按 m4.large 執行個體類型的隨需費率收費。

| Hour                               | 1             | 2             | 3             | 4             | 5             | 6             | Total cost    |
|------------------------------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| <b>Unused Capacity Reservation</b> | \$0.10        | \$0.00        | \$0.00        | \$0.00        | \$0.00        | \$0.00        | <b>\$0.10</b> |
| <b>On-demand Instance Usage</b>    | \$0.00        | \$0.10        | \$0.10        | \$0.10        | \$0.10        | \$0.10        | <b>\$0.50</b> |
| <b>Hourly cost</b>                 | <b>\$0.10</b> | <b>\$0.10</b> | <b>\$0.10</b> | <b>\$0.10</b> | <b>\$0.10</b> | <b>\$0.10</b> | <b>\$0.60</b> |

## 帳單折扣

Savings Plans 和區域預留執行個體的帳單折扣適用於容量保留。AWS 會自動將這些折扣套用至具有相符屬性的「產能保留」。當執行個體使用 容量預留，折扣會套用至該執行個體。在涵蓋未使用的 容量預留 之前，折扣會優先套用至執行個體用量。

用於區域 預留執行個體 的帳單折扣不適用於 容量預留。

如需詳細資訊，請參閱下列內容：

- [Reserved Instances](#)
- [Savings Plans 使用者指南](#)
- [帳單和購買選項](#)

## 檢視您的帳單

您可以在 AWS Billing and Cost Management 主機上查看帳戶的費用和費用。

- Dashboard (儀表板) 會顯示您帳戶的支出摘要。
- 在 Bills (帳單) 頁面上，於 Details (詳細資訊) 下方，展開 Elastic Compute Cloud 區段及區域，以取得容量預留的帳單資訊。

您可在線上檢視費用，或是下載 CSV 檔案。如需詳細資訊，請參閱 AWS Billing and Cost Management 使用者指南中的 [容量預留明細項目](#)。

## 使用 容量預留

若要開始使用 容量預留，您可以在所需的可用區域中建立容量保留。然後，您可以使用預留容量啟動執行個體，檢視其容量使用率即時狀況，並視需要增加或減少其容量。

根據預設，容量保留會自動比對具有相符屬性 (執行個體類型、平台、可用區域和租用) 的新執行個體和執行中執行個體。這代表任何屬性相符的執行個體會 在 容量預留 中自動執行。不過，您也可以針對特定工作負載的容量預留。這可讓您明確控制哪些執行個體可以在該預留容量中執行。

您可以指定保留的結束方式。您可以選擇取消容量預留或在指定的時間自動結束。如果指定結束時間，則容量預留會在指定時間的一小時內取消。例如，如果您指定 2019/5/31 13:30:55，則容量預留保證會在 2019/5/31 的 13:30:55 和 14:30:55 之間結束。在保留結束後，您就不能再以針對容量預留的執行個體。使用預留容量執行的執行個體會持續不間斷執行。如果針對容量預留的執行個體停止，在移除其容量預留針對的偏好設定或設定它們針對不同的容量預留之前，都無法重新啟動它們。

## 內容

- [建立 容量預留](#)
- [使用現有的 容量預留 啟動執行個體](#)
- [修改 容量預留](#)
- [修改執行個體的 容量預留 設定](#)
- [檢視 容量預留](#)
- [取消 容量預留](#)

## 建立 容量預留

如果您建立容量保留的請求成功，則可立即使用容量。只要容量預留在作用中，就有預留容量供您使用，而且您可以隨時使用它啟動執行個體。如果容量預留為開啟中，則具有相符屬性的新執行個體和現

有執行個體都會使用容量預留的容量自動執行。如果容量預留為 `targeted`，則執行個體必須特別針對它使用預留容量執行。

如果符合以下條件之一，則您建立容量預留的請求可能會失敗：

- Amazon EC2 沒有足夠的容量可滿足請求。請稍後再試、嘗試其他可用區域，或是嘗試較小的請求。如果您的應用程式能因應各種執行個體類型和大小，請嘗試不同的執行個體屬性。
- 請求的數量超過所選執行個體系列的隨需執行個體限制。請提高執行個體系列的隨需執行個體限制，然後再試一次。如需詳細資訊，請參閱 [隨需執行個體配額](#)。

## 使用主控台建立容量預留

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 選擇 容量預留，接著選擇 Create 容量預留 (建立 容量預留)。
3. 在 Create a 容量預留 (建立容量預留) 頁面的 Instance details (執行個體詳細資訊) 區段中設定以下設定。您啟動的執行個體類型、平台、可用區域和租用必須符合您在此處指定的執行個體類型、平台、可用區域和租用，否則不會套用容量保留。例如，如果開啟的 容量預留 不相符，則執行個體啟動 容量預留 明確目標將會失敗。
  - a. Instance Type (執行個體類型)—使用預留容量啟動的執行個體類型。
  - b. Launch EBS-optimized instances (啟動 EBS 最佳化執行個體) – 指定是否為 EBS 最佳化執行個體預留容量。根據預設，某些執行個體類型會選取此選項。如需詳細資訊，請參閱 [the section called “EBS 最佳化”](#)。
  - c. Platform (平台)—執行個體的作業系統。如需詳細資訊，請參閱 [支援的平台](#)。
  - d. Availability Zone (可用區域)—要預留容量的可用區域。
  - e. Tenancy (租用)—指定是要執行共用硬體執行個體 (預設) 或是專用執行個體。
  - f. (選用) Placement group ARN (置放群組 ARN) – 要在其中建立容量保留的集群放置群組的 ARN。  
  
如需詳細資訊，請參閱 [集群放置群組中的容量保留](#)。
  - g. Quantity (數量) – 要預留容量的執行個體數目。如果您指定的數量超過所選執行個體類型的剩餘隨需執行個體限制，請求會被拒絕。
4. 在 Reservation details (保留詳細資訊) 區段中設定以下設定：
  - a. Reservation Ends (保留結束) – 選擇下列其中一個選項：
    - Manually (手動) – 預留容量直到您明確取消為止。

- Specific time (特定時間) – 在指定的日期和時間自動取消容量保留。
  - b. Instance eligibility (執行個體資格) – 選擇下列其中一個選項：
    - open — (預設) 容量保留符合任何具有相符屬性 (執行個體類型、平台、可用區域和租用) 的執行個體。如果啟動具有相符屬性的執行個體，它會自動置放到預留容量。
    - target — 容量保留僅接受具有相符屬性 (執行個體類型、平台、可用區域和租用) 且明確鎖定保留項目的執行個體。
5. 選擇 Request reservation (請求保留)。

若要使用建立容量保留 AWS CLI

使用 [create-capacity-reservation](#) 命令。如需詳細資訊，請參閱 [支援的平台](#)。

下列命令會建立一個容量保留，以保留三個在us-east-1a可用區域中m5.2xlarge執行 RHEL AMI 的執行個體的容量保留。

```
aws ec2 create-capacity-reservation --instance-type m5.2xlarge --instance-platform Red Hat Enterprise Linux --availability-zone us-east-1a --instance-count 3
```

下列命令會建立容量保留，以保留三個m5.2xlarge執行 Windows 並在us-east-1a可用區域中使用 SQL Server AMI 執行 Windows 的執行個體的容量保留。

```
aws ec2 create-capacity-reservation --instance-type m5.2xlarge --instance-platform Windows with SQL Server --availability-zone us-east-1a --instance-count 3
```

使用現有的 容量預留 啟動執行個體

當您啟動執行個體時，您可以指定是否在任何 open 容量預留、特定 容量預留、或容量預留群組中啟動執行個體。您只能在具有相符屬性 (執行個體類型、平台、可用區域和租用) 和足夠容量的容量保留中啟動執行個體。或者，您可以設定執行個體以避免在容量預留中執行，即使您的 open 容量預留 有相符屬性和可用容量。

使用容量預留啟動執行個體可依啟動的執行個體數量降低其可用容量。例如，如果啟動三個執行個體，則容量預留的可用容量會減少三個。

使用主控台以現有的容量預留啟動執行個體

1. 請遵循程序[啟動執行個體](#)，但在完成下列步驟以指定置放群組和容量保留的設定之前，請勿啟動執行個體。

## 2. 展開進階詳細資料並執行下列操作：

- a. 在「放置」群組中，選取要在其中啟動執行個體的叢集置放群組。
  - b. 對於 Capacity Reservation (容量保留)，根據容量保留的組態選取下列其中一個選項：
    - 無 — 防止執行個體啟動到容量保留。執行個體以隨需容量執行。
    - 開啟 — 將執行個體啟動至任何具有相符屬性和足夠容量供您選取的執行個體數量使用的容量保留。如果沒有容量足夠的相符容量預留，執行個體將使用隨需容量。
    - 按 ID 鎖定目標 — 將執行個體啟動至選取的容量保留。如果選取的容量預留在您所選之執行個體數量沒有足夠的容量，執行個體啟動會失敗。
    - 按群組鎖定 — 使用所選容量保留群組中相符屬性和可用容量，將執行個體啟動至任何容量保留。如果所選群組沒有容量預留具有相符屬性和可用容量的，則執行個體會啟動至隨需容量。
3. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。如需詳細資訊，請參閱 [使用新的啟動執行個體精靈啟動執行個體](#)。

### 使用將執行個體啟動至現有容量保留 AWS CLI

使用 [run-instances](#) 命令，並指定 `--capacity-reservation-specification` 參數。

以下範例會使用具有相符屬性和可用容量的任何開啟中容量預留，來啟動 t2.micro 執行個體：

```
aws ec2 run-instances --image-id ami-abc12345 --count 1 --instance-type t2.micro
--key-name MyKeyPair --subnet-id subnet-1234567890abcdef1 --capacity-reservation-
specification CapacityReservationPreference=open
```

以下範例會使用 targeted 容量預留 啟動 t2.micro 執行個體：

```
aws ec2 run-instances --image-id ami-abc12345 --count 1 --instance-type t2.micro
--key-name MyKeyPair --subnet-id subnet-1234567890abcdef1 --capacity-reservation-
specification CapacityReservationTarget={CapacityReservationId=cr-a1234567}
```

以下範例會在 容量預留 群組啟動 t2.micro 執行個體：

```
aws ec2 run-instances --image-id ami-abc12345 --count 1
--instance-type t2.micro --key-name MyKeyPair --subnet-
id subnet-1234567890abcdef1 --capacity-reservation-specification
```

```
CapacityReservationTarget={CapacityReservationResourceGroupArn=arn:aws:resource-groups:us-west-1:123456789012:group/my-cr-group}
```

## 修改 容量預留

您可以在建立後變更啟用 容量預留 的屬性。您不能在容量預留過期後或明確取消後修改它。

修改容量預留時，您只能增加或減少數量以及變更其發佈的方式。您不能變更容量預留的執行個體類型、EBS 最佳化、平台、可用區域或執行個體容量保留資格。如果您需要修改這些屬性的任何一項，建議您取消保留，然後建立具有所需屬性的新保留。

如果您指定的新數量超過所選執行個體類型的剩餘隨需執行個體限制，更新會失敗。

### 使用主控台修改容量預留

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 選取 容量預留 (容量預留)，選取要修改的容量預留，然後選取 Edit (編輯)。
3. 視需要修改 Quantity (數量) 或 Reservation ends (保留結束) 選項，然後選擇 Save changes (儲存變更)。

### 若要使用修改容量保留 AWS CLI

使用 [modify-capacity-reservation](#) 命令：

例如，下列命令會修改 容量預留 以保留八個執行個體的容量。

```
aws ec2 modify-capacity-reservation --capacity-reservation-id cr-1234567890abcdef0 --instance-count 8
```

## 修改執行個體的 容量預留 設定

您可以隨時修改已停止執行個體的 容量預留 設定：

- 從任何具有相符屬性 (執行個體類型、平台、可用區域和租用) 和可用容量的容量保留開始。
- 在特定容量預留中啟動執行個體。
- 在 Capacity Reservation (容量預留) 群組內具有相符屬性和可用容量的任何 Capacity Reservation (容量預留) 中啟動
- 防止在容量預留中開始執行個體。

## 使用主控台修改執行個體的容量預留設定

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 選取 Instances (執行個體)，並選取要修改的執行個體。停止執行個體 (若還未停止的話)。
3. 選擇動作、執行個體設定、修改容量保留設定。
4. 針對 容量預留，請選擇下列其中一種選項：
  - Open (開啟) — 使用具有相符屬性以及針對所選執行個體數量具有足夠容量的容量預留來啟動執行個體。如果沒有容量足夠的相符容量預留，執行個體將使用隨需容量。
  - None (無) — 防止使用容量預留啟動執行個體。執行個體以隨需容量執行。
  - Specify Capacity Reservation (指定容量保留) – 在選取的容量保留中啟動執行個體。如果選取的容量預留在您所選之執行個體數量沒有足夠的容量，執行個體啟動會失敗。
  - Specify Capacity Reservation group (指定容量保留群組) – 在選取的容量保留群組內具有相符屬性和可用容量的任何容量保留中啟動執行個體。如果所選群組沒有 容量預留 具有相符屬性和可用容量的，則執行個體會啟動至隨需容量。

## 若要修改執行個體的容量保留設定 AWS CLI

使用 [modify-instance-capacity-reservation-attributes](#) 命令。

例如，下列命令會將執行個體的 容量預留 設定變更為 open 或 none。

```
aws ec2 modify-instance-capacity-reservation-attributes --instance-id i-1234567890abcdef0 --capacity-reservation-specification CapacityReservationPreference=none | open
```

例如，下列命令會修改執行個體，以將特定 容量預留 鎖定為目標。

```
aws ec2 modify-instance-capacity-reservation-attributes --instance-id i-1234567890abcdef0 --capacity-reservation-specification CapacityReservationTarget={CapacityReservationId=cr-1234567890abcdef0}
```

例如，下列命令會修改執行個體，以將特定 容量預留 群組鎖定為目標。

```
aws ec2 modify-instance-capacity-reservation-attributes --instance-id i-1234567890abcdef0 --capacity-reservation-specification CapacityReservationTarget={CapacityReservationResourceGroupArn=arn:aws:resource-groups:us-west-1:123456789012:group/my-cr-group}
```



## 檢視 容量預留

容量預留 擁有以下可能的狀態：

- **active**—容量可供使用。
- **expired** – 容量保留在保留請求指定的日期和時間自動過期。預留容量將不再提供使用。
- **cancelled** – 容量保留已取消。預留容量將不再提供使用。
- **pending** – 容量保留請求成功，但容量佈建仍待定。
- **failed** – 容量保留請求失敗。請求會因為請求參數無效、容量限制或執行個體限制而失敗。失敗的請求可供您檢視 60 分鐘。

### Note

由於 Amazon EC2 API 遵守**最終一致性**模式，因此在建立容量保留後，主控台和 [describe-capacity-reservations](#) 回應最長需 5 分鐘的時間才能表示容量保留處於 active 狀態。在這段期間，主控台和 `describe-capacity-reservations` 回應可能表示容量保留處於 pending 狀態。但是，容量保留可能已經可用，您可以嘗試在其中啟動執行個體。

使用主控台檢視您的容量預留

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 選取 容量預留 (容量預留)，然後選取要檢視的容量預留。
3. 選擇 View launched instances for this reservation (檢視用於此保留的啟動執行個體)。

若要使用檢視您的「容量保留」AWS CLI

使用 [describe-capacity-reservations](#) 命令：

例如，下面的命令描述所有 容量預留。

```
aws ec2 describe-capacity-reservations
```

範例輸出。

```
{
```

```

"CapacityReservations": [
 {
 "CapacityReservationId": "cr-1234abcd56EXAMPLE ",
 "EndDateType": "unlimited",
 "AvailabilityZone": "eu-west-1a",
 "InstanceMatchCriteria": "open",
 "Tags": [],
 "EphemeralStorage": false,
 "CreateDate": "2019-08-16T09:03:18.000Z",
 "AvailableInstanceCount": 1,
 "InstancePlatform": "Linux/UNIX",
 "TotalInstanceCount": 1,
 "State": "active",
 "Tenancy": "default",
 "EbsOptimized": true,
 "InstanceType": "a1.medium",
 "PlacementGroupArn": "arn:aws:ec2:us-east-1:123456789012:placement-group/
MyPG"
 },
 {
 "CapacityReservationId": "cr-abcdEXAMPLE9876ef ",
 "EndDateType": "unlimited",
 "AvailabilityZone": "eu-west-1a",
 "InstanceMatchCriteria": "open",
 "Tags": [],
 "EphemeralStorage": false,
 "CreateDate": "2019-08-07T11:34:19.000Z",
 "AvailableInstanceCount": 3,
 "InstancePlatform": "Linux/UNIX",
 "TotalInstanceCount": 3,
 "State": "cancelled",
 "Tenancy": "default",
 "EbsOptimized": true,
 "InstanceType": "m5.large"
 }
]
}

```

## 取消 容量預留

如果不再需要預留容量，您可以隨時取消容量預留。取消容量預留時，會立即釋出容量，而且不會再預留給您使用。

您可以取消空的容量預留和有執行中之執行個體的容量預留。如果您取消有執行中之執行個體的容量預留，執行個體會繼續在容量保留以外正常執行，依標準 隨需執行個體 費率計價，如有作用中相符的 Savings Plan 或區域預留執行個體，則依折扣費率計價。

在您取消容量預留後，即不再啟動以其為目標的執行個體。修改這些執行個體，讓它們針對不同的容量預留、使用具相符屬性及足夠容量的任何開啟中容量預留啟動，或避免使用容量預留啟動。如需詳細資訊，請參閱 [修改執行個體的 容量預留 設定](#)。

### 使用主控台取消容量預留

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 選取容量預留 (容量預留)，然後選取要取消的容量預留。
3. 選擇 Cancel reservation (取消保留) 和 Cancel reservation (取消保留)。

### 若要使用取消容量保留 AWS CLI

使用 [cancel-capacity-reservation](#) 命令：

例如，下列命令會取消有 cr-1234567890abcdef0 ID 的 容量預留。

```
aws ec2 cancel-capacity-reservation --capacity-reservation-id cr-1234567890abcdef0
```

### 使用 容量預留 群組

您可以使用 AWS Resource Groups 來建立容量保留的邏輯集合，稱為資源群組。資源群組是全部位於相同 AWS 區域中的 AWS 資源的邏輯群組。如需資源群組的詳細資訊，請參閱 AWS Resource Groups 使用者指南中的 [什麼是資源群組？](#)。

您可以包含您在帳戶中擁有的「產能保留」，以及單一資源群組中其他 AWS 帳號與您共用的「產能保留」。您也可以包含在單一資源群組中包含具有不同屬性 (執行個體類型、平台、可用區域和租用) 的容量保留。

為容量保留建立資源群組時，可以選定一個容量保留群組做為容量保留的目標，不要選擇個別的容量保留。以容量保留群組為目標的執行個體，符合群組中具有相符屬性 (執行個體類型、平台、可用區域和租用) 和可用容量的任何容量保留。如果群組沒有具有相符屬性和可用容量的，則執行個體 容量預留 會使用隨需容量執行。如果稍後階段將相符的 容量預留 新增至目標群組，則執行個體會自動比對並移入其預留容量。

若要避免在群組中意外使用 容量預留，請將群組中的 容量預留 設定為僅接受明確以容量保留為目標的執行個體。若要這樣做，請在使用 Amazon EC2 主控台建立 容量預留 時，將執行個體資格設定為已

鎖定目標 (舊主控台) 或僅指定此保留的執行個體 (新主控台)。使用時 AWS CLI，請在建立容量保留 `--instance-match-criteria targeted` 時指定。這樣做可確保只有明確鎖定目標群組或群組中的容量預留 執行個體才能在群組中執行。

如果群組中的 容量預留 在其有執行中的執行個體時被取消或過期，則這些執行個體會自動移至群組中具有相符屬性和可用容量的另一個 容量預留。如果群組中剩餘的 容量預留 沒有相符的屬性和可用容量，則執行個體會以隨需容量執行。如果稍後階段將相符的 容量預留 新增至目標群組，則執行個體會自動移至其預留容量中。

## 主題

- [建立容量保留群組](#)
- [在群組新增容量保留](#)
- [檢視群組中的容量保留](#)
- [檢視容量保留所屬的群組](#)
- [從群組移除容量保留](#)
- [刪除容量保留群組](#)

## 建立容量保留群組

### 為容量保留建立群組

使用「[建立群組](#)」指令 AWS CLI。在 `name` 提供群組的描述性名稱，並在 `configuration` 指定兩個 `Type` 請求參數：

- `AWS::EC2::CapacityReservationPool` 可確保將資源群組鎖定為執行個體啟動的目標
- `AWS::ResourceGroups::Generic` 將 `allowed-resource-types` 設為 `AWS::EC2::CapacityReservation` 確保資源群組僅接受「容量預留」

例如，以下命令會建立名為 `MyCRGroup` 的群組。

```
aws resource-groups create-group --name MyCRGroup --configuration
'{"Type":"AWS::EC2::CapacityReservationPool"}'
'{"Type":"AWS::ResourceGroups::Generic", "Parameters": [{"Name": "allowed-resource-
types", "Values": ["AWS::EC2::CapacityReservation"]}]]'
```

下面顯示了範例輸出。

```
{
 "GroupConfiguration": {
 "Status": "UPDATE_COMPLETE",
 "Configuration": [
 {
 "Type": "AWS::EC2::CapacityReservationPool"
 },
 {
 "Type": "AWS::ResourceGroups::Generic",
 "Parameters": [
 {
 "Values": [
 "AWS::EC2::CapacityReservation"
],
 "Name": "allowed-resource-types"
 }
]
 }
]
 },
 "Group": {
 "GroupArn": "arn:aws:resource-groups:sa-east-1:123456789012:group/MyCRGroup",
 "Name": "MyCRGroup"
 }
}
```

## 在群組新增容量保留

若將與您共用的容量保留新增到群組，當該容量保留取消共用，該容量保留會自動從群組中移除。

## 在群組新增容量保留

使用 [group-resources](#) AWS CLI 命令。在 `group` 指定要新增 容量預留 的群組名稱，並在 `resources` 指定要新增 容量預留 的 ARN。若要新增多個 容量預留，請使用空格分隔 ARN。若要取得要新增之容量保留的 ARN，請使用 [描述-容量-保留指 AWS CLI 令並指定容量保留的識別碼](#)。

例如，下列命令會在名為 `MyCRGroup` 的群組中新增兩個 容量預留。

```
aws resource-groups group-resources --group MyCRGroup --resource-arns arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-1234567890abcdef1 arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-54321abcdef567890
```

下面顯示了範例輸出。

```
{
 "Failed": [],
 "Succeeded": [
 "arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-1234567890abcdef1",
 "arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-54321abcdef567890"
]
}
```

## 檢視群組中的容量保留

### 檢視特定群組中的 容量預留

使用[列表組資源](#) AWS CLI 命令。對於 `group`，指定群組的名稱。

例如，下列命令會在名為 `MyCRGroup` 的群組列出 容量預留。

```
aws resource-groups list-group-resources --group MyCRGroup
```

下面顯示了範例輸出。

```
{
 "QueryErrors": [],
 "ResourceIdentifiers": [
 {
 "ResourceType": "AWS::EC2::CapacityReservation",
 "ResourceArn": "arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-1234567890abcdef1"
 },
 {
 "ResourceType": "AWS::EC2::CapacityReservation",
 "ResourceArn": "arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-54321abcdef567890"
 }
]
}
```

### Note

畫面會列出您擁有的容量保留和與您共用的容量保留。

## 檢視容量保留所屬的群組

### AWS CLI

檢視已新增特定容量保留的群組

使用 [get-groups-for-capacity-reservation](#) AWS CLI 命令。

例如，下列指令會列出已新增 容量預留 `cr-1234567890abcdef1` 的群組。

```
aws ec2 get-groups-for-capacity-reservation --capacity-reservation-id cr-1234567890abcdef1
```

下面顯示了範例輸出。

```
{
 "CapacityReservationGroups": [
 {
 "OwnerId": "123456789012",
 "GroupArn": "arn:aws:resource-groups:sa-east-1:123456789012:group/MyCRGroup"
 }
]
}
```

#### Note

若指定與您共用的容量保留，命令只會傳回您擁有的容量保留群組。

### Amazon EC2 console

檢視已新增特定容量保留的群組

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選取 容量預留，選取要檢視的 容量預留，然後選取檢視。

已新增 容量預留 的群組會列在群組卡片中。

**Note**

若選擇與您共用的容量保留，主控台只會顯示您擁有的容量保留群組。

## 從群組移除容量保留

### 從群組中移除 容量預留

使用取[消分組](#) AWS CLI 資源命令。在 `group` 指定要從中移除 容量預留 的群組 ARN，並在 `resources` 指定要移除的 容量預留 的 ARN。若要移除多個 容量預留，請使用空格分隔 ARN。

下列範例會從名為 `MyCRGroup` 的群組中移除兩個 容量預留。

```
aws resource-groups ungroup-resources --group MyCRGroup --resource-arns arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-0e154d26a16094dd arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-54321abcdef567890
```

下面顯示了範例輸出。

```
{
 "Failed": [],
 "Succeeded": [
 "arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-0e154d26a16094dd",
 "arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-54321abcdef567890"
]
}
```

## 刪除容量保留群組

### 刪除群組

使用[刪除群組命令](#) AWS CLI。在 `group` 提供要刪除的群組名稱。

例如，以下命令會刪除名為 `MyCRGroup` 的檔案。

```
aws resource-groups delete-group --group MyCRGroup
```

下面顯示了範例輸出。



```
{
 "Group": {
 "GroupArn": "arn:aws:resource-groups:sa-east-1:123456789012:group/MyCRGroup",
 "Name": "MyCRGroup"
 }
}
```

## 集群放置群組中的容量保留

您可以在集群放置群組中建立容量保留，以便為您的工作負載預留 Amazon EC2 運算容量。集群放置群組具有低網路延遲和高網路輸送量的優點。

在集群放置群組中建立容量保留，可確保在需要時可隨時存取集群放置群組中的運算容量。這非常適合為需要運算擴展的高效能 (HPC) 工作負載預留容量。它可讓您縮小叢集的規模，同時確保容量仍然可供您使用，以便在需要時進行擴展。

### 主題

- [限制](#)
- [使用集群放置群組中的容量保留](#)

### 限制

在集群放置群組中建立容量保留時，請注意下列事項：

- 如果現有的容量保留不在放置群組中，您無法修改容量保留以保留放置群組中的容量。若要預留置放群組中的容量，您必須在置放群組中建立容量保留。
- 在置放群組中建立容量保留後，您不能將其修改為置放群組之外的預留容量。
- 您可以透過修改置放群組中的現有容量保留或在置放群組中建立附加容量保留來增加置放群組中的預留容量。但是，這會增加出現容量不足錯誤的可能性。
- 您不能分享已在集群放置群組中建立的容量保留。
- 您無法刪除具有 active 容量預留的集群放置群組。您必須取消您在集群放置群組中建立的所有容量保留之後，才可將其刪除。

## 使用集群放置群組中的容量保留

要開始對集群放置群組使用容量保留，請執行以下步驟。

**Note**

如果要在現有集群放置群組中建立容量保留，請跳過步驟 1。然後，對於步驟 2 和 3，指定現有集群放置群組的 ARN。如需如何尋找現有叢集置放群組之 ARN 的相關資訊，請參閱[檢視放置群組資訊](#)。

**主題**

- [步驟 1：\(有條件\) 建立與容量保留搭配使用的集群放置群組](#)
- [步驟 2：在集群放置群組中建立容量保留](#)
- [步驟 3：在集群放置群組中啟動執行個體](#)

**步驟 1：(有條件) 建立與容量保留搭配使用的集群放置群組**

僅當您需要建立新的集群放置群組時，才執行此步驟。要使用現有的集群放置群組，請跳過此步驟，然後對於步驟 2 和 3，請使用該集群放置群組的 ARN。如需如何尋找現有叢集置放群組之 ARN 的相關資訊，請參閱[檢視放置群組資訊](#)。

您可以使用下列其中一種方法來建立集群放置群組。

**Console****使用主控台來建立集群放置群組**

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Placement Groups (置放群組)，然後選擇 Create placement group (建立置放群組)。
3. 對於 Name (名稱)，指定置放群組的描述性名稱。
4. 對於 Placement strategy (置放策略)，選擇 Cluster (叢集)。
5. 選擇 Create group (建立群組)。
6. 在「放置群組」表格的「群組 ARN」資料欄中，記下您所建立之叢集置放群組的 ARN。您將需要它來進行下一步。

**AWS CLI****使用建立叢集置放群組 AWS CLI**

使用 `create-placement-group` 命令。對於 `--group-name`，指定置放群組的描述性名稱，然後為 `--strategy` 指定 `cluster`。

下列範例會建立一個名為 MyPG 且使用 `cluster` 置放策略的置放群組。

```
aws ec2 create-placement-group \
 --group-name MyPG \
 --strategy cluster
```

記下命令輸出中傳回的置放群組 ARN，因為下一個步驟會需要此 ARN。

## 步驟 2：在集群放置群組中建立容量保留

在集群放置群組中建立容量保留的方式與建立任何容量保留的方式相同。但是，您還必須指定要在其中建立容量保留的集群放置群組的 ARN。如需詳細資訊，請參閱 [建立容量預留](#)。

### 考量事項

- 指定的集群放置群組必須處於 `available` 狀態。如果集群放置群組處於 `pending`、`deleting` 或 `deleted` 狀態，請求將失敗。
- 容量保留和集群放置群組必須位於相同的可用區域中。如果建立容量保留請求指定的可用區域不同於集群放置群組的可用區域，則請求將失敗。
- 您只能為集群放置群組支援的執行個體類型建立容量保留。若您指定不支援的執行個體類型，請求就會失敗。如需詳細資訊，請參閱 [集群放置群組的規則與限制](#)。
- 如果您在集群放置群組中建立 `open` 容量保留，並且存在具有相符屬性 (置放群組 ARN、執行個體類型、可用區域、平台和租期) 的現有執行中執行個體，這些執行個體就會自動在容量保留中執行。
- 如果符合以下條件之一，則您建立容量預留的請求可能會失敗：
  - Amazon EC2 沒有足夠的容量可滿足請求。請稍後再試、嘗試其他可用區域，或是嘗試較小的容量。如果您的工作負載因應各種執行個體類型和大小，請嘗試不同的執行個體屬性。
  - 請求的數量超過所選執行個體系列的隨需執行個體限制。請提高執行個體系列的隨需執行個體限制，然後再試一次。如需詳細資訊，請參閱 [隨需執行個體配額](#)。

您可以使用下列其中一種方法來在集群放置群組中建立容量保留。

## Console

### 使用主控台建立容量預留

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 選擇 容量預留，接著選擇 Create 容量預留 (建立 容量預留)。
3. 在 [建立容量保留] 頁面上，視需要指定執行個體類型、平台、可用區域、租用、數量和結束日期。
4. 在放置群組中，選取要在其中建立容量保留的叢集置放群組的 ARN。
5. 選擇建立。

如需詳細資訊，請參閱 [建立 容量預留](#)。

## AWS CLI

### 若要使用建立容量保留 AWS CLI

使用 [create-capacity-reservation](#) 命令。對於 `--placement-group-arn`，指定要在其中建立容量保留的叢集置放群組的 ARN。

```
$ aws ec2 create-capacity-reservation \
 --instance-type instance_type \
 --instance-platform platform \
 --availability-zone az \
 --instance-count quantity \
 --placement-group-arn placement_group_ARN
```

如需詳細資訊，請參閱 [建立 容量預留](#)。

### 步驟 3：在叢集置放群組中啟動執行個體

在叢集置放群組中的容量保留內啟動執行個體，其方式與在任何容量保留中啟動執行個體相同。但是，您還必須指定要在其中啟動執行個體的叢集置放群組的 ARN。如需詳細資訊，請參閱 [建立 容量預留](#)。

### 考量事項

- 如果容量保留為 open，您無需在執行個體啟動請求中指定容量保留。如果執行個體具備與指定置放群組中的容量保留相符的屬性 (置放群組 ARN、執行個體類型、可用區域、平台和租期)，則執行個體將自動在容量保留中執行。

- 如果容量保留僅接受目標執行個體啟動，則除了請求中的集群放置群組之外，還必須指定目標容量保留。
- 如果容量保留位於容量保留群組中，則除了請求中的集群放置群組之外，還必須指定目標容量保留群組。如需詳細資訊，請參閱 [使用 容量預留 群組](#)。

您可以使用下列其中一種方法，將執行個體啟動至集群放置群組中的容量保留。

## Console

### 使用主控台以現有的容量預留啟動執行個體

1. 請遵循程序 [啟動執行個體](#)，但在完成下列步驟以指定置放群組和容量保留的設定之前，請勿啟動執行個體。
2. 展開進階詳細資料並執行下列操作：
  - a. 在「放置」群組中，選取要在其中啟動執行個體的叢集置放群組。
  - b. 對於 Capacity Reservation (容量保留)，根據容量保留的組態選取下列其中一個選項：
    - 開啟 — 將執行個體啟動至叢集置放群組中具有相符屬性和足夠容量的任何 open 容量保留。
    - 按 ID 鎖定 — 將執行個體啟動至僅接受目標執行個體啟動的容量保留。
    - 按群組鎖定 — 在選取的容量保留群組中，將執行個體啟動至具有相符屬性和可用容量的任何容量保留。
3. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。如需詳細資訊，請參閱 [使用新的啟動執行個體精靈啟動執行個體](#)。

如需詳細資訊，請參閱 [使用現有的 容量預留 啟動執行個體](#)。

## AWS CLI

### 使用將執行個體啟動至現有的容量保留 AWS CLI

使用 [run-instances](#) 命令。如果您需要針對特定容量保留或容量保留群組，請指定 `--capacity-reservation-specification` 參數。對於 `--placement`，指定 `GroupName` 參數，然後指定您在上述步驟建立的置放群組的名稱。

以下命令將執行個體啟動到集群放置群組中的 `targeted` 容量保留內。

```
$ aws ec2 run-instances \
```

```
--image-id ami_id \
--count quantity \
--instance-type instance_type \
--key-name key_pair_name \
--subnet-id subnetid \
--capacity-reservation-specification
CapacityReservationTarget={CapacityReservationId=capacity_reservation_id} \
--placement "GroupName=cluster_placement_group_name"
```

如需詳細資訊，請參閱 [使用現有的 容量預留 啟動執行個體](#)。

## Local Zones 中的容量保留

本地區域是地理位置靠近您的用戶的 AWS 區域的擴展。在 Local Zone 中建立的資源可以為本機使用者提供非常低延遲的通訊服務。如需詳細資訊，請參閱 [AWS Local Zones](#)。

您可以在該本機 AWS 區域中建立新的子網路，將 VPC 從其父區域延伸到本機區域。當您在 Local Zone 中建立子網時，VPC 也會擴展到該 Local Zone。Local Zone 中的子網與 VPC 中的其他子網的運作方式相同。

透過使用 Local Zones，您可以將容量預留放置在更接近使用者的多個位置。您在 Local Zones 中建立和使用容量預留的方式，與在一般可用區域中建立和使用容量預留的方式相同。適用相同的功能和執行個體比對行為。如需 Local Zones 中支援的定價模型的詳細資訊，請參閱 [AWS Local Zones 常見問答集](#)。

### 考量事項

您無法在 Local Zone 中使用容量預留群組。

若要在 Local Zone 中使用容量預留

1. 啟用本地區域以在您的 AWS 帳戶中使用。如需詳細資訊，請參閱 [選擇加入 Local Zones](#)。
2. 在 Local Zone 中建立容量預留。對於 Availability Zone (可用區域)，請選擇 Local Zone。例如，「本地區域」由「地 AWS 區」代碼表示，後面接著指示位置的識別碼 us-west-2-lax-1a。如需詳細資訊，請參閱 [建立 容量預留](#)。
3. 在 Local Zone 中建立子網。對於 Availability Zone (可用區域)，請選擇 Local Zone。如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的 [在您的 VPC 中建立子網路](#)。
4. 啟動執行個體。對於子網，選擇 Local Zone 中的子網 (例如 subnet-123abc | us-west-2-lax-1a)，對於容量預留，請選擇您在 Local Zone 中建立容量預留所必需的規格 (open 或透過 ID 將其鎖定為目標)。如需詳細資訊，請參閱 [使用現有的 容量預留 啟動執行個體](#)。

## 在 Wavelength 區域中的容量保留

AWS Wavelength 可讓開發人員建立提供極低延遲的應用程式給行動裝置與最終使用者。Wavelength 將標準的 AWS 運算與儲存服務部署至電信業者 5G 網路的邊緣。您可將 Amazon Virtual Private Cloud (VPC) 擴充至一或多個 Wavelength 區域。然後，您可以使用 Amazon EC2 執行個體等 AWS 資源來執行需要超低延遲和區域 AWS 服務連線的應用程式。如需詳細資訊，請參閱[AWS Wavelength 區域](#)。

當您建立隨需容量預留時，可以選擇 Wavelength 區域，並透過指定與 Wavelength 區域相關聯的子網，在 Wavelength 區域中對容量預留啟動執行個體。Wavelength 區域是以一個 AWS 區域代碼加上一個識別符來表示，用以代表位置，例如 us-east-1-wl1-bos-wlz-1。

並非每個區域都有 Wavelength 區域。如需支援 Wavelength 區域的區域相關資訊，請參閱 AWS Wavelength 開發人員指南中的[可用 Wavelength 區域](#)。

### 考量事項

您無法使用在 Wavelength 區域的容量預留群組。

### 在 Wavelength 區域中使用容量預留

1. 啟用 Wavelength 區域以便在您的 AWS 帳戶中使用。如需詳細資訊，請參閱 [the section called “啟用 Wavelength 區域”](#)。
2. 在 Wavelength 區域中建立容量預留。對於可用區域，請選擇 Wavelength。例 us-east-1-wl1-bos-wlz-1 如，「Wavelength」由「AWS 區域」代碼表示，後面接著指示位置的識別碼。如需詳細資訊，請參閱 [建立容量預留](#)。
3. 在 Wavelength 區域中建立子網。對於可用區域，請選擇 Wavelength 區域。如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的[在您的 VPC 中建立子網路](#)。
4. 啟動執行個體。在子網中，選擇在 Wavelength 區域中的子網 (例如 subnet-123abc | us-east-1-wl1-bos-wlz-1)，對於容量預留，請選擇規格 (open 或透過 ID 將其鎖定為目標)，這對於您在 Wavelength 中建立的容量預留為必要條件。如需詳細資訊，請參閱 [使用現有的容量預留 啟動執行個體](#)。

### 容量保留 AWS Outposts

AWS Outposts 是一項全受管服務，可將 AWS 基礎架構、服務、API 和工具延伸至客戶場所。透過提供 AWS 受管理基礎架構的本機存取權，AWS Outposts 讓客戶能夠使用與 AWS 區域相同的程式設計介面在內部部署建置和執行應用程式，同時使用本機運算和儲存資源來降低延遲和本機資料處理需求。

Outpost 是部署在客戶站點的 AWS 計算和儲存容量集區。AWS 作為 AWS 區域的一部分來操作、監控和管理此容量。

您可以在帳戶中建立的 Outposts 上建立容量保留。這可讓您在站點的 Outpost 上預留運算容量。您在 Outposts 上建立和使用容量預留的方式，與在一般可用區域中建立和使用容量預留的方式相同。適用相同的功能和執行個體比對行為。

您也可以使用與組織內的其他 AWS 帳戶共用 AWS Resource Access Manager Outposts 上的容量保留。如需共用容量保留的資訊，請參閱 [使用共用的 容量預留](#)。

## 先決條件

您的站點必須安裝 Outpost。如需詳細資訊，請參閱 AWS Outposts 使用者指南中的 [建立 Outpost 並訂購 Outpost 容量](#)。

## 考量

- 您無法在 Outpost 上使用容量預留群組。

若要在 Outpost 上使用容量預留

1. 在 Outpost 上建立子網。如需詳細資訊，請參閱 AWS Outposts 使用者指南中的 [建立子網](#)。
2. 在 Outpost 上建立容量保留。
  - a. 開啟主 AWS Outposts 控制台，網址為 <https://console.aws.amazon.com/outposts/>。
  - b. 在導覽窗格中，選擇 Outposts，然後選擇 Actions (動作)、Create Capacity Reservation (建立容量預留)。
  - c. 視需要設定容量預留，然後選擇 Create (建立)。如需詳細資訊，請參閱 [建立 容量預留](#)。

### Note

Instance Type (執行個體類型) 下拉式清單只會列出選取的 Outpost 支援的執行個體類型，而 Availability Zone (可用區域) 下拉式清單只會列出與選取的 Outpost 相關聯的可用區域。

3. 在容量預留中啟動執行個體。對於子網，選取您在步驟 1 中建立的子網，對於容量預留，選取您在步驟 2 中建立的容量預留。如需詳細資訊，請參閱 AWS Outposts 使用者指南中的 [在 Outpost 上啟動執行個體](#)。



## 使用共用的 容量預留

容量保留共用可讓容量保留擁有者與其他 AWS 帳戶或 AWS 組織內共用其保留容量。這可讓您集中建立和管理容量保留，並在多個 AWS 帳戶或 AWS 組織內共用保留容量。

在此模型中，擁有容量保留 (擁有者) 的 AWS 帳戶會與其他 AWS 帳戶 (取用者) 共用該帳戶。消費者可以在與他們共用的 容量預留 中啟動執行個體，就好像他們在自己的帳戶中擁有的 容量預留 中啟動執行個體一樣。容量預留擁有者負責管理 容量預留和他們在其中啟動的執行個體。擁有者在他們共用的 容量預留中無法修改消費者啟動的執行個體。消費者在分享給他們的 容量預留中負責管理他們啟動的執行個體。消費者無法檢視或修改其他消費者或 容量預留擁有者所擁有的執行個體。

容量預留擁有者可以將 容量預留 分享給下列對象：

- AWS 組織內部或外部的特定 AWS 帳戶
- 組織內的組 AWS 織單位
- 它的整個 AWS 組織

## 目錄

- [共用容量預留的先決條件](#)
- [相關服務](#)
- [跨可用區域共用](#)
- [共用 容量預留](#)
- [停止共用容量預留](#)
- [識別並檢視共享的容量保留](#)
- [檢視共用的 容量預留 用量](#)
- [共用的容量預留許可](#)
- [計費和計量](#)
- [執行個體限制](#)

## 共用容量預留的先決條件

- 若要共用容量保留，您必須在 AWS 帳戶中擁有該容量保留。您無法將已分享給您的 容量預留 再分享出去。
- 針對租用執行個體您只能共用 容量預留。針對專用租用執行個體您無法共用 容量預留。

- 容量保留共用不適用於 AWS 帳單歷史記錄有限的新 AWS 帳戶或帳戶。
- 若要與您的組織或 AWS 組織中的組織單位共用容量保留，您 AWS 必須啟用與共用 AWS Organizations。如需詳細資訊，請參閱《AWS RAM 使用者指南》中的[透過 AWS Organizations 啟用共用](#)。

## 相關服務

容量保留共用與 AWS Resource Access Manager (AWS RAM) 整合。AWS RAM 是一項服務，可讓您與任何 AWS 帳戶或透過其他帳戶共用 AWS 資源 AWS Organizations。您可以透過建立資源共用 AWS RAM 來共用您所擁有的資源。資源共享指定要共用的資源，以及共用它們的消費者。消費者可以是個人 AWS 帳戶，也可以是組織單位或整個組織 AWS Organizations。

若要取得有關的更多資訊 AWS RAM，請參閱[AWS RAM 使用者指南](#)。

## 跨可用區域共用

為確保資源分配至區域中的所有可用區域，可用區域會獨立對應至各個帳戶的名稱。這可能導致帳戶之間的可用區域命名出現差異。例如，您 AWS 帳戶的可 us-east-1a 用區域可能與其他 AWS 帳戶 us-east-1a 的位置不同。

若要基於您的帳戶來識別 容量預留的相對位置，您必須使用可用區域 ID (AZ ID)。AZ ID 是所有 AWS 帳戶中可用區域的唯一且一致的識別碼。例如，use1-az1 是 us-east-1 區域的 AZ ID，每個 AWS 帳戶中的位置都相同。

## 檢視您帳戶中可用區域的 AZ ID

1. [請在以下位置開啟 AWS RAM 主控台](https://console.aws.amazon.com/ram)。 <https://console.aws.amazon.com/ram>
2. 畫面右側的 Your AZ ID (您的 AZ ID) 面板中會顯示目前區域的 AZ ID。

## 共用 容量預留

當您與其他 AWS 帳戶共用您擁有的容量保留時，您可以讓他們將執行個體啟動到您的預留容量中。如果您共用開啟的容量預留，請謹記下列事項，因為有可能導致容量預留用量超出預料：

- 如果消費者的執行中執行個體符合 容量預留的屬性、將 CapacityReservationPreference 參數設為 open，以及尚未執行預留容量，他們會自動使用共用的 容量預留。
- 如果取用者啟動具有相符屬性 (執行個體類型、平台、可用區域和租用) 的執行個體，並將 CapacityReservationPreference 參數設定為 open，則會自動啟動至共用容量保留。

若要分享 容量預留，您必須將它新增至資源共享。資源共用是一 AWS RAM 種可讓您跨 AWS 帳號共用資源的資源。資源共享指定要共用的資源，以及共用它們的消費者。當您使用 Amazon EC2 主控台分享 容量預留時，您需要將它新增至現有的資源共享。若要將容量預留新增至新的資源共享，您必須使用 [AWS RAM 主控台](#) 建立資源共享。

如果您是組織中的一員，AWS Organizations 且已啟用組織內的共用功能，則您組織中的用戶會被授與存取共用「容量保留」(如果符合[共用的前提網站](#))。如果容量保留與外部帳戶共享，他們會收到加入資源共享的邀請，並且在接受邀請後便能存取共享的容量保留。

### Important

將執行個體啟動至與您共用的容量保留之前，請先在主控台中檢視共用容量保留，或使用描述容量保留的 [AWS CLI](#) 命令說明共用容量保留，以確認您是否具有存取共用容量保留的權限。如果您可以在主控台中檢視共用的容量保留或使用說明共用的容量保留 [AWS CLI](#)，則可供您使用，而且您可以在其中啟動執行個體。如果您嘗試將執行個體啟動至容量保留中，但由於共享失敗而無法存取，則執行個體將啟動至隨需容量中。

您可以使用 Amazon EC2 主控台、AWS RAM 主控台或 AWS CLI 來分享您擁有的容量保留。

使用 Amazon EC2 主控台來分享您擁有的 容量預留

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 容量預留。
3. 選擇要分享的 容量預留，然後選擇 Actions (動作)、Share reservation (共用保留)。
4. 選取資源共享以新增 容量預留，然後選取 Share 容量預留 (共用 容量預留)。

可能需要經過幾分鐘，消費者才能存取共用的 容量預留。

共用您使用 AWS RAM 主控台擁有的容量保留

請參閱《AWS RAM 使用者指南》中的[建立資源共享](#)。

若要共用您所擁有的容量保留 AWS CLI

使用 [create-resource-share](#) 命令。

停止共用容量預留

容量預留擁有者可以隨時停止共用容量預留。適用的規定如下：

- 共用停止時在共用容量中執行的執行個體 (由消費者擁有) 會在預留容量外繼續正常運作，而容量會根據 Amazon EC2 容量可用性還原到容量預留。
- 共用 容量預留的消費者不能再於預留容量中啟動新的執行個體。

若要停止共用您擁有的容量預留，您必須從資源共用中移除它。您可以使用 Amazon EC2 主控台、AWS RAM 主控台或 AWS CLI 進行此操作。

如何使用 Amazon EC2 主控台停止共用您擁有的容量預留

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 容量預留。
3. 選取容量預留，然後選取共用索引標籤。
4. Sharing (共用) 索引標籤列出已新增 容量預留的資源共享。選取要從中移除 容量預留的資源共享，然後選取 Remove from resource share (從資源共享中移除)。

停止共用您使用 AWS RAM 主控台擁有的容量保留

請參閱《AWS RAM 使用者指南》中的[更新資源共享](#)。

若要停止共用您所擁有的容量保留，請使用 AWS CLI

使用 [disassociate-resource-share](#) 命令。

識別並檢視共享的容量保留

#### Important

在將執行個體啟動至與您共享的容量保留中之前，請透過在主控台中檢視共享容量保留或使用 AWS CLI 對其進行描述，來驗證您是否有權存取該共享容量保留。如果您可以在主控台中檢視共用的容量保留或使用說明共用的容量保留 AWS CLI，則可供您使用，而且您可以在其中啟動執行個體。如果您嘗試將執行個體啟動至容量保留中，但由於共享失敗而無法存取，則執行個體將啟動至隨需容量中。

擁有者和消費者可以使用 Amazon EC2 主控台和 AWS CLI 來識別並檢視共享的容量保留。

使用 Amazon EC2 主控台來識別共用的 容量預留

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。

2. 在導覽窗格中，選擇 容量預留。畫面會列出您擁有的 容量預留和已分享給您的 容量預留。擁有者欄會顯示容量保留擁有者的 AWS 帳戶 ID。(me) AWS 帳號 ID 旁邊表示您是擁有者。

若要使用識別共用容量保留 AWS CLI

使用 [describe-capacity-reservations](#) 命令。此命令會傳回您擁有的容量保留以及與您共用的容量保留。OwnerId顯示容量保留擁有者的 AWS 帳戶 ID。

檢視共用的 容量預留 用量

共用的容量保留擁有者可以隨時使用 Amazon EC2 主控台和 AWS CLI來檢視其用量。

使用 Amazon EC2 主控台來檢視容量預留用量

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 容量預留。
3. 選取要檢視用量的 容量預留，然後選取 Usage (用量) 索引標籤。

AWS 帳戶 ID 資料欄顯示目前使用容量預留的消費者的帳戶 ID。Launched instances (啟動的執行個體數) 欄顯示每個消費者目前在預留容量中執行的執行個體數。

若要使用檢視容量保留使用狀況 AWS CLI

使用 [get-capacity-reservation-usage](#) 命令。AccountId 顯示使用 容量預留的帳戶的帳戶 ID。UsedInstanceCount 顯示消費者目前在預留容量中執行的執行個體數。

共用的容量預留許可

擁有者的許可

擁有者負責管理和取消他們共用的 容量預留。擁有者無法修改共用的 容量預留中由其他帳戶擁有的執行中執行個體。擁有者仍然負責管理他們在共用的 容量預留中啟動的執行個體。

消費者的許可

消費者負責管理他們在共用的 容量預留中執行的執行個體。消費者完全無法修改共用的 容量預留，也無法檢視或修改其他消費者或 容量預留擁有者所擁有的執行個體。

計費和計量

共用 容量預留無須額外收費。

容量預留擁有者對於他們在 容量預留內執行的執行個體及未用的預留容量，需要付費。消費者對於他們在共用的 容量預留內執行的執行個體，需要付費。

如果容量保留擁有者屬於不同的付款人帳戶，且區域預留執行個體或 Savings Plan 涵蓋容量保留，則會繼續針對區域預留執行個體或 Savings Plan 向容量保留擁有者收取費用。在這些情況下，容量保留擁有者會支付區域預留執行個體或 Savings Plan 的費用，並且會針對在共用容量保留中執行的執行個體向消費者收費。

## 執行個體限制

所有 容量預留用量都計入 容量預留擁有者的 隨需執行個體 限制。其中包括：

- 未用的預留容量
- 容量預留擁有者所擁有者的執行個體的用量
- 消費者擁有的執行個體的用量

消費者在共享容量中啟動的執行個體計入 容量預留擁有者的 隨需執行個體 限制。消費者的執行個體限制等於他們自己的 隨需執行個體 限制和他們在共用的 容量預留中可存取的可用容量的總和。

## 容量預留機群

隨需容量預留機群是一組容量預留。

容量預留機群要求包含啟動容量預留機群所需的所有組態資訊。使用單一要求，您可以為多個執行個體類型的工作負載預留大量 Amazon EC2 容量，最多至您指定的目標容量。

建立容量預留機群之後，您可以修改或取消容量預留機群，共同管理機群中的容量預留。

## 主題

- [容量預留機群的運作方式](#)
- [考量事項](#)
- [定價](#)
- [容量預留機群概念](#)
- [使用容量預留機群](#)
- [容量預留機群組態範例](#)
- [使用容量保留機群的服務連結角色](#)

## 容量預留機群的運作方式

建立容量預留機群時，機群會嘗試建立個別容量預留，符合您在機群要求中指定的總目標容量。

機群預留容量的執行個體數量，取決於您指定的[總目標容量](#)與[執行個體類型權重](#)。預留容量的執行個體類型，取決於您使用的[分配策略](#)和[執行個體類型優先順序](#)。

如果建立機群時容量不足，且無法立即滿足總目標容量，則機群會非同步嘗試建立容量預留，直到已預留要求的容量數量為止。

機群達到總目標容量時，會嘗試維持該容量。如果取消機群中的容量預留，機群會根據您的機群組態，自動建立一或多個容量預留，取代失去的容量並維持總目標容量。

無法個別管理機群中的容量預留。必須修改機群來集體管理。修改機群時，機群中的容量預留會自動更新以反映變更。

容量預留機群目前支援 open 執行個體符合條件，而機群啟動的所有容量預留會自動使用此執行個體符合條件。根據此準則，具有相符屬性 (執行個體類型、平台、可用區域和租用) 的新執行個體和現有執行個體會自動在叢集建立的容量保留中執行。容量預留機群不支援 target 執行個體符合條件。

## 考量事項

使用容量預留機群時請記住下列事項：

- 您可以使用和 AWS API 建立、修改、檢視及取消容量保留叢集。AWS CLI
- 無法個別管理機群中的容量預留。必須修改或取消機群來集體管理。
- 容量預留機群無法跨區域。
- 容量預留機群無法跨可用區域。
- 由容量保留叢集建立的容量保留會自動標記為下列 AWS 產生的標記：
  - 索引鍵 – `aws:ec2-capacity-reservation-fleet`
  - 值 – `fleet_id`

您可以使用此標籤，識別容量預留機群建立的容量預留。

## 定價

使用容量預留機群無須額外收費。系統會針對您容量預留機群建立的個別容量預留計費。如需容量預留計費方式的詳細資訊，請參閱[容量預留定價和帳單](#)。

## 容量預留機群概念

本主題說明容量預留機群的一些概念。

### 主題

- [總目標容量](#)
- [分配策略](#)
- [執行個體類型權重](#)
- [執行個體類型優先順序](#)

### 總目標容量

總目標容量定義容量預留機群預留的運算容量總量。建立容量預留機群時，您可以指定總目標容量。建立機群之後，Amazon EC2 會自動建立容量預留，最多預留至總目標容量。

容量預留機群為其預留容量的執行個體數量，取決於總目標容量，以及您為容量預留機群中每個執行個體類型指定的執行個體類型權重 (total target capacity/instance type weight=number of instances)。

您可以根據對您工作負載有意義的單位數量，指派總目標容量。舉例來說，如果您的工作負載需要特定數量 vCPU，您可以根據所需的 vCPU 數量，指派總目標容量。如果您的工作負載需要 2048 vCPU，請將總目標容量指派為 2048，然後根據機群中執行個體類型提供的 vCPU 數量，指派執行個體類型權重。如需範例，請參閱[執行個體類型權重](#)。

### 分配策略

容量預留機群中的分配策略，決定如何從容量預留機群組態中的執行個體類型規格，履行您的預留容量要求。

目前只支援 prioritized 分配策略。使用此策略，容量預留機群會使用您指派給容量預留機群組態中每個執行個體類型規格的優先順序，建立容量預留。優先順序值越低，表示使用的優先順序越高。例如，假設您建立使用下列執行個體類型和優先順序的容量預留機群：

- m4.16xlarge – 優先順序 = 1
- m5.16xlarge – 優先順序 = 3
- m5.24xlarge – 優先順序 = 2



機群會先嘗試為 m4.16xlarge 建立容量保留。如果 Amazon EC2 m4.16xlarge 容量不足時，機群會嘗試為 m5.24xlarge 建立容量保留。如果 Amazon EC2 m5.24xlarge 容量不足時，機群會為 m5.16xlarge 建立容量保留。

## 執行個體類型權重

執行個體類型權重是您指派給容量預留機群中每個執行個體類型的權重。權重決定計入機群總目標容量的該特定執行個體類型每個執行個體容量單位量。

您可以根據對您工作負載有意義的單位數量來指派權重。例如，如果您的工作負載需要特定數量 vCPU，您可以根據容量預留機群中每個執行個體類型提供的 vCPU 數量來指派權重。在此情況下，如果您使用 m4.16xlarge 和 m5.24xlarge 執行個體建立容量預留機群，您可以指派對應於每個執行個體 vCPU 數量的權重，如下所示：

- m4.16xlarge — 64 個 vCPU，權重 = 64 個單位
- m5.24xlarge — 96 個 vCPU，權重 = 96 個單位

執行個體類型權重決定容量預留機群為其預留容量的執行個體數量。例如，如果總目標容量為 384 單位的容量預留機群，使用先前範例中的執行個體類型和權重，則機群可以為 6 個 m4.16xlarge 執行個體 ( $384 \text{ 總目標容量} / 64 \text{ 執行個體類型權重} = 6 \text{ 個執行個體}$ )，或 4 個 m5.24xlarge 執行個體 ( $384 / 96 = 4$ ) 預留容量。

如果您未指派執行個體類型權重，或將執行個體類型權重指派為 1，則總目標容量會只根據執行個體數量。舉例來說，如果總目標容量為 384 個單位的容量預留機群使用先前範例中的執行個體類型，但省略權重或對這兩個執行個體類型指定 1，機群可為 384 個 m4.16xlarge 執行個體或 384 個 m5.24xlarge 執行個體預留容量。

## 執行個體類型優先順序

執行個體類型優先順序是您指派給機群中執行個體類型的值。優先順序是用來決定應優先使用哪些為機群指定的執行個體類型。

優先順序值越低，表示使用的優先順序越高。

## 使用容量預留機群

### 主題

- [開始之前](#)

- [容量預留機群狀態](#)
- [建立容量預留機群](#)
- [檢視容量預留機群](#)
- [修改容量預留機群](#)
- [取消容量預留機群](#)

## 開始之前

建立容量預留機群之前：

1. 決定工作負載所需的運算容量。
2. 決定要使用的執行個體類型和可用區域。
3. 根據您的需求和偏好，為每個執行個體類型指派優先順序。如需詳細資訊，請參閱 [執行個體類型優先順序](#)。
4. 建立適合您工作負載的容量加權系統。為每個執行個體類型指派權重，並決定總目標容量。如需詳細資訊，請參閱 [執行個體類型權重](#) 及 [總目標容量](#)。
5. 決定您是否無限期需要容量保留，或是只需要一段特定時間。

## 容量預留機群狀態

容量預留機群可以是下列任一狀態：

- `submitted` – 已提交容量保留機群請求，Amazon EC2 正準備建立容量保留。
- `modifying` – 正在修改容量保留機群。修改完前，機群會一直維持在此狀態。
- `active` – 容量保留機群已達到其總目標容量，並嘗試維持此容量。機群會一直維持在該狀態，直到遭修改或刪除。
- `partially_fulfilled` – 容量保留機群已部分達到總目標容量。Amazon EC2 容量不足以達到總目標容量。機群正嘗試以非同步方式達到總目標容量。
- `expiring` – 容量保留機群已到結束日期，且即將過期。其一或多個容量預留可能仍作用中。
- `expired` – 容量保留機群已到結束日期。機群及其容量預留已過期。機群無法建立新的容量預留。
- `cancelling` – 容量保留機群正遭取消。其一或多個容量預留可能仍作用中。
- `cancelled` – 容量保留機群已手動取消。機群及其容量預留已取消，且機群無法建立新的容量預留。
- `failed` – 容量保留機群無法為指定的執行個體類型預留容量。

## 建立容量預留機群

建立容量預留機群時，它會自動為機群請求中指定的執行個體類型建立容量預留，最多達指定的總目標容量。容量預留機群預留容量的執行個體數量，取決於您在請求中指定的總目標容量和執行個體類型權重。如需詳細資訊，請參閱[執行個體類型權重](#)及[總目標容量](#)。

建立機群時，您必須指定要使用的執行個體類型，以及每個執行個體類型的優先順序。如需詳細資訊，請參閱[分配策略](#)及[執行個體類型優先順序](#)。

### Note

第一次建立容量保留叢集時，會在您的帳戶中自動建立AWSServiceRoleForEC2CapacityReservationFleet服務連結角色。如需詳細資訊，請參閱[使用容量保留機群的服務連結角色](#)。

容量預留機群目前支援 open 執行個體符合條件。

您只能使用命令列建立容量預留機群。

若要建立容量預留機群

使用 [[建立容量](#)- AWS CLI 保留-艦隊] 指令。

```
aws ec2 create-capacity-reservation-fleet \
--total-target-capacity capacity_units \
--allocation-strategy prioritized \
--instance-match-criteria open \
--tenancy dedicated/default \
--end-date yyyy-mm-ddThh:mm:ss.000Z \
--instance-type-specifications file://instanceTypeSpecification.json
```

instanceTypeSpecification.json 內容如下。

```
[
 {
 "InstanceType": "instance_type",
 "InstancePlatform": "platform",
 "Weight": instance_type_weight,
 "AvailabilityZone": "availability_zone",
 "AvailabilityZoneId" : "az_id",
```

```

 "EbsOptimized": true/false,
 "Priority" : instance_type_priority
 }
]

```

預期的輸出結果。

```

{
 "Status": "status",
 "TotalFulfilledCapacity": fulfilled_capacity,
 "CapacityReservationFleetId": "cr_fleet_id",
 "TotalTargetCapacity": capacity_units
}

```

## 範例

```

aws ec2 create-capacity-reservation-fleet \
--total-target-capacity 24 \
--allocation-strategy prioritized \
--instance-match-criteria open \
--tenancy default \
--end-date 2021-12-31T23:59:59.000Z \
--instance-type-specifications file://instanceTypeSpecification.json

```

## instanceTypeSpecification.json

```

[
 {
 "InstanceType": "m5.xlarge",
 "InstancePlatform": "Linux/UNIX",
 "Weight": 3.0,
 "AvailabilityZone": "us-east-1a",
 "EbsOptimized": true,
 "Priority" : 1
 }
]

```

範例輸出。

```

{
 "Status": "submitted",

```

```
"TotalFulfilledCapacity": 0.0,
"CapacityReservationFleetId": "crf-abcdef01234567890",
"TotalTargetCapacity": 24
}
```

## 檢視容量預留機群

您可以隨時檢視容量預留機群的組態和容量資訊。檢視機群也會提供有關機群內個別容量預留的詳細資訊。

您只能使用命令列檢視容量預留機群。

## 若要檢視容量預留機群

使用[描述-容量-保留-叢](#) AWS CLI 集指令。

```
aws ec2 describe-capacity-reservation-fleets \
--capacity-reservation-fleet-ids cr_fleet_ids
```

## 預期的輸出結果

```
{
 "CapacityReservationFleets": [
 {
 "Status": "status",
 "EndDate": "yyyy-mm-ddThh:mm:ss.000Z",
 "InstanceMatchCriteria": "open",
 "Tags": [],
 "CapacityReservationFleetId": "cr_fleet_id",
 "Tenancy": "dedicated/default",
 "InstanceTypeSpecifications": [
 {
 "CapacityReservationId": "cr1_id",
 "AvailabilityZone": "cr1_availability_zone",
 "FulfilledCapacity": cr1_used_capacity,
 "Weight": cr1_instance_type_weight,
 "CreateDate": "yyyy-mm-ddThh:mm:ss.000Z",
 "InstancePlatform": "cr1_platform",
 "TotalInstanceCount": cr1_number of instances,
 "Priority": cr1_instance_type_priority,
 "EbsOptimized": true/false,
 "InstanceType": "cr1_instance_type"
 },
],
 },
],
}
```

```

{
 "CapacityReservationId": "cr2_id",
 "AvailabilityZone": "cr2_availability_zone",
 "FulfilledCapacity": cr2_used_capacity,
 "Weight": cr2_instance_type_weight,
 "CreateDate": "yyyy-mm-ddThh:mm:ss.000Z",
 "InstancePlatform": "cr2_platform",
 "TotalInstanceCount": cr2_number_of_instances,
 "Priority": cr2_instance_type_priority,
 "EbsOptimized": true/false,
 "InstanceType": "cr2_instance_type"
 },
],
"TotalTargetCapacity": total_target_capacity,
"TotalFulfilledCapacity": total_target_capacity,
"CreateTime": "yyyy-mm-ddThh:mm:ss.000Z",
"AllocationStrategy": "prioritized"
}
]
}

```

## 範例

```

aws ec2 describe-capacity-reservation-fleets \
--capacity-reservation-fleet-ids crf-abcdef01234567890

```

## 範例輸出

```

{
 "CapacityReservationFleets": [
 {
 "Status": "active",
 "EndDate": "2021-12-31T23:59:59.000Z",
 "InstanceMatchCriteria": "open",
 "Tags": [],
 "CapacityReservationFleetId": "crf-abcdef01234567890",
 "Tenancy": "default",
 "InstanceTypeSpecifications": [
 {
 "CapacityReservationId": "cr-1234567890abcdef0",
 "AvailabilityZone": "us-east-1a",
 "FulfilledCapacity": 5.0,
 "Weight": 1.0,

```

```
 "CreateDate": "2021-07-02T08:34:33.398Z",
 "InstancePlatform": "Linux/UNIX",
 "TotalInstanceCount": 5,
 "Priority": 1,
 "EbsOptimized": true,
 "InstanceType": "m5.xlarge"
 }
],
"TotalTargetCapacity": 5,
"TotalFulfilledCapacity": 5.0,
"CreateTime": "2021-07-02T08:34:33.397Z",
"AllocationStrategy": "prioritized"
}
]
```

## 修改容量預留機群

您可以隨時修改容量預留機群的總目標容量和日期。修改容量預留叢集的總目標容量時，機群會自動建立新的容量預留，或者修改或取消機群中的現有容量預留，以滿足新的總目標容量。修改機群的結束日期時，會相應地更新所有個別容量預留的結束日期。

修改機群後，其狀態會轉換為 `modifying`。您無法在機群處於 `modifying` 狀態時，嘗試對機群進行額外修改。

您無法修改容量預留機群使用的租用、可用區域、執行個體類型、執行個體平台、優先順序或權重。如果您必須變更任一參數，可能需要取消現有機群，並以所需參數建立新機群。

您只能使用命令列修改容量預留機群。

## 若要修改容量預留機群

使用 [修改容量-保](#) AWS CLI 留-艦隊指令。

### Note

您無法在相同命令中指定 `--end-date` 和 `--remove-end-date`。

```
aws ec2 modify-capacity-reservation-fleet \
--capacity-reservation-fleet-id cr_fleet_ids \

```

```
--total-target-capacity capacity_units \
--end-date yyyy-mm-ddThh:mm:ss.000Z \
--remove-end-date
```

### 預期的輸出結果

```
{
 "Return": true
}
```

### 範例：修改總目標容量

```
aws ec2 modify-capacity-reservation-fleet \
--capacity-reservation-fleet-id crf-01234567890abcdef \
--total-target-capacity 160
```

### 範例：修改結束日期

```
aws ec2 modify-capacity-reservation-fleet \
--capacity-reservation-fleet-id crf-01234567890abcdef \
--end-date 2021-07-04T23:59:59.000Z
```

### 範例：移除結束日期

```
aws ec2 modify-capacity-reservation-fleet \
--capacity-reservation-fleet-id crf-01234567890abcdef \
--remove-end-date
```

### 範例輸出

```
{
 "Return": true
}
```

### 取消容量預留機群

不再需要容量預留機群及其預留的容量時，您可以取消該機群。取消機群時，其狀態會變更為 `cancelled`，且無法再建立新的容量預留。此外，系統會取消機群中的所有個別容量預留，且先前在預留容量中執行的執行個體，會繼續在共用容量中正常執行。



您只能使用命令列取消容量預留機群。

若要取消容量預留機群

使用[取消容量-保留-艦隊命令](#) AWS CLI。

```
aws ec2 cancel-capacity-reservation-fleets \
--capacity-reservation-fleet-ids cr_fleet_ids
```

預期的輸出結果

```
{
 "SuccessfulFleetCancellations": [
 {
 "CurrentFleetState": "state",
 "PreviousFleetState": "state",
 "CapacityReservationFleetId": "cr_fleet_id_1"
 },
 {
 "CurrentFleetState": "state",
 "PreviousFleetState": "state",
 "CapacityReservationFleetId": "cr_fleet_id_2"
 }
],
 "FailedFleetCancellations": [
 {
 "CapacityReservationFleetId": "cr_fleet_id_3",
 "CancelCapacityReservationFleetError": [
 {
 "Code": "code",
 "Message": "message"
 }
]
 }
]
}
```

範例：成功取消

```
aws ec2 cancel-capacity-reservation-fleets \
--capacity-reservation-fleet-ids crf-abcdef01234567890
```

## 範例輸出

```
{
 "SuccessfulFleetCancellations": [
 {
 "CurrentFleetState": "cancelling",
 "PreviousFleetState": "active",
 "CapacityReservationFleetId": "crf-abcdef01234567890"
 }
],
 "FailedFleetCancellations": []
}
```

## 容量預留機群組態範例

### 主題

- [範例 1：根據 vCPU 數量預留容量](#)

### 範例 1：根據 vCPU 數量預留容量

以下範例會建立容量預留機群，此機群使用兩種執行個體類型：m5.4xlarge 和 m5.12xlarge。

它會根據指定執行個體類型提供的 vCPU 數量使用加權系統。總目標容量為 480 個 vCPU。m5.4xlarge 提供 16 個 vCPU，取得權重 16，而 m5.12xlarge 提供 48 個 vCPU，取得權重 48。此加權系統會設定容量預留機群，為 30 個 m5.4xlarge 執行個體 ( $480/16=30$ )，或 10 個 m5.12xlarge 執行個體 ( $480/48=10$ ) 預留容量。

機群設定為優先預留 m5.12xlarge 容量，取得優先順序 1，而 m5.4xlarge 取得較低的優先順序 2。這表示機群會嘗試先預留 m5.12xlarge 容量，並只在 Amazon EC2 m5.12xlarge 容量不足時，嘗試預留 m5.4xlarge 容量。

機群會為 Windows 執行個體預留容量，且該預留會在 October 31, 2021 23:59:59 UTC 自動過期。

```
aws ec2 create-capacity-reservation-fleet \
--total-target-capacity 480 \
--allocation-strategy prioritized \
--instance-match-criteria open \
--tenancy default \
--end-date 2021-10-31T23:59:59.000Z \
--instance-type-specifications file://instanceTypeSpecification.json
```

instanceTypeSpecification.json 內容如下。

```
[
 {
 "InstanceType": "m5.4xlarge",
 "InstancePlatform": "Windows",
 "Weight": 16,
 "AvailabilityZone": "us-east-1a",
 "EbsOptimized": true,
 "Priority" : 2
 },
 {
 "InstanceType": "m5.12xlarge",
 "InstancePlatform": "Windows",
 "Weight": 48,
 "AvailabilityZone": "us-east-1a",
 "EbsOptimized": true,
 "Priority" : 1
 }
]
```

### 使用容量保留機群的服務連結角色

隨需容量保留叢集使用 AWS Identity and Access Management (IAM) [服務連結角色](#)。服務連結角色是直接連結至容量預留機群的一種特殊 IAM 角色類型。服務連結角色由容量保留叢集預先定義，並包含服務代表您呼叫其他 AWS 服務所需的所有權限。

服務連結角色可讓設定容量預留機群更為簡單，因為您不必手動新增必要的許可。容量預留機群定義其服務連結角色的許可，除非另有定義，否則僅有容量預留機群可以擔任其角色。定義的許可包括信任政策和許可政策，且該許可政策無法附加至其他 IAM 實體。

您必須先刪除服務連結角色的相關資源，才能將其刪除。如此可保護您的容量預留機群資源，避免您不小心移除資源的存取許可。

### 容量保留機群的服務連結角色許可

容量保留叢集使用名為的服務連結角色 `AWSServiceRoleForEC2CapacityReservationFleet` 來代表您建立、描述、修改及取消先前由容量保留叢集建立的容量保留。

`AWSServiceRoleForEC2CapacityReservationFleet` 服務連結角色會信任下列實體擔任該角色：`capacity-reservation-fleet.amazonaws.com`

角色使用包含下列權限的AWSEC2CapacityReservationFleetRolePolicy原則：

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:DescribeCapacityReservations",
 "ec2:DescribeInstances"
],
 "Resource": "*"
 },
 {
 "Effect": "Allow",
 "Action": [
 "ec2:CreateCapacityReservation",
 "ec2:CancelCapacityReservation",
 "ec2:ModifyCapacityReservation"
],
 "Resource": [
 "arn:aws:ec2:*:*:capacity-reservation/*"
],
 "Condition": {
 "StringLike": {
 "ec2:CapacityReservationFleet": "arn:aws:ec2:*:*:capacity-
reservation-fleet/crf-*"
 }
 }
 },
 {
 "Effect": "Allow",
 "Action": [
 "ec2:CreateTags"
],
 "Resource": [
 "arn:aws:ec2:*:*:capacity-reservation/*"
],
 "Condition": {
 "StringEquals": {
 "ec2:CreateAction": "CreateCapacityReservation"
 }
 }
 }
]
}
```

```
]
}
```

您必須設定許可，IAM 實體 (如使用者、群組或角色) 才可建立、編輯或刪除服務連結角色。如需詳細資訊，請參閱《IAM 使用者指南》中的[服務連結角色許可](#)。

### 建立容量保留機群的服務連結角色

您不需要手動建立一個服務連結角色。當您使用 `create-capacity-reservation-fleet` AWS CLI 命令或 `CreateCapacityReservationFleet` API 建立容量保留叢集時，會自動為您建立服務連結角色。

若您刪除此服務連結角色，之後需要再次建立，您可以在帳戶中使用相同程序重新建立角色。建立容量預留機群時，容量預留機群會再次為您建立服務連結角色。

### 編輯容量保留機群的服務連結角色

容量保留叢集不允許您編輯 `AWSServiceRoleForEC2CapacityReservationFleet` 服務連結角色。因為有各種實體可能會參考服務連結角色，所以您無法在建立角色之後變更角色名稱。然而，您可使用 IAM 來編輯角色描述。如需更多資訊，請參閱 IAM 使用者指南中的[編輯服務連結角色](#)。

### 刪除容量保留機群的服務連結角色

若您不再使用需要服務連結角色的功能或服務，我們建議您刪除該角色。如此一來，您就沒有未主動監控或維護的未使用實體。然而，在手動刪除服務連結角色之前，您必須先刪除資源。

#### Note

若容量預留機群服務在您試圖刪除資源時正在使用該角色，刪除可能會失敗。若此情況發生，請等待數分鐘後並再次嘗試操作。

### 若要刪除 `AWSServiceRoleForEC2CapacityReservationFleet` 服務連結角色

1. 使用 `delete-capacity-reservation-fleet` AWS CLI 命令或 `DeleteCapacityReservationFleet` API 刪除帳戶中的容量保留叢集。
2. 使用 IAM 主控台或 AWS API 刪除 `AWSServiceRoleForEC2CapacityReservationFleet` 服務連結角色。AWS CLI 如需詳細資訊，請參閱《IAM 使用者指南》中的[刪除服務連結角色](#)。

## 容量保留機群服務連結角色的支援區域

容量預留機群支援在所有提供服務的區域中，使用服務連結角色。如需詳細資訊，請參閱 [AWS 區域與端點](#)。

## 監控容量保留

您可以使用下列功能來監控容量保留：

### 主題

- [使用指標監視容 CloudWatch 量保留](#)
- [使用監視容量保留 EventBridge](#)
- [使用率通知](#)

## 使用指標監視容 CloudWatch 量保留

透過 CloudWatch 指標，您可以透過設定 CloudWatch 警示來在滿足使用量閾值時通知您，有效率地監控容量保留並識別未使用的容量。這可協助您維護經常使用的 容量預留 磁碟區，並達到更高層級的使用率。

隨需容量保留會 CloudWatch 每五分鐘傳送一次指標資料。作用中時間少於五分鐘的 容量預留 不支援指標。

如需有關在 CloudWatch 主控台中檢視指標的詳細資訊，請參閱 [使用 Amazon CloudWatch 指標](#)。如需建立警示的詳細資訊，請參閱 [建立 Amazon CloudWatch 警示](#)。

### 目錄

- [容量預留 用量指標](#)
- [容量預留 指標維度](#)
- [檢視容 CloudWatch 量保留的指標](#)

## 容量預留 用量指標

AWS/EC2CapacityReservations 命名空間包含下列用量指標，可讓您用來監控和維護您針對預留指定之閾值內的隨需容量。

| 指標                | 描述 |
|-------------------|----|
| UsedInstanceCount |    |

| 指標                     | 描述                            |
|------------------------|-------------------------------|
|                        | 目前正在使用的執行個體數量。<br>單位：計數       |
| AvailableInstanceCount | 可供使用的執行個體數量。<br>單位：計數         |
| TotalInstanceCount     | 您已預留的執行個體總數。<br>單位：計數         |
| InstanceUtilization    | 目前正在使用的預留容量執行個體百分比。<br>單位：百分比 |

### 容量預留 指標維度

您可以使用下列維度來強化上表中列出的指標。

| 維度                    | 描述                            |
|-----------------------|-------------------------------|
| CapacityReservationId | 此全域唯一的維度會篩選您僅針對已識別之容量預留請求的資料。 |

### 檢視容 CloudWatch 量保留的指標

指標會先依服務命名空間分組，再依支援的維度分組。您可以使用下列程序檢視 容量預留 的指標。

使用 CloudWatch 主控台檢視容量保留測量結果

1. 開啟主 CloudWatch 控制台，[網址為 https://console.aws.amazon.com/cloudwatch/](https://console.aws.amazon.com/cloudwatch/)。
2. 如有必要請變更區域。從導覽列，選取您 容量預留 所在的「區域」。如需詳細資訊，請參閱[區域與端點](#)。

3. 在導覽窗格中，選擇 Metrics (指標)。
4. 針對所有指標，選擇 EC2 容量預留。
5. 選擇指標維度依據容量預留。指標將依 CapacityReservationId 分組。
6. 若要排序指標，請使用直欄標題。若要將指標圖形化，請選取指標旁的核取方塊。

### 檢視容量保留指標 (AWS CLI)

使用下列 [list-metrics](#) 命令：

```
aws cloudwatch list-metrics --namespace "AWS/EC2CapacityReservations"
```

### 使用監視容量保留 EventBridge

AWS Health EventBridge 當您帳戶中的容量保留在特定期間的使用量低於 20% 時，會將事件傳送至 Amazon。使用 EventBridge，您可以建立規則來觸發程式設計動作以回應此類事件。例如您可以建立一個「容量保留在過去 7 天的使用率低於 20% 時，自動取消該容量保留」的規則。

中的事件 EventBridge 會以 JSON 物件表示。事件的獨特欄位會包含在 JSON 物件的 "detail" 區段中。"event" 欄位則包含事件名稱。"result" 欄位包含觸發事件之動作的完成狀態。如需詳細資訊，請參閱 [Amazon EventBridge 使用者指南中的 Amazon EventBridge 事件模式](#)。

如需詳細資訊，請參閱 [Amazon EventBridge 使用者指南](#)。

中不支援此功能 AWS GovCloud (US)。

### 目錄

- [事件](#)
- [建立 EventBridge 規則](#)

### 事件

AWS Health 當容量保留的容量使用量低於 20% 時，會傳送下列事件。

### 事件

- [AWS\\_EC2\\_ODCR\\_UNDERUTILIZATION\\_NOTIFICATION](#)
- [AWS\\_EC2\\_ODCR\\_UNDERUTILIZATION\\_NOTIFICATION\\_SUMMARY](#)



## AWS\_EC2\_ODCR\_UNDERUTILIZATION\_NOTIFICATION

以下例子顯示新建立的容量保留在過去 24 小時的使用量低於 20% 時產生的事件。

```
{
 "version": "0",
 "id": "b3e00086-f271-12a1-a36c-55e8ddaa130a",
 "detail-type": "AWS Health Event",
 "source": "aws.health",
 "account": "123456789012",
 "time": "2023-03-10T12:03:38Z",
 "region": "ap-south-1",
 "resources": [
 "cr-01234567890abcdef"
],
 "detail": {
 "eventArn": "arn:aws:health:ap-south-1::event/EC2/
AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION/
AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION_cr-01234567890abcdef-6211-4d50-9286-0c9fbc243f04",
 "service": "EC2",
 "eventTypeCode": "AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION",
 "eventTypeCategory": "accountNotification",
 "startTime": "Fri, 10 Mar 2023 12:03:38 GMT",
 "endTime": "Fri, 10 Mar 2023 12:03:38 GMT",
 "eventDescription": [
 {
 "language": "en_US",
 "latestDescription": "A description of the event will be provided here"
 }
],
 "affectedEntities": [
 {
 "entityValue": "cr-01234567890abcdef"
 }
]
 }
}
```

## AWS\_EC2\_ODCR\_UNDERUTILIZATION\_NOTIFICATION\_SUMMARY

以下例子顯示一個或多個容量保留在過去 7 天的使用量低於 20% 時產生的事件。

```
{
```

```

"version": "0", "id": "7439d42b-3c7f-ad50-6a88-25e2a70977e2",
"detail-type": "AWS Health Event",
"source": "aws.health",
"account": "123456789012",
"time": "2023-03-07T06:06:01Z",
"region": "us-east-1",
"resources": [
 "cr-01234567890abcdef | us-east-1b | t3.medium | Linux/UNIX | 0.0%",
 "cr-09876543210fedcba | us-east-1a | t3.medium | Linux/UNIX | 0.0%"
],
"detail": {
 "eventArn": "arn:aws:health:us-east-1::event/EC2/
AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION_SUMMARY/
AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION_SUMMARY_726c1732-d6f6-4037-b9b8-
bec3c2d3ba65",
 "service": "EC2",
 "eventTypeCode": "AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION_SUMMARY",
 "eventTypeCategory": "accountNotification",
 "startTime": "Tue, 7 Mar 2023 06:06:01 GMT",
 "endTime": "Tue, 7 Mar 2023 06:06:01 GMT",
 "eventDescription": [
 {
 "language": "en_US",
 "latestDescription": "A description of the event will be provided
here"
 }
],
 "affectedEntities": [
 {
 "entityValue": "cr-01234567890abcdef | us-east-1b | t3.medium | Linux/
UNIX | 0.0%"
 },
 {
 "entityValue": "cr-09876543210fedcba | us-east-1a | t3.medium | Linux/
UNIX | 0.0%"
 }
]
}
}

```

## 建立 EventBridge 規則

若要在容量保留使用率降至 20% 以下時接收電子郵件通知，請建立 Amazon SNS 主題，然後為 AWS\_EC2\_ODCR\_UNDERUTILIZATION\_NOTIFICATION 事件建立 EventBridge 規則。

### 建立 Amazon SNS 主題

1. 在 <https://console.aws.amazon.com/sns/v3/home> 開啟 Amazon SNS 主控台。
2. 在導覽窗格中選擇 Topics (主題)，然後選擇 Create topic (建立主題)。
3. 針對類型，選擇標準。
4. 在名稱中輸入新主題的名稱。
5. 請選擇建立主題。
6. 選擇建立訂閱。
7. 在通訊協定中選擇電子郵件，然後在端點中輸入接收通知的電子郵件地址。
8. 選擇建立訂閱。
9. 您會在上方輸入的電子郵件地址收到主旨列如下的電子郵件訊息：AWS Notification - Subscription Confirmation。請依照指示來確認訂閱。

### 若要建立 EventBridge 規則

1. 在以下位置打開 Amazon EventBridge 控制台 <https://console.aws.amazon.com/events/>。
2. 在導覽窗格中，選擇 Rules (規則)，然後選擇 Create rule (建立規則)。
3. 在名稱中輸入新規則的名稱。
4. 針對規則類型，選擇具有事件模式的規則。
5. 選擇下一步。
6. 在事件模式中執行下列動作：
  - a. 在事件來源欄位中，選擇 AWS 服務。
  - b. 在 AWS 服務中選擇 AWS Health。
  - c. 在事件類型中選擇 EC2 ODCR 使用率不足通知。
7. 選擇下一步。
8. 對目標 1 執行下列動作：
  - a. 在目標類型欄位中，選擇 AWS 服務。

- b. 對於 Select a target (選取目標)，選擇 SNS topic (SNS 主題)。
  - c. 為主題選擇您剛剛建立的主題。
9. 選擇下一步，然後再選擇一次下一步。
  10. 選擇建立規則。

## 使用率通知

AWS Health 當您帳戶中「容量保留」的容量使用率降至 20% 以下時，會傳送下列電子郵件與 AWS Health Dashboard 通知。

- 每個新建立的容量保留在過去 24 小時使用率低於 20% 的個別通知。
- 所有容量保留在過去 7 天使用率低於 20% 的摘要通知。

電子郵件通知和 AWS Health Dashboard 通知會傳送至與擁有容量保留的 AWS 帳戶相關聯的電子郵件地址。通知包含以下資訊：

- 容量保留 ID。
- 容量保留的可用區域。
- 容量保留的平均使用率。
- 容量保留的執行個體類型和平台 (作業系統)。

此外，當您帳戶中「容量保留」的產能使用率在 24 小時與 7 天期間內降至 20% 以下時，AWS Health 會將事件傳送至 EventBridge。使用 EventBridge，您可以建立規則來啟動自動動作，例如傳送電子郵件通知或觸發 AWS Lambda 函數，以回應此類事件。如需詳細資訊，請參閱 [使用監視容量保留 EventBridge](#)。

## ML 的容量區塊

ML 容量區塊可讓您在未來保留需求高的 GPU 執行個體，支援短期機器學習 (ML) 工作負載。在容量區塊內執行的執行個體會自動放置在 [Amazon EC2](#) 內，以實現低延遲 UltraClusters、PB 級、無阻塞的聯網。

使用容量區塊，您可以查看未來何時可使用 GPU 執行個體容量，並且可以排定容量區塊在您最方便的時間啟動。保留容量區塊時，您可以獲得 GPU 執行個體的可預測容量保證，同時只須支付所需時間的費用。如果您需要 GPU 一次支援 ML 工作負載數天或數週，且不想在未使用 GPU 執行個體的情況下支付保留費用，建議您使用容量區塊。

以下是容量區塊的一些常見使用案例：

- ML 模型訓練和微調 – 不間斷存取保留的 GPU 執行個體，以完成 ML 模型訓練和微調。
- ML 實驗和原型 – 進行實驗並建置短時間需要 GPU 執行個體的原型。

容量區塊目前可用於 p5.48xlarge 和 p4d.24xlarge 執行個體。這些 p5.48xlarge 執行個體可在美國東部 (俄亥俄) 和美國東部 (維吉尼亞北部) 區域使用。這些 p4d.24xlarge 執行個體可在美國東部 (俄亥俄) 和美國西部 (奧勒岡) 區域使用。您可以保留最多未來八週內的容量區塊。

您可以使用容量區塊來預留具有下列保留持續時間 p5 和 p4d 執行個體數量選項的執行個體。

- 保留時間長度以 1 天為單位增加，最多總共 14 天
- 1、2、4、8、16、32 或 64 個執行個體的保留執行個體數量選項

若要預留容量區塊，請先指定容量需求，包括執行個體類型、執行個體數量、時間量、最早開始日期，以及所需的最晚結束日期。您就可以看到符合您規格的可用容量區塊方案。容量區塊方案包含詳細資訊，例如開始時間、可用區域和保留價格。容量區塊方案的價格取決於交付方案時的可用供給與需求。保留容量區塊後，價格不會變更。如需詳細資訊，請參閱 [容量區塊定價和計費](#)。

購買容量區塊方案時，系統會根據您選取的日期和執行個體數量保留。容量區塊保留開始時，您可以在啟動請求中指定保留 ID，鎖定執行個體啟動。

您可以使用所有保留的執行個體，直到容量區塊結束時間前 30 分鐘為止。容量區塊保留剩 30 分鐘時，我們就會開始終止容量區塊中執行的所有執行個體。我們會利用這段時間清理您的執行個體，再將容量區塊交付給下一位客戶。保留的最後 30 分鐘不會以容量區塊的價格收費。我們會在終止程序開始前 EventBridge 10 分鐘發出事件。如需詳細資訊，請參閱 [監視容量區塊 EventBridge](#)。

## 主題

- [支援平台](#)
- [考量事項](#)
- [相關資源](#)
- [容量區塊定價和計費](#)
- [使用容量區塊](#)
- [監控容量區塊](#)

## 支援平台

ML 的容量區塊目前支援，以 p5.48xlarge 及具有預設租用的 p4d.24xlarge 執行個體。當您使用購買「AWS Management Console 容量區塊」時，預設平台選項為 Linux/Unix。當您使用 AWS Command Line Interface (AWS CLI) 或購買 AWS SDK 容量區塊時，可以使用下列平台選項：

- Linux/UNIX
- Red Hat Enterprise Linux
- RHEL with HA
- SUSE Linux
- Ubuntu Pro

## 考量事項

使用容量區塊之前，請考慮下列詳細資訊和限制。

- 容量區塊開始並在國際標準時間 (UTC) 上午 11 點 30 分結束。
- 終止容量區塊中執行的執行個體程序會在保留的最後一天國際標準時間 (UTC) 上午 11 點開始。
- 可以保留最多未來 8 週內的容量區塊。
- 不可修改和取消容量區塊。
- 容量區塊無法跨 AWS 帳戶或 AWS 組織內共用。
- 容量區塊無法在容量保留群組中使用。
- 在特定日期，AWS 組織中所有帳戶的容量區塊中可保留的執行個體總數不得超過 64 個執行個體。
- 若要使用容量區塊，執行個體必須專門鎖定保留 ID。
- 容量區塊中的執行個體不會計入您的隨需執行個體限制。
- 對於使用自訂 AMI 的 P5 執行個體，請確定您擁有 [EFA 所需的軟體和組態](#)。
- 容量區塊目前無法與 Amazon EKS 受管節點群組或 Karpenter 使用。如需如何建立 Amazon EKS 自我管理節點群組的詳細資訊，請參閱 Amazon EKS 使用者指南中的 [ML 適用容量區塊](#)。

## 相關資源

建立容量區塊之後，您可以使用容量區塊執行下列作業：

- 將執行個體啟動至容量區塊。如需詳細資訊，請參閱 [在容量區塊中啟動執行個體](#)。

- 創建一個 Amazon EC2 Auto Scaling 組。如需詳細資訊，請參閱 [Amazon EC2 Auto Scaling 使用者指南中的機器學習工作負載使用容量區塊](#)。

#### Note

如果您使用 Amazon EC2 Auto Scaling 或 Amazon EKS，您可以排程擴展以在容量區塊保留開始時執行。透過排定的擴充功能，AWS 會自動為您處理重試，因此您不必擔心實作重試邏輯來處理暫時性失敗。

- 使用 AWS ParallelCluster。如需詳細資訊，請參閱 [使用 AWS ParallelCluster 和適用於 ML 的 Amazon EC2 容量區塊增強機器學習工作流程](#)。

如需有關的詳細資訊 AWS ParallelCluster，請參閱 [什麼是 AWS ParallelCluster](#)。

## 容量區塊定價和計費

### 主題

- [定價](#)
- [帳單](#)

### 定價

使用適用於 ML 的 Amazon EC2 容量區塊，您只需為保留的容量付費。容量區塊的價格取決於購買時容量區塊的可用供給與需求。您可以在保留前檢視容量區塊方案的價格。我們會在保留時預先收取容量區塊的費用。當您在某個日期範圍內搜尋容量區塊時，我們會傳回可用的最低價格容量區塊方案。保留容量區塊之後，價格不會變更。

使用容量區塊時，您會為執行個體執行時使用的作業系統付費。如需有關作業系統價格的詳細資訊，請參閱 [適用於 ML 定價的 Amazon EC2 容量區塊](#)。

### 帳單

我們會預先收取容量區塊方案的費用。您購買容量區塊後 12 小時內，我們會向您的 AWS 帳戶收取費用。處理您的付款時，您的容量區塊保留資源會維持 payment-pending 狀態。如果無法在 12 小時內處理您的付款，您的容量區塊就會釋出，且保留狀態會變更為 payment-failed。

成功處理您的付款後，容量區塊資源狀態會從 payment-pending 變更為 scheduled。您會收到反映一次性預付款的發票。在發票中，您可以將已付金額與容量區塊保留 ID 相關聯。

容量區塊保留開始時，系統只會根據保留中執行個體執行時，您使用的作業系統向您收費。您可以在 AWS Cost and Usage Report 的年度帳單中查看使用月份的使用量和相關費用。

### Note

Savings Plans 和預留執行個體折扣不適用於容量區塊。

## 檢視您的帳單

您可以在 AWS Billing and Cost Management 主控台中檢視帳單。容量區塊的預付款會顯示在您購買保留的月份。

保留開始後，您的帳單會分別顯示區塊保留已使用和未使用時間的明細。您可以透過這些明細項目，查看保留中使用了多少時間。如果您使用付費作業系統，只會在明細中看到已使用時間的使用費用。如需詳細資訊，請參閱 [定價](#)。未使用時間不會收費。

如需詳細資訊，請參閱 AWS Billing and Cost Management 使用者指南中的 [檢視您的帳單](#)。

如果您的容量區塊在與購買保留時不同的月份開始，預付費用和保留用量會顯示在不同的帳單月份下。在您的產能區塊預留識別碼中 AWS Cost and Usage Report，會列在預付費用的 ReservationARN 明細行項目中，以及週年紀念帳單中的明細行項目/ResourceID 中，以便您可以將用量關聯至對應的預付價格。

## 使用容量區塊

若要開始使用容量區塊，您要先尋找並購買符合您保留大小、時間長度和時間需求的可用容量區塊。接著，保留開始時，您可以透過啟動鎖定保留 ID 的執行個體，使用容量區塊。在保留到期前三十分鐘，我們會開始終止任何仍在容量區塊中執行的執行個體。

容量區塊會在單一可用區域中以 targeted 容量保留的形式提供。若要在容量區塊中執行執行個體，您必須在啟動執行個體時指定保留 ID。如果您自行停止執行個體且容量區塊到期，則必須先鎖定狀態為 active 的另一個容量區塊，才能重新啟動執行個體。

容量區塊預設會在容量區塊內的執行個體之間提供低延遲、高輸送量的網路連線，因此不需要搭配容量區塊使用叢集置放群組。

## 主題

- [必要條件](#)
- [尋找和購買容量區塊](#)



- [在容量區塊中啟動執行個體](#)
- [檢視容量區塊](#)

### 必要條件

您必須 AWS 區域 針對要使用的執行個體類型使用對應的。如需詳細資訊，請參閱 [區域](#)。

下列提供具有p5.48xlarge執行個體的容量區塊 AWS 區域。

| 區域名稱          | 區域代碼      |
|---------------|-----------|
| 美國東部 (俄亥俄)    | us-east-2 |
| 美國東部 (維吉尼亞北部) | us-east-1 |

下列提供具有p4d.24xlarge執行個體的容量區塊 AWS 區域。

| 區域名稱       | 區域代碼      |
|------------|-----------|
| 美國東部 (俄亥俄) | us-east-2 |
| 美國西部 (奧勒岡) | us-west-2 |

#### Note

並非所有執行個體類型都支援 64 個執行個體的容量區塊大小 AWS 區域。

### 尋找和購買容量區塊

若要保留容量區塊，您必須先找到符合您需求的可用容量時間區塊。若要尋找可保留的容量區塊，請指定。

- 您需要的執行個體數量
- 您需要執行個體的時間長度
- 您需要保留的日期範圍

若要搜尋可用的容量區塊方案，請指定保留時間長度和執行個體數量。您必須選取下列任一選項。

- 保留時間長度 – 最多 14 天，以 1 天為單位增加
- 例如計數 — 1、2、4、8、16、32 或 64 個執行個體

如果可用的容量區塊符合您的規格，我們會傳回單一容量區塊方案的詳細資訊。方案詳細資訊包括保留的開始時間、保留的可用區域，以及保留的價格。如需詳細資訊，請參閱 [定價](#)。

您可以購買顯示的容量區塊方案，也可以修改搜尋條件以查看其他可用選項。方案沒有預先設定的到期時間，但僅採先到先得方式。

購買容量區塊方案時，您會立即獲得回應，確認已保留您的容量區塊。確認後，您會在帳戶中看到新的容量保留，保留類型為 `capacity-block`，且 `start-date` 設定為您所購買方案的開始時間。保留您的容量區塊時，其狀態為 `payment-pending`。成功處理預付款後，保留狀態會變更為 `scheduled`。如需詳細資訊，請參閱 [帳單](#)。


您可以使用下列任一方法來尋找和購買容量區塊。

## Console

### 使用主控台尋找和購買容量區塊

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在畫面頂端的導覽列中，選取一個 AWS 區域。此選項很重要，因為並非所有區域中的所有執行個體類型都支援 64 個執行個體的容量區塊大小。
3. 在導覽窗格中，依序選擇容量保留和購買容量區塊。
4. 在容量屬性中，您可以設定容量區塊搜尋參數。默認情況下，該平台是 Linux。如果您要選取不同的作業系統，請使用 AWS CLI。如需詳細資訊，請參閱 [支援平台](#)。
5. 在總容量中，選取您要保留的執行個體數量。
6. 在持續時間中，輸入您需要保留的天數。
7. 在要搜尋容量區塊的日期範圍中，輸入保留的最早可能開始日期及最晚的可接受結束日期。
8. 選擇尋找容量區塊。
9. 如果可用的容量區塊符合您的規格，您會在建議的容量區塊下看到方案。如果有多個符合您規格的方案，則會顯示可用的最低價格容量區塊方案。若要檢視其他容量區塊方案，請調整您的搜尋輸入內容，並再次選擇尋找容量區塊。
10. 找到要購買的容量區塊方案時，請選擇下一步。

11. (選用) 在新增標籤頁面上，選擇新增標籤。
12. 檢閱和購買頁面會列出開始和結束日期、時間長度、執行個體總數和價格。

 Note

容量區塊在您保留之後就無法修改或取消。

13. 在購買容量區塊快顯視窗中，輸入「confirm」，然後選擇購買。

## AWS CLI

若要使用尋找容量區塊 AWS CLI

使用 `describe-capacity-block-offerings` 命令。

下列範例會搜尋具有 16 個 p5.48xlarge 執行個體的容量區塊，日期範圍從 2023-08-14 到 2023-10-22，時間長度為 48 小時。執行個體數量必須為預先設定選項 1、2、4、8、16、32、64 之中的整數。容量時間長度必須是介於 24 到 336 間，24 的倍數整數，以小時表示天數。

```
aws ec2 describe-capacity-block-offerings --instance-type p5.48xlarge \
--instance-count 16 --start-date-range 2023-08-14T00:00:00Z \
--end-date-range 2023-10-22-T00:00:00Z --capacity-duration 48
```

若要使用購買容量區塊 AWS CLI

使用 `purchase-capacity-block` 命令，然後指定您要購買之容量區塊的方案 ID，以及執行個體平台。

```
aws ec2 purchase-capacity-block \
--capacity-block-offering-id cbr-0123456789abcdefg \
--instance-platform Linux/UNIX
```

## 在容量區塊中啟動執行個體

保留容量區塊後，您可以在 AWS 帳戶中檢視容量區塊保留。您可以檢視 `start-date` 和 `end-date`，查看保留開始和結束時間。在容量區塊保留開始之前，可用容量會顯示為零。您可以依標籤索引鍵 `aws:ec2capacityreservation:incrementalRequestedQuantity` 的標籤值，查看容量區塊中可用的執行個體數量。

容量區塊保留開始時，保留狀態會從 `scheduled` 變更為 `active`。我們透過 Amazon 發出事件 EventBridge，通知您可以使用容量區塊。如需詳細資訊，請參閱 [監控容量區塊](#)。

若要使用容量區塊，您必須在啟動執行個體時指定容量區塊保留 ID。在容量區塊中啟動執行個體，會依啟動的執行個體數量減少可用容量。舉例來說，如果您購買的執行個體容量為八個執行個體，而您啟動四個執行個體，則可用容量會減少四個。

如果您在保留結束之前終止在容量區塊中執行的執行個體，可以在其位置啟動新執行個體。當您停止或終止容量區塊中的執行個體時，需要幾分鐘時間清理執行個體，才能啟動另一個執行個體加以取代。在此期間，您的執行個體將處於 `stopping` 或 `shutting-down` 狀態。完成此程序後，您的執行個體狀態會變更為 `stopped` 或 `terminated`。接著，容量區塊中的可用容量將會更新，顯示另一個可用的執行個體。

下列步驟說明如何使用 AWS Management Console 或將執行個體啟動到 `active` 狀態中的容量區塊 AWS CLI。

如需如何設定 EKS 節點群組，在開始時自動使用容量區塊的詳細資訊，請參閱《Amazon EKS 使用者指南》中的 [Capacity Blocks for ML](#)。

如需如何使用 EC2 Fleet 在容量區塊中啟動執行個體的資訊，請參閱 [教學課程：在容量區塊中啟動執行個體](#)。

如需如何建立鎖定容量區塊的啟動範本資訊，請參閱：[從啟動範本啟動執行個體](#)。

您可以使用下列任一種方法，在容量區塊中啟動執行個體。

## Console

使用主控台在容量區塊中啟動執行個體

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在螢幕上方的導覽列上，為容量區塊保留選取區域。
3. 從 Amazon EC2 主控台儀表板選擇 Launch Instance (啟動執行個體)。
4. (選用) 在名稱和標籤中，您可以為執行個體命名並標記執行個體。如需標籤的相關資訊，請參閱 [標記您的 Amazon EC2 資源](#)
5. 在應用程式和作業系統映像中，選取 Amazon Machine Image (AMI)。
6. 在執行個體類型中，選取適合您容量區塊保留的執行個體類型。
7. 在金鑰對 (登入) 中，選擇現有的金鑰對，或選擇建立新的金鑰對以建立新的金鑰對。如需詳細資訊，請參閱 [Amazon EC2 金鑰配對和 Amazon EC2 執行個體](#)。

8. 在 Network settings (網路設定) 下，使用預設設定，或視需要選擇 Edit (編輯) 以便設定網路設定。

**⚠ Important**

您的執行個體無法在與容量區塊所在可用區域不同的可用區域子網路中啟動。

9. 在進階詳細資訊中，設定執行個體，如下所示。
  - a. 在購買選項 (市場類型) 中，選取容量區塊。
  - b. 在容量保留中，選取依據 ID 的目標。
  - c. 選取容量區塊保留的容量保留 ID。
10. 在 Summary (摘要) 面板上，針對 Number of instances (執行個體的數量)，輸入要啟動的執行個體數量。
11. 選擇啟動執行個體。

## AWS CLI

### 使用將執行個體啟動到容量區塊 AWS CLI

- 使用 run-instances 命令，然後在 instance-market-options 結構中將 MarketType 指定為 capacity-block。您也必須指定 capacity-reservation-specification 參數。

以下範例會在具有相符屬性和可用容量的作用中容量區塊中，啟動單一 p5.48xlarge 執行個體。

```
aws ec2 run-instances --image-id ami-abc12345 --count 1 \
 --instance-type p5.48xlarge --key-name MyKeyPair \
 --subnet-id subnet-1234567890abcdef1 \
 --instance-market-options MarketType='capacity-block' \
 --capacity-reservation-specification \
 CapacityReservationTarget={CapacityReservationId=cr-a1234567}
```

## 檢視容量區塊

容量區塊具有下列狀態：

- payment-pending – 尚未處理預付款。

- `payment-failed` – 無法在 12 小時內處理付款。您的容量區塊已釋出。
- `scheduled` – 已處理付款，容量區塊保留尚未開始。
- `active` – 預留容量可供使用。
- `expired` – 容量區塊保留在保留請求中指定的日期和時間自動過期。預留容量將不再提供使用。

您可以使用下列任一種方法來檢視容量區塊預留。

## Console

### 使用主控台檢視容量區塊

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 容量預留。
3. 在容量保留概觀頁面上，您會看到一個資源表格，其中包含所有容量保留資源的詳細資訊。若要尋找您的容量區塊保留，請從容量保留 ID 上方的下拉式清單中選取容量區塊。在表格中，您可以查看容量區塊的相關資訊，例如開始和結束日期、時間長度和狀態。
4. 如需容量區塊的詳細資訊，請選取要檢視之容量區塊的保留 ID。容量保留詳細資訊頁面會顯示保留項目的所有屬性，以及容量區塊中使用中和可用的執行個體數量。

#### Note

在容量區塊保留開始之前，可用容量會顯示為零。您可以為標籤鍵使用下列標籤值，以查看容量區塊保留開始時可用的執行個體數量：`aws:ec2capacityreservation:incrementalRequestedQuantity`。

## AWS CLI

### 使用檢視容量區塊的步驟 AWS CLI

使用 `describe-capacity-reservations` 命令時，預設會同時列出隨需容量保留和容量區塊保留。若只要檢視容量區塊保留，請為 `capacity-reservation-type` 參數使用 `capacity-block` 來篩選。

例如，下列命令說明目前的一或多個容量區塊保留項目 AWS 區域。

```
aws ec2 describe-capacity-reservations --reservation-type capacity-block
```

範例輸出。

```
{
 "CapacityReservations": [
 {
 "CapacityReservationId": "cr-12345678",
 "EndDateType": "limited",
 "ReservationType": "capacity-block",
 "AvailabilityZone": "eu-east-2a",
 "InstanceMatchCriteria": "targeted",
 "EphemeralStorage": false,
 "CreateDate": "2023-11-29T14:22:45Z",
 "StartDate": "2023-12-15T12:00:00Z",
 "EndDate": "2023-08-19T12:00:00Z",
 "AvailableInstanceCount": 0,
 "InstancePlatform": "Linux/UNIX",
 "TotalInstanceCount": 16,
 "State": "payment-pending",
 "Tenancy": "default",
 "EbsOptimized": true,
 "InstanceType": "p5.48xlarge"
 },
 ...
]
}
```

## 監控容量區塊

### 主題

- [監視容量區塊 EventBridge](#)
- [記錄容量封鎖 API 呼叫 AWS CloudTrail](#)

## 監視容量區塊 EventBridge

當您的容量區塊保留開始時，Amazon EC2 會透過發出事件，表示您 EventBridge 的容量已準備就緒可供使用。在容量區塊保留結束前四十分鐘，您會收到另一個 EventBridge 事件，告訴您在保留區中執行的任何執行個體將在 10 分鐘內開始終止。如需有關 EventBridge 事件的詳細資訊，請參閱 [Amazon EventBridge 活動](#)。

以下事件為針對容量區塊所發出事件的結構：

### 已提供容量區塊

以下範例說明已提供容量區塊的事件。

```
{
 "customer_event_id": "[Capacity Reservation Id]-delivered",
 "detail_type": "Capacity Block Reservation Delivered",
 "source": "aws.ec2",
 "account": "[Customer Account ID]",
 "time": "[Current time]",
 "resources": [
 "[ODCR ARN]"
],
 "detail": {
 "capacity-reservation-id": "[ODCR ID]",
 "end-date": "[ODCR End Date]"
 }
}
```

### 容量區塊到期警告

以下範例說明容量區塊到期警告的事件。

```
{
 "customer_event_id": "[Capacity Reservation Id]-approaching-expiry",
 "detail_type": "Capacity Block Reservation Expiration Warning",
 "source": "aws.ec2",
 "account": "[Customer Account ID]",
 "time": "[Current time]",
 "resources": [
 "[ODCR ARN]"
],
 "detail": {
 "capacity-reservation-id": "[ODCR ID]",
 "end-date": "[ODCR End Date]"
 }
}
```

### 記錄容量封鎖 API 呼叫 AWS CloudTrail

容量區塊與一項服務整合 AWS CloudTrail，可提供使用者、角色或服務在容量區塊中所採取的動作記錄的 AWS 服務。CloudTrail 將容量區塊的 API 呼叫擷取為事件。這些擷取的呼叫包括來自容量區塊主控台的呼叫，以及針對容量區塊 API 操作的程式碼呼叫。如果您建立追蹤，您可以啟用持續交付 CloudTrail 事件到 Amazon S3 儲存貯體，包括容量區塊的事件。如果您未設定追蹤，您仍然可以在



[事件歷程記錄] 中檢視 CloudTrail 主控台中最近的事件。使用收集的資訊 CloudTrail，您可以判斷對容量區塊提出的要求、提出要求的來源 IP 位址、提出要求的人員、提出要求的時間，以及其他詳細資訊。

若要進一步了解 CloudTrail，請參閱[AWS CloudTrail 使用者指南](#)。

## 容量區塊資訊 CloudTrail

CloudTrail 在您創建帳戶 AWS 帳戶 時啟用。當活動發生在容量區塊中時，該活動會與 CloudTrail 事件歷史記錄中的其他 AWS 服務事件一起記錄在事件中。您可以查看，搜索和下載最近的事件在您的 AWS 帳戶。如需詳細資訊，請參閱[使用 CloudTrail 事件歷程記錄檢視事件](#)。

如需您 AWS 帳戶的事件的持續記錄 (包括容量區塊的事件)，請建立追蹤。追蹤可 CloudTrail 將日誌檔交付到 Amazon S3 儲存貯體。依預設，當您在主控台中建立追蹤時，該追蹤會套用至所有的 AWS 區域。追蹤記錄來自 AWS 分區中所有區域的事件，並將日誌檔傳送到您指定的 Amazon S3 儲存貯體。此外，您還可以設定其他 AWS 服務，以進一步分析 CloudTrail 記錄中收集的事件資料並採取行動。如需詳細資訊，請參閱下列內容：

- [建立追蹤的概觀](#)
- [CloudTrail 支援的服務與整合](#)
- [設定 CloudTrail 的 Amazon SNS 通知](#)
- [從多個區域接收 CloudTrail 日誌文件並從多個帳戶接收 CloudTrail 日誌文件](#)

所有容量區塊動作都會記錄在 Amazon EC2 API 參考中，CloudTrail 並記錄在其中。例如，呼叫和CapacityBlockActive動作會CapacityBlockScheduled在 CloudTrail 記錄檔中產生項目。

每一筆事件或日誌專案都會包含產生請求者的資訊。身分資訊可協助您判斷下列事項：

- 要求是使用根使用者登入資料還是 AWS Identity and Access Management (IAM) 使用者登入資料提出。
- 提出該請求時，是否使用了特定角色或聯合身分使用者的暫時安全憑證。
- 請求是否由其他 AWS 服務提出。

如需詳細資訊，請參閱[CloudTrail 使 userIdentity 元素](#)。


## 了解容量區塊日誌檔案項目

追蹤是一種組態，可讓事件以日誌檔的形式傳遞到您指定的 Amazon S3 儲存貯體。CloudTrail 記錄檔包含一或多個記錄項目。事件代表來自任何來源的單一請求，包括有關請求的操作，動作的日期和時

間，請求參數等信息。CloudTrail 日誌文件不是公共 API 調用的有序堆棧跟踪，因此它們不會以任何特定順序顯示。

下列範例顯示下列項目的 CloudTrail 記錄項目：

- [TerminateCapacityBlocksInstances](#)
- [CapacityBlockPaymentFailed](#)
- [CapacityBlock已排程](#)
- [CapacityBlock作用中](#)
- [CapacityBlock失敗](#)
- [CapacityBlock已過期](#)

 Note

已修改資料隱私權範例的某些欄位。

## TerminateCapacityBlocksInstances

```
{
 "eventVersion": "1.05",
 "userIdentity": {
 "accountId": "123456789012",
 "invokedBy": "AWS Internal;"
 },
 "eventTime": "2023-10-02T00:06:08Z",
 "eventSource": "ec2.amazonaws.com",
 "eventName": "TerminateCapacityBlockInstances",
 "awsRegion": "us-east-1",
 "sourceIPAddress": "203.0.113.25",
 "userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
 "requestParameters": null,
 "responseElements": null,
 "eventID": "a1b2c3d4-EXAMPLE",
 "readOnly": false,
 "resources": [
 {
 "accountId": "123456789012",
 "type": "AWS::EC2::Instance",
```

```

 "ARN": "arn:aws:ec2:US East (N. Virginia):123456789012:instance/
i-1234567890abcdef0"
 }
 {
 "accountId": "123456789012",
 "type": "AWS::EC2::Instance",
 "ARN": "arn:aws::ec2:US East (N. Virginia):123456789012:instance/
i-0598c7d356eba48d7"
 }
],
"eventType": "AwsServiceEvent",
"recipientAccountId": "123456789012",
"serviceEventDetails": {
 "capacityReservationId": "cr-12345678",
}
}

```

## CapacityBlockPaymentFailed

```

{
 "eventVersion": "1.05",
 "userIdentity": {
 "accountId": "123456789012",
 "invokedBy": "AWS Internal;"
 },
 "eventTime": "2023-10-02T00:06:08Z",
 "eventSource": "ec2.amazonaws.com",
 "eventName": "CapacityBlockPaymentFailed",
 "awsRegion": "us-east-1",
 "sourceIPAddress": "203.0.113.25",
 "userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
 "requestParameters": null,
 "responseElements": null,
 "eventID": "a1b2c3d4-EXAMPLE",
 "readOnly": false,
 "resources": [
 {
 "ARN": "arn:aws:ec2:US East (N. Virginia):123456789012:capacity-reservation/
cr-12345678",
 "accountId": "123456789012",
 "type": "AWS::EC2::CapacityReservation"
 }
],
}

```

```

"eventType": "AwsServiceEvent",
"recipientAccountId": "123456789012",
"serviceEventDetails": {
 "capacityReservationId": "cr-12345678",
 "capacityReservationState": "payment-failed"
}
}

```

## CapacityBlock已排程

```

{
 "eventVersion": "1.05",
 "userIdentity": {
 "accountId": "123456789012",
 "invokedBy": "AWS Internal;"
 },
 "eventTime": "2023-10-02T00:06:08Z",
 "eventSource": "ec2.amazonaws.com",
 "eventName": "CapacityBlockScheduled",
 "awsRegion": "us-east-1",
 "sourceIPAddress": "203.0.113.25",
 "userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
 "requestParameters": null,
 "responseElements": null,
 "eventID": "a1b2c3d4-EXAMPLE",
 "readOnly": false,
 "resources": [
 {
 "ARN": "arn:aws:ec2:US East (N. Virginia):123456789012:capacity-reservation/cr-12345678",
 "accountId": "123456789012",
 "type": "AWS::EC2::CapacityReservation"
 }
],
 "eventType": "AwsServiceEvent",
 "recipientAccountId": "123456789012",
 "serviceEventDetails": {
 "capacityReservationId": "cr-12345678",
 "capacityReservationState": "scheduled"
 }
}

```

## CapacityBlock作用中

```
{
 "eventVersion": "1.05",
 "userIdentity": {
 "accountId": "123456789012",
 "invokedBy": "AWS Internal;"
 },
 "eventTime": "2023-10-02T00:06:08Z",
 "eventSource": "ec2.amazonaws.com",
 "eventName": "CapacityBlockActive",
 "awsRegion": "us-east-1",
 "sourceIPAddress": "203.0.113.25",
 "userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
 "requestParameters": null,
 "responseElements": null,
 "eventID": "a1b2c3d4-EXAMPLE",
 "readOnly": false,
 "resources": [
 {
 "ARN": "arn:aws:ec2:US East (N. Virginia):123456789012:capacity-reservation/cr-12345678",
 "accountId": "123456789012",
 "type": "AWS::EC2::CapacityReservation"
 }
],
 "eventType": "AwsServiceEvent",
 "recipientAccountId": "123456789012",
 "serviceEventDetails": {
 "capacityReservationId": "cr-12345678",
 "capacityReservationState": "active"
 }
}
```

## CapacityBlock失敗

```
{
 "eventVersion": "1.05",
 "userIdentity": {
 "accountId": "123456789012",
 "invokedBy": "AWS Internal;"
 },
 "eventTime": "2023-10-02T00:06:08Z",
```

```

"eventSource": "ec2.amazonaws.com",
"eventName": "CapacityBlockFailed",
"awsRegion": "us-east-1",
"sourceIPAddress": "203.0.113.25",
"userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
"requestParameters": null,
"responseElements": null,
"eventID": "a1b2c3d4-EXAMPLE",
"readOnly": false,
"resources": [
 {
 "ARN": "arn:aws:ec2:US East (N. Virginia):123456789012:capacity-reservation/
cr-12345678",
 "accountId": "123456789012",
 "type": "AWS::EC2::CapacityReservation"
 }
],
"eventType": "AwsServiceEvent",
"recipientAccountId": "123456789012",
"serviceEventDetails": {
 "capacityReservationId": "cr-12345678",
 "capacityReservationState": "failed"
}
}

```

## CapacityBlock已過期

```

{
 "eventVersion": "1.05",
 "userIdentity": {
 "accountId": "123456789012",
 "invokedBy": "AWS Internal;"
 },
 "eventTime": "2023-10-02T00:06:08Z",
 "eventSource": "ec2.amazonaws.com",
 "eventName": "CapacityBlockExpired",
 "awsRegion": "us-east-1",
 "sourceIPAddress": "203.0.113.25",
 "userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
 "requestParameters": null,
 "responseElements": null,
 "eventID": "a1b2c3d4-EXAMPLE",
 "readOnly": false,
}

```

```

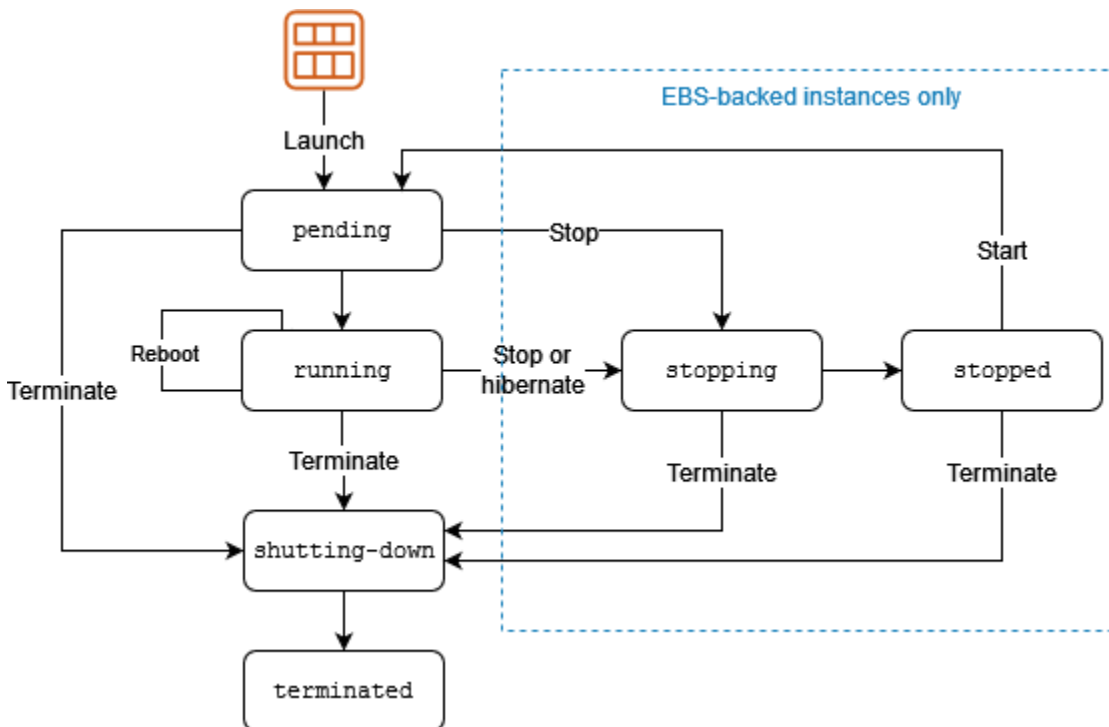
"resources": [
 {
 "ARN": "arn:aws:ec2:US East (N. Virginia):123456789012:capacity-reservation/
cr-12345678",
 "accountId": "123456789012",
 "type": "AWS::EC2::CapacityReservation"
 }
],
"eventType": "AwsServiceEvent",
"recipientAccountId": "123456789012",
"serviceEventDetails": {
 "capacityReservationId": "cr-12345678",
 "capacityReservationState": "expired"
}
}

```

## 執行個體生命週期

Amazon EC2 執行個體會轉換到不同狀態，從您將其啟動的那一刻起到其終止為止。

下圖顯示執行個體狀態之間的轉換。請注意，您無法停止和啟動執行個體後端執行個體。如需執行個體後端執行個體的詳細資訊，請參閱[根設備儲存](#)。



下表提供每個執行個體狀態的簡短說明，並指出是否要計費執行個體用量。某些 AWS 資源 (例如 Amazon EBS 磁碟區和彈性 IP 地址) 會產生費用，無論執行個體的状态為何。如需詳細資訊，請參閱 AWS Billing 使用者指南中的[避免非預期的費用](#)。

| 執行個體狀態        | 描述                                                                 | 執行個體用量計費 |
|---------------|--------------------------------------------------------------------|----------|
| pending       | 執行個體準備進入 running 狀態。當執行個體啟動時，或者在處於 stopped 狀態之後啟動時，將進入 pending 狀態。 | 不計費      |
| running       | 執行個體正在執行中且可供使用。                                                    | 計費       |
| stopping      | 執行個體已準備停止。                                                         | 不計費      |
| stopped       | 執行個體已關閉且無法使用。執行個體可以隨時啟動。                                           | 不計費      |
| shutting down | 執行個體準備終止。                                                          | 不計費      |
| terminated    | 執行個體已永久刪除且無法啟動。                                                    | 不計費      |

 **Note**

適用於已終止執行個體的預留執行個體，根據其付款選項計費至期滿。如需詳細資訊，請參閱 [Reserved Instances](#)

## 目錄

- [執行個體啟動](#)
- [執行個體停止和啟動 \(僅限 Amazon EBS 支援的執行個體\)](#)



- [執行個體休眠 \(僅限 Amazon EBS 支援的執行個體\)](#)
- [重新啟動執行個體](#)
- [執行個體終止](#)
- [重新啟動、停止、休眠和終止之間的差異](#)
- [啟動您的執行個體](#)
- [停止並啟動 Amazon EC2 執行個體](#)
- [休眠您的 Amazon EC2 實例](#)
- [重新啟動您的執行個體](#)
- [終止 Amazon EC2 實例](#)
- [執行個體淘汰](#)
- [執行個體恢復](#)

## 執行個體啟動

當您啟動執行個體時，其將進入 pending 狀態。您在啟動時指定的執行個體類型決定託管執行個體的電腦硬體。您在啟動時指定的 Amazon Machine Image (AMI) 將用來啟動執行個體。執行個體便已準備就緒，可供使用時，其將變為 running 狀態。接著您可連接到執行中的執行個體，並以一般使用電腦的方式使用。

您的執行個體轉換為 running 狀態後，會針對該執行個體執行的每秒計費 (最少為一分鐘)，即使執行個體保持閒置而且您未連接它仍會計費。

## 執行個體停止和啟動 (僅限 Amazon EBS 支援的執行個體)

如果您的執行個體未能通過狀態檢查或未如預期執行您的應用程式，而且執行個體的根磁碟區為 Amazon EBS 磁碟區，您可以停止和啟動執行個體，以嘗試修正此問題。

停止執行個體時，其會變成 stopping 狀態，然後再變成 stopped 狀態。執行個體若為 stopped，您不需支付使用量或資料傳輸費用。任何 Amazon EBS 磁碟區的儲存都會產生費用。當您的執行個體變為 stopped 狀態時，您可修改執行個體的部分屬性，包括執行個體類型。

當您啟動執行個體時，它會進入 pending 狀態，執行個體會移至新的主機電腦 (但有時會保留在目前的主機上)。停止並啟動執行個體時，連接原本主機電腦的執行個體儲存體磁碟區內的所有資料可能都會遺失。

您的執行個體會保留其私有 IPv4 地址，這表示與該私有 IPv4 地址或網路介面相關聯的彈性 IP 地址會與您的執行個體保留關聯。如果您的執行個體有 IPv6 地址，則會保留 IPv6 地址。

每次將執行個體從 stopped 轉換為 running 時，只要執行個體在執行就會按秒收費，每次啟動執行個體至少會收費一分鐘。

如需停止和啟動執行個體的詳細資訊，請參閱：[停止並啟動 Amazon EC2 執行個體](#)。

## 執行個體休眠 (僅限 Amazon EBS 支援的執行個體)

當您讓執行個體進入休眠時，我們會對作業系統發出執行休眠 (暫停磁碟機) 的訊號，將執行個體記憶體 (RAM) 的內容儲存於您的 Amazon EBS 根磁碟區。我們會保留執行個體的 Amazon EBS 根磁碟區，以及任何已連接的 Amazon EBS 資料磁碟區。當您啟動執行個體時，Amazon EBS 根磁碟區會還原至其先前的狀態，並重新載入 RAM 的內容。先前連接的資料磁碟區會重新連接，且執行個體會保留其執行個體 ID。

執行個體休眠時會變成 stopping 狀態，然後再變成 stopped 狀態。當執行個體處於 stopped 狀態時，不收取休眠執行個體的使用費，但當執行個體處於 stopping 狀態時，我們會收費，這與[停止執行個體](#)但不休眠的情況不同。我們不會收取資料傳輸費用，但是會收取任何 Amazon EBS 磁碟區的儲存 (包含 RAM 資料的儲存) 費用。

當您啟動休眠的執行個體時，它會進入 pending 狀態，而且我們會將執行個體移至新的主機電腦 (但有時候會保留在目前的主機上)。

您的執行個體會保留其私有 IPv4 地址，這表示與該私有 IPv4 地址或網路介面相關聯的彈性 IP 地址仍會與您的執行個體相關聯。如果您的執行個體擁有 IPv6 地址，其將保留 IPv6 地址。

如需詳細資訊，請參閱 [休眠您的 Amazon EC2 實例](#)。

## 重新啟動執行個體

您可以使用 Amazon EC2 主控台、命令列工具和 Amazon EC2 API 來重新啟動執行個體。建議您使用 Amazon EC2 重新啟動執行個體，而不是從執行個體執行作業系統重新啟動命令。

重新啟動執行個體等同於重新啟動作業系統。執行個體會留在相同的主機電腦上，並保有其公有 DNS 名稱、私有 IP 地址和執行個體存放區磁碟區上的所有資料。重新啟動通常需要幾分鐘時間才能完成，但重新啟動實際需要時間仍取決於執行個體組態。

重新啟動執行個體不會啟動新的執行個體計費期間；會繼續以秒計費，但不會收取其他的最少一分鐘收費。

如需詳細資訊，請參閱 [重新啟動您的執行個體](#)。

## 執行個體終止

當您決定不再需要執行個體時，您可終止執行個體。執行個體的狀態變更為 `shutting-down` 或 `terminated` 時刻起，該執行個體便停止收取費用。

如果啟用終止保護，您將無法以主控台、CLI 或 API 終止執行個體。

終止執行個體後，仍會短暫顯示於主控台，之後該項目會自動刪除。您也可以使用 CLI 和 API 來描述終止的執行個體。資源 (例如標籤) 會逐漸與該終止的執行個體取消關聯，過一段時間後，將不再顯示於該已終止的執行個體。您無法連接或復原已終止的執行個體。

每個 Amazon EBS 支援的執行個體都支援該 `InstanceInitiatedShutdownBehavior` 屬性，該屬性可控制執行個體在您從執行個體本身啟動關機時停止還是終止 (例如，在 Linux 上使用 `shutdown` 命令)。預設行為是停止執行個體。您可在執行個體運作中或停止時，修改此屬性設定。

每個 Amazon EBS 磁碟區均支援 `DeleteOnTermination` 屬性，其控制當您終止磁碟區所連接的執行個體時要刪除或保留磁碟區。預設值為刪除根設備磁碟區，並保留任何其他 EBS 磁碟區。

如需詳細資訊，請參閱 [終止 Amazon EC2 實例](#)。

## 重新啟動、停止、休眠和終止之間的差異

下表摘要列出正在重新啟動、正在停止、正在休眠和正在終止執行個體之間的主要差異。

| 特性            | 重新開機            | 停止/啟動 (僅適用於 Amazon EBS 後端執行個體)                                         | 休眠 (僅限 Amazon EBS 後端的執行個體)                                             | 終止 |
|---------------|-----------------|------------------------------------------------------------------------|------------------------------------------------------------------------|----|
| 主機電腦          | 執行個體保持在同一個主機電腦上 | 我們將執行個體移至新的主機電腦 (但有時候會保留在目前主機上)。                                       | 我們將執行個體移至新的主機電腦 (但有时候會保留在目前主機上)。                                       | 無  |
| 公有及私有 IPv4 位址 | 這些地址將保持不變       | 執行個體將保留其私有 IPv4 地址。執行個體將取得新的公有 IPv4 地址，除非其擁有的是彈性 IP 地址，該地址在停止/啟動時不會改變。 | 執行個體將保留其私有 IPv4 地址。執行個體將取得新的公有 IPv4 地址，除非其擁有的是彈性 IP 地址，該地址在停止/啟動時不會改變。 | 無  |

| 特性              | 重新開機               | 停止/啟動 (僅適用於 Amazon EBS 後端執行個體)                                                                        | 休眠 (僅限 Amazon EBS 後端的執行個體)                                                                                           | 終止                                 |
|-----------------|--------------------|-------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------|------------------------------------|
| 彈性 IP 地址 (IPv4) | 彈性 IP 地址仍與執行個體關聯   | 彈性 IP 地址仍與執行個體關聯                                                                                      | 彈性 IP 地址仍與執行個體關聯                                                                                                     | 彈性 IP 地址將從執行個體解除關聯                 |
| IPv6 地址         | 執行個體將保留其私有 IPv6 地址 | 執行個體將保留其私有 IPv6 地址                                                                                    | 執行個體將保留其私有 IPv6 地址                                                                                                   | 無                                  |
| 執行個體存放區磁碟區      | 資料將保留              | 資料將清除                                                                                                 | 資料將清除                                                                                                                | 資料將清除                              |
| 根設備磁碟區          | 磁碟區將保留             | 磁碟區將保留                                                                                                | 磁碟區將保留                                                                                                               | 磁碟區預設為刪除                           |
| RAM (記憶體內容)     | 將清除 RAM            | 將清除 RAM                                                                                               | RAM 會儲存到根磁碟區的檔案                                                                                                      | 將清除 RAM                            |
| 帳單              | 執行個體帳單時間不會變更       | 執行個體的狀態變更為 stopping 時刻起，該執行個體便停止收取費用。每次執行個體從 stopped 轉移為 running 時，我們都會啟動新的執行個體計費期間，每次啟動執行個體時最少收費一分鐘。 | 執行個體的狀態為 stopping 時會產生費用，但執行個體的狀態為 stopped 時即停止產生費用。每次執行個體從 stopped 轉移為 running 時，我們都會啟動新的執行個體計費期間，每次啟動執行個體時最少收費一分鐘。 | 一旦執行個體的狀態變更為，就不再產生費用 shutting-down |

作業系統關機命令每次都會終止執行個體後端執行個體。您可控制作業系統關機命令要停止或終止 Amazon EBS 後端執行個體。如需詳細資訊，請參閱 [變更執行個體啟動的關機行為](#)。

## 啟動您的執行個體

執行個體是 AWS 雲端中的虛擬伺服器。您從 Amazon Machine Image (AMI) 啟動執行個體。AMI 可為您的執行個體提供作業系統、應用程式伺服器和應用程式。

註冊後 AWS，您可以使用免費方案免費開始使用 AWS Amazon EC2。您可以使用免費方案來啟動並免費使用 t2.micro 執行個體 12 個月 (在無法使用 t2.micro 的區域中，您可以根據免費方案使用 t3.micro 執行個體)。如果您啟動的執行個體不包含在免費方案內，您將需要為執行個體支付標準 Amazon EC2 使用費。如需詳細資訊，請參閱 [Amazon EC2 定價](#)。

您可以使用下列方法來啟動執行個體。

| 方法                                                                                | 文件                                                              |
|-----------------------------------------------------------------------------------|-----------------------------------------------------------------|
| [Amazon EC2 主控台] 使用啟動執行個體精靈指定啟動參數。                                                | <a href="#">使用舊版啟動執行個體精靈啟動執行個體</a>                              |
| [Amazon EC2 主控台] 建立啟動範本並從啟動範本啟動執行個體。                                              | <a href="#">從啟動範本啟動執行個體</a>                                     |
| [Amazon EC2 主控台] 使用現有執行個體做為基礎。                                                    | <a href="#">使用現有執行個體的參數啟動執行個體</a>                               |
| [Amazon EC2 主控台] 使用您從 AWS Marketplace 購買的 AMI。                                    | <a href="#">啟動 AWS Marketplace 執行個體</a>                         |
| [AWS CLI] 使用您選取的 AMI。                                                             | <a href="#">透過 AWS CLI 使用 Amazon EC2</a>                        |
| [AWS Tools for Windows PowerShell] 使用您選取的 AMI。                                    | <a href="#">Amazon EC2 來自 AWS Tools for Windows PowerShell</a>  |
| [AWS CLI] 使用 EC2 Fleet 跨越不同的 EC2 執行個體類型和可用區域及跨越隨需執行個體、預留執行個體和 Spot 執行個體購買模型來佈建容量。 | <a href="#">EC2 Fleet</a>                                       |
| [AWS CloudFormation] 使用 AWS CloudFormation 模板指定實例。                                | <a href="#">AWS::EC2::Instance</a> 《AWS CloudFormation 使用者指南》中的 |

| 方法                               | 文件                                      |
|----------------------------------|-----------------------------------------|
| [AWS SDK] 使用特定於語言的 AWS SDK 啟動實例。 | <a href="#">AWS SDK for .NET</a>        |
|                                  | <a href="#">AWS SDK for C++</a>         |
|                                  | <a href="#">AWS SDK for Go</a>          |
|                                  | <a href="#">AWS 適用於 Java 的開發套件</a>      |
|                                  | <a href="#">AWS 適用於的 SDK JavaScript</a> |
|                                  | <a href="#">AWS SDK for PHP</a>         |
|                                  | <a href="#">AWS 適用於 Python 的 SDK</a>    |
|                                  | <a href="#">AWS 適用於紅寶石 V3 的 SDK</a>     |

**Note**

若要將 EC2 執行個體啟動到僅限 IPv6 的子網路中，您必須使用 Nitro 系統 [上建置的執行個體 AWS](#)。

**Note**

啟動僅限 IPv6 執行個體時，DHCPv6 可能無法立即向執行個體提供 IPv6 DNS 名稱伺服器。在此初始延遲期間，執行個體可能無法解析公有網域。對於在 Amazon Linux 2 上執行的執行個體，如果您想要使用 IPv6 DNS 名稱伺服器立即更新 /etc/resolv.conf 檔案，請在啟動時執行下列雲端初始化指令：

```
#cloud-config
bootcmd:
- /usr/bin/sed -i -E 's,^nameserver\s+[\.:digit:]]+$/,nameserver
fd00:ec2::253,' /etc/resolv.conf
```

另一個選擇是更改組態檔案並重新製作 AMI 的映像，以便在啟動時立即具有 IPv6 DNS 名稱伺服器地址。

當您啟動執行個體時，您可以在與下列其中一個資源相關聯的子網中啟動執行個體：

- 可用區域 - 此選項是預設值。
- Local Zone - 若要啟動 Local Zone 中的執行個體，您必須選擇加入 Local Zone，然後在區域中建立子網。如需詳細資訊，請參閱[開始使用 Local Zones](#)
- Wavelength 區域 - 若要啟動「Wavelength 區域」中的執行個體，您必須選擇加入「Wavelength 區域」，然後在區域中建立子網。如需如何在「Wavelength 區域」中啟動執行個體的詳細資訊，請參閱[開始使用 AWS Wavelength](#)。
- Outpost - 若要啟動 Outpost 中的執行個體，您必須建立 Outpost。如需如何建立前哨的詳細資訊，請參閱[開始使用 AWS Outposts](#)。

啟動您的執行個體之後，您便可以連線及使用它。開始時，執行個體的狀態為 pending。當執行個體的状态為 running 時，執行個體便會開始開機。在您連線到執行個體前可能需要短暫的時間。請注意，裸機執行個體類型可能需要更長的時間來啟動。

執行個體會獲得公有 DNS 名稱，您可以使用該名稱，從網際網路聯絡執行個體。執行個體也會獲得私有 DNS 名稱，讓其他位於相同 VPC 中的執行個體使用該名稱來聯絡執行個體。

當您完成使用執行個體後，請務必終止它。如需詳細資訊，請參閱[終止 Amazon EC2 實例](#)。

## 使用新的啟動執行個體精靈啟動執行個體

您可以使用新的啟動執行個體精靈啟動執行個體。啟動執行個體精靈會指定啟動執行個體所需的所有啟動參數。其中，啟動執行個體精靈會提供預設值，您可以接受預設值或指定自己的值。如果您接受預設值，則可以透過僅選取金鑰對來啟動執行個體。

### Important

當您啟動的執行個體不包含在 [AWS 免費方案](#) 內時，則需要為執行個體執行的時間付費，即使其保持閒置狀態仍會計費。

## 主題

- [快速啟動執行個體](#)
- [使用定義的參數啟動執行個體](#)
- [使用舊版啟動執行個體精靈啟動執行個體](#)

## 快速啟動執行個體

若要快速設定執行個體以進行測試，請遵循下列步驟。您將選取作業系統和金鑰對，並接受預設值。如需啟動執行個體精靈中所有參數的相關資訊，請參閱 [使用定義的參數啟動執行個體](#)。

## 快速啟動執行個體

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在畫面頂端的導覽列中，會顯示目前的「AWS 區域」(例如，美國東部 (俄亥俄))。選取要在其中啟動執行個體的區域。這項選擇非常重要，因為有些 Amazon EC2 資源可在不同區域間共享，其他則無法。如需詳細資訊，請參閱 [資源位置](#)。
3. 從 Amazon EC2 主控台儀表板選擇 Launch Instance (啟動執行個體)。
4. (選用) 在 Name and tags (名稱和標籤) 下，針對 Name (名稱)，輸入執行個體的描述性名稱。
5. 在 Application and OS Images (Amazon Machine Image) (應用程式和作業系統映像 (Amazon Machine Image)) 下，選擇 Quick Start (快速入門)，然後選擇執行個體的作業系統。
6. 在 Key pair (login) (金鑰對 (登入)) 下，針對 Key pair name (金鑰對名稱)，選擇現有的金鑰對或建立新的金鑰對。
7. 在 Summary (摘要) 面板中，選擇 Launch instance (啟動執行個體)。

## 使用定義的參數啟動執行個體

除了金鑰對之外，啟動執行個體精靈還會提供所有參數的預設值。您可以接受任何或全部預設值，或者透過為每個參數指定自己的值來設定執行個體。這些參數會在啟動執行個體精靈中分組。下列指示會引導您完成每個參數群組。

## 執行個體組態的參數

- [起始執行個體啟動](#)
- [名稱和標籤](#)
- [應用程式和作業系統映像 \(Amazon Machine Image\)](#)
- [執行個體類型](#)
- [金鑰對 \(登入\)](#)
- [Network settings \(網路設定\)](#)
- [設定儲存](#)
- [進階詳細資訊](#)
- [Summary](#)



## 起始執行個體啟動

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在畫面頂端的導覽列中，會顯示目前的「AWS 區域」(例如，美國東部 (俄亥俄))。選取要在其中啟動執行個體的區域。這項選擇非常重要，因為有些 Amazon EC2 資源可在不同區域間共享，其他則無法。如需詳細資訊，請參閱 [資源位置](#)。
3. 從 Amazon EC2 主控台儀表板選擇 Launch Instance (啟動執行個體)。

## 名稱和標籤

執行個體名稱是一個標籤，其中鍵是 Name (名稱)，而值是您指定的名稱。您可以標記執行個體、磁碟區和網路介面。對於 Spot 執行個體，您只能標記 Spot 執行個體請求。如需標籤的相關資訊，請參閱 [標記您的 Amazon EC2 資源](#)。

指定執行個體名稱和其他標籤是選用的。

- 對於 Name (名稱)，輸入執行個體的描述性名稱。如果您未指定名稱，則可以透過其 ID 來標識執行個體，該 ID 將在您啟動執行個體時自動產生。
- 若要新增其他標籤，請選擇 Add additional tags (新增其他標籤)。選取 Add tag (新增標籤)，然後輸入鍵和值，然後選取要標記的資源類型。為每個要新增的其他標籤重新選擇 Add tag (新增標籤)。

## 應用程式和作業系統映像 (Amazon Machine Image)

Amazon Machine Image (AMI) 包含建立執行個體所需的資訊。例如，AMI 可能包含做為網路伺服器所需的軟體，例如 Linux、Apache 和您的網站。

您可以尋找適當的 AMI，如下所示。對於尋找 AMI 的每個選項，可以選擇右上方的 Cancel (取消)，以返回啟動執行個體精靈，而不選擇 AMI。

## 搜尋列

若要搜尋所有可用的 AMI，請在 AMI 搜尋列中輸入關鍵字，然後按 Enter 鍵。選取 Select (選取) 來選取 AMI。

## Recents (最近)

最近使用的 AMI。

選取 Recently launched (最近啟動) 或 Currently in use (目前使用中)，然後從 Amazon Machine Image (AMI) 中，選取 AMI。

## My AMIs (我的 AMI)

您擁有的私有 AMI，或是與您共享的私有 AMI。

選取 Owned by me (由我擁有) 或 Shared with me (與我共享)，然後從 Amazon Machine Image (AMI) 中，選取 AMI。

## Quick Start

AMI 會依作業系統分組，以協助您快速開始使用。

首先選取您需要的操作系統，然後從 Amazon Machine Image (AMI) 中，選取 AMI。若要選取具有免費方案資格的 AMI，請確定 AMI 已標示為 Free tier eligible (具有免費方案資格)。

## Browse more AMIs (瀏覽更多 AMI)

選擇 Browse more AMIs (瀏覽更多 AMI) 以瀏覽完整的 AMI 目錄。

- 若要搜尋所有可用的 AMI，請在搜尋列中輸入關鍵字，然後按 Enter 鍵。
- 若要使用 Systems Manager 參數尋找 AMI，請選擇搜尋列右側的箭頭按鈕，然後選擇 Search by Systems Manager parameter (依 Systems Manager 參數搜尋)。如需詳細資訊，請參閱 [使用 Systems Manager 參數尋找 AMI](#)。
- 若要依類別搜尋，請選擇 Quickstart AMIs (AMI 快速入門)、My AMIs (我的 AMI)、AWS Marketplace AMI, 或 Community AMIs (社群 AMI)。

這 AWS Marketplace 是一個在線商店，您可以在其中購買運行的軟件 AWS，包括 AMI。如需有關從啟動執行個體的詳細資訊 AWS Marketplace，請參閱 [啟動 AWS Marketplace 執行個體](#)。在 Community AMIs (社群 AMI) 中，您可以尋找 AWS 社群成員已提供給其他人使用的 AMI。來自 Amazon 或已驗證的合作夥伴的 AMI 標記為 Verified provider (已驗證的供應商)。

- 若要篩選 AMI 清單，請在畫面左側的 Refine results (精簡結果) 下選取一或多個核取方塊。篩選條件選項會有所不同，取決於選取的搜尋類別。
- 檢查每個 AMI 列出的 Root device type (根設備類型)。注意哪些 AMI 是您需要的類型：ebs (Amazon EBS 後端) 或 instance-store (執行個體存放區後端)。如需詳細資訊，請參閱 [根設備儲存](#)。
- 檢查每個 AMI 列出的 Virtualization type (虛擬化類型)。注意哪些 AMI 是您需要的類型：hvm 或 paravirtual。例如，有些執行個體類型需要 HVM。如需 Linux 虛擬化類型的詳細資訊，請參閱 [AMI 虛擬化類型](#)。
- 檢查每個 AMI 列出的開機模式。注意哪些 AMI 使用您需要的開機模式：legacy-bios、uefi 或 uefi-preferred。如需詳細資訊，請參閱 [Amazon EC2 啟動模式](#)。
- 選取符合您需求的 AMI，然後選取 Select (選取)。

## 變更 AMI 時發出警示

如果修改了與選定 AMI 關聯的任何磁碟區或安全群組的組態，然後選取不同的 AMI，則會開啟一個視窗，警示您目前的某些設定將會變更或移除。您可以檢閱對安全群組和磁碟區做出的變更。此外，您可以檢視要新增和刪除的磁碟區，或者僅檢視待新增的磁碟區。

## 執行個體類型

執行個體類型定義執行個體的硬體組態和大小。較大的執行個體類型具有較多的 CPU 和記憶體。如需詳細資訊，請參閱 [Amazon EC2 執行個體類型](#)。

- 針對 Instance type (執行個體類型)，選取執行個體的執行個體類型。

**免費方案** — 如果您的 AWS 帳戶使用時間少於 12 個月，您可以透過選取 t2.micro 執行個體類型 (或在無法使用 t2.micro 的區域中使用 t3.micro 執行個體類型)，在免費方案下使用 Amazon EC2。如果某個執行個體類型符合免費方案的條件，則標記為 Free tier eligible (符合免費方案條件)。如需 t2.micro 和 t3.micro 的詳細資訊，請參閱 [爆量效能執行個體](#)。

- Compare instance types (比較執行個體類型)：您可以依下列屬性比較不同的執行個體類型：vCPU 數目、架構、記憶體數量 (GiB)、儲存數量 (GB)、儲存類型，以及網路效能。
- 取得建議：您可以從 Amazon Q EC2 執行個體類型選取器取得執行個體類型的指導和建議。如需詳細資訊，請參閱 [取得新工作負載的執行個體類型建議](#)。

## 金鑰對 (登入)

針對 Key pair name (金鑰對名稱)，選擇現有的金鑰對，或選擇 Create new key pair (建立新的金鑰對) 以建立新的金鑰對。如需詳細資訊，請參閱 [Amazon EC2 金鑰配對](#) 和 [Amazon EC2 執行個體](#)。

### Important

如果您選擇 Proceed without key pair (Not recommended) (繼續而不使用金鑰對 (不建議)) 選項，您將無法連線到執行個體，除非您選擇已設定為允許使用者透過其他方式登入的 AMI。

## Network settings (網路設定)

視需要設定網路設定。

- VPC：為您的執行個體選擇現有的 VPC。您可以選擇預設 VPC 或您建立的 VPC。如需詳細資訊，請參閱 [the section called “Virtual Private Cloud \(VPC\)”](#)。

- Subnet (子網)：您可以在與可用區域、Local Zone、Wavelength 區域或 Outpost 相關聯的子網中啟動執行個體。

若要在可用區域中啟動執行個體，請選取要在當中啟動執行個體的字網。若要建立新的子網，請選擇 Create new subnet (建立新的子網)，以前往 Amazon VPC 主控台。完成後，請返回啟動執行個體精靈並選擇 Refresh (重新整理) 圖示，載入清單中的子網。

若要在僅限 IPv6 的子網路中啟動執行個體，該執行個體必須[建置在 Nitro 系統上](#)。

在 Local Zone 中啟動執行個體，選取您在 Local Zone 中建立的子網。

若要在 Outpost 中啟動執行個體，請在與 Outpost 相關聯的 VPC 中選取子網。

- Auto-assign Public IP (自動指派公有 IP)：指定您的執行個體是否會收到公有 IPv4 地址。根據預設，預設子網中的執行個體會收到公有 IPv4 地址，非預設子網中的執行個體則不會。您可以選取 Enable (啟用) 或 Disable (停用) 覆寫子網的預設設定。如需詳細資訊，請參閱 [公有 IPv4 地址](#)。
- Firewall (security groups) (防火牆 (安全群組))：使用安全群組定義執行個體的防火牆規則。這些規則會指定應交付至您執行個體的傳入網路流量。所有其他流量都會遭到忽略。如需安全群組的詳細資訊，請參閱[適用於 Amazon EC2 執行個體的亞馬遜 EC2 安全群組](#)。

如果新增網路介面，您必須在網路介面中指定相同的安全群組。

選取或建立安全群組，如下所示：

- 若要選取 VPC 的現有安全群組，請選取 Select existing security group (選取現有安全群組)，然後從 Common security groups (常見安全群組) 中選取您的安全群組。
- 若要建立新的 VPC 安全群組，請選擇 Create security group (建立安全群組)。啟動執行個體精靈會自動定義 launch-wizard-x 安全群組，並提供以下核取方塊以快速新增安全群組規則：

(Linux) 允許 SSH 流量來自 — 建立輸入規則以允許您透過安全殼層 (連接埠 22) 連線至執行個體。

(Windows) 允許 RDP 流量來源 — 建立輸入規則以允許您透過 RDP (通訊埠 3389) 連線至執行個體。

指定流量來自 Anywhere (任意位置)、Custom (自訂) 還是 My IP (我的 IP)。

Allow HTTPs traffic from the internet (允許來自網際網路的 HTTPS 流量) – 建立開啟連接埠 443 (HTTPS) 的傳入規則，以允許來自任意位置的網際網路流量。如果您的執行個體將是 Web 伺服器，則需要此規則。

Allow HTTP traffic from the internet (允許來自網際網路的 HTTP 流量) – 建立開啟連接埠 80 (HTTP) 的傳入規則，以允許來自任意位置的網際網路流量。如果您的執行個體將是 Web 伺服器，則需要此規則。

您可以編輯這些規則並新增符合您需求的規則。

若要編輯或新增規則，請選擇 Edit (編輯) (位於右上角)。若要新增規則，請選擇 Add security group rule (新增安全群組規則)。對於 Type (類型)，選取網路流量類型。Protocol (通訊協定) 欄位會自動填入要對網路流量開啟的通訊協定。對於 Source type (來源類型)，選取來源類型。若要讓啟動執行個體精靈新增電腦的公有 IP 地址，請選擇 My IP (我的 IP)。但是，如果您透過 ISP 或是從防火牆後方進行連線，不具備靜態 IP 地址，則需要找到用戶端電腦使用的 IP 地址範圍。

#### Warning

如果您短暫啟動測試執行個體，而且很快就會將其停止或終止，則啟用所有 IP 地址 (0.0.0.0/0)，透過 SSH 或 RDP 存取您執行個體的規則是可接受的，但在生產環境中並不安全。建議您僅授權特定 IP 地址或地址範圍存取您的執行個體。

- Advanced network configuration (進階網路組態) – 僅在選擇子網時才能使用。

#### 網路介面

- Device index (設備索引)：網路卡的索引。主要網路介面必須指派給網路卡索引 0。某些執行個體類型支援多個網路卡。
- Network interface (網路介面)：選取 New interface (新介面)，讓 Amazon EC2 建立新的介面，或是選取現有的可用網路介面。
- Description (描述)：(選用) 新網路介面的描述。
- Subnet (子網)：要建立新網路介面的子網。對主要網路介面 (eth0) 而言，這是要啟動執行個體的字網。如果您已輸入 eth0 的現有網路介面，則會在網路介面所在的子網中啟動執行個體。
- 安全群組：VPC 中要與網路介面建立關聯的一或多個安全性群組。
- Primary IP (主要 IP)：您子網範圍內的私有 IPv4 地址。保留空白讓 Amazon EC2 為您選擇私有 IPv4 地址。
- Secondary IP (次要 IP)：來自子網範圍的一或多個其他私有 IPv4 地址。選擇 Manually assign (手動指派)，然後輸入 IP 地址。選擇 Add IP (新增 IP) 以新增另一個 IP 地址。或者，選擇 Automatically assign (自動指派)，讓 Amazon EC2 為您選擇一個，然後輸入一值以指示要新增的 IP 地址數目。

- (僅限 IPv6) IPv6 IPs (IPv6 IP)：子網範圍內的 IPv6 地址。選擇 Manually assign (手動指派)，然後輸入 IP 地址。選擇 Add IP (新增 IP) 以新增另一個 IP 地址。或者，選擇 Automatically assign (自動指派)，讓 Amazon EC2 為您選擇一個，然後輸入一值以指示要新增的 IP 地址數目。
- IPv4 Prefixes (IPv4 字首)：網路介面的 IPv4 字首。
- IPv6 Prefixes (IPv6 字首)：網路介面的 IPv6 字首。
- (雙堆疊和僅限 IPv6) 指派主要 IPv6 IP：(選用) 如果要將執行個體啟動至雙堆疊或僅限 IPv6 的子網路，您可選擇指派主要 IPv6 IP。指派主要 IPv6 地址可讓您避免中斷執行個體或 ENI 的流量。如果此執行個體依賴於其 IPv6 地址未改變，請選擇啟用。當您啟動執行個體時，AWS 會自動將與 ENI 相關聯的 IPv6 位址指派為主要 IPv6 位址。在啟用 IPv6 GUA 地址作為主要 IPv6 地址後，您便無法停用該地址。當您啟用 IPv6 GUA 地址作為主要 IPv6 地址時，第一個 IPv6 GUA 地址會成為主要 IPv6 地址，直到執行個體終止或網路介面中斷連接為止。如果有多個與執行個體連接之 ENI 相關聯的 IPv6 地址，並且啟用主要 IPv6 地址，則與 ENI 相關聯的第一個 IPv6 GUA 地址會成為主要 IPv6 地址。
- Delete on termination (終止時刪除)：是否在刪除執行個體時刪除網路介面。
- Elastic Fabric Adapter：指出網路介面是否為 Elastic Fabric Adapter。如需詳細資訊，請參閱 [Elastic Fabric Adapter](#)。
- ENA 快遞：ENA Express 由可 AWS 擴展的可靠數據報 (SRD) 技術提供支持。SRD 技術使用封包噴塗機制來分配負載並避免網路擁塞。啟用 ENA Express 便可支援的執行個體在可能的情況下，在一般 TCP 流量之上使用 SRD 進行通訊。除非您從清單中選取啟用或停用，否則啟動執行個體精靈不會包含執行個體的 ENA Express 組態。
- ENA Express UDP：如果您已啟用 ENA Express，您可以選擇性用於 UDP 流量。除非您選取啟用或停用，否則啟動執行個體精靈不會包含執行個體的 ENA Express 組態。

選擇 [新增網路介面] 以新增其他網路介面。其他網路介面可以位於相同 VPC 的不同子網路中，或位於您擁有的不同 VPC 中的子網路中 (只要子網路與您的執行個體位於相同的可用區域)。如果您選擇新增位於其他 VPC 子網路中的其他網路介面，則在選取子網路時會看到「多重虛擬私人雲端子網路」選項。如果您在其他 VPC 中選取子網路，多重虛擬私人雲端標籤會顯示在您新增的網路介面旁邊。如此一來，您就能在不同 VPC 建立多重主目錄執行個體，分別使用不同的網路和安全組態。請注意，如果您從另一個 VPC 連接其他 ENI，則必須從該 VPC 為 ENI 選擇一個安全群組。

如需詳細資訊，請參閱 [彈性網路介面](#)。若您指定超過一個網路介面，您的執行個體便無法收到公有 IPv4 地址。此外，若您為 eth0 指定現有的網路介面，您便無法使用 Auto-assign Public IP (自動指派公有 IP) 來覆寫子網的公有 IPv4 設定。如需詳細資訊，請參閱 [在啟動執行個體期間指派公有 IPv4 地址](#)。

## 設定儲存

您選取的 AMI 包含一或多個儲存體磁碟區，包括根磁碟區。您可以指定要連接至執行個體的其他磁碟區。

您可以使用 Simple (簡單) 或 Advanced (進階) 檢視。使用 Simple (簡單) 檢視，您可以指定磁碟區的大小和類型。若要指定所有磁碟區參數，請選擇卡片右上方的 Advanced (進階) 檢視。

使用 Advanced (進階) 檢視，您可以設定每個磁碟區，如下所示：

- **Storage type (儲存類型)**：選取要與您的執行個體相關聯的 Amazon EBS 磁碟區或執行個體存放區磁碟區。此清單中可用的磁碟區類型取決於您選擇的執行個體類型。如需詳細資訊，請參閱 [Amazon EC2 執行個體存放區](#) 和 [Amazon EBS 磁碟區](#)。
- **Device name (設備名稱)**：從磁碟區可用的設備名稱清單中選取。
- **Snapshot (快照)**：選取要從中還原磁碟區的快照。您可以在 Snapshot (快照) 欄位中輸入文字，搜尋可用的共享和公有快照。
- **Size (GiB) (大小 (GiB))**：針對 EBS 磁碟區，您可以指定儲存體大小。如果已選取符合免費方案資格的 AMI 和執行個體，請注意，若要維持在免費方案中，您的總儲存體必須低於 30 GiB。
- **Volume Type (磁碟區類型)**：針對 EBS 磁碟區，選取磁碟區類型。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的 Amazon EBS 磁碟區類型](#)。
- **IOPS**：如果您已選取 Provisioned IOPS SSD 磁碟區類型，就可以輸入磁碟區能支援的每秒 I/O 操作數 (IOPS)。
- **Delete on termination (終止時刪除)**：針對 Amazon EBS 磁碟區，請選擇 Yes (是) 以執行個體終止時刪除磁碟區，或選擇 No (否) 保留磁碟區。如需詳細資訊，請參閱 [在執行個體終止時保留資料](#)。
- **Encrypted (已加密)**：如果執行個體類型支援 EBS 加密，您可以選擇 Yes (是) 以啟用磁碟區的加密。如果您在此區域中預設啟用加密，則會為您啟用預設加密。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的 Amazon EBS 加密](#)。
- **KMS Key (KMS 金鑰)**：如果您為 Encrypted (已加密) 選取了 Yes (是)，則必須選取要用來加密磁碟區的客戶受管金鑰。如果在此區域中預設啟用加密，則會為您選取預設客戶受管金鑰。您可以選取不同的金鑰，或指定您已建立之任何客戶受管金鑰的 ARN。
- **File systems (檔案系統)**：將 Amazon EFS 或 Amazon FSx 檔案系統掛載到執行個體。如需有關掛載 Amazon EFS 檔案系統的詳細資訊，請參閱 [將 Amazon EFS 搭配 Linux 執行個體](#)。如需有關掛載 Amazon FSx 檔案系統的詳細資訊，請參閱 [使用 Amazon FSx 搭配 Amazon EC2](#)。

## 進階詳細資訊

針對 Advanced Details (進階詳細資訊)，展開此區段來檢視欄位，指定執行個體的其他參數。

- Purchasing option (購買選項)：選擇 Request Spot Instances (請求 Spot 執行個體) 以 Spot 價格請求 Spot 執行個體，但是不超過隨需價格，並選擇 Customize (自訂) 變更預設的 Spot 執行個體設定。您可以設定最高價 (不建議)，並變更請求類型、請求持續時間和中斷行為。如不請求 Spot 執行個體，Amazon EC2 預設會啟動隨需執行個體。如需詳細資訊，請參閱 [建立 Spot 執行個體請求](#)。
- 網域加入目錄：選取啟動後要加入執行個體的 AWS Directory Service 目錄 (網域)。若您選取網域，您必須選取具備必要許可的 IAM 角色。如需有關網域加入 Linux 執行個體的詳細資訊，請參閱 [將 Linux EC2 執行個體無縫加入您的 AWS 受管 Microsoft AD 目錄](#)。如需有關加入 Windows 執行個體的網域的詳細資訊，請參閱 [將 Windows EC2 執行個體無縫加入 AWS 管理的 AD 目錄](#)。
- IAM 執行個體設定檔：選取要與執行個體建立關聯的 AWS Identity and Access Management (IAM) 執行個體設定檔。如需詳細資訊，請參閱 [Amazon EC2 的 IAM 的角色](#)。
- Hostname type (主機名稱類型)：選取執行個體的客體作業系統主機名稱是否要包含資源名稱或 IP 名稱。如需詳細資訊，請參閱 [Amazon EC2 執行個體主機名稱類型](#)。
- DNS Hostname (DNS 主機名稱)：決定資源名稱或 IP 名稱的 DNS 查詢 (根據您在 Hostname type (主機名稱類型) 所選的項目) 是以 IPv4 地址 (A 記錄)、IPv6 地址 (AAAA 記錄) 還是以兩者進行回應。如需詳細資訊，請參閱 [Amazon EC2 執行個體主機名稱類型](#)。
- Shutdown behavior (關機行為)：選取執行個體是否應於關機時停止或終止。如需詳細資訊，請參閱 [變更執行個體啟動的關機行為](#)。
- Stop - Hibernate behavior (停止 - 休眠行為)：若要啟用休眠，請選取 Enable (啟用)。此欄位只在您的執行個體符合休眠先決條件時才可使用。如需詳細資訊，請參閱 [休眠您的 Amazon EC2 實例](#)。
- Termination protection (終止保護)：若要防止意外性的終止，請選取 Enable (啟用)。如需詳細資訊，請參閱 [啟用終止保護](#)。
- Stop protection (停止保護)：若要防止意外性的停止，請選擇 Enable (啟用)。如需詳細資訊，請參閱 [啟用停止保護](#)。
- 詳細 CloudWatch 監控：選擇啟用以開啟使用 Amazon 對執行個體的詳細監控 CloudWatch。需支付額外費用。如需詳細資訊，請參閱 [使用監控執行個體 CloudWatch](#)。
- 彈性 GPU：Amazon 彈性圖形於 2024 年 1 月 8 日終止使用壽命。對於需要圖形加速的工作負載，我們建議您使用 Amazon EC2 G4ad、G4dn 或 G5 執行個體。
- Elastic inference (彈性推斷)：一個彈性推斷加速器，用於連接到您的 EC2 CPU 執行個體。如需詳細資訊，請參閱 Amazon Elastic Inference 開發人員指南中的 [Working with Amazon Elastic Inference](#)。

#### Note

自 2023 年 4 月 15 日起，不 AWS 會將新客戶加入 Amazon Elastic Inference (EI)，並協助目前客戶將工作負載遷移到提供更優惠價格和效能的選項。2023 年 4 月 15 日之後，新客戶



將無法在 Amazon、Amazon ECS 或 Amazon Amazon EC2 使用亞馬遜 SageMaker EI 加速器啟動執行個體。但是，在過去 30 天內至少使用過一次 Amazon EI 的客戶將被視為目前客戶，並且可以繼續使用該服務。

- Credit specification (額度規格)：選擇 Unlimited (無限制)，只要有需要，就讓應用程式爆量超過基線。此欄位僅對 T 執行個體有效。可能需支付額外費用。如需詳細資訊，請參閱 [爆量效能執行個體](#)。
- Placement group name (置放群組名稱)：指定要在其中啟動執行個體的置放群組。可以選取現有的置放群組或建立新的置放群組。並非所有執行個體類型都支援在置放群組中啟動執行個體。如需詳細資訊，請參閱 [置放群組](#)。
- EBS 最佳化執行個體：針對 Amazon EBS 最佳化的執行個體使用最佳化組態堆疊，並為 Amazon EBS I/O 額外提供專屬的容量。如果執行個體類型支援此功能，請選取 Enable (啟用) 以啟用它。需支付額外費用。如需詳細資訊，請參閱 [the section called “EBS 最佳化”](#)。
- 容量保留：指定是否要在任何開啟的容量保留 (Open (開啟))、特定容量保留 (Target by ID (按 ID 分類的目標)) 或容量保留群組 (Target by group (按群組分類的目標)) 中啟動執行個體。若要指定不應使用容量保留，請選擇 None (無)。如需詳細資訊，請參閱 [使用現有的 容量預留 啟動執行個體](#)。
- Tenancy (租用)：選擇在共享硬體 (Shared (共享))、隔離的、專用硬體 (Dedicated (專用)) 或專用執行個體 (Dedicated host (專用執行個體)) 上執行您的執行個體。如果您選擇將執行個體啟動至專用執行個體，則可以指定是否將執行個體啟動至主機資源群組，或是以特定專用執行個體為目標。可能需支付額外費用。如需詳細資訊，請參閱 [專用執行個體](#) 及 [Dedicated Hosts](#)。
- RAM 磁碟 ID：(僅適用於半虛擬 (PV) AMIs) 選取執行個體的 RAM 磁碟。若您已選取核心，則可能需要選取具有可支援之驅動程式的特定 RAM 磁碟。
- 核心 ID：(僅適用於全虛擬 (PV) AMIs) 選取執行個體的核心。
- Nitro Enclave：可讓您從 Amazon EC2 執行個體建立稱為隔離區的獨立執行環境。選取「啟用」以啟用「AWS 硝基隔離區」的執行個體。如需詳細資訊，請參閱 [什麼是 AWS 硝基飛地？](#) 在 AWS 硝基飛地用戶指南。
- 授權組態：您可以根據指定的授權配置啟動執行個體，以追蹤您的授權使用情況。如需詳細資訊，請參閱《AWS License Manager 使用者指南》中的 [建立授權組態](#)。
- 可存取的中繼資料：您可以啟用或停用對執行處理中繼資料的存取。如需詳細資訊，請參閱 [設定新執行個體的執行個體中繼資料選項](#)。
- 中繼資料 IPv6 端點：您可以讓執行個體使用 IMDS IPv6 位址 [fd00:ec2::254] 擷取執行個體中繼資料。只有當您將 [AWS Nitro 系統上建置的執行個體啟動到支援 IPv6 的子網路 \(僅限雙堆疊或 IPv6\)](#) 時，才能使用此選項。如需有關擷取執行個體中繼資料的詳細資訊，請參閱 [擷取執行個體中繼資料](#)。

- 中繼資料版本：如果啟用對執行處理中繼資料的存取，則可以選擇請求執行處理中繼資料時請求使用執行個體中繼資料服務第 2 版。如需詳細資訊，請參閱 [設定新執行個體的執行個體中繼資料選項](#)。
- 中繼資料回應躍點限制：如果您啟用執行個體中繼資料，您可以為中繼資料字符設定允許的網路躍點數量。如需詳細資訊，請參閱 [設定新執行個體的執行個體中繼資料選項](#)。
- Allow tags in metadata (允許中繼資料中的標籤)：如果選取 Enable (啟用)，則執行個體將允許從其中繼資料存取其所有標籤。如果未指定任何值，則預設情況下，不允許存取執行個體中繼資料中的標籤。如需詳細資訊，請參閱 [允許存取執行個體中繼資料中的標籤](#)。
- User data (使用者資料)：您可以指定使用者資料，以在啟動期間設定執行個體或執行組態指令碼。如需 Linux 執行個體之使用者資料的詳細資訊，請參閱 [啟動時在您的 Amazon EC2 執行個體上執行命令](#)。如需 Windows 執行個體之使用者資料的詳細資訊，請參閱 [Amazon EC2 如何處理 Windows 執行個體的使用者資料](#)。

## Summary

使用 Summary (摘要) 面板，以指定要啟動的執行個體數量、檢閱您的執行個體組態，以及啟動您的執行個體。

- Number of instances (執行個體數目)：輸入要啟動的執行個體數目。所有執行個體都會以相同的組態啟動。

### Tip

為了確保執行個體能更快啟動，請將大型請求分成較小的批次。例如，分別建立五個啟動請求，每個請求各啟動 100 個執行個體，而不要以單一請求啟動 500 個執行個體。

- (選用) 若指定了多個執行個體，為了確保足以處理您應用程式需求的正確執行個體數目，您可以選擇 consider EC2 Auto Scaling (考慮 EC2 Auto Scaling) 以建立啟動範本及 Auto Scaling 群組。Auto Scaling 會根據您的規格擴展群組中的執行個體數目。如需詳細資訊，請參閱 [Amazon EC2 Auto Scaling 使用者指南](#)。

### Note

如果 Amazon EC2 Auto Scaling 將 Auto Scaling 群組中的執行個體標記為「狀況不良」，將自動排程更換該執行個體，並終止該執行個體，啟動另一個執行個體，而且您會遺失原始執行個體上的資料。如果您停止或重新開機執行個體，或另一個事件將此執行個體標記為「狀

況不良」，則執行個體會標記為「狀況不良」。如需詳細資訊，請參閱 [Amazon EC2 Auto Scaling 使用者指南](#) 中的 [自動擴展群組中執行個體的運作 Health 態檢查](#)。

- 檢閱執行個體的詳細資訊，並進行任何必要的變更。您可以直接導覽至某個區段，方法是在 Summary (摘要) 面板中選擇其連結。
- 當您準備好啟動執行個體時，請選擇 Launch instance (啟動執行個體)。

如果執行個體無法啟動或狀態立即進入 terminated 而非 running，請參閱 [針對執行個體啟動問題進行故障診斷](#)。

(選用) 您可以建立執行個體的帳單提醒。在確認畫面上的 Next Steps (後續步驟) 下方，選擇 Create billing alerts (建立帳單提醒)，然後遵循指示進行。您還可以在啟動執行個體後建立帳單提醒。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的 [建立帳單警示以監控估計 AWS 費用](#)。

### 使用舊版啟動執行個體精靈啟動執行個體

只有當您所在的區域支援舊啟動體驗時，才能使用舊啟動執行個體精靈來啟動執行個體。啟動執行個體精靈會指定啟動執行個體所需的所有啟動參數。其中，啟動執行個體精靈會提供預設值，您可以接受預設值或指定自己的值。您必須指定 AMI 和金鑰對，才能啟動執行個體。

如需使用新版啟動執行個體精靈的說明，請參閱 [使用新的啟動執行個體精靈啟動執行個體](#)。

#### Important

當您啟動的執行個體不包含在 [AWS 免費方案](#) 內時，則需要為執行個體執行的時間付費，即使其保持閒置狀態仍會計費。

啟動執行個體的步驟：

- [起始執行個體啟動](#)
- [步驟 1：選擇 Amazon Machine Image \(AMI\)](#)
- [步驟 2：選擇執行個體類型](#)
- [步驟 3：設定執行個體詳細資訊](#)
- [步驟 4：新增儲存體](#)
- [步驟 5：新增標籤](#)
- [步驟 6：設定安全群組](#)

- [步驟 7：檢閱執行個體啟動並選取金鑰對](#)

## 起始執行個體啟動

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在螢幕上方的導覽列中會顯示目前的區域 (例如, US East (Ohio))。選取一個符合您執行個體需求的區域。這項選擇非常重要, 因為有些 Amazon EC2 資源可在不同區域間共享, 其他則無法。如需詳細資訊, 請參閱 [資源位置](#)。
3. 從 Amazon EC2 主控台儀表板選擇 Launch Instance (啟動執行個體)。

### 步驟 1：選擇 Amazon Machine Image (AMI)

當您啟動執行個體時, 您必須選取組態 (又稱為 Amazon Machine Image (AMI))。AMI 包含建立新執行個體的必要資訊。例如, AMI 可能包含充當網路伺服器所需的軟體, 例如 Linux、Apache 和您的網站。

在您啟動執行個體時, 您可以從清單中選取 AMI, 也可以選取指向 AMI ID 的 Systems Manager 參數。如需詳細資訊, 請參閱 [the section called “使用 Systems Manager 參數尋找 AMI”](#)。

在 Choose an Amazon Machine Image (AMI) (選擇 Amazon Machine Image (AMI)) 頁面上, 使用兩個選項中的一個選項來選擇 AMI。使用 [搜尋 AMI 清單](#), 或 [依 Systems Manager 參數搜尋](#)。

### 透過搜尋 AMI 清單

1. 在左側窗格中選取要使用的 AMI 類型：

#### Quick Start

選取熱門的 AMI 可協助您快速開始使用。若要選取符合免費方案資格的 AMI, 請在左側窗格中選取 Free tier only (僅限免費方案)。這些 AMI 會標記為 Free tier eligible (符合免費方案)。

#### My AMIs (我的 AMI)

您擁有的私有 AMI, 或是與您共享的私有 AMI。若要檢視與您共享的 AMI, 請在左側窗格中選擇 Shared with me (與我共享)。

#### AWS Marketplace

一個在線商店, 您可以在其中購買可運行的軟件 AWS, 包括 AMI。如需有關從啟動執行個體的詳細資訊 AWS Marketplace, 請參閱 [啟動 AWS Marketplace 執行個體](#)。

## Community AMIs (社群 AMI)

AWS 社群成員提供給其他人使用的 AMI。若要透過作業系統篩選 AMI 的清單，請在 Operating system (作業系統) 下方選擇適當的核取方塊。您也可以依架構和根設備類型篩選。

2. (Linux 執行個體) 檢查針對每個 AMI 列出的根裝置類型。注意哪些 AMI 是您需要的類型 (ebs (Amazon EBS 後端) 或 instance-store (執行個體存放區後端))。如需詳細資訊，請參閱 [根設備儲存](#)。
3. 檢查每個 AMI 列出的 Virtualization type (虛擬化類型)。注意哪些 AMI 是您需要的類型 (hvm 或 paravirtual)。例如，有些執行個體類型需要 HVM。如需 Linux 虛擬化類型的詳細資訊，請參閱 [AMI 虛擬化類型](#)。
4. 檢查每個 AMI 列出的開機模式。請注意哪些 AMI 使用您需要的開機模式，legacy-bios 或 uefi。如需詳細資訊，請參閱 [Amazon EC2 啟動模式](#)。
5. 選取符合您需求的 AMI，然後選取 Select (選取)。

## 依 Systems Manager 參數

1. 選擇 Search by Systems Manager parameter (依 Systems Manager 參數搜尋) (位於右上角)。
2. 對於 Systems Manager parameter (Systems Manager 參數)，請選取一個參數。對應的 AMI ID 會出現在 Currently resolves to (目前解析為) 旁邊。
3. 選擇 Search (搜尋)。與 AMI ID 相符的 AMI 會顯示在清單中。
4. 從清單中選取 AMI，然後選取 Select (選取)。

## 步驟 2：選擇執行個體類型

在 Choose an Instance Type (選取執行個體類型) 頁面上，選取要啟動之執行個體的硬體組態和大小。較大的執行個體類型具有較多的 CPU 和記憶體。如需詳細資訊，請參閱 [Amazon EC2 執行個體類型](#)。

若要維持符合免費方案的資格，請選擇 t2.micro 執行個體類型 (或無法使用 t2.micro 的區域中的 t3.micro 執行個體類型)。如果某個執行個體類型符合免費方案的條件，則標記為 Free tier eligible (符合免費方案條件)。如需 t2.micro 和 t3.micro 的詳細資訊，請參閱 [爆量效能執行個體](#)。

根據預設，精靈會顯示目前世代的執行個體類型，然後根據您選取的 AMI 選取第一個可用的執行個體類型。若要檢視前代的執行個體類型，請從篩選條件清單中選擇 All generations (所有世代)。

**Note**

若要快速設定執行個體以進行測試，請選擇 Review and Launch (檢閱和啟動) 以接受預設組態設定，然後啟動您的執行個體。否則，若要進一步設定您的執行個體，請選擇 Next: Configure Instance Details (下一步：設定執行個體詳細資訊)。

**步驟 3：設定執行個體詳細資訊**

在 Configure Instance Details (設定執行個體詳細資訊) 頁面上，根據需要變更下列設定 (展開 Advanced Details (進階詳細資訊) 以查看所有設定)，然後選擇 Next: Add Storage (下一步：新增儲存體)：

- Number of instances (執行個體數目)：輸入要啟動的執行個體數目。

**Tip**

為了確保執行個體能更快啟動，請將大型請求分成較小的批次。例如，分別建立五個啟動請求，每個請求各啟動 100 個執行個體，而不要以單一請求啟動 500 個執行個體。

- (選用) 若要協助確認您維持足以處理您應用程式的正確執行個體數目，您可以選擇 Launch into Auto Scaling Group (啟動到 Auto Scaling 群組) 以建立啟動組態及 Auto Scaling 群組。Auto Scaling 會根據您的規格擴展群組中的執行個體數目。如需詳細資訊，請參閱 [Amazon EC2 Auto Scaling 使用者指南](#)。

**Note**

如果 Amazon EC2 Auto Scaling 將 Auto Scaling 群組中的執行個體標記為「狀況不良」，將自動排程更換該執行個體，並終止該執行個體，啟動另一個執行個體，而且您會遺失原始執行個體上的資料。如果您停止或重新開機執行個體，或另一個事件將此執行個體標記為「狀況不良」，則執行個體會標記為「狀況不良」。如需詳細資訊，請參閱《Amazon EC2 Auto Scaling 使用者指南》中的 [Auto Scaling 執行個體的運作狀態檢查](#)。

- Purchasing option (購買選項)：選擇 Request Spot instances (請求 Spot 執行個體) 啟動 Spot 執行個體。這會從此頁面新增及移除選項。您可以選擇是否設定最高價 (不建議)，選擇是否變更請求類型、中斷行為及請求有效期限。如需詳細資訊，請參閱 [建立 Spot 執行個體請求](#)。

- Network (網路)：選取 VPC，或者若要建立新的 VPC，請選取 Create new VPC (建立新 VPC) 以前往 Amazon VPC 主控台。完成後，請返回啟動執行個體精靈並選擇 Refresh (重新整理)，然後載入清單中的 VPC。
- Subnet (子網)：您可以在與可用區域、Local Zone、Wavelength 區域或 Outpost 相關聯的子網中啟動執行個體。

若要在可用區域中啟動執行個體，請選取要在當中啟動執行個體子網。您可以選取 [無偏好設定]，讓您在任何可用區域中 AWS 選擇預設子網路。若要建立新的子網，請選擇 Create new subnet (建立新的子網)，以前往 Amazon VPC 主控台。當您完成時，請返回精靈並選擇 Refresh (重新整理)，將您的子網載入至清單中。

在 Local Zone 中啟動執行個體，選取您在 Local Zone 中建立的子網。

若要在 Outpost 中啟動執行個體，請在與 Outpost 相關聯的 VPC 中選取子網。

- Auto-assign Public IP (自動指派公有 IP)：指定您的執行個體是否會收到公有 IPv4 地址。根據預設，預設子網中的執行個體會收到公有 IPv4 地址，非預設子網中的執行個體則不會。您可以選取 Enable (啟用) 或 Disable (停用) 覆寫子網的預設設定。如需詳細資訊，請參閱 [公有 IPv4 地址](#)。
- Auto-assign IPv6 IP (自動指派 IPv6 IP)：指定您的執行個體是否會從子網範圍中收到 IPv6 地址。選取 Enable (啟用) 或 Disable (停用) 覆寫子網的預設設定。此選項僅適用於您已經將 IPv6 CIDR 區塊與您的 VPC 和子網相關聯。如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的[將 IPv6 CIDR 區塊新增至 VPC](#)。
- Hostname type (主機名稱類型)：選取執行個體的客體作業系統主機名稱是否要包含資源名稱或 IP 名稱。如需詳細資訊，請參閱 [Amazon EC2 執行個體主機名稱類型](#)。
- DNS Hostname (DNS 主機名稱)：決定資源名稱或 IP 名稱的 DNS 查詢 (根據您在 Hostname type (主機名稱類型) 所選的項目) 是以 IPv4 地址 (A 記錄)、IPv6 地址 (AAAA 記錄) 還是以兩者進行回應。如需詳細資訊，請參閱 [Amazon EC2 執行個體主機名稱類型](#)。
- 網域加入目錄：選取執行個體在啟動後加入的 AWS Directory Service 目錄 (網域)。若您選取網域，您必須選取具備必要許可的 IAM 角色。如需有關網域加入 Linux 執行個體的詳細資訊，請參閱[將 Linux EC2 執行個體無縫加入您的 AWS 受管 Microsoft AD 目錄](#)。如需有關網域加入 Windows 執行個體的詳細資訊，請[無縫加入 Windows EC2 執行個體](#)。
- Placement group (置放群組)：置放群組會判斷您執行個體的置放策略。選取現有的置放群組或建立新的置放群組。此選項僅適用於您已經選取支援置放群組的執行個體類型。如需詳細資訊，請參閱 [置放群組](#)。
- 容量預留：指定要在共用容量、任何 open 容量預留、特定的 容量預留 或 容量預留 群組中啟動執行個體。如需詳細資訊，請參閱 [使用現有的 容量預留 啟動執行個體](#)。

- IAM 角色：選取要與執行個體建立關聯的 AWS Identity and Access Management (IAM) 角色。如需詳細資訊，請參閱 [Amazon EC2 的 IAM 的角色](#)。
- CPU options (CPU 選項)：選擇 Specify CPU options (指定 CPU 選項) 在啟動期間指定自訂數量的 vCPU。設定 CPU 核心數量及每個核心的執行緒數量。如需詳細資訊，請參閱 [最佳化 CPU 選項](#)。
- Shutdown behavior (關機行為)：選取執行個體是否應於關機時停止或終止。如需詳細資訊，請參閱 [變更執行個體啟動的關機行為](#)。
- Stop - Hibernate behavior (停止 - 休眠行為)：若要啟用休眠，請選取此核取方塊。此選項只在您的執行個體符合休眠必要條件時才可使用。如需詳細資訊，請參閱 [休眠您的 Amazon EC2 實例](#)。
- Enable termination protection (啟用終止保護)：若要防止意外性的終止，請選取此核取方塊。如需詳細資訊，請參閱 [啟用終止保護](#)。
- Enable stop protection (啟用停止保護)：若要防止意外性的停止，請選取此核取方塊。如需詳細資訊，請參閱 [啟用停止保護](#)。
- 監控：選取此核取方塊可開啟使用 Amazon 對執行個體的詳細監控 CloudWatch。需支付額外費用。如需詳細資訊，請參閱 [使用監控執行個體 CloudWatch](#)。
- EBS 最佳化執行個體：Amazon EBS 最佳化執行個體使用最佳化組態堆疊，並為 Amazon EBS I/O 額外提供專屬的容量。如果執行個體類型支援此功能，請選取此核取方塊以啟用它。需支付額外費用。如需詳細資訊，請參閱 [Amazon EBS – 最佳化執行個體](#)。
- Tenancy (租用)：若您要在 VPC 中啟動您的執行個體，您可以選擇在隔離的專用硬體 (Dedicated (專用)) 上或專用執行個體 (Dedicated host (專用執行個體)) 上執行您的執行個體。可能需支付額外費用。如需詳細資訊，請參閱 [專用執行個體](#) 及 [Dedicated Hosts](#)。
- T2/T3 Unlimited (T2/T3 無限制)：選取此核取方塊，允許您的應用程式在需要的時候爆量並超過基線。可能需支付額外費用。如需詳細資訊，請參閱 [爆量效能執行個體](#)。
- 檔案系統：若要建立新的檔案系統來掛載至執行處理，請選擇 Create new file system (建立新檔案系統)，輸入新檔案系統的名稱，然後選擇 Create (建立)。檔案系統是使用 Amazon EFS Quick Create 建立而成的，這會套用服務建議的設定。啟用檔案系統存取所需的安全群組會自動建立並連接至檔案系統的執行處理和掛載目標。您也可以選擇手動建立並連接至必要的安全群組。若要將一個或多個現有的 Amazon EFS 檔案系統掛載至您的執行個體，請選擇 Add file system (新增檔案系統)，然後選擇要掛載的檔案系統和要使用的掛載點。如需詳細資訊，請參閱 [將 Amazon EFS 搭配 Linux 執行個體](#)。
- Network interfaces (網路介面)：若您選取特定的子網，您可以為您的執行個體指定最多兩個網路介面：
  - 在 [網路介面] 中，選取 [新增網路介面] 以 AWS 建立新的介面，或選取現有的可用網路介面。
  - 在 [主要 IP] 中，輸入子網路範圍內的私人 IPv4 位址，或保留 [自動指派 AWS] 讓您選擇私人 IPv4 位址。



- 針對 Secondary IP addresses (輔助 IP 地址)，選取 Add IP (新增 IP) 指派超過一個私有 IPv4 地址給選取的網路介面。
- (僅限 IPv6) 如果是 IPv6 IP，請選擇「新增 IP」，然後輸入子網路範圍內的 IPv6 位址，或保留「自動指派 AWS」讓您選擇一個 IP 位址。
- 網路卡索引：網路卡的索引。主要網路介面必須指派給網路卡索引 0。某些執行個體類型支援多個網路卡。
- 選擇 Add Device (新增裝置) 新增次要網路介面。次要網路介面可位於 VPC 的不同子網中，只要其位於與您的執行個體相同的可用區域內即可。

如需詳細資訊，請參閱 [彈性網路介面](#)。若您指定超過一個網路介面，您的執行個體便無法收到公有 IPv4 地址。此外，若您為 eth0 指定現有的網路介面，您便無法使用 Auto-assign Public IP (自動指派公有 IP) 來覆寫子網的公有 IPv4 設定。如需詳細資訊，請參閱 [在啟動執行個體期間指派公有 IPv4 地址](#)。

- Kernel ID (核心 ID)：(僅對全虛擬化 (PV) AMIs 有效) 除非您希望使用特定的核心，否則請選取 Use default (使用預設)。
- RAM disk ID (RAM 磁碟 ID)：(僅對全虛擬化 (PV) AMIs 有效) 除非您希望使用特定的 RAM 磁碟，否則請選取 Use default (使用預設)。若您已選取核心，則可能需要選取具有可支援之驅動程式的特定 RAM 磁碟。
- 「飛地區」：選取「啟用」以啟用「AWS 硝基隔離區」的執行個體。如需詳細資訊，請參閱 [什麼是 AWS 硝基飛地？](#) 在 AWS 硝基飛地用戶指南。
- 可存取的中繼資料：您可以啟用或停用對執行個體中繼資料服務 (IMDS) 的存取。如需詳細資訊，請參閱 [使用 IMDSv2](#)。
- 中繼資料 IPv6 端點：您可以讓執行個體使用 IMDS IPv6 位址 [fd00:ec2::254] 擷取執行個體中繼資料。只有當您將 [AWS Nitro 系統上建置的執行個體啟動到支援 IPv6 的子網路 \(僅限雙堆疊或 IPv6\)](#) 時，才能使用此選項。如需有關擷取執行個體中繼資料的詳細資訊，請參閱 [擷取執行個體中繼資料](#)。
- 中繼資料版本：如果啟用對 IMDS 的存取，則可以選擇請求執行處理中繼資料時請求使用執行個體中繼資料服務第 2 版。如需詳細資訊，請參閱 [設定新執行個體的執行個體中繼資料選項](#)。
- 中繼資料字符回應躍點限制：如果啟用 IMDS，可以為中繼資料字符設定允許的網路躍點數量。如需詳細資訊，請參閱 [使用 IMDSv2](#)。
- User data (使用者資料)：您可以指定使用者資料，以在啟動期間設定執行個體或執行組態指令碼。若要連接檔案，請選取 As file (做為檔案) 選項並瀏覽要連接的檔案。

## 步驟 4：新增儲存體

您選取的 AMI 包含一或多個儲存體磁碟區，包含根設備磁碟區。在 Add Storage (新增儲存體) 頁面上，您可以透過選擇 Add New Volume (新增新磁碟區)，指定要連接到執行個體的額外磁碟區。依下列方式設定每個磁碟區，然後選擇 Next: Add Tags (下一步：新增標籤)。

- **Type (類型)**：選取執行個體存放區或 Amazon EBS 磁碟區，以與您的執行個體相關聯。此清單中可用的磁碟區類型取決於您選擇的執行個體類型。如需詳細資訊，請參閱 [Amazon EC2 執行個體存放區](#) 和 [Amazon EBS 磁碟區](#)。
- **Device (裝置)**：從磁碟區可用的裝置名稱清單中選取。
- **Snapshot (快照)**：輸入要還原磁碟區的快照名稱或 ID。您也可以在此 Snapshot (快照) 欄位中輸入文字，搜尋可用的共享和公有快照。快照說明會區分大小寫。
- **Size (大小)**：針對 EBS 磁碟區，您可以指定儲存體大小。即使您已選取符合免費方案資格的 AMI 和執行個體，若要維持在免費方案中，您的總儲存體必須低於 30 GiB。
- **Volume Type (磁碟區類型)**：針對 EBS 磁碟區，選取磁碟區類型。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的 Amazon EBS 磁碟區類型](#)。
- **IOPS**：如果您已選取 Provisioned IOPS SSD 磁碟區類型，就可以輸入磁碟區能支援的每秒 I/O 操作數 (IOPS)。
- **Delete on Termination (終止時刪除)**：針對 Amazon EBS 磁碟區，選取此核取方塊以在執行個體終止時刪除磁碟區。如需詳細資訊，請參閱 [在執行個體終止時保留資料](#)。
- **Encrypted (加密)**：如果執行個體類型支援 EBS 加密，您可以指定磁碟區的加密狀態。如果在此區域中預設啟用加密，則會為您選取預設客戶受管金鑰。您可以選取不同的金鑰或停用加密。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的 Amazon EBS 加密](#)。

## 步驟 5：新增標籤

在 Add Tags (新增標籤) 頁面上，透過提供鍵和值的組合，指定 [標籤](#)。您可以為執行個體、磁碟區或兩者建立標籤。對於 Spot 執行個體，您只能標記 Spot 執行個體請求。選擇 Add another tag (新增另一個標籤) 以為您的資源新增超過一個標籤。當完成時，請選擇 Next: Configure Security Group (下一步：設定安全群組)。

## 步驟 6：設定安全群組

在 Configure Security Group (設定安全群組) 頁面上，使用安全群組定義您執行個體的防火牆規則。這些規則會指定應交付至您執行個體的傳入網路流量。所有其他流量都會遭到忽略。(如需安全群組的詳細資訊，請參閱 [適用於 Amazon EC2 執行個體的亞馬遜 EC2 安全群組](#)。) 選取或建立安全群組，如下所示，然後選取 Review and Launch (檢閱和啟動)。

- 若要選取現有的安全群組，請選取 **Select an existing security group** (選取現有安全群組)，然後選取您的安全群組。您無法編輯現有安全群組的規則，但您可以透過選擇 **Copy to new** (複製到新的群組)，將它們複製到新的群組。接著您便可以新增規則，如下步驟中所述。
- 若要建立新的安全群組，請選擇 **Create a new security group** (建立新安全群組)。精靈會自動定義啟動精靈 x 安全性群組，並建立輸入規則，讓您連線至執行個體。Linux 執行個體針對安全殼層 (連接埠 22) 使用輸入規則，而 Windows 執行個體則使用 RDP (連接埠 3389) 的輸入規則。
- 您可以新增符合您需求的規則。例如，若您的執行個體為 Web 伺服器，請開啟連接埠 80 (HTTP) 和 443 (HTTPS) 以允許網際網路流量。

若要新增規則，請選取 **Add Rule** (新增規則)，選取要向網路流量開啟的通訊協定，然後指定來源。從 **Source** (來源) 清單中選擇 **My IP** (我的 IP)，讓精靈新增您電腦的公有 IP 地址。但是，如果您透過 ISP 或是從防火牆後方進行連線，不具備靜態 IP 地址，則需要找到用戶端電腦使用的 IP 地址範圍。

#### Warning

您可以在此簡短的練習中使用讓所有 IP 地址 (0.0.0.0/0) 透過 SSH 或 RDP 存取您執行個體的規則，但在生產環境中則並不安全。建議您僅授權特定 IP 地址或地址範圍存取您的執行個體。

### 步驟 7：檢閱執行個體啟動並選取金鑰對

在 **Review Instance Launch** (檢閱執行個體啟動) 頁面上，檢查您執行個體的詳細資訊，並透過選擇適當的 **Edit** (編輯) 連結進行任何必要的變更。

當您就緒後，選擇 **Launch** (啟動)。

在選取現有金鑰對或建立新的金鑰對對話方塊中，您可以選取現有的金鑰對或建立新的金鑰對。例如，選取 **Choose an existing key pair** (選取現有金鑰對)，然後選取您在設定時建立的金鑰對。如需詳細資訊，請參閱 [Amazon EC2 金鑰配對](#) 和 [Amazon EC2 執行個體](#)。

#### Important

若您選擇繼續而不使用金鑰對選項，您將無法連線到執行個體，除非您選擇已設定為允許使用者透過其他方式登入的 AMI。

若要啟動執行個體，請選取確認核取方塊，然後選取啟動執行個體。

(選用) 您可以建立執行個體的狀態檢查警示 (可能需支付額外費用)。在確認畫面上，選擇 **Create status check alarms** (建立狀態檢查警示)，然後遵循指示進行。還可以在啟動執行個體後建立狀態檢查警示。如需詳細資訊，請參閱 [建立和編輯狀態檢查警示](#)。

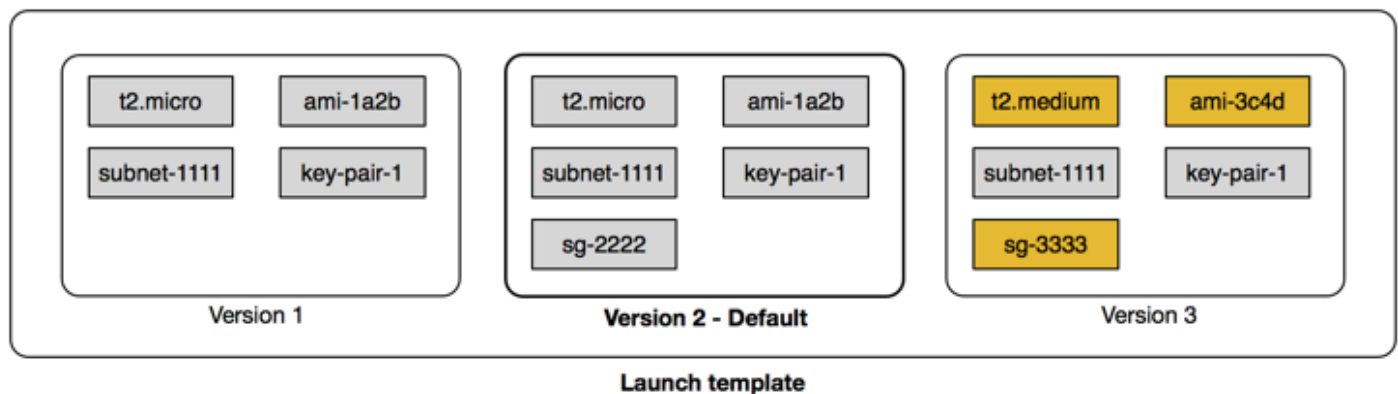
如果執行個體無法啟動或狀態立即進入 `terminated` 而非 `running`，請參閱 [針對執行個體啟動問題進行故障診斷](#)。

## 從啟動範本啟動執行個體

您可以使用啟動範本來儲存執行個體啟動參數，這樣您就不必在每次啟動執行個體時指定它們。例如，您可以使用您通常用於啟動執行個體的 AMI ID、執行個體類型和網路設定來建立啟動範本。使用 Amazon EC2 主控台、AWS SDK 或命令列工具啟動執行個體時，您可以指定啟動範本，而不必再次輸入參數。

針對每個啟動範本，您可以建立一或多個已編號的 啟動範本版本。每個版本都可以有不同的啟動參數。當您從啟動範本啟動執行個體時，可以使用任意版本的啟動範本。如果您未指定版本，即使用預設版本。您可以設定任意版本的啟動範本做為預設—版本，而該版本即預設為啟動範本的第一個版本。

下圖顯示具有三個版本的啟動範本。第一個版本指定用來啟動執行個體的執行個體類型、AMI ID、子網和金鑰對。第二個版本以第一個版本為基礎，同時指定執行個體的安全群組。第三個版本針對部分參數使用不同的值。第 2 版設定為預設版本。如果您已從此啟動範本啟動執行個體，則會在未指定其他版本時使用第 2 版的啟動參數。



## 目錄

- [啟動範本限制](#)
- [使用 IAM 許可控制啟動範本的存取權](#)
- [使用啟動範本來控制啟動執行個體](#)
- [建立啟動範本](#)
- [修改啟動範本 \(管理啟動範本版本\)](#)

- [刪除啟動範本](#)
- [從啟動範本啟動執行個體](#)

## 啟動範本限制

啟動範本和啟動範本版本的使用規則如下：

- **配額** — 若要檢視啟動範本的配額和啟動範本版本，請開啟 [Service Quotas](#) 主控台或使用 [list-service-quotas](#) AWS CLI 指令。每個 AWS 帳戶每個區域最多可以有 5,000 個啟動範本，每個啟動範本最多可有 10,000 個版本。您帳戶的配額可能不太一樣，視年齡和使用記錄而定。
- **參數是選用值** – 啟動範本參數是選用的。不過，您必須確定啟動執行個體的請求包括所有必要參數。例如，如果您的啟動範本未包括 AMI ID，則必須在啟動執行個體時同時指定啟動範本和 AMI ID。
- **參數尚未驗證** – 在您建立啟動範本時，啟動範本參數尚未完整驗證。如果您為參數指定了不正確的數值，或是未使用支援的參數組合，則無法使用此啟動範本啟動任何執行個體。請確定您指定正確的參數值，以及您使用支援的參數組合。例如，若要在置放群組中啟動執行個體，您必須指定支援的執行個體類型。
- **標籤** – 您可以為啟動範本建立標籤，但無法為啟動範本版本建立標籤。
- **固定** – 啟動範本是不可變的。若要修改啟動範本，您必須建立新版本的啟動範本。
- **版本編號** – 啟動範本版本會依其建立順序進行編號。當您建立啟動範本版本時，無法自行指定版本號碼。

## 使用 IAM 許可控制啟動範本的存取權

可使用 IAM 許可控制使用者可以執行哪些啟動範本動作，例如檢視、建立或刪除啟動範本。

當您授與使用者建立啟動範本和啟動範本版本的權限時，您無法使用資源層級權限來限制他們可以在啟動範本中指定的資源。因此，請確定您僅授與適當管理員建立啟動範本和啟動範本版本的權限。

您必須將使用啟動範本的任何人授與建立和存取啟動範本中指定之資源所需的權限。例如：

- 若要從共用私有 Amazon 機器映像 (AMI) 啟動執行個體，使用者必須擁有 AMI 的啟動權限。
- 若要使用現有快照的標籤建立 EBS 磁碟區，使用者必須擁有快照的讀取權限，以及建立和標記磁碟區的權限。

## 目錄

- [ec2 : CreateLaunch模板](#)
- [ec2 : DescribeLaunch模板](#)
- [ec2 : DescribeLaunchTemplateVersions](#)
- [ec2 : DeleteLaunch模板](#)
- [控制版本控制許可](#)
- [控制啟動範本上標籤的存取權](#)

## ec2 : CreateLaunch模板

若要在主控台中建立啟動範本或使用 API，主體必須具有 IAM 政策中的 `ec2:CreateLaunchTemplate` 許可。盡可能使用標籤來協助控制帳戶中啟動範本的存取權。

例如，下列 IAM 政策陳述式僅在範本使用指定標籤 (*purpose=testing*) 時才授予主體建立啟動範本的許可。

```
{
 "Sid": "IAMPolicyForCreatingTaggedLaunchTemplates",
 "Action": "ec2:CreateLaunchTemplate",
 "Effect": "Allow",
 "Resource": "*",
 "Condition": {
 "StringEquals": {
 "aws:ResourceTag/purpose": "testing"
 }
 }
}
```

建立啟動範本的主體可能需要一些相關許可，例如：

- `ec2:CreateTags` — 要在 `CreateLaunchTemplate` 操作期間向啟動模板添加標籤，`CreateLaunchTemplate` 調用者必須在 IAM 策略中具有 `ec2:CreateTags` 許可。
- `ec2:RunInstances` — 若要從他們建立的啟動範本啟動 EC2 執行個體，主體還必須具有 IAM 政策中的 `ec2:RunInstances` 許可。

針對套用標籤的資源建立動作，使用者必須具有 `ec2:CreateTags` 許可。如下 IAM 政策陳述式使用 `ec2:CreateAction` 條件索引鍵允許使用者僅在 `CreateLaunchTemplate` 的內容中建立標籤。使用者無法為現有啟動範本或任何其他資源套用標籤。如需詳細資訊，請參閱 [在建立期間授予標籤資源的許可](#)。

```
{
 "Sid": "IAMPolicyForTaggingLaunchTemplatesOnCreation",
 "Action": "ec2:CreateTags",
 "Effect": "Allow",
 "Resource": "arn:aws:ec2:region:account-id:launch-template/*",
 "Condition": {
 "StringEquals": {
 "ec2:CreateAction": "CreateLaunchTemplate"
 }
 }
}
```

建立啟動範本的 IAM 使用者不會自動擁有使用其所建立啟動範本的許可。與任何其他主體一樣，啟動範本建立者必須透過 IAM 政策取得許可。如果 IAM 使用者想要從啟動範本啟動 EC2 執行個體，他們必須具有 `ec2:RunInstances` 許可。授予這些許可時，您可以指定使用者只能使用具有特定標籤或特定 ID 的啟動範本。您也可以控制 AMI 和其他資源，讓使用啟動範本的任何人都可以在啟動執行個體時參照和使用，方法是指定 `RunInstances` 呼叫的資源層級許可。如需範例政策，請參閱 [啟動範本](#)。

#### ec2 : DescribeLaunch模板

若要列出帳戶中的啟動範本，主體必須具有 IAM 政策中的 `ec2:DescribeLaunchTemplates` 許可。因為 `Describe` 動作不支援資源層級許可，您必須在沒有條件的情況下指定它們，且政策中資源元素的值必須為 `"*"`。

例如，下列 IAM 政策陳述式會授予主體許可，以列出帳戶中的所有啟動範本。

```
{
 "Sid": "IAMPolicyForDescribingLaunchTemplates",
 "Action": "ec2:DescribeLaunchTemplates",
 "Effect": "Allow",
 "Resource": "*"
}
```

#### ec2 : DescribeLaunchTemplateVersions

檢視啟動範本的主體也應具有擷取組成啟動範本之整組屬性的 `ec2:DescribeLaunchTemplateVersions` 許可。

若要列出帳戶中的啟動範本版本，主體必須具有 IAM 政策中的 `ec2:DescribeLaunchTemplateVersions` 許可。因為 `Describe` 動作不支援資源層級許可，您必須在沒有條件的情況下指定它們，且政策中資源元素的值必須為 `"*"`。

例如，下列 IAM 政策陳述式會授予主體許可，以列出帳戶中的所有啟動範本。

```
{
 "Sid": "IAMPolicyForDescribingLaunchTemplateVersions",
 "Effect": "Allow",
 "Action": "ec2:DescribeLaunchTemplateVersions",
 "Resource": "*"
}
```

## ec2 : DeleteLaunch 模板

### Important

在授予主體刪除資源的許可時，您應該小心謹慎。刪除啟動範本可能會導致依賴啟動範本的 AWS 資源失敗。

若要刪除啟動範本，主體必須具有 IAM 政策中的 `ec2:DeleteLaunchTemplate` 許可。盡可能使用以標籤為基礎的條件索引鍵以限制許可。

例如，下列 IAM 政策陳述式僅在範本使用指定標籤 (*purpose=testing*) 時才授予主體刪除啟動範本的許可。

```
{
 "Sid": "IAMPolicyForDeletingLaunchTemplates",
 "Action": "ec2:DeleteLaunchTemplate",
 "Effect": "Allow",
 "Resource": "*",
 "Condition": {
 "StringEquals": {
 "aws:ResourceTag/purpose": "testing"
 }
 }
}
```

或者，您可以使用 ARN 來辨識 IAM 政策適用的啟動範本。



啟動範本具有下列 ARN。

```
"Resource": "arn:aws:ec2:region:account-id:launch-template/lt-09477bcd97b0d310e"
```

您可以透過將其包含在清單中來指定多個 ARN，也可以指定 "\*" 的 Resource 值 (沒有 Condition 元素)，以允許主體者刪除帳戶中的任何啟動範本。

### 控制版本控制許可

對於受信任的管理員，您可以使用類似下列範例的 IAM 政策，授予建立和刪除啟動範本版本的存取權，以及變更啟動範本預設版本的存取權。

#### Important

授與主參與者建立啟動範本版本或修改啟動範本的權限時，請務必小心。

- 建立啟動範本版本時，會影響任何允許 Amazon EC2 代表您使用該 Latest 版本啟動執行個體的 AWS 資源。
- 修改啟動範本時，您可以變更哪個版本，因此會影響允許 Amazon EC2 以此修改版本代表您啟動執行個體的任何 AWS 資源。Default

您還需要謹慎處理與範本版本互Default動Latest或啟動範本版本的 AWS 資源的方式，例如 EC2 叢集和 Spot 叢集。當針對 Latest 或 Default 使用不同的啟動範本版本時，Amazon EC2 在啟動新執行個體以滿足機群的目標容量時，不會重新檢查完成動作所需的使用者許可，因此沒有使用者與 AWS 資源的互動。透過授與使用者呼叫 CreateLaunchTemplateVersion 和 ModifyLaunchTemplate API 的許可，如果使用者將機群指向含有執行個體設定檔 (IAM 角色的容器) 的不同啟動範本版本，則該使用者實際上也會獲得 iam:PassRole 許可。這表示使用者可能會更新啟動範本，以將 IAM 角色傳遞給執行個體 (即使執行個體沒有 iam:PassRole 許可)。在將許可授予可建立和管理啟動範本版本的使用者時保持謹慎，即可妥善管理此風險。

ec2 : CreateLaunchTemplateVersion

若要建立啟動範本的新版本，主體必須具有 IAM 政策中啟動範本的 ec2:CreateLaunchTemplateVersion 許可。

例如，下列 IAM 政策陳述式僅在版本使用指定標籤 (*environment=production*) 時才授予主體建立啟動範本版本的許可。或者，您可以指定一或多個啟動範本 ARN，也可以指定 "\*" 的 Resource 值 (沒有 Condition 元素)，以允許主體在帳戶中建立任何啟動範本的版本。

```
{
 "Sid": "IAMPolicyForCreatingLaunchTemplateVersions",
 "Action": "ec2:CreateLaunchTemplateVersion",
 "Effect": "Allow",
 "Resource": "*",
 "Condition": {
 "StringEquals": {
 "aws:ResourceTag/environment": "production"
 }
 }
}
```

#### ec2 : DeleteLaunchTemplateVersion

##### Important

一如往常，在授予委託人刪除資源的許可時，您應該小心謹慎。刪除啟動範本版本可能會導致依賴啟動範本版本的 AWS 資源失敗。

若要刪除啟動範本版本，主體必須具有 IAM 政策中啟動範本的 `ec2:DeleteLaunchTemplateVersion` 許可。

例如，下列 IAM 政策陳述式僅在版本使用指定標籤 (*environment=production*) 時才授予主體刪除啟動範本版本的許可。或者，您可以指定一或多個啟動範本 ARN，也可以指定 "\*" 的 Resource 值 (沒有 Condition 元素)，以允許主體在帳戶中刪除任何啟動範本的版本。

```
{
 "Sid": "IAMPolicyForDeletingLaunchTemplateVersions",
 "Action": "ec2:DeleteLaunchTemplateVersion",
 "Effect": "Allow",
 "Resource": "*",
 "Condition": {
 "StringEquals": {
 "aws:ResourceTag/environment": "production"
 }
 }
}
```

```
}

```

## ec2 : ModifyLaunchTemplate

若要變更與啟動範本相關聯的 Default 版本，主體必須具有 IAM 政策中啟動範本的 `ec2:ModifyLaunchTemplate` 許可。

例如，下列 IAM 政策陳述式僅在啟動範本使用指定標籤 (*`environment=production`*) 時才授予主體修改啟動範本的許可。或者，您可以指定一或多個啟動範本 ARN，也可以指定 "\*" 的 Resource 值 (沒有 Condition 元素)，以允許主體在帳戶中修改任何啟動範本。

```
{
 "Sid": "IAMPolicyForModifyingLaunchTemplates",
 "Action": "ec2:ModifyLaunchTemplate",
 "Effect": "Allow",
 "Resource": "*",
 "Condition": {
 "StringEquals": {
 "aws:ResourceTag/environment": "production"
 }
 }
}
```

## 控制啟動範本上標籤的存取權

當資源為啟動範本時，您可以使用條件索引鍵來限制標記許可。例如，下列 IAM 政策只允許從指定帳戶或區域中的啟動範本移除具有 *`temporary`* 索引鍵的標籤。

```
{
 "Sid": "IAMPolicyForDeletingTagsOnLaunchTemplates",
 "Action": "ec2:DeleteTags",
 "Effect": "Allow",
 "Resource": "arn:aws:ec2:region:account-id:launch-template/*",
 "Condition": {
 "ForAllValues:StringEquals": {
 "aws:TagKeys": [temporary]
 }
 }
}
```

如需可用於控制可套用至 Amazon EC2 資源之標籤索引鍵和值的條件索引鍵的詳細資訊，請參閱 [控制對特定標籤的存取](#)。

## 使用啟動範本來控制啟動執行個體

如果您使用啟動範本，您可以指定使用者只能使用啟動執行個體，而且使用者只能使用特定的啟動範本。您也可以控制哪些人員可以建立、修改、描述與刪除啟動範本和啟動範本版本。

## 使用啟動範本控制啟動參數

啟動範本可以包含啟動執行個體的所有或部分參數。當您使用啟動範本啟動執行個體時，您可以覆寫啟動範本中指定的參數。或者，您可以指定啟動範本中沒有的其他參數。

### Note

您無法在啟動期間移除啟動範本參數 (例如，您無法為參數指定 null 值)。若要移除參數，請在沒有參數的情況下建立新版本的啟動範本，並使用該版本來啟動執行個體。

若要啟動執行個體，使用者必須具備使用 `ec2:RunInstances` 動作的許可。使用者也必須具備許可，才能建立或使用由該執行個體所建立或與其相關聯的資源。您可以使用 `ec2:RunInstances` 動作上的資源層級許可，以控制使用者可指定的啟動參數。或者，您可以將使用啟動範本啟動執行個體的許可授予使用者。這可讓您在啟動範本中 (而非在 IAM 政策中) 管理啟動參數，而且可讓您使用啟動範本做為啟動執行個體的授權媒介。例如，您可以指定使用者只能使用啟動範本來啟動執行個體，而且使用者只能使用特定的啟動範本。您也可以控制使用者能在啟動範本中覆寫的啟動參數。如需範例政策，請參閱 [啟動範本](#)。

## 控制啟動範本的使用

根據預設，使用者不具使用啟動範本的許可。您可以建立使用者政策，授予使用者建立、修改、描述與刪除啟動範本和啟動範本版本的許可。您也可以將資源層級許可套用至一些啟動範本動作，藉以控制使用者將特定資源用於那些動作的能力。如需詳細資訊，請參閱下列範例政策：[範例：使用啟動範本](#)。

將使用 `ec2:CreateLaunchTemplate` 和 `ec2:CreateLaunchTemplateVersion` 動作的許可授予使用者時請小心。您無法使用資源層級許可來控制使用者能於啟動範本中指定的資源。若要限制用來啟動執行個體的資源，請確定您只將建立啟動範本和啟動範本版本的許可授予合適的管理員。

## 搭配 EC2 機群或 Spot 機群使用啟動範本時的重要安全考量

若要使用啟動範本，您必須授予使用者許可，以便建立、修改、描述和刪除啟動範本及啟動範本版本。透過控制對 `ec2:CreateLaunchTemplate` 和 `ec2:CreateLaunchTemplateVersion` 動作的存取權，您可以控制誰可以建立啟動範本和啟動範本版本。您也可以透過控制 `ec2:ModifyLaunchTemplate` 動作的存取權來控制可修改啟動範本的對象。

### Important

如果 EC2 機群或 Spot 機群設定為使用「最新」或「預設」啟動範本版本，則機群不會知道「最新」或「預設」稍後是否會變更為不同的啟動範本版本。當「最新」或「預設」使用不同的啟動範本版本時，Amazon EC2 在啟動新執行個體以滿足機群的目標容量時，不會重新檢查完成動作所需的許可。將許可授與可建立和管理啟動範本版本的使用者 (尤其是允許使用者變更「預設」啟動範本版本的 `ec2:ModifyLaunchTemplate` 動作) 時，請務必考量到這點。

透過授與使用者針對啟動範本 API 使用 EC2 動作的許可，如果使用者建立或更新 EC2 機群或 Spot 機群以指向含有執行個體設定檔 (IAM 角色的容器) 的不同啟動範本版本，則該使用者實際上也會獲得 `iam:PassRole` 許可。這表示使用者可能會更新啟動範本，以將 IAM 角色傳遞給執行個體 (即使執行個體沒有 `iam:PassRole` 許可)。如需詳細資訊和範例 IAM 政策，請參閱《IAM 使用者指南》中的 [利用 IAM 角色來授予許可給在 Amazon EC2 執行個體上執行的應用程式](#)。

如需詳細資訊，請參閱 [控制啟動範本的使用](#) 及 [範例：使用啟動範本](#)。

## 建立啟動範本

使用您定義的參數建立啟動範本，或使用現有的啟動範本或執行個體做為新啟動範本的基礎。

### 任務

- [從參數建立啟動範本](#)
- [從現有啟動範本建立啟動範本](#)
- [從執行個體建立啟動範本](#)
- [使用 Systems Manager 參數而非 AMI ID](#)

## 從參數建立啟動範本

若要建立啟動範本，您必須指定啟動範本名稱和至少一個執行個體組態參數。

### 控制台方向

若要使用主控台建立啟動範本

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Launch Templates (啟動範本) 接著選擇 Create launch template (建立啟動範本)。

3. 啟動範本參數會被分組。有關每個組的詳細信息，請參閱以下部分。
4. 使用「摘要」面板來檢閱您的啟動範本組態。您可以通過選擇其鏈接導航到任何部分，然後進行任何必要的更改。
5. 當您準備好建立啟動範本時，請選擇 Create launch template (建立啟動範本)。

### 啟動範本名稱、說明和標籤

1. 針對 Launch template name (啟動範本名稱)，輸入啟動範本的描述性名稱。
2. 在 Template version description (範本版本描述) 中，提供此啟動範本版本的簡短描述。
3. 若要在建立時標記啟動範本，請展開 Template tags (範本標籤)，接著選擇 Add Tag (新增標籤)，然後輸入標籤鍵值組。為每個要新增的其他標籤重新選擇 Add tag (新增標籤)。

#### Note

要標記啟動執行個體時建立的資源，您必須在 Resource tags (資源標籤) 下指定標籤。如需詳細資訊，請參閱 [資源標籤](#)。

### 應用程式和作業系統映像 (Amazon Machine Image)

Amazon Machine Image (AMI) 包含建立執行個體所需的資訊。例如，AMI 可能包含做為網路伺服器所需的軟體，例如 Linux、Apache 和您的網站。

您可以尋找適當的 AMI，如下所示。對於尋找 AMI 的每個選項，可以選擇右上方的 Cancel (取消)，以返回啟動範本，而不選擇 AMI。

#### 搜尋列

若要搜尋所有可用的 AMI，請在 AMI 搜尋列中輸入關鍵字，然後按 Enter 鍵。選取 Select (選取) 來選取 AMI。

#### Recents (最近)

最近使用的 AMI。

選取 Recently launched (最近啟動) 或 Currently in use (目前使用中)，然後從 Amazon Machine Image (AMI) 中，選取 AMI。

#### My AMIs (我的 AMI)

您擁有的私有 AMI，或是與您共享的私有 AMI。

選取 Owned by me (由我擁有) 或 Shared with me (與我共享)，然後從 Amazon Machine Image (AMI) 中，選取 AMI。

## Quick Start

AMI 會依作業系統分組，以協助您快速開始使用。

首先選取您需要的操作系統，然後從 Amazon Machine Image (AMI) 中，選取 AMI。若要選取具有免費方案資格的 AMI，請確定 AMI 已標示為 Free tier eligible (具有免費方案資格)。

## Browse more AMIs (瀏覽更多 AMI)

選擇 Browse more AMIs (瀏覽更多 AMI) 以瀏覽完整的 AMI 目錄。

- 若要搜尋所有可用的 AMI，請在搜尋列中輸入關鍵字，然後按 Enter 鍵。
- 若要使用 Systems Manager 參數尋找 AMI，請選擇搜尋列右側的箭頭按鈕，然後選擇 Search by Systems Manager parameter (依 Systems Manager 參數搜尋)。如需詳細資訊，請參閱 [使用 Systems Manager 參數尋找 AMI](#)。
- 若要指定在從啟動範本中啟動執行個體時解析為 AMI 的 Systems Manager 參數，請選擇搜尋列右側的箭頭按鈕，然後選擇指定自訂值/Systems Manager 參數。如需詳細資訊，請參閱 [使用 Systems Manager 參數而非 AMI ID](#)。
- 若要依類別搜尋，請選擇 Quickstart AMIs (AMI 快速入門)、My AMIs (我的 AMI)、AWS Marketplace AMI, 或 Community AMIs (社群 AMI)。

這 AWS Marketplace 是一個在線商店，您可以在其中購買運行的軟件 AWS，包括 AMI。如需有關從啟動執行個體的詳細資訊 AWS Marketplace，請參閱 [啟動 AWS Marketplace 執行個體](#)。在社群 AMI 中，您可以找到 AWS 社群成員提供給其他人使用的 AMI。來自 Amazon 或已驗證的合作夥伴的 AMI 標記為 Verified provider (已驗證的供應商)。

- 若要篩選 AMI 清單，請在畫面左側的 Refine results (精簡結果) 下選取一或多個核取方塊。篩選條件選項會有所不同，取決於選取的搜尋類別。
- 檢查每個 AMI 列出的 Root device type (根設備類型)。注意哪些 AMI 是您需要的類型：ebs (Amazon EBS 後端) 或 instance-store (執行個體存放區後端)。如需詳細資訊，請參閱 [根設備儲存](#)。
- 檢查每個 AMI 列出的 Virtualization type (虛擬化類型)。注意哪些 AMI 是您需要的類型：hvm 或 paravirtual。例如，有些執行個體類型需要 HVM。如需詳細資訊，請參閱 [AMI 虛擬化類型](#)。
- 檢查每個 AMI 列出的開機模式。注意哪些 AMI 使用您需要的開機模式：legacy-bios、uefi 或 uefi-preferred。如需詳細資訊，請參閱 [Amazon EC2 啟動模式](#)。
- 選取符合您需求的 AMI，然後選取 Select (選取)。

## 執行個體類型

執行個體類型定義執行個體的硬體組態和大小。較大的執行個體類型具有較多的 CPU 和記憶體。如需詳細資訊，請參閱 [Amazon EC2 執行個體類型](#)。

對於 Instance type (執行個體類型)，您可以選取執行個體類型，也可以指定執行個體屬性，並讓 Amazon EC2 利用這些屬性識別執行個體類型。

### Note

只有在使用 Auto Scaling 群組、EC2 Fleet 和 Spot Fleet 啟動執行個體時，才支援指定執行個體屬性。如需詳細資訊，請參閱 [使用屬性型執行個體類型選取來建立 Auto Scaling 群組](#)、[EC2 Fleet 的屬性型執行個體類型選擇](#)，以及 [Spot Fleet 的屬性型執行個體類型選擇](#)。如果您打算在啟動 [執行個體精靈中或搭配 RunInstancesAPI 使用啟動範本](#)，則必須選取執行個體類型。

- Instance type (執行個體類型)：確定執行個體類型與您指定的 AMI 相容。如需詳細資訊，請參閱 [Amazon EC2 執行個體類型](#)。
- Compare instance types (比較執行個體類型)：您可以依下列屬性比較不同的執行個體類型：vCPU 數目、架構、記憶體數量 (GiB)、儲存數量 (GB)、儲存類型，以及網路效能。
- 取得建議：您可以從 Amazon Q EC2 執行個體類型選取器取得執行個體類型的指導和建議。如需詳細資訊，請參閱 [取得新工作負載的執行個體類型建議](#)。
- Advanced (進階)：若要指定執行個體屬性，並讓 Amazon EC2 利用這些屬性識別執行個體類型，請選擇 Advanced (進階)，然後選擇 Specify instance type attributes (指定執行個體類型屬性)。
  - Number of vCPUs (vCPU 數量)：輸入運算需求的 vCPU 數量下限和上限。若要表示沒有限制，請輸入下限 0，並將上限保留為空白。
  - Amount of memory (MiB) (記憶體數量 (MiB))：輸入運算需求的記憶體數量 (MiB) 下限和上限。若要表示沒有限制，請輸入下限 0，並將上限保留為空白。
  - 展開 Optional instance type attributes (選用執行個體類型屬性)，然後選擇 Add attribute (新增屬性)，以更詳細地表達您的運算需求。如需每個屬性的相關資訊，請參閱 [InstanceRequirements](#) 請參閱 Amazon EC2 API 參考中的請求。
- Resulting instance types (產生的執行個體類型)：您可以預覽符合所指定屬性的執行個體類型。若要排除執行個體類型，請選擇 Add attribute (新增屬性)，然後從 Attribute (屬性) 清單中，選擇 Excluded instance types (排除的執行個體類型)。從 Attribute value (屬性值) 清單中，選取要排除的執行個體類型。



## 金鑰對 (登入)

執行個體的金鑰對。

針對 Key pair name (金鑰對名稱)，選擇現有的金鑰對，或選擇 Create new key pair (建立新的金鑰對) 以建立新的金鑰對。如需詳細資訊，請參閱 [Amazon EC2 金鑰配對](#) 和 [Amazon EC2 執行個體](#)。

## Network settings (網路設定)

視需要設定網路設定。

- Subnet (子網)：您可以在與可用區域、Local Zone、Wavelength 區域或 Outpost 相關聯的子網中啟動執行個體。

若要在可用區域中啟動執行個體，請選取要在當中啟動執行個體子網。若要建立新的子網，請選擇 Create new subnet (建立新的子網)，以前往 Amazon VPC 主控台。當您完成時，請返回精靈並選擇 Refresh (重新整理) 圖示，將您的子網載入清單中。

在 Local Zone 中啟動執行個體，選取您在 Local Zone 中建立的子網。

若要在 Outpost 中啟動執行個體，請在與 Outpost 相關聯的 VPC 中選取子網。

- Firewall (security groups) (防火牆 (安全群組))：使用一或多個安全群組定義執行個體的防火牆規則。這些規則會指定應交付至您執行個體的傳入網路流量。所有其他流量都會遭到忽略。如需安全群組的詳細資訊，請參閱 [適用於 Amazon EC2 執行個體的亞馬遜 EC2 安全群組](#)。

如果新增網路介面，您必須在網路介面中指定相同的安全群組。

選取或建立安全群組，如下所示：

- 若要選取現有的安全群組，請選取 Select existing security group (選取現有安全群組)，然後從 Common security groups (常見安全群組) 選取您的安全群組。
- 若要建立新的安全群組，請選擇 Create security group (建立安全群組)。

您可以新增符合您需求的規則。例如，若您的執行個體為 Web 伺服器，請開啟連接埠 80 (HTTP) 和 443 (HTTPS) 以允許網際網路流量。

若要新增規則，請選擇 Add security group rule (新增安全群組規則)。對於 Type (類型)，選取網路流量類型。Protocol (通訊協定) 欄位會自動填入要對網路流量開啟的通訊協定。對於 Source type (來源類型)，選取來源類型。若要讓啟動範本新增您電腦的公有 IP 地址，請選擇 My IP (我的 IP)。但是，如果您透過 ISP 或是從防火牆後方進行連線，不具備靜態 IP 地址，則需要找到用戶端電腦使用的 IP 地址範圍。

**⚠ Warning**

如果您短暫啟動測試執行個體，而且很快就會將其停止或終止，則啟用所有 IP 地址 (0.0.0.0/0)，透過 SSH 或 RDP 存取您執行個體的規則是可接受的，但在生產環境中並不安全。建議您僅授權特定 IP 地址或地址範圍存取您的執行個體。

## • 進階網路組態：

## 網路介面

- Device index (裝置索引)：網路介面的裝置號碼，例如，主要網路介面為 eth0。如果您將此欄位保留空白，AWS 會建立主要網路介面。
- Network interface (網路介面)：選取 New interface (新介面)，讓 Amazon EC2 建立新的介面，或是選取現有的可用網路介面。
- Description (描述)：(選用) 新網路介面的描述。
- Subnet (子網)：要建立新網路介面的子網。對主要網路介面 (eth0) 而言，這是要啟動執行個體子網。如果您已輸入 eth0 的現有網路介面，則會在網路介面所在的子網中啟動執行個體。
- 安全群組：VPC 中要與網路介面建立關聯的一或多個安全性群組。
- Auto-assign public IP (自動指派公有 IP)：指定您的執行個體是否會收到公有 IPv4 地址。根據預設，預設子網中的執行個體會收到公有 IPv4 地址，非預設子網中的執行個體則不會。您可以選取 Enable (啟用) 或 Disable (停用) 覆寫子網的預設設定。如需詳細資訊，請參閱 [公有 IPv4 地址](#)。
- Primary IP (主要 IP)：您子網範圍內的私有 IPv4 地址。保留空白讓 Amazon EC2 為您選擇私有 IPv4 地址。
- Secondary IP (次要 IP)：來自子網範圍的一或多個其他私有 IPv4 地址。選擇 Manually assign (手動指派)，然後輸入 IP 地址。選擇 Add IP (新增 IP) 以新增另一個 IP 地址。或者，選擇 Automatically assign (自動指派)，讓 Amazon EC2 為您選擇一個，然後輸入一值以指示要新增的 IP 地址數目。
- (僅限 IPv6) IPv6 IPs (IPv6 IP)：子網範圍內的 IPv6 地址。選擇 Manually assign (手動指派)，然後輸入 IP 地址。選擇 Add IP (新增 IP) 以新增另一個 IP 地址。或者，選擇 Automatically assign (自動指派)，讓 Amazon EC2 為您選擇一個，然後輸入一值以指示要新增的 IP 地址數目。
- IPv4 Prefixes (IPv4 字首)：網路介面的 IPv4 字首。
- IPv6 Prefixes (IPv6 字首)：網路介面的 IPv6 字首。
- (選用) 指派主要 IPv6 IP：如果要將執行個體啟動至雙堆疊或僅限 IPv6 的子網路，您可選擇指派主要 IPv6 IP。指派主要 IPv6 地址可讓您避免中斷執行個體或 ENI 的流量。如果此執行個體依賴於其 IPv6 地址未改變，請選擇啟用。當您啟動執行個體時，AWS 會自動將與 ENI 相關聯的

IPv6 位址指派為主要 IPv6 位址。在啟用 IPv6 GUA 地址作為主要 IPv6 地址後，您便無法停用該地址。當您啟用 IPv6 GUA 地址作為主要 IPv6 地址時，第一個 IPv6 GUA 地址會成為主要 IPv6 地址，直到執行個體終止或網路介面中斷連接為止。如果有多個與執行個體連接之 ENI 相關聯的 IPv6 地址，並且啟用主要 IPv6 地址，則與 ENI 相關聯的第一個 IPv6 GUA 地址會成為主要 IPv6 地址。

- Delete on termination (終止時刪除)：是否在刪除執行個體時刪除網路介面。
- Elastic Fabric Adapter：指出網路介面是否為 Elastic Fabric Adapter。如需詳細資訊，請參閱 [the section called “Elastic Fabric Adapter”](#)。
- 網路卡索引：網路卡的索引。主要網路介面必須指派給網路卡索引 0。某些執行個體類型支援多個網路卡。
- ENA 快遞：ENA Express 由可 AWS 擴展的可靠數據報 (SRD) 技術提供支持。SRD 技術使用封包噴塗機制來分配負載並避免網路擁塞。啟用 ENA Express 便可支援的執行個體在可能的情況下，在一般 TCP 流量之上使用 SRD 進行通訊。除非您選取啟用或停用，否則啟動範本不會包含執行個體的 ENA Express 組態。
- ENA Express UDP：如果您已啟用 ENA Express，您可以選擇性用於 UDP 流量。除非您選取啟用或停用，否則啟動範本不會包含執行個體的 ENA Express 組態。

選擇 Add network interface (新增網路介面) 以新增更多網路介面。您可新增的網路介面數取決於選取執行個體類型支援的數量。其他網路介面可以位於相同 VPC 的不同子網路中，或位於您擁有的不同 VPC 中的子網路中 (只要子網路與您的執行個體位於相同的可用區域)。如果您在其他 VPC 中選取子網路，多重虛擬私人雲端標籤會顯示在您新增的網路介面旁邊。如此一來，您就能在不同 VPC 建立多重主目錄執行個體，分別使用不同的網路和安全組態。請注意，如果您從另一個 VPC 連接其他 ENI，則必須從該 VPC 為 ENI 選擇一個安全群組。

如需詳細資訊，請參閱 [彈性網路介面](#)。若您指定超過一個網路介面，您的執行個體便無法收到公有 IPv4 地址。此外，若您為 eth0 指定現有的網路介面，您便無法使用 Auto-assign Public IP (自動指派公有 IP) 來覆寫子網的公有 IPv4 設定。如需詳細資訊，請參閱 [在啟動執行個體期間指派公有 IPv4 地址](#)。

## 設定儲存

如果您為啟動範本指定 AMI，AMI 將包含一或多個儲存磁碟區，包含根磁碟區 (Volume 1 (AMI Root))。您可以指定要連接至執行個體的其他磁碟區。

您可以使用 Simple (簡單) 或 Advanced (進階) 檢視。搭配 Simple (簡單) 檢視，您可以指定磁碟區的大小和類型。若要指定所有磁碟區參數，請選擇卡片右上方的 Advanced (進階) 檢視。

若要新增磁碟區，請選擇 Add new volume (新增磁碟區)。

使用 Advanced (進階) 檢視，您可以設定每個磁碟區，如下所示：

- **Storage type (儲存體類型)**：要與您的執行個體相關聯的磁碟區類型 (EBS 或暫時性)。執行個體存放區 (暫時性) 磁碟區類型僅適用於您選取支援的執行個體類型。如需詳細資訊，請參閱 [Amazon EC2 執行個體存放區](#) 和 [Amazon EBS 磁碟區](#)。
- **Device name (設備名稱)**：從磁碟區可用的設備名稱清單中選取。
- **Snapshot (快照)**：選取要從中建立磁碟區的快照。您可以在 Snapshot (快照) 欄位中輸入文字，搜尋可用的共享和公有快照。
- **Size (GiB) (大小 (GiB))**：針對 EBS 磁碟區，您可以指定儲存體大小。如果已選取符合免費方案資格的 AMI 和執行個體，請注意，若要維持在免費方案中，您的總儲存體必須低於 30 GiB。
- **Volume Type (磁碟區類型)**：針對 EBS 磁碟區，選取磁碟區類型。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的 Amazon EBS 磁碟區類型](#)。
- **IOPS**：如果您選取佈建 IOPS SSD (io1 和 io2) 或一般用途 SSD (gp3) 磁碟區類型，則您可輸入磁碟區可支援的每秒輸入/輸出操作數 (IOPS)。對於 io1、io2 和 gp3 磁碟區，這是必要項目。gp2、st1、sc1 或標準磁碟區不支援此功能。如果您針對啟動範本省略此參數，則必須在從啟動範本啟動執行個體時為其指定一個值。
- **Delete on termination (終止時刪除)**：針對 Amazon EBS 磁碟區，請選擇 Yes (是) 以執行個體終止時刪除磁碟區，或選擇 No (否) 保留磁碟區。如需詳細資訊，請參閱 [在執行個體終止時保留資料](#)。
- **Encrypted (已加密)**：如果執行個體類型支援 EBS 加密，您可以選擇 Yes (是) 以啟用磁碟區的加密。如果您在此區域中預設啟用加密，則會為您啟用預設加密。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的 Amazon EBS 加密](#)。
- **KMS Key (KMS 金鑰)**：如果您為 Encrypted (已加密) 選取了 Yes (是)，則必須選取要用來加密磁碟區的客戶受管金鑰。如果在此區域中預設啟用加密，則會為您選取預設客戶受管金鑰。您可以選取不同的金鑰，或指定您已建立之任何客戶受管金鑰的 ARN。

## 資源標籤

若要標記啟動執行個體時建立的資源，在 Resource tags (資源標籤) 下，選擇 Add tag (新增標籤)，然後輸入標籤鍵值組。針對 Resource types (資源類型)，請指定建立時要標記的資源。您可以為所有資源指定相同的標籤，或者為不同的資源指定不同的標籤。為每個要新增的其他標籤重新選擇 Add tag (新增標籤)。

您可以針對使用啟動範本時建立的以下資源指定標籤：

- 執行個體
- 磁碟區
- Spot 執行個體請求
- 網路介面

#### Note

若要標記啟動範本本身，您必須在 Template tags (範本標籤) 下指定標籤。如需詳細資訊，請參閱 [啟動範本名稱、說明和標籤](#)。

### 進階詳細資訊

針對 Advanced Details (進階詳細資訊)，展開此區段來檢視欄位，指定執行個體的其他參數。

- Purchasing option (購買選項)：選擇 Request Spot Instances (請求 Spot 執行個體) 以 Spot 價格請求 Spot 執行個體，但是不超過隨需價格，並選擇 Customize (自訂) 變更預設的 Spot 執行個體設定。您可以設定最高價 (不建議)，並變更請求類型、請求持續時間和中斷行為。如不請求 Spot 執行個體，EC2 預設會啟動隨需執行個體。如需詳細資訊，請參閱 [Spot 執行個體](#)。
- IAM instance profile (IAM 執行個體設定檔)：選取要與執行個體建立關聯的 AWS Identity and Access Management (IAM) 執行個體設定檔。如需詳細資訊，請參閱 [Amazon EC2 的 IAM 的角色](#)。
- Hostname type (主機名稱類型)：選取執行個體的客體作業系統主機名稱是否要包含資源名稱或 IP 名稱。如需詳細資訊，請參閱 [Amazon EC2 執行個體主機名稱類型](#)。
- DNS Hostname (DNS 主機名稱)：決定資源名稱或 IP 名稱的 DNS 查詢 (根據您在 Hostname type (主機名稱類型) 所選的項目) 是以 IPv4 地址 (A 記錄)、IPv6 地址 (AAAA 記錄) 還是以兩者進行回應。如需詳細資訊，請參閱 [Amazon EC2 執行個體主機名稱類型](#)。
- Shutdown behavior (關機行為)：選取執行個體是否應於關機時停止或終止。如需詳細資訊，請參閱 [變更執行個體啟動的關機行為](#)。
- Stop - Hibernate behavior (停止 - 休眠行為)：若要啟用休眠，請選取 Enable (啟用)。此欄位僅對符合休眠事前準備的執行個體有效。如需詳細資訊，請參閱 [休眠您的 Amazon EC2 實例](#)。
- Termination protection (終止保護)：若要防止意外性的終止，請選取 Enable (啟用)。如需詳細資訊，請參閱 [啟用終止保護](#)。
- Stop protection (停止保護)：若要防止意外性的停止，請選擇 Enable (啟用)。如需詳細資訊，請參閱 [啟用停止保護](#)。

- 詳細 CloudWatch 監控：選擇啟用以啟用使用 Amazon 對執行個體進行詳細監控 CloudWatch。需支付額外費用。如需詳細資訊，請參閱 [使用監控執行個體 CloudWatch](#)。
- 彈性 GPU：Amazon 彈性圖形於 2024 年 1 月 8 日終止使用壽命。對於需要圖形加速的工作負載，我們建議您使用 Amazon EC2 G4ad、G4dn 或 G5 執行個體。
- Elastic inference (彈性推斷)：一個彈性推斷加速器，用於連接到您的 EC2 CPU 執行個體。如需詳細資訊，請參閱 Amazon Elastic Inference 開發人員指南中的 [Working with Amazon Elastic Inference](#)。

#### Note

自 2023 年 4 月 15 日起，不 AWS 會將新客戶加入 Amazon Elastic Inference (EI)，並協助目前客戶將工作負載遷移到提供更優惠價格和效能的選項。2023 年 4 月 15 日之後，新客戶將無法在 Amazon、Amazon ECS 或 Amazon Amazon EC2 使用亞馬遜 SageMaker EI 加速器啟動執行個體。但是，在過去 30 天內至少使用過一次 Amazon EI 的客戶將被視為目前客戶，並且可以繼續使用該服務。

- Credit specification (額度規格)：選擇 Unlimited (無限制)，只要有需要，就讓應用程式爆量超過基線。此欄位僅對 T 執行個體有效。可能需支付額外費用。如需詳細資訊，請參閱 [爆量效能執行個體](#)。
- Placement group name (置放群組名稱)：指定要在其中啟動執行個體的置放群組。可以選取現有的置放群組或建立新的置放群組。並非所有執行個體類型都可以在置放群組中啟動。如需詳細資訊，請參閱 [置放群組](#)。
- EBS-optimized instance (EBS 最佳化執行個體)：選取 Enable (啟用) 為 Amazon EBS I/O 提供額外的專用容量。但並非所有執行個體類型都支援此功能。需支付額外費用。如需詳細資訊，請參閱 [the section called "EBS 最佳化"](#)。
- 容量保留：指定是否要在任何開啟的容量保留 (Open (開啟))、特定容量保留 (Target by ID (按 ID 分類的目標)) 或容量保留群組 (Target by group (按群組分類的目標)) 中啟動執行個體。若要指定不應使用容量保留，請選擇 None (無)。如需詳細資訊，請參閱 [使用現有的 容量預留 啟動執行個體](#)。
- Tenancy (租用)：選擇在共享硬體 (Shared (共享))、隔離的、專用硬體 (Dedicated (專用)) 或專用執行個體 (Dedicated host (專用執行個體)) 上執行您的執行個體。如果您選擇將執行個體啟動至專用執行個體，則可以指定是否將執行個體啟動至主機資源群組，或是以特定專用執行個體為目標。可能需支付額外費用。如需詳細資訊，請參閱 [專用執行個體](#) 及 [Dedicated Hosts](#)。
- RAM 磁碟 ID：(僅適用於半虛擬 (PV) AMIs) 選取執行個體的 RAM 磁碟。若您已選取核心，則可能需要選取具有可支援之驅動程式的特定 RAM 磁碟。
- 核心 ID：(僅適用於全虛擬 (PV) AMIs) 選取執行個體的核心。

- Nitro Enclave：可讓您從 Amazon EC2 執行個體建立稱為隔離區的獨立執行環境。選取 Enable (啟用)，以啟用 AWS Nitro Enclaves 的執行個體。如需詳細資訊，請參閱 [AWS Nitro Enclaves 使用者指南中的什麼是AWS Nitro Enclaves？](#)
- 授權組態：您可以根據指定的授權配置啟動執行個體，以追蹤您的授權使用情況。如需詳細資訊，請參閱《AWS License Manager 使用者指南》中的[建立授權組態](#)。
- Specify CPU options (指定 CPU 選項)：選擇 Specify CPU options (指定 CPU 選項) 在啟動期間指定自訂數量的 vCPU。設定 CPU 核心數量及每個核心的執行緒數量。如需詳細資訊，請參閱[最佳化 CPU 選項](#)。
- 中繼資料 IPv6 端點：您可以讓執行個體使用 IMDS IPv6 位址[fd00:ec2::254]擷取執行個體中繼資料。只有當您將[AWS Nitro 系統上建置的執行個體啟動到支援 IPv6 的子網路 \(僅限雙堆疊或 IPv6\)](#)時，才能使用此選項。如需詳細資訊，請參閱[擷取執行個體中繼資料](#)。
- 可存取的中繼資料：您可以啟用或停用對 IMDS 的存取。如需詳細資訊，請參閱[設定新執行個體的中繼資料中繼資料選項](#)。
- 中繼資料版本：如果啟用對 IMDS 的存取，則可以選擇請求執行處理中繼資料時請求使用執行個體中繼資料服務第 2 版。如需詳細資訊，請參閱[設定新執行個體的中繼資料中繼資料選項](#)。
- 中繼資料回應躍點限制：如果啟用 IMDS，可以為中繼資料字符設定允許的網路躍點數量。如需詳細資訊，請參閱[設定新執行個體的中繼資料中繼資料選項](#)。
- 允許中繼資料中的標籤：如果選取 Enable (啟用)，則執行個體將允許從其中繼資料存取其執行個體的所有標籤。如果未在範本中包含此設定，則預設情況下，不允許存取執行個體中繼資料中的標籤。如需詳細資訊，請參閱[允許存取執行個體中繼資料中的標籤](#)。
- User data (使用者資料)：您可以指定使用者資料，以在啟動期間設定執行個體或執行組態指令碼。如需詳細資訊，請參閱[啟動時在您的 Amazon EC2 執行個體上執行命令](#)。

## AWS CLI 例子

下列範例會使用 [Create-launch-範本](#) 命令來建立具有指定名稱與執行個體組態的啟動範本。

```
aws ec2 create-launch-template \
 --launch-template-name TemplateForWebServer \
 --version-description WebVersion1 \
 --tag-specifications 'ResourceType=launch-
template,Tags=[{Key=purpose,Value=production}]' \
 --launch-template-data file://template-data.json
```

下列範例 JSON 會指定執行個體組態的啟動範本資料。將 JSON 儲存至檔案，並將其包含在 `--launch-template-data` 參數中，如範例命令所示。

```
{
 "NetworkInterfaces": [{
 "AssociatePublicIpAddress": true,
 "DeviceIndex": 0,
 "Ipv6AddressCount": 1,
 "SubnetId": "subnet-7b16de0c"
 }],
 "ImageId": "ami-8c1be5f6",
 "InstanceType": "r4.4xlarge",
 "TagSpecifications": [{
 "ResourceType": "instance",
 "Tags": [{
 "Key": "Name",
 "Value": "webserver"
 }]
 }],
 "CpuOptions": {
 "CoreCount": 4,
 "ThreadsPerCore": 2
 }
}
```

下列為範例輸出。

```
{
 "LaunchTemplate": {
 "LatestVersionNumber": 1,
 "LaunchTemplateId": "lt-01238c059e3466abc",
 "LaunchTemplateName": "TemplateForWebServer",
 "DefaultVersionNumber": 1,
 "CreatedBy": "arn:aws:iam::123456789012:root",
 "CreateTime": "2017-11-27T09:13:24.000Z"
 }
}
```

## AWS Tools for Windows PowerShell 例子

下列範例會使用指[New-EC2LaunchTemplate](#) 令程式，建立具有指定名稱和執行個體組態的啟動範本。

```
$launchTemplateData = [Amazon.EC2.Model.RequestLaunchTemplateData]@{
 ImageId = 'ami-8c1be5f6'
 InstanceType = 'r4.4xlarge'
```



```

NetworkInterfaces = @(
 [Amazon.EC2.Model.LaunchTemplateInstanceNetworkInterfaceSpecificationRequest]@{
 AssociatePublicIpAddress = $true
 DeviceIndex = 0
 Ipv6AddressCount = 1
 SubnetId = 'subnet-7b16de0c'
 }
)
TagSpecifications = @(
 [Amazon.EC2.Model.LaunchTemplateTagSpecificationRequest]@{
 ResourceType = 'instance'
 Tags = [Amazon.EC2.Model.Tag]@{
 Key = 'Name'
 Value = 'webserver'
 }
 }
)
CpuOptions = [Amazon.EC2.Model.LaunchTemplateCpuOptionsRequest]@{
 CoreCount = 4
 ThreadsPerCore = 2
}
}
$tagSpecificationData = [Amazon.EC2.Model.TagSpecification]@{
 ResourceType = 'launch-template'
 Tags = [Amazon.EC2.Model.Tag]@{
 Key = 'purpose'
 Value = 'production'
 }
}
}
New-EC2LaunchTemplate -LaunchTemplateName 'TemplateForWebServer' -VersionDescription
'WebVersion1' -LaunchTemplateData $launchTemplateData -TagSpecification
$tagSpecificationData

```

下列為範例輸出。

```

CreatedBy : arn:aws:iam::123456789012:root
CreateTime : 9/19/2023 16:57:55
DefaultVersionNumber : 1
LatestVersionNumber : 1
LaunchTemplateId : lt-01238c059eEXAMPLE
LaunchTemplateName : TemplateForWebServer
Tags : {purpose}

```

## 從現有啟動範本建立啟動範本

您可以複製現有啟動範本，然後調整參數以建立新版啟動範本。不過，您只能在使用 Amazon EC2 主控台時執行此操作；AWS CLI 不支援複製範本。

### Console

若要從現有啟動範本建立啟動範本

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Launch Templates (啟動範本) 接著選擇 Create launch template (建立啟動範本)。
3. 針對 Launch template name (啟動範本名稱)，輸入啟動範本的描述性名稱。
4. 在 Template version description (範本版本描述) 中，提供此啟動範本版本的簡短描述。
5. 若要在建立時標記啟動範本，請展開 Template tags (範本標籤)，接著選擇 Add Tag (新增標籤)，然後輸入標籤鍵值組。
6. 展開來源範本，並針對啟動範本名稱，選擇要以新啟動範本為基礎的啟動範本。
7. 針對 Source template version (來源範本版本)，選擇要做為新啟動範本基礎的啟動範本版本。
8. 視需要調整任何啟動參數，然後選擇 Create launch template (建立啟動範本)。

## 從執行個體建立啟動範本

### Console

若要從執行個體建立啟動範本

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選取 Actions (動作)、Create template From instance (從執行個體建立範本)。
4. 提供名稱、描述和標籤，並視需要調整啟動參數。

#### Note

當您從執行個體建立啟動範本時，範本不包含執行個體的網路介面 ID 和 IP 地址。

5. 選擇 Create launch template (建立啟動範本)。

## AWS CLI

您可以使用從現有執行個體建立啟動範本，方法是先從執行個體取得啟動範本資料，然後使用啟動範本資料建立啟動範本。AWS CLI

若要從執行個體啟動範本資料

- 使用 [get-launch-template-data](#) 命令，並指定執行個體 ID。您能以輸出做為基礎，用以建立新啟動範本或啟動範本版本。根據預設，輸出包括最上層 LaunchTemplateData 物件，而此物件無法於啟動範本資料中指定。使用 `--query` 選項排除此物件。

```
aws ec2 get-launch-template-data \
 --instance-id i-0123d646e8048babc \
 --query "LaunchTemplateData"
```

下列為範例輸出。

```
{
 "Monitoring": {},
 "ImageId": "ami-8c1be5f6",
 "BlockDeviceMappings": [
 {
 "DeviceName": "/dev/xvda",
 "Ebs": {
 "DeleteOnTermination": true
 }
 }
],
 "EbsOptimized": false,
 "Placement": {
 "Tenancy": "default",
 "GroupName": "",
 "AvailabilityZone": "us-east-1a"
 },
 "InstanceType": "t2.micro",
 "NetworkInterfaces": [
 {
 "Description": "",
 "NetworkInterfaceId": "eni-35306abc",
 "PrivateIpAddresses": [
 {
 "Primary": true,

```

```

 "PrivateIpAddress": "10.0.0.72"
 }
],
 "SubnetId": "subnet-7b16de0c",
 "Groups": [
 "sg-7c227019"
],
 "Ipv6Addresses": [
 {
 "Ipv6Address": "2001:db8:1234:1a00::123"
 }
],
 "PrivateIpAddress": "10.0.0.72"
 }
]
}

```

例如，您可以將輸出直接寫入檔案：

```

aws ec2 get-launch-template-data \
 --instance-id i-0123d646e8048babc \
 --query "LaunchTemplateData" >> instance-data.json

```

### 使用啟動範本資料建立啟動範本

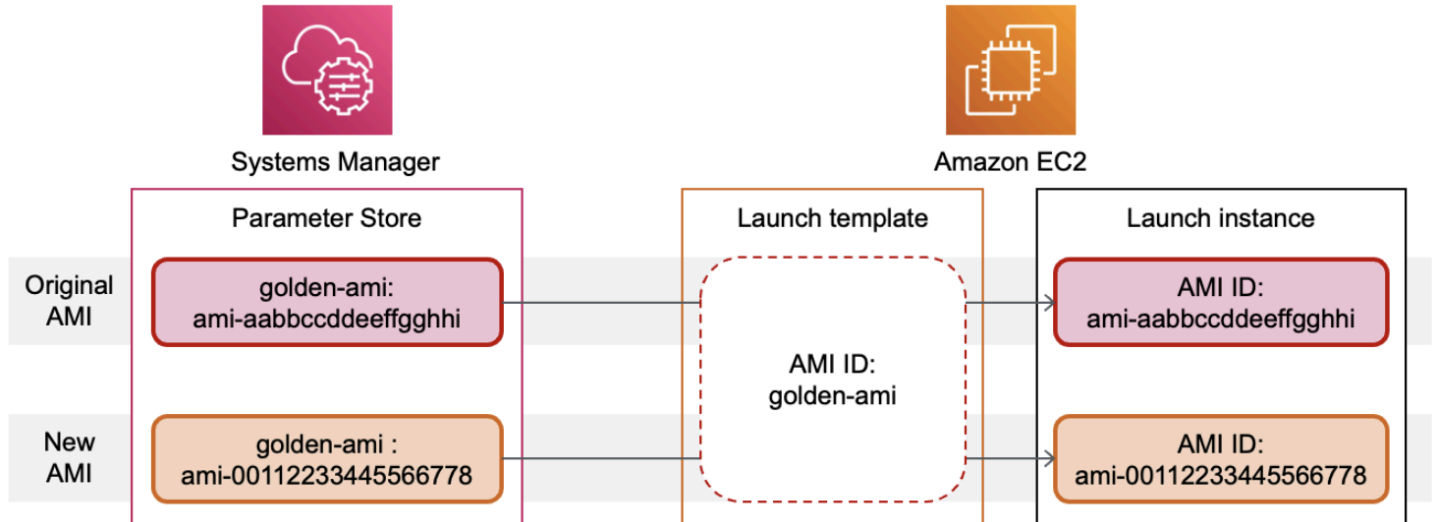
- 使用 [create-launch-template](#) 命令，使用上一個程序的輸出建立啟動範本。如需使用建立啟動範本的詳細資訊 AWS CLI，請參閱[從參數建立啟動範本](#)。

### 使用 Systems Manager 參數而非 AMI ID

您可以指定 AWS Systems Manager 參數，而非在啟動範本中指定 AMI ID。如果 AMI ID 變更，則您可以透過更新 Systems Manager 參數存放區中的 Systems Manager 參數，在一個位置更新 AMI ID。參數也可以與其他參數共用 AWS 帳戶。您可以在一個帳戶中集中存儲和管理 AMI 參數，並與需要引用它們的所有其他帳戶共享它們。透過使用 Systems Manager 參數，您可以在單一動作中更新所有啟動範本。

Systems Manager 參數是您可以在 Systems Manager 參數存放區中建立的使用者定義的索引鍵/值對。參數存放區提供一個中央位置來存放應用程式組態值。如需詳細資訊，請參閱《AWS Systems Manager 使用者指南》中的 [AWS Systems Manager 參數存放區](#)。

在下圖中，golden-ami 參數首先映射到參數存放區中的原始 AMI ami-aabbccddeeffgghhi。在啟動範本中，AMI ID 的值為 golden-ami。使用此啟動範本啟動執行個體時，AMI ID 會解析為 ami-aabbccddeeffgghhi。稍後，會更新 AMI，從而產生新的 AMI ID。在參數存放區中，golden-ami 參數會映射至新的 ami-00112233445566778。啟動範本保持不變。使用此啟動範本啟動執行個體時，AMI ID 會解析為新的 ami-00112233445566778。



### AMI ID 的 Systems Manager 參數格式

啟動範本要求使用者定義的 Systems Manager 參數在用於取代 AMI ID 時必須遵循下列格式：

- 參數類型：String
- 參數資料類型：aws:ec2:image - 這可確保參數存放區驗證您輸入的值是否為格式正確的 AMI ID。

如需有關為 AMI ID 建立有效參數的詳細資訊，請參閱《AWS Systems Manager 使用者指南》中的[建立 Systems Manager 參數](#)。

### 啟動範本中的 Systems Manager 參數格式

若要使用 Systems Manager 參數取代啟動範本中的 AMI ID，在啟動範本中指定參數時，必須使用下列其中一種格式：

若要參照公用參數：

- resolve:ssm:*public-parameter*

若要參照儲存在同一帳戶中的參數：

- `resolve:ssm:parameter-name`
- `resolve:ssm:parameter-name:version-number` - 版本編號本身是預設標籤
- `resolve:ssm:parameter-name:label`

若要參考從另一個參數共用的參數 AWS 帳戶：

- `resolve:ssm:parameter-ARN`
- `resolve:ssm:parameter-ARN:version-number`
- `resolve:ssm:parameter-ARN:label`

## 參數版本

Systems Manager 參數是版本化資源。更新參數時，會建立新的連續參數版本。Systems Manager 支援 [參數標籤](#)，您可以將其映射至特定參數版本。

例如，`golden-ami` 參數可以有三個版本：1、2 和 3。您可以建立映射至版本 2 的參數標籤 `beta`，以及映射至版本 3 的參數標籤 `prod`。

在啟動範本中，可以使用下列其中一種格式來指定 `golden-ami` 參數的版本 3：

- `resolve:ssm:golden-ami:3`
- `resolve:ssm:golden-ami:prod`

指定版本或標籤是可選操作。如果未指定版本或標籤，則會使用最新版本的參數。

在啟動範本中指定 Systems Manager 參數

建立啟動範本或新版本的啟動範本時，可以在啟動範本中指定 Systems Manager 參數，而非 AMI ID。

## Console

在啟動範本中指定 Systems Manager 參數

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Launch Templates (啟動範本) 接著選擇 Create launch template (建立啟動範本)。
3. 針對 Launch template name (啟動範本名稱)，輸入啟動範本的描述性名稱。

4. 在 Application and OS Images (Amazon Machine Image) (應用程式和作業系統映像 (Amazon Machine Image)) 下，選擇 Browse more AMIs (瀏覽更多 AMI)。
5. 選擇搜尋列右側的箭頭按鈕，然後選擇指定自訂值/Systems Manager 參數。
6. 在指定自訂值或 Systems Manager 參數對話方塊中，執行下列操作：
  - a. 對於 AMI ID 或 Systems Manager 參數字串，請使用下列其中一種格式輸入 Systems Manager 參數名稱：

若要參照公用參數：

- **resolve:ssm:*public-parameter***

若要參照儲存在同一帳戶中的參數：

- **resolve:ssm:*parameter-name***
- **resolve:ssm:*parameter-name:version-number***
- **resolve:ssm:*parameter-name:label***

若要參考從另一個參數共用的參數 AWS 帳戶：

- **resolve:ssm:*parameter-ARN***
- **resolve:ssm:*parameter-ARN:version-number***
- **resolve:ssm:*parameter-ARN:label***

- b. 選擇儲存。

7. 視需要指定任何其他啟動範本參數，然後選擇建立啟動範本。

如需詳細資訊，請參閱 [從參數建立啟動範本](#)。

## AWS CLI

在啟動範本中指定 Systems Manager 參數

- 使用 [create-launch-template](#) 命令，建立啟動範本。若要指定要使用的 AMI，請使用下列其中一種格式輸入 Systems Manager 參數名稱：

若要參照公用參數：

- **resolve:ssm:*public-parameter***

若要參照儲存在同一帳戶中的參數：

- `resolve:ssm:parameter-name`
- `resolve:ssm:parameter-name:version-number`
- `resolve:ssm:parameter-name:label`

若要參考從另一個參數共用的參數 AWS 帳戶：

- `resolve:ssm:parameter-ARN`
- `resolve:ssm:parameter-ARN:version-number`
- `resolve:ssm:parameter-ARN:label`

以下範例建立的啟動範本會指定下列項目：

- 啟動範本的名稱 (*TemplateForWebServer*)
- 啟動範本的標籤 (*purpose=production*)
- 在 JSON 檔案中指定的執行個體組態的資料：
  - 要使用的 AMI (`resolve:ssm:golden-ami`)
  - 要啟動的執行個體類型 (*m5.4xlarge*)
  - 執行個體的標籤 (*Name=webserver*)

```
aws ec2 create-launch-template \
 --launch-template-name TemplateForWebServer \
 --tag-specifications 'ResourceType=launch-\
template,Tags=[{Key=purpose,Value=production}]' \
 --launch-template-data file://template-data.json
```

以下是一個範例 JSON 檔案，其中包含執行個體組態的啟動範本資料。ImageId 的值為 Systems Manager 參數名稱，以所需格式 `resolve:ssm:golden-ami` 輸入。

```
{"LaunchTemplateData": {
 "ImageId": "resolve:ssm:golden-ami",
 "InstanceType": "m5.4xlarge",
 "TagSpecifications": [{
```



```

 "ResourceType": "instance",
 "Tags": [{
 "Key": "Name",
 "Value": "webserver"
 }]
 }]
}
}

```

確認啟動範本取得正確的 AMI ID

若要將 Systems Manager 參數解析為實際的 AMI ID

使用[描述啟動範本版本指令](#)並包含參數。--resolve-alias

```

aws ec2 describe-launch-template-versions \
 --launch-template-name my-launch-template \
 --versions $Default \
 --resolve-alias

```

回應包含的 AMI 識別碼 ImageId。在此範例中，當使用此啟動範本啟動執行個體時，AMI ID 會解析為 `ami-0ac394d6a3example`。

```

{
 "LaunchTemplateVersions": [
 {
 "LaunchTemplateId": "lt-089c023a30example",
 "LaunchTemplateName": "my-launch-template",
 "VersionNumber": 1,
 "CreateTime": "2022-12-28T19:52:27.000Z",
 "CreatedBy": "arn:aws:iam::123456789012:user/Bob",
 "DefaultVersion": true,
 "LaunchTemplateData": {
 "ImageId": "ami-0ac394d6a3example",
 "InstanceType": "t3.micro",
 }
 }
]
}

```

## 相關資源

若要取得有關使用 Systems Manager 參數的更多資訊，請參閱「Systems Manager」文件中的下列參考資料。

- 如需如何查詢 Amazon EC2 支援的 AMI 公用參數的相關資訊，請參閱[呼叫 AMI 公用參數](#)。
- 如需與其他 AWS 帳戶共用參數或透過共用參數的相關資訊 AWS Organizations，請參閱[使用共用參數](#)。
- 如需監控參數是否成功建立的相關資訊，請參閱[Amazon 機器映像 ID 的原生參數支援](#)。

## 限制

- 目前，EC2 機群和 Spot 機群不支援使用具有指定 Systems Manager 參數來取代 AMI ID 的啟動範本。對於 EC2 機群和 Spot 機群，如果您在啟動範本中指定 AMI，則必須指定 AMI ID。
- Amazon EC2 Auto Scaling 提供了其他限制。如需詳細資訊，請[AWS Systems Manager 參閱 Amazon EC2 Auto Scaling 使用者指南中的啟動範本中使用參數而非 AMI ID](#)。

## 修改啟動範本 (管理啟動範本版本)

啟動範本是不可變的；啟動範本建立後便無法再修改。但是，您可以建立新版本的啟動範本，來包含您需要的任何變更。

您可以建立啟動範本的不同版本、設定預設版本、說明啟動範本版本，以及刪除您不再需要的版本。

## 任務

- [建立啟動範本版本](#)
- [設定預設啟動範本版本](#)
- [描述啟動範本版本](#)
- [刪除啟動範本版本](#)

## 建立啟動範本版本

當您建立啟動範本版本時，可以指定新的啟動參數，或使用現有版本做為新版本的基礎。如需啟動參數的詳細資訊，請參閱[建立啟動範本](#)。

## Console

若要建立啟動範本版本

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Launch Templates (啟動範本)。
3. 選取啟動範本，然後選取 Actions (動作)、Modify template (Create new version) (修改範本 (建立新版本))。
4. 在 Template version description (範本版本描述) 中，輸入此啟動範本版本的描述。
5. (選取性) 展開 Source template (來源範本)，並選取啟動範本的版本，以作為新啟動範本版本的基礎。新啟動範本版本會繼承此啟動範本版本的啟動參數。
6. 視需要修改啟動參數，然後選擇 Create launch template (建立啟動範本)。

## AWS CLI

若要建立啟動範本版本

- 使用 [create-launch-template-version](#) 命令。您可以指定要做為新版本基礎的來源版本。新版本會繼承此版本的啟動參數，但您可以使用 `--launch-template-data` 來覆寫參數。下列範例會建立以第 1 版啟動範本為基礎的新版本，並指定不同的 AMI ID。

```
aws ec2 create-launch-template-version \
 --launch-template-id lt-0abcd290751193123 \
 --version-description WebVersion2 \
 --source-version 1 \
 --launch-template-data "ImageId=ami-c998b6b2"
```

## 設定預設啟動範本版本

您可以設定啟動範本的預設版本。當您從啟動範本啟動執行個體而且未指定版本時，會使用預設版本的參數來啟動執行個體。

## Console

若要設定預設啟動範本版本

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Launch Templates (啟動範本)。

3. 選取啟動範本，然後選取 Actions (動作)、Set default version (設定預設版本)。
4. 對於 Template version (範本版本)，請選取要設為預設版本的版本號碼，然後選取 Set as default version (設為預設版本)。

## AWS CLI

若要設定預設啟動範本版本

- 使用 [modify-launch-template](#) 命令，並指定您要設定為預設值的版本。

```
aws ec2 modify-launch-template \
 --launch-template-id lt-0abcd290751193123 \
 --default-version 2
```

## 描述啟動範本版本

您可以使用主控台，檢視所選啟動範本的所有版本，或取得其最新或預設版本符合特定版本號碼之啟動範本的清單。使用 AWS CLI，您可以描述指定啟動範本的所有版本、個別版本或某個範圍的版本。您也可以說明帳戶中所有啟動範本的所有最新版本或所有預設版本。

## Console

若要描述啟動範本版本

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Launch Templates (啟動範本)。
3. 您可以檢視特定啟動範本版本，或取得其最新或預設版本符合特定版本號碼之啟動範本的清單。
  - 若要檢視某個版本的啟動範本：請選取啟動範本。在 Versions (版本) 索引標籤上，從 Version (版本) 中選取某個版本以檢視其詳細資料。
  - 若要取得其最新版本符合特定版本號碼之所有啟動範本的清單：請從搜尋列中選擇 Latest version (最新版本)，然後選擇版本號碼。
  - 若要取得其預設版本符合特定版本號碼之所有啟動範本的清單：請從搜尋列中選擇 Default version (預設版本)，然後選擇版本號碼。

## AWS CLI

### 若要描述啟動範本版本

- 使用 [describe-launch-template-versions](#) 命令，並指定版本號碼。在下列範例中，指定的是第 1 版和第 3 版。

```
aws ec2 describe-launch-template-versions \
 --launch-template-id lt-0abcd290751193123 \
 --versions 1 3
```

### 若要描述帳戶中所有最新和預設啟動範本版本

- 使用 [describe-launch-template-versions](#) 命令，並指定 \$Latest、\$Default 或兩者。您必須省略呼叫中的啟動範本 ID 和名稱。您無法指定版本號碼。

```
aws ec2 describe-launch-template-versions \
 --versions "$Latest,$Default"
```

## 刪除啟動範本版本

如果您不再需要啟動範本版本，可以將其刪除。

### 考量事項

- 版本號碼在刪除之後就無法取代。
- 您無法刪除啟動範本的預設版本；您必須先指派不同的版本作為預設值。如果預設版本是啟動範本的唯一版本，則須 [刪除整個啟動範本](#)。
- 使用主控台時，您一次可以刪除一個啟動範本版本。使用時 AWS CLI，您最多可以在單一要求中刪除 200 個啟動範本版本。若要在單個請求中刪除超過 200 個版本，您可以 [刪除啟動範本](#)，而這也會刪除啟動範本的所有版本。

## Console

### 若要刪除啟動範本版本

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Launch Templates (啟動範本)。

3. 選取啟動範本，然後選取 Actions (動作)、Delete template version (刪除範本版本)。
4. 選取要刪除的版本，然後選取 Delete (刪除)。

## AWS CLI

若要刪除啟動範本版本

- 使用 [delete-launch-template-versions](#) 命令，並指定要刪除的版本號碼。您在單個請求中最多可刪除 200 個啟動範本版本。

```
aws ec2 delete-launch-template-versions \
 --launch-template-id lt-0abcd290751193123 \
 --versions 1
```

## 刪除啟動範本

如果您不再需要啟動範本，即可將其刪除。刪除啟動範本會刪除它的所有版本。若要刪除啟動範本的特定版本，請參閱 [刪除啟動範本版本](#)。

刪除啟動範本時，不會影響您從啟動範本啟動的任何執行個體。

## Console

若要刪除啟動範本

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Launch Templates (啟動範本)。
3. 選取啟動範本，然後選取 Actions (動作)、Delete template (刪除範本)。
4. 輸入 **Delete** 以確認刪除，然後選擇 Delete (刪除)。

## AWS CLI

若要刪除啟動範本

- 使用 [delete-launch-template](#) AWS CLI 命令，並指定啟動範本。

```
aws ec2 delete-launch-template --launch-template-id lt-01238c059e3466abc
```

## 從啟動範本啟動執行個體

數個執行個體啟動服務支援啟動範本。此主題說明如何在使用 EC2 啟動執行個體精靈、Amazon EC2 Auto Scaling、EC2 機群和 Spot 機群啟動執行個體時使用啟動範本。

### 主題

- [從啟動範本啟動執行個體](#)
- [透過 Amazon EC2 Auto Scaling 使用啟動範本](#)
- [透過 EC2 Fleet 使用啟動範本](#)
- [搭配使用啟動範本與 Spot Fleet](#)

## 從啟動範本啟動執行個體

您可以使用啟動範本中所含的參數來啟動執行個體。您可以選擇覆寫或新增啟動參數，再啟動執行個體。

使用啟動範本所啟動的執行個體會自動獲指派兩個具有索引鍵 `aws:ec2launchtemplate:id` 和 `aws:ec2launchtemplate:version` 的標籤。您無法移除或編輯這些標籤。

### Console

#### 使用主控台從啟動範本啟動執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Launch Templates (啟動範本)。
3. 選取啟動範本，然後選取 Actions (動作)、Launch instance from template (從範本啟動執行個體)。
4. 對於 Source template version (來源範本版本)，請選取要使用的啟動範本版本。
5. 對於 Number of instances (執行個體數量)，指定要啟動的執行個體數量。
6. (選用) 您可以在 Instance details (執行個體詳細資訊) 區段中變更和新增參數，以覆寫或新增啟動範本參數。
7. 選擇 Launch instance from template (從範本啟動執行個體)。

## AWS CLI

### 使用 AWS CLI 從啟動範本啟動執行個體

- 使用 [run-instances](#) 命令，並指定 `--launch-template` 參數。選擇指定要使用的啟動範本版本。如果您未指定版本，即使用預設版本。

```
aws ec2 run-instances \
 --launch-template LaunchTemplateId=lt-0abcd290751193123,Version=1
```

- 若要覆寫啟動範本參數，請在 [run-instances](#) 命令中指定參數。下列範例會覆寫啟動範本中指定的執行個體類型 (如果有的話)。

```
aws ec2 run-instances \
 --launch-template LaunchTemplateId=lt-0abcd290751193123 \
 --instance-type t2.small
```

- 如果您指定屬複雜結構一部分的巢狀參數，則會使用啟動範本中所指定的複雜結構以及您指定的其他巢狀參數，來啟動執行個體。

在下列範例中，會使用 `Owner=TeamA` 標籤以及啟動範本中指定的其他標籤，來啟動執行個體。如果啟動範本的現有標籤包含 `Owner` 索引鍵，則會將值取代為 `TeamA`。

```
aws ec2 run-instances \
 --launch-template LaunchTemplateId=lt-0abcd290751193123 \
 --tag-specifications "ResourceType=instance,Tags=[{Key=Owner,Value=TeamA}]"
```

在下列範例中，會使用裝置名稱為 `/dev/xvdb` 的磁碟區以及啟動範本中指定的其他區塊型設備映射，來啟動執行個體。如果啟動範本具有針對 `/dev/xvdb` 所定義的現有磁碟區，則會將其值取代為指定的值。

```
aws ec2 run-instances \
 --launch-template LaunchTemplateId=lt-0abcd290751193123 \
 --block-device-mappings "DeviceName=/dev/xvdb,Ebs={VolumeSize=20,VolumeType=gp2}"
```

如果執行個體無法啟動或狀態立即進入 `terminated` 而非 `running`，請參閱 [針對執行個體啟動問題進行故障診斷](#)。



## PowerShell

### 使用 AWS Tools for PowerShell 從啟動範本啟動執行個體

- 使用指 [New-EC2Instance](#) 令並指定 `-LaunchTemplate` 參數。選擇指定要使用的啟動範本版本。如果您未指定版本，即使用預設版本。

```
Import-Module AWS.Tools.EC2
New-EC2Instance `
 -LaunchTemplate (
 New-Object -TypeName Amazon.EC2.Model.LaunchTemplateSpecification -
Property @{
 LaunchTemplateId = 'lt-0abcd290751193123';
 Version = '4'
}
)
```

- 若要覆寫啟動範本參數，請在指 [New-EC2Instance](#) 令中指定參數。下列範例會覆寫啟動範本中指定的執行個體類型 (如果有的話)。

```
Import-Module AWS.Tools.EC2
New-EC2Instance `
 -InstanceType t4g.small `
 -LaunchTemplate (
 New-Object -TypeName Amazon.EC2.Model.LaunchTemplateSpecification -
Property @{
 LaunchTemplateId = 'lt-0abcd290751193123';
 Version = '4'
}
)
```

- 如果您指定屬複雜結構一部分的巢狀參數，則會使用啟動範本中所指定的複雜結構以及您指定的其他巢狀參數，來啟動執行個體。

在下列範例中，會使用 `Owner=TeamA` 標籤以及啟動範本中指定的其他標籤，來啟動執行個體。如果啟動範本的現有標籤包含 `Owner` 索引鍵，則會將值取代為 `TeamA`。

```
Import-Module AWS.Tools.EC2
New-EC2Instance `
 -InstanceType t4g.small `
 -LaunchTemplate (
```

```

 New-Object -TypeName Amazon.EC2.Model.LaunchTemplateSpecification -
Property @{
 LaunchTemplateId = 'lt-0abcd290751193123';
 Version = '4'
}
) `
-TagSpecification (
 New-Object -TypeName Amazon.EC2.Model.TagSpecification -Property @{
 ResourceType = 'instance';
 Tags = @(
 @{key = "Owner"; value = "TeamA" },
 @{key = "Department"; value = "Operations" }
)
}
)
)

```

在下列範例中，會使用裝置名稱為 `/dev/xvdb` 的磁碟區以及啟動範本中指定的其他區塊型設備映射，來啟動執行個體。如果啟動範本具有針對 `/dev/xvdb` 所定義的現有磁碟區，則會將其值取代為指定的值。

```

Import-Module AWS.Tools.EC2
New-EC2Instance `
 -InstanceType t4g.small `
 -LaunchTemplate (
 New-Object -TypeName Amazon.EC2.Model.LaunchTemplateSpecification -
Property @{
 LaunchTemplateId = 'lt-0abcd290751193123';
 Version = '4'
}
) `
 -BlockDeviceMapping (
 New-Object -TypeName Amazon.EC2.Model.BlockDeviceMapping -Property @{
 DeviceName = '/dev/xvdb';
 EBS = (
 New-Object -TypeName Amazon.EC2.Model.EbsBlockDevice -Property @{
 VolumeSize = 25;
 VolumeType = 'gp3'
}
)
}
)
)

```

如果執行個體無法啟動或狀態立即進入 `terminated` 而非 `running`，請參閱[針對執行個體啟動問題進行故障診斷](#)。

## 透過 Amazon EC2 Auto Scaling 使用啟動範本

您可以建立 Auto Scaling 群組，並指定要用於群組的啟動範本。Amazon EC2 Auto Scaling 在 Auto Scaling 群組中啟動執行個體時，會使用相關聯之啟動範本中定義的啟動參數。如需詳細資訊，請參閱 Amazon EC2 [Auto Scaling 使用者指南中的為 Auto Scaling 群組建立啟動範本和使用進階設定](#) 建立啟動範本。

在您可以使用啟動範本建立 Auto Scaling 群組之前，您必須建立啟動範本，其中包含在 Auto Scaling 群組中啟動執行個體所需的參數，例如 AMI 的 ID。主控台提供指導，協助您建立可與 Amazon EC2 Auto Scaling 搭配使用的範本。

### 建立啟動範本以與使用主控台搭配 Auto Scaling 使用

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Launch Templates (啟動範本) 接著選擇 Create launch template (建立啟動範本)。
3. 針對 Launch template name (啟動範本名稱)，輸入啟動範本的描述性名稱。
4. 在 Template version description (範本版本描述) 中，提供此啟動範本版本的簡短描述。
5. 在 Auto Scaling guidance (Auto Scaling 指引) 中，選取核取方塊，使 Amazon EC2 提供指引，協助建立要搭配 Auto Scaling 使用的範本。
6. 視需要修改啟動參數。由於您選取了 Auto Scaling 指引，因此某些欄位為必填欄位，而某些欄位則無法使用。如需如何設定 Amazon EC2 Auto Scaling 的啟動參數的相關資訊，請參閱 Amazon EC2 [自動擴展使用者指南中的為 Auto Scaling 群組建立啟動範本和使用進階設定](#) 建立 Auto Scaling 範本。
7. 選擇 Create launch template (建立啟動範本)。
8. (選用) 若要使用此啟動範本建立 Auto Scaling 群組，請在 Next steps (後續步驟) 頁面上選擇 Create Auto Scaling group (建立 Auto Scaling 群組)。

如需示範如何使用各種參數組合建立啟動範本的範例，請參閱 [Amazon EC2 Auto Scaling 使用者指南中的 AWS Command Line Interface \(AWS CLI\) 建立和管理啟動範本的範例](#)。AWS CLI

若要使用啟動範本建立或更新「Auto Scaling」群組 AWS CLI

- 使用 [create-auto-scaling-group](#) 或 [update-auto-scaling-group](#) 命令，並指定 `--launch-template` 參數。

如需使用啟動範本建立或更新 Auto Scaling 群組的詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南中的以下主題。

- [使用啟動範本建立 Auto Scaling 群組](#)
- [更新 Auto Scaling 群組](#)

透過 EC2 Fleet 使用啟動範本

您可以建立 EC2 Fleet 請求，並在執行個體組態中指定啟動範本。當 Amazon EC2 滿足 EC2 Fleet 請求時，會使用相關聯之啟動範本中定義的啟動參數。您可以覆寫啟動範本中指定的部分參數。

如需詳細資訊，請參閱 [建立 EC2 Fleet](#)。

若要使用啟動範本建立 EC2 叢集 AWS CLI

- 使用 [create-fleet](#) 命令。使用 `--launch-template-configs` 參數指定啟動範本以及啟動範本的任何覆寫。

搭配使用啟動範本與 Spot Fleet

您可以建立 Spot Fleet 請求，並在執行個體組態中指定啟動範本。當 Amazon EC2 滿足 Spot Fleet 請求時，會使用相關聯之啟動範本中定義的啟動參數。您可以覆寫啟動範本中指定的部分參數。

如需詳細資訊，請參閱 [建立 Spot Fleet 請求](#)。

若要使用主控台透過啟動範本來建立 Spot 機群請求

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選擇 Request Spot Instances (要求 Spot 執行個體)。
4. 在 Launch parameters (啟動參數) 下，選擇 Use a launch template (使用啟動範本)。
5. 針對 Launch template (啟動範本)，選擇啟動範本，然後從右邊的欄位選擇啟動範本版本。

6. 透過在此畫面上選取不同的選項，設定 Spot 機群。如需選項的詳細資訊，請參閱 [使用已定義的參數建立 Spot Fleet 請求 \(主控台\)](#)。
7. 當您準備好建立 Spot 機群時，選擇 Launch (啟動)。

若要使用啟動範本建立競價型叢集請求 AWS CLI

- 使用 [request-spot-fleet](#) 命令。使用 LaunchTemplateConfigs 參數指定啟動範本以及啟動範本的任何覆寫。

## 使用現有執行個體的參數啟動執行個體

Amazon EC2 主控台提供 Launch more like this (啟動更多相似執行個體) 選項，可讓您使用目前的執行個體做為啟動其他執行個體的基礎。此選項會自動使用選取執行個體的特定組態詳細資訊，填入 Amazon EC2 啟動執行個體精靈。

### 考量事項

- 我們不會複製您的執行個體；我們只會複寫部分的組態詳細資訊。若要建立您執行個體的複本，首先請從它建立 AMI，然後從 AMI 啟動更多執行個體。建立 [啟動範本](#)，以確保您使用相同的啟動詳細資訊來啟動執行個體。
- 目前的執行個體必須處於 running 狀態。

### 已複製的詳細資訊

下列組態詳細資訊會從選取的執行個體複製到啟動執行個體精靈：

- AMI ID
- 執行個體類型
- 可用區域，或是選取執行個體所在的 VPC 及子網
- 公有 IPv4 地址。若選取的執行個體目前具有公有 IPv4 地址，新的執行個體便會收到公有 IPv4 地址 – 無論選取執行個體的預設公有 IPv4 地址設定為何。如需公有 IPv4 地址的詳細資訊，請參閱 [公有 IPv4 地址](#)。
- 置放群組 (如適用)
- 與執行個體相關聯的 IAM 角色 (如適用)
- 關機行為設定 (停止或終止)
- 終止保護設定 (true 或 false)

- CloudWatch 監視 (啟用或停用)
- Amazon EBS 最佳化設定 (true 或 false)
- 租用設定 (若在 VPC 中啟動的話 (共享或專用))
- 核心 ID 和 RAM 磁碟區 ID (如適用)
- 使用者資料 (若指定)
- 與執行個體相關聯的標籤 (如適用)
- 與執行個體相關聯的安全群組
- [視窗執行個體] 關聯資訊。若選取的執行個體與組態檔相關聯，則相同的檔案便會自動與新的執行個體相關聯。若組態檔包含加入的網域組態，新的執行個體便會加入相同網域。如需加入網域的詳細資訊，請參閱 AWS Directory Service 管理指南中的[無縫加入 Windows EC2 執行個體](#)。

### 未複製的詳細資訊

下列組態詳細資訊不是從您選取的執行個體複製的。相反地，精靈會套用其預設設定或行為：

- 網路介面數 – 預設為一個網路介面，即主要網路介面 (eth0)。
- 儲存體 – 預設儲存體組態會由 AMI 和執行個體類型判定。

依據現有執行個體的操作方式，啟動更多執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體，然後選取動作、映像和範本、啟動更多相似執行個體。
4. 啟動執行個體精靈隨即開啟。透過在此畫面上選取不同的選項，對執行個體組態進行任何必要的變更。

當您準備好啟動執行個體時，請選擇 Launch instance (啟動執行個體)。

5. 如果執行個體無法啟動或狀態立即進入 terminated 而非 running，請參閱[針對執行個體啟動問題進行故障診斷](#)。

## 啟動 AWS Marketplace 執行個體

您可以使用 Amazon EC2 啟動精靈訂閱 AWS Marketplace 產品，並從產品的 AMI 啟動執行個體啟動。如需已支付 AMI 的詳細資訊，請參閱[已支付 AMI](#)。若要在啟動之後取消您的訂閱，您需要先終止從中執行的所有執行個體。如需詳細資訊，請參閱[管理您的 AWS Marketplace 訂閱](#)。

## New console

若要從 AWS Marketplace 使用啟動精靈啟動執行個體


1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 從 Amazon EC2 主控台儀表板選擇 Launch Instance (啟動執行個體)。
3. (選用) 在 Name and tags (名稱和標籤) 下，針對 Name (名稱)，輸入執行個體的描述性名稱。
4. 在 Application and OS Images (Amazon Machine Image) (應用程式和作業系統映像 (Amazon Machine Image)) 中，選擇 Browse more AMIs (瀏覽更多 AMI)，然後選擇 AWS Marketplace AMIs 標籤。透過瀏覽類別或使用搜尋功能，來尋找適合的 AMI。若要選取產品，請選取 Select (選取)。
5. 隨即開啟一個視窗，其中顯示您所選取產品的概觀。您可以檢視定價資訊，以及廠商已提供的任何其他資訊。準備就緒後，請選擇下列其中一個按鈕：
  - 執行個體啟動時訂閱 — 當您選擇 Launch 執行個體 (在步驟 10) 時即會開始訂閱。
  - 立即訂閱 — 您的訂閱立即開始。訂閱正在進行中時，您可以繼續執行此程序中的步驟來設定執行個體。如果您的信用卡詳細資訊有任何問題，系統會要求您更新帳戶詳細資訊。

### Note

在您使用 AMI 啟動執行個體之前，不會向您收取使用產品的費用。當您選取執行個體類型時，請記下每個受支援執行個體類型的定價。可能也需支付產品的額外稅金。


6. 針對 Instance type (執行個體類型)，選取執行個體的執行個體類型。執行個體類型定義執行個體的硬體組態和大小。
7. 在 Key pair (login) (金鑰對 (登入)) 下，針對 Key pair name (金鑰對名稱)，選擇現有的金鑰對或建立新的金鑰對。
8. 在 Network settings (網路設定)、Firewall (security groups) (防火牆 (安全群組)) 中，記下已根據廠商的產品規格所建立的新安全群組。安全群組可能會包括規則，以允許 Linux 上之 SSH (連接埠 22) 或 Windows 上之 RDP (連接埠 3389) 的所有 IPv4 地址 (0.0.0.0/0) 存取。建議您調整這些規則，改為僅允許特定地址或某範圍的地址透過這些連接埠存取您的執行個體。
9. 您可以使用畫面上的其他欄位來設定執行個體、新增儲存以及新增標籤。如需有關您可設定之不同選項的資訊，請參閱 [使用定義的參數啟動執行個體](#)。

10. 在 Summary (摘要) 面板的 Software Image (AMI) (軟體映像 (AMI)) 中，檢查即將要從中啟動執行個體之 AMI 的詳細資訊。同時檢查您指定的其他組態詳細資訊。當您準備好啟動執行個體時，請選擇 Launch instance (啟動執行個體)。
11. 根據您已訂閱的產品，執行個體可能需要幾分鐘以上的時間啟動。如果您在步驟 5 選擇「執行個體啟動時訂閱」，則在執行個體啟動之前，您會先訂閱產品。如果您的信用卡詳細資訊有任何問題，系統會要求您更新帳戶詳細資訊。顯示啟動確認頁面時，請選擇 View all instances (檢視所有執行個體) 以前往 Instances (執行個體) 頁面。

 Note

只要您的執行個體處於 running 狀態，就會向您收取訂閱費用，即使它處於閒置狀態。如果您的執行個體已停止，仍可能會向您收取儲存體的費用。

12. 當您的執行個體處於 running 狀態時，即可連線。若要執行此操作，請選取清單中的執行個體，然後選取 Connect (連線)，並選取連線選項。如需連線至執行個體的詳細資訊，請參閱[連接至您的 Linux 執行個體連線至 Windows 執行個體](#)。

 Important

請仔細檢查廠商使用說明，因為您可能需要使用特定使用者名稱連線到執行個體。如需有關存取訂閱的資訊，請參閱[管理您的 AWS Marketplace 訂閱](#)。

13. 如果執行個體無法啟動或狀態立即進入 terminated 而非 running，請參閱[針對執行個體啟動問題進行故障診斷](#)。

## Old console

若要從 AWS Marketplace 使用啟動精靈啟動執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在 Amazon EC2 儀表板中，選擇 Launch Instance (啟動執行個體)。
3. 在 Choose an Amazon Machine Image (AMI) (選擇 Amazon Machine Image (AMI)) 頁面上，選擇左側的 AWS Marketplace 類別。瀏覽類別或使用搜尋功能，來尋找適合的 AMI。選取 Select (選取) 來選取產品。
4. 對話方塊會顯示選取之產品的概觀。您可以檢視定價資訊，以及廠商已提供的任何其他資訊。當您就緒時，請選擇 Continue (繼續)。



**Note**

在您使用 AMI 啟動執行個體之前，不會向您收取使用產品的費用。請記下每個支援之執行個體類型的定價，因為系統會提示您於精靈的下一頁選取執行個體類型。可能也需支付產品的額外稅金。

5. 在 Choose an Instance Type (選取執行個體類型) 頁面上，選取要啟動之執行個體的硬體組態和大小。完成後，請選擇 Next: Configure Instance Details (下一步：設定執行個體詳細資訊)。
6. 在精靈的後續頁面上，您可以設定執行個體、新增儲存體，以及新增標籤。如需您可設定之不同選項的詳細資訊，請參閱[使用舊版啟動執行個體精靈啟動執行個體](#)。在您到 Configure Security Group (設定安全群組) 頁面前，請皆選擇 Next (下一步)。

此精靈會根據廠商的產品規格來建立新的安全群組。安全群組可能會包括規則，以允許 Linux 上之 SSH (連接埠 22) 或 Windows 上之 RDP (連接埠 3389) 的所有 IPv4 地址 (0.0.0.0/0) 存取。建議您調整這些規則，改為僅允許特定地址或某範圍的地址透過這些連接埠存取您的執行個體。

準備就緒後，請選擇 Review and Launch (檢閱和啟動)。

7. 在 Review Instance Launch (檢閱執行個體啟動) 頁面上，檢查即將要啟動執行個體之 AMI 的詳細資訊，以及您在精靈中設定的其他組態詳細資訊。準備就緒後，請選取 Launch (啟動) 選取或建立金鑰對，並啟動執行個體。
8. 根據您已訂閱的產品，執行個體可能需要幾分鐘以上的時間啟動。您要先訂閱產品，才能啟動執行個體。如果您的信用卡詳細資訊有任何問題，系統會要求您更新帳戶詳細資訊。顯示啟動確認頁面時，請選擇 View Instances (檢視執行個體) 前往 Instances (執行個體) 頁面。

**Note**

只要您的執行個體正在執行，就會向您收取訂閱費用，即使執行個體處於閒置狀態亦同。如果您的執行個體已停止，仍可能會向您收取儲存體的費用。

9. 當您的執行個體處於 running 狀態時，即可連線。若要執行此作業，請選取清單中的執行個體，然後選取 Connect (連線)。請遵循對話方塊中的說明。如需連線至執行個體的詳細資訊，請參閱[連接至您的 Linux 執行個體連線至 Windows 執行個體](#)。

**⚠ Important**

請仔細檢查廠商使用說明，因為您可能需要使用特定使用者名稱登入執行個體。如需存取訂閱詳細資訊的詳細資訊，請參閱[管理您的 AWS Marketplace 訂閱](#)。

10. 如果執行個體無法啟動或狀態立即進入 `terminated` 而非 `running`，請參閱[針對執行個體啟動問題進行故障診斷](#)。

## 使用 API 和 CLI 啟動 AWS Marketplace AMI 執行個體

若要使用 API 或命令列工具從產 AWS Marketplace 品啟動執行個體，請先確定您已訂閱產品。您接著可以使用下列方法以產品的 AMI ID 來啟動執行個體：

| 方法                               | 文件                                                                                                             |
|----------------------------------|----------------------------------------------------------------------------------------------------------------|
| AWS CLI                          | 使用 <a href="#">run-instances</a> 命令，或者參閱下列主題，查看詳細資訊： <a href="#">啟動執行個體</a> 。                                  |
| AWS Tools for Windows PowerShell | 使用命 <a href="#">New-EC2Instance</a> 令，或參閱下列主題以取得更多資訊： <a href="#">使用 Windows 啟動 Amazon EC2 執行個體 PowerShell</a> |
| 查詢 API                           | 使用 <a href="#">RunInstances</a> 請求。                                                                            |

## 停止並啟動 Amazon EC2 執行個體

如果執行個體使用 Amazon EBS 磁碟區做為其根設備，則您可以停止和啟動此執行個體。當您停止執行個體時，執行個體就會關閉。啟動執行個體時，通常會將其移轉至新的基礎主機電腦，並指派新的公用 IPv4 位址。

當您停止執行個體時，該執行個體不會刪除。如果您決定不再需要使用執行個體時，可將其終止。如需詳細資訊，請參閱 [終止 Amazon EC2 實例](#)。如果要讓執行個體休眠以儲存執行個體記憶體 (RAM) 中的內容，請參閱：[休眠您的 Amazon EC2 實例](#)。有關執行個體生命週期操作之間的區別，請參閱：[重新啟動、停止、休眠和終止之間的差異](#)。

### 目錄

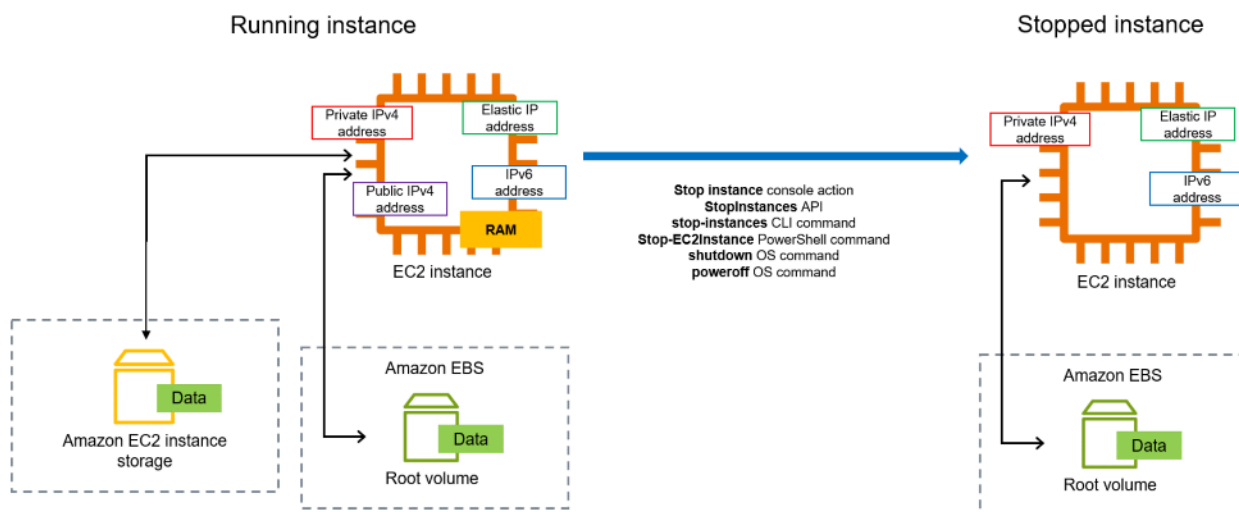
- [執行個體停止和啟動如何運作](#)
- [手動停止和啟動執行個體](#)

- [自動停止及啟動執行個體](#)
- [尋找所有執行中和停止的執行個體](#)
- [為執行個體啟用停止保護](#)

## 執行個體停止和啟動如何運作

當您停止執行個體時，變更會在執行個體的作業系統層級註冊，部分資源會遺失，而且部分資源會持續存在。啟動執行個體時，會在執行個體層級註冊變更。

下圖顯示 Amazon EC2 執行個體停止時遺失及持續存在的內容。執行個體停止時，如果彈性 IP 地址與執行個體沒有關聯，則會遺失所有連接的執行個體儲存體磁碟區和儲存在這些磁碟區上的資料、儲存在執行個體 RAM 上的資料，以及指派的公有 IPv4 地址。執行個體會保留指派的私有 IPv4 地址、與執行個體關聯的彈性 IP 地址、所有 IPv6 地址，以及所有連接的 Amazon EBS 磁碟區以及這些磁碟區上的資料。



## 若您停止執行個體會如何

### 在作業系統層級註冊的變更

- API 請求會將按鈕事件傳送給訪客。
- 由於按鈕事件而停止各種系統服務。順利關機由從虛擬化管理程序按下 ACPI 關機按鈕事件觸發。
- 啟動 ACPI 關機。
- 當順利關機程序結束時，執行個體就會關機。OS 關機時間無法設定。
- 如果執行個體作業系統在幾分鐘內沒有完全關閉，便會執行硬關機。

- 執行個體會停止執行。
- 執行個體的狀態會變成 `stopping` 然後再變成 `stopped`。
- [Auto Scaling] 如果您的執行是 Auto Scaling 群組，當執行個體的狀態處於 `running` 以外的任何 Amazon EC2 狀態，或是狀態檢查的狀態變為 `impaired`，則 Amazon EC2 Auto Scaling 會認定此執行個體運作狀態不佳，並將其取代。如需詳細資訊，請參閱《Amazon EC2 Auto Scaling 使用者指南》中的 [Auto Scaling 執行個體的運作狀態檢查](#)。
- [Windows 執行個體] 停止並啟動 Windows 執行個體時，啟動代理程式會在執行個體上執行任務，例如變更任何連接 Amazon EBS 磁碟區的磁碟機代號。如需這些預設值以及如何變更它們的詳細資訊，請參閱 [the section called “EC2Launch v2”](#)。

### 資源丟失

- 存放在 RAM 的資料。
- 存放在執行個體儲存體磁碟區的資料。
- Amazon EC2 在啟動或開始時自動指派給執行個體的公有 IPv4 地址。若要保留永遠不變的公有 IPv4 地址，可將 [彈性 IP 地址](#) 與您的執行個體建立關聯。

### 持續存在的資源

- 所有連接的 Amazon EBS 磁碟區。
- 連接的 Amazon EBS 磁碟區上存放的資料。
- 私有 IPv4 地址。
- IPv6 地址。
- 與執行個體有關聯的彈性 IP 地址。請注意，當執行個體停用時，系統會 [針對相關聯的彈性 IP 地址向您收費](#)。

如需停止 Mac 執行個體時會發生什麼情況的詳細資訊，請參閱 [the section called “停止並終止您的 Mac 執行個體”](#)。

### 啟動執行個體時會出現的情況

#### 在作業系統層級註冊的變更

- 大多數情況下，執行個體會遷移至新的基礎主機電腦（不過有時候會保留在目前的主機上，例如當執行個體是配置到 [專用執行個體](#) 組態中的主機時）。

- 如果執行個體設定為接收公有 IPv4 地址，Amazon EC2 會為執行個體指派新的公有 IPv4 地址。若要保留永遠不變的公有 IPv4 地址，可將[彈性 IP 地址](#)與您的執行個體建立關聯。

## 測試應用程式對停止和啟動的回應

您可以用 AWS Fault Injection Service 來測試執行個體停止和啟動時應用程式的回應方式。如需詳細資訊，請參閱 [AWS Fault Injection Service 使用者指南](#)。

## 與執行個體停止和啟動相關的成本

停止和啟動執行個體的相關成本如下。

**停止** — 一旦執行個體狀態變更為 `shutting-down` 或 `terminated`，執行個體就不會再產生費用。您不需為已停止的執行個體支付使用量或資料傳輸費用。存放 Amazon EBS 儲存磁碟區會產生費用。

**啟動** — 每次啟動已停止的執行個體時，我們會收取至少一分鐘的使用費。如果超過一分鐘，只會依使用的秒數收費。例如，如果您執行執行個體 20 秒便停止，我們會收取完整一分鐘的費用。如果您執行執行個體 3 分 40 秒，則只會收取 3 分 40 秒的使用費。

## 手動停止和啟動執行個體

您可以停止並啟動 Amazon EBS 支援的執行個體 (具有 EBS 根裝置的執行個體)。您無法使用實例存儲根設備停止和啟動實例。

### Warning

停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。停止執行個體之前，請確認您已將所需的任何資料從執行個體存放磁碟區複製到持續性儲存，例如 Amazon EBS 或 Amazon S3。

## Console

### 停止和啟動 Amazon EBS 支援的執行個體

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽窗格中，選擇 [執行個體]，然後選取執行個體。
3. 在儲存區索引標籤上，確認根裝置類型為 EBS。否則，您無法停止執行個體。

4. 選擇 Instance state (執行個體狀態)、Stop instance (停止執行個體)。如果此選項已停用，則執行個體已停止或其根設備是執行個體存放磁碟區。
5. 出現確認提示時，請選擇 Stop (停止)。停止執行個體可能需要幾分鐘。
6. 若要啟動停止的執行個體，請選取執行個體，然後選取執行個體狀態、啟動執行個體。
7. 執行個體進入 running 狀態可能需要數分鐘。
8. 如果停止了 Amazon EBS 支援的執行個體，而該執行個體似乎「凍結」於 stopping 狀態，則您可以強制將其停止。如需詳細資訊，請參閱 [停用執行個體時的疑難排解](#)。

## Command line

### 必要條件

確認執行個體的根裝置是 EBS 磁碟區。例如，執行[描述執行個體](#) AWS CLI 命令，並確認 RootDeviceType 是 ebs 否為 instance-store

### 停止和啟動 Amazon EBS 支援的執行個體

請使用以下其中一個命令：

- AWS CLI—[stop-instances](#) 與 [start-instances](#)。
- AWS Tools for PowerShell— [Stop-EC2Instance](#) 和 [Start-EC2Instance](#)。
- OS 命令—您可以使用 shutdown 或 poweroff 命令啟動關機作業。使用 OS 命令時，執行個體預設會停止。您可以變更這項預設動作，讓執行個體改為終止。如需詳細資訊，請參閱 [變更執行個體啟動的關機行為](#)。

[Linux 執行個體] 使用執行個體的 OS halt 指令不會啟動關機。如果您使用 halt 命令，執行個體不會終止，而是會將 CPU 放入 HLT，從而暫停 CPU 作業。執行個體會維持執行中狀態。

## 自動停止及啟動執行個體

您可以使用下列服務自動停止及啟動執行個體：

### 開啟例項排程器 AWS

您可以使用執行個體排程器 on AWS 來自動執行 EC2 執行個體的啟動和停止。如需詳細資訊，請參閱[如何搭配使用執行個體排程器 CloudFormation 來排程 EC2 執行個體？](#) 請注意，[這會額外收費](#)。

## AWS Lambda 和 Amazon 的 EventBridge 規則

您可以使用 Lambda 和 EventBridge 規則按排程停止和啟動執行個體。如需詳細資訊，請參閱[如何使用 Lambda 定期停止和啟動 Amazon EC2 執行個體？](#)

## Amazon EC2 Auto Scaling

為確保您有正確數量的 Amazon EC2 執行個體可用來處理應用程式的負載，請建立 Auto Scaling 群組。Amazon EC2 Auto Scaling 可確保您的應用程式始終具有處理流量需求的適當容量，且只在需要執行個體時啟動執行個體來節省成本。請注意，Amazon EC2 Auto Scaling 會終止不需要的執行個體，而不是停止。若要設定 Auto Scaling 群組，請參閱[開始使用 Amazon EC2 Auto Scaling](#)。

## 尋找所有執行中和停止的執行個體

您可以使用 [Amazon EC2 全域檢視](#)，在單一頁面 AWS 區域上找到所有執行中和已停止的執行個體。此功能對清查和查找遺忘的執行個體尤其有用。如需有關如何建立全域檢視的資訊，請參閱 [Amazon EC2 全域檢視](#)。

## 為執行個體啟用停止保護

若要避免執行個體意外停止，您可啟用執行個體的停止保護功能。停止保護還可以保護您的執行個體免遭意外終止。

Amazon EC2 [ModifyInstanceAttribute](#) API 的 `DisableApiStop` 屬性可控制執行個體是否可以使用 Amazon EC2 主控台 AWS CLI、或 Amazon EC2 API 停止執行個體。您可在執行個體啟動時、運作中或停止時，設定此屬性值。

### 考量事項

- 啟用停止保護功能無法防止您在使用作業系統命令 (例如 shutdown 或 poweroff) 從執行個體啟動關機作業，而意外停止執行個體。
- 啟用停止防護並不會防 AWS 止在[排程事件](#)停止執行個體時停止執行個體。
- 啟用停止保護不會阻止 Amazon EC2 Auto Scaling 在執行個體狀態不良時或在縮減事件期間終止執行個體。您可以使用[執行個體停止保護功能](#)，控制 Auto Scaling 群組是否能在縮減時終止特定的執行個體。
- 停止保護不僅可防止執行個體意外停止，還可避免在使用主控台 AWS CLI、或 API 時意外終止。但是此功能不會自動設定 `DisableApiTermination` 屬性。請注意，當 `DisableApiStop` 屬性設定

為時false，DisableApiTermination屬性設定會決定是否可以使用主控台 AWS CLI、或 API 終止執行個體。如需更多資訊，請參閱[終止 Amazon EC2 實例](#)。

- 您無法為執行個體儲存體所備份的執行個體啟用停止保護。
- 您無法為 Spot 執行個體啟用停止保護。
- 當您啟用或停用停止保護時，Amazon EC2 API 會遵循最終一致性模式。這表示執行命令來設定停止保護屬性的結果，可能不會立即顯示在您執行的所有後續命令中。如需詳細資訊，請參閱 Amazon EC2 開發人員指南中的[最終一致性](#)。

## 停止保護任務

- [在執行個體啟動時啟用停止保護](#)
- [針對執作中或已停止的執行個體啟用停止保護](#)
- [針對執作中或已停止的執行個體停用停止保護](#)

## 在執行個體啟動時啟用停止保護

您可以在執行個體啟動時，使用下列其中一種方法來啟用停止保護。

### Console

#### 在執行個體啟動時啟用停止保護

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在儀表板上，選擇啟動執行個體。
3. 在[新版啟動執行個體精靈](#)中設定執行個體。
4. 在精靈中，在進階詳細資訊下的停止保護選擇啟用，即可啟用停止保護功能。

### AWS CLI

#### 在執行個體啟動時啟用停止保護

使用[執行個體指 AWS CLI 令啟動執行個體](#)，並指定參數disable-api-stop。

```
aws ec2 run-instances \
 --image-id ami-a1b2c3d4e5example \
 --instance-type t3.micro \
 --key-name MyKeyPair \
 --stop-protection-enabled
```



```
--disable-api-stop \
...
```

## 針對執行中或已停止的執行個體啟用停止保護

您可以在執行個體為執行中或已停止狀態時，使用下列其中一種方法來啟用停止保護。

### Console

#### 針對執行中或已停止的執行個體啟用停止保護

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽窗格中選擇 (執行個體)。
3. 選取執行個體，然後選擇動作、執行個體設定、變更停止保護。
4. 勾選 Enable (啟用) 核取方塊，然後選取 Save (儲存)。

### AWS CLI

#### 針對執行中或已停止的執行個體啟用停止保護

使用 [修改-執行個體屬性](#) AWS CLI 指令並指定參數。disable-api-stop

```
aws ec2 modify-instance-attribute \
 --instance-id i-1234567890abcdef0 \
 --disable-api-stop
```

## 針對執行中或已停止的執行個體停用停止保護

您可以使用下列其中一種方法，來對執行中或已停止的執行個體停用停止保護。

### Console

#### 針對執行中或已停止的執行個體停用停止保護

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽窗格中選擇 (執行個體)。
3. 選取執行個體，然後選取 Actions (動作)、Instance Settings (執行個體設定)、Change Stop Protection (變更停止保護)。

4. 清除 Enable (啟用) 核取方塊，然後選擇 Save (儲存)。

## AWS CLI

針對執行中或已停止的執行個體停用停止保護

使用[修改-執行個體屬性](#) AWS CLI 指令並指定參數。no-disable-api-stop

```
aws ec2 modify-instance-attribute \
 --instance-id i-1234567890abcdef0 \
 --no-disable-api-stop
```

## 休眠您的 Amazon EC2 實例

當您讓執行個體休眠時，Amazon EC2 訊號會對作業系統執行休眠 (暫停磁碟機)。休眠會將執行個體記憶體 (RAM) 的內容儲存於您的 Amazon Elastic Block Store (Amazon EBS) 根磁碟區。Amazon EC2 會保留執行個體的 EBS 根磁碟區，以及任何已連接的 EBS 資料磁碟區。啟動執行個體時：

- EBS 根磁碟區會還原至其先前的狀態
- 重新載入 RAM 的內容
- 恢復先前在執行個體上執行的程序
- 先前連接的資料磁碟區會重新連接，且執行個體會保留其執行個體 ID

只有當您的執行個體已[啟用休眠](#)並符合[休眠必要條件](#)，才能讓執行個體進入休眠。

如果執行個體或應用程式需要花費長時間來引導並建置記憶體使用量，以完全發揮功效，則您可以使用休眠來「預先培養」執行個體。若要「預先培養」執行個體，您需要執行下列作業：

1. 在已啟用休眠的情況下將其啟動。
2. 使其進入所需的狀態。
3. 讓其休眠，以便準備好可在需要時恢復為所需狀態。

若執行個體處於 stopped 狀態而休眠，或 RAM 的內容要傳輸到 EBS 根磁碟區，這兩種情況的資料傳輸費用都不需要您支付。不過您需支付任何 EBS 磁碟區的儲存費用，包括 RAM 內容的儲存。

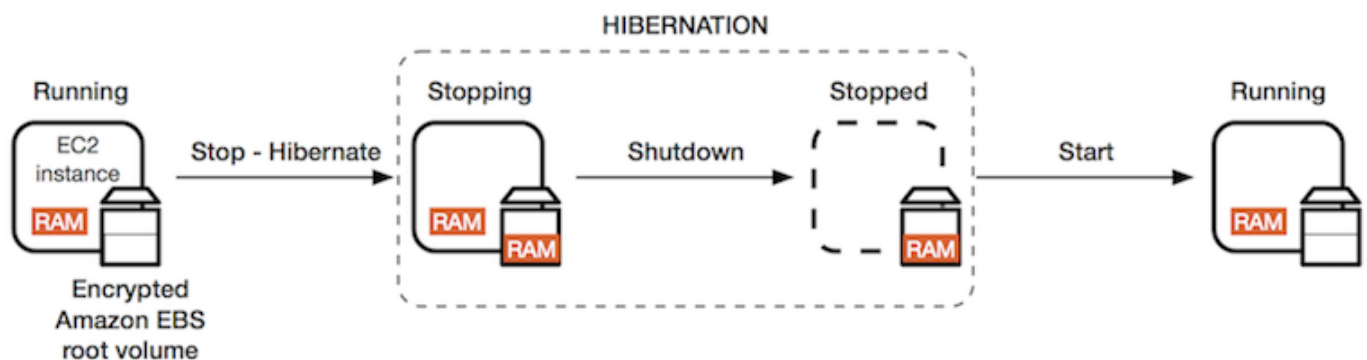
如果您認為不再需要某個執行個體，您可以隨時予以終止，即使該執行個體處於 stopped (休眠) 狀態也可以終止。如需詳細資訊，請參閱[終止 Amazon EC2 實例](#)。

## 目錄

- [亞馬遜 EC2 實例休眠如何工作](#)
- [亞馬遜 EC2 執行個體休眠的先決條件](#)
- [設定一個 AMI 以支援休眠](#)
- [啟用 Amazon EC2 執行個體的休眠](#)
- [在執行個體上停用 KASLR \(僅限 Ubuntu\)](#)
- [休眠亞馬遜 EC2 實例](#)
- [啟動休眠狀態的 Amazon EC2 執行個體](#)
- [亞馬遜 EC2 執行個體休眠的疑難](#)

## 亞馬遜 EC2 實例休眠如何工作

下圖顯示 EC2 執行個體休眠程序的基本概觀。



當您休眠執行個體時會發生什麼情況

當您休眠執行個體時，會發生下列情況：

- 執行個體會移至 `stopping` 狀態。Amazon EC2 會對作業系統發出執行休眠 (暫停磁碟機) 的訊號。休眠會凍結所有程序、將執行個體記憶體 (RAM) 的內容儲存於 EBS 根磁碟區，然後執行一般關機。
- 完成關機之後，執行個體會進入 `stopped` 狀態。
- 所有的 EBS 磁碟區仍會繼續連接至執行個體，其中的資料也會持續存在，包括儲存在 RAM 上的內容。
- 任何 Amazon EC2 執行個體存放區磁碟區都會維持連接到執行個體的状态，但執行個體存放區磁碟區上的資料會遺失。
- 當您的執行個體變為 `stopped` 狀態時，您可修改執行個體的部分屬性，包括執行個體類型或大小。

- 在大部分的情況中，執行個體會在啟動時轉移到新的基礎主機電腦。這也是您在停止並啟動執行個體時會發生的情況。
- 當您啟動執行個體時，執行個體會開機，作業系統會從 EBS 根磁碟區讀取 RAM 的內容，然後將程序恢復為先前的狀態。
- 執行個體會保留其私有 IPv4 位址及任何彈性 IPv6 位址。當您啟動執行個體時，執行個體會繼續保留其私有 IPv4 位址和任何 IPv6 位址。
- Amazon EC2 會釋出公用 IPv4 位址。當您啟動執行個體時，Amazon EC2 會將新公用 IPv4 位址指派給執行個體。
- 執行個體仍會保持與彈性 IP 地址的關聯。我們會針對和已休眠執行個體建立關聯的任何彈性 IP 地址向您收費。

如需休眠與重新開機、停止和終止之間差異的資訊，請參閱[重新啟動、停止、休眠和終止之間的差異](#)。

## 限制

- 讓執行個體休眠時，任何執行個體存放區磁碟區上的資料都會遺失。
- (Linux 執行個體) 您無法讓擁有超過 150 GB 記憶體體的 Linux 執行個體進行休眠。
- (視窗執行個體) 您無法讓擁有超過 16 GB 記憶體體的 Windows 執行個體進行休眠。
- 如果您從休眠或已啟用休眠的執行個體中建立快照或 AMI，您可能無法連線至從 AMI 或從快照建立的 AMI 啟動的新執行個體。
- (僅限 Spot 執行個體) 如果 Amazon EC2 讓您的 Spot 執行個體休眠，則只有 Amazon EC2 可以恢復您的執行個體。如果是您讓 Spot 執行個體休眠 ([使用者啟動的休眠](#))，您可以恢復執行個體。只有在有可用的容量，且 Spot 價格低於或等於您指定的最高價格時，才能恢復休眠的 Spot 執行個體。
- 您無法讓位於 Auto Scaling 群組或由 Amazon ECS 使用的執行個體進入休眠。若您的執行個體位於 Auto Scaling 群組中，且您嘗試讓其進入休眠，Amazon EC2 Auto Scaling 服務會將停止的執行個體標記為運作狀態不良，且可能會將其終止並啟動取代用執行個體。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南中的[自動擴展群組中執行個體的運作 Health 態檢查](#)。
- 您無法在已啟用 UEFI [安全開機](#)的情況下，將設定為在 UEFI 模式下開機的執行個體休眠。
- 如果您休眠啟動到 容量預留 的執行個體，容量預留 不確保休眠執行個體可以在您嘗試啟動之後恢復。
- 如果啟用了聯邦資訊處理標準 (FIPS) 模式，則無法將使用 5.10 以下核心的執行個體休眠。
- 我們不支援讓執行個體休眠超過 60 天。若要保留執行個體 60 天以上，您必須啟動休眠的執行個體、停止執行個體並將其啟動。

- 我們持續透過升級與安全修補程式來更新我們的平台，可能會與現有的已休眠執行個體衝突。在進行需要啟動已休眠執行個體的重大更新時，我們會通知您，以便我們執行關機或重新啟動來套用必要的升級和安全修補程式。

### 讓 Spot 執行個體休眠的考量

- 如果「您」讓 Spot 執行個體休眠，只要容量可用且 Spot 價格低於或等於您指定的最高價格，就可以重新啟動該執行個體。
- 如果「Amazon EC2」讓您的 Spot 執行個體休眠：
  - 只有 Amazon EC2 才能恢復您的執行個體。
  - 當 Spot 價格低於或等於您所指定的最高價格，而且有可用的容量時，Amazon EC2 會恢復已休眠的 Spot 執行個體。
  - 在 Amazon EC2 讓 Spot 執行個體休眠之前，您會在休眠開始前兩分鐘收到中斷通知。

如需詳細資訊，請參閱 [Spot 執行個體中斷](#)。

- 您可以透過多種方式讓 Spot 執行個體休眠。如需詳細資訊，請參閱 [指定中斷行為](#)。

### 亞馬遜 EC2 執行個體休眠的先決條件

您可以在啟動隨需執行個體或 Spot 執行個體時啟用休眠支援。您無法在現有執行個體上啟用休眠功能，無論執行中或已停止。如需詳細資訊，請參閱 [啟用執行個體休眠](#)。

#### 休眠執行個體的需求

- [AWS 區域](#)
- [AMI](#)
- [例證族群](#)
- [執行個體 RAM 大小](#)
- [根磁碟區類型](#)
- [根磁碟區大小](#)
- [根磁碟區加密](#)
- [EBS volume type \(EBS 磁碟區類型\)](#)
- [Spot 執行個體請求](#)

## AWS 區域

您可以在所 AWS 區域有實例中使用休眠。

## AMI

您必須使用支援休眠的 HVM AMI。下列 AMI 支援休眠：

## AMI

- 2023.09.20 或之後發行的 AL2023 AMI
- 2019.08.29 或更新版本發行的 Amazon Linux 2 AMI。
- 2018.11.16 或更新版本發行的 Amazon Linux AMI 2018.03。
- CentOS 版本 8 AMI <sup>1</sup> (需要設定[其他組態](#))
- Fedora 版本 34 或更新版本 AMI <sup>1</sup> (需要設定[其他組態](#))
- Red Hat Enterprise Linux (RHEL) 9 AMI <sup>1</sup> (需要設定[其他組態](#))
- Red Hat Enterprise Linux (RHEL) 8 AMI <sup>1</sup> (需要設定[其他組態](#))
- Ubuntu 22.04.2 LTS (Jammy Jellyfish) AMI，發行序號為 20230303 或更新版本 <sup>2</sup>
- Ubuntu 20.04 LTS (Focal Fossa) AMI，發行序號為 20210820 或更新版本 <sup>2</sup>
- Ubuntu 18.04 LTS (Bionic Beaver) AMI，發行序號為 20190722.1 或更新版本 <sup>2 4</sup>
- Ubuntu 16.04 LTS (Xenial Xerus) AMI <sup>2 3 4</sup> (需要設定[其他組態](#))

<sup>1</sup> 對於 CentOS、Fedora 和 Red Hat Enterprise Linux，只有以 Nitro 為基礎的執行個體才支援休眠。

<sup>2</sup> 建議您在採用 Ubuntu 22.04.2 LTS (Jammy Jellyfish)、Ubuntu 20.04 LTS (Focal Fossa)、Ubuntu 18.04 LTS (Bionic Beaver) 和 Ubuntu 16.04 LTS (Xenial Xerus) 的執行個體上停用 KASLR。如需詳細資訊，請參閱 [在執行個體上停用 KASLR \(僅限 Ubuntu\)](#)。

<sup>3</sup> 對於 Ubuntu 16.04 LTS (Xenial Xerus) AMI，t3.nano 執行個體類型上不支援休眠。由於 Ubuntu (Xenial Xerus) 已於 2021 年 4 月終止支援，因此不會提供任何修補程式。如果想要使用 t3.nano 執行個體類型，則建議升級至 Ubuntu 22.04.2 LTS (Jammy Jellyfish)、Ubuntu 20.04 LTS (Focal Fossa) AMI 或 Ubuntu 18.04 LTS (Bionic Beaver) AMI。

<sup>4</sup> 對 Ubuntu 18.04 LTS (Bionic Beaver) 和 Ubuntu 16.04 LTS (Xenial Xerus) 的支援已接近尾聲。

若要設定您自己的 AMI 以支援休眠，請參閱[設定一個 AMI 以支援休眠](#)。

即將推出其他版本的 Ubuntu 及其他作業系統的支援。

## 視窗 AMI

- 2023.09.13 或更新版本的 Windows Server 2022 AMI
- 2019.09.11 或更新版本的 Windows Server 2019 AMI
- 2019.09.11 或更新版本的 Windows Server 2016 AMI
- 2019.09.11 或更新版本的 Windows Server 2012 R2 AMI
- 2019.09.11 或更新版本的 Windows Server 2012 AMI

## 例證族群

您必須使用支援休眠的執行個體系列。

- 一般用途：立方米、M4、M5、M5 安培、M5 安培、M5D、M6i、M7i-柔性、T2、T3、T3A
- 計算最佳化：C3、C4、C5、C5d、C6i、C6id、C7a、C7 I-彈性
- 最佳化記憶體：R3、R4、R5、R5、R5 阿、R5、R5 分、R7 安
- 儲存最佳化：I3、i3en

Nitro 執行個體 — 不支援裸機執行個體。

查看特定區域中支援休眠的可用執行個體類型

可用的執行個體類型因區域而異。若要查看區域中支援休眠的可用執行個體類型，請使用 [describe-instance-types](#) 命令，搭配 `--region` 參數。包含 `--filters` 參數以將結果範圍限定為支援休眠的執行個體類型，以及包含 `--query` 參數以將輸出範圍限定為 InstanceType 的值。

```
aws ec2 describe-instance-types --filters Name=hibernation-supported,Values=true --query "InstanceTypes[*].[InstanceType]" --output text | sort
```

## 範例輸出

```
c3.2xlarge
c3.4xlarge
c3.8xlarge
c3.large
c3.xlarge
c4.2xlarge
c4.4xlarge
```

```
c4.8xlarge
```

```
...
```

## 執行個體 RAM 大小

Linux 執行個體 — 必須小於 150 GB。

視窗執行個體 — 最多可容納 16 GB。若要讓 T3 或 T3A 視窗執行個體休眠，我們建議至少有 1 GB 的記憶體。

## 根磁碟區類型

根磁碟區必須是 EBS 磁碟區，而非執行個體存放區。

## 根磁碟區大小

根磁碟區的大小必須足以儲存 RAM 內容並容納您預期的使用情況，例如作業系統或應用程式。如果您啟用休眠，則會於啟動時在根磁碟區上配置空間以存放 RAM。

## 根磁碟區加密

根磁碟區必須加密，以確保在休眠時保護記憶體中的敏感內容。當 RAM 的資料移至 EBS 根磁碟區時，一律會加密。根磁碟區的加密會在執行個體啟動時執行。

使用下列三個選項中的一個來確保根磁碟區是加密的 EBS 磁碟區：

- EBS 預設加密 - 您可以啟用 EBS 預設加密，以確保會加密在您 AWS 帳戶中建立的所有新 EBS 磁碟區。以此方式，您可以為您的執行個體啟用休眠，而不在執行個體啟動時指定加密意圖。如需詳細資訊，請參閱[預設啟用加密](#)。
- EBS 「單一步驟」加密 - 您可以從未加密的 AMI 啟動加密的 EBS 後端 EC2 執行個體，並可同時啟用休眠。如需詳細資訊，請參閱[對 EBS 後端 AMI 使用加密](#)。
- 加密 AMI - 您可以使用加密的 AMI 來啟動您的執行個體，藉此啟用 EBS 加密。如果您的 AMI 沒有加密的根快照，您可以將資料複製到新的 AMI，然後請求加密。如需詳細資訊，請參閱[複製期間加密未加密的映像](#) 及 [複製 AMI](#)。

## EBS volume type (EBS 磁碟區類型)

EBS 磁碟區必須使用下列其中一種 EBS 磁碟區類型：

- 一般用途 SSD (gp2 和 gp3)
- 佈建 IOPS SSD (io1 和 io2)



如果您選擇佈建 IOPS SSD 磁碟區類型，必須使用適當的 IOPS 佈建 EBS 磁碟區，才能獲得休眠的最佳效能。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的 Amazon EBS 磁碟區類型](#)。

## Spot 執行個體請求

對於 Spot 執行個體，下列需求適用：

- Spot 執行個體請求類型必須為 persistent。
- 您無法在 Spot 執行個體請求中指定啟動群組。

## 設定一個 AMI 以支援休眠

下列 Linux AMI 支援休眠，但若要讓使用其中一個 AMI 啟動的執行個體休眠，則需要額外的設定，才能讓執行個體休眠。

需要設定其他組態：

- [2019.08.29 或之後發佈的 Amazon Linux 2 最低 AMI](#)
- [2019.08.29 前發佈的 Amazon Linux 2](#)
- [2018.11.16 前發佈的 Amazon Linux](#)
- [CentOS 版本 8 或更新版本](#)
- [Fedora 版本 34 或更新版本](#)
- [Red Hat Enterprise Linux 版本 8 或 9](#)
- [Ubuntu 20.04 LTS \(Focal Fossa\) 在序號 20210820 前發佈的版本](#)
- [Ubuntu 18.04 \(Bionic Beaver\) 在序號 20190722.1 前發佈的版本](#)
- [Ubuntu 16.04 \(Xenial Xerus\)](#)

如需詳細資訊，請參閱[更新 Amazon Linux 2 執行個體上的執行個體軟體](#)。

以下 AMI 不需要設定其他組態，因為它們已經設定為支援休眠：

- 2023.09.20 或之後發行的 AL2023 AMI
- 2019.08.29 或之後發佈的 Amazon Linux 2 完整 AMI
- 2018.11.16 或更新版本的 Amazon Linux AMI 2018.03
- Ubuntu 22.04.2 LTS (Jammy Jellyfish) AMI，發行序號為 20230303 或更新版本
- Ubuntu 20.04 LTS (Focal Fossa) AMI，發行序號為 20210820 或更新版本

- Ubuntu 18.04 LTS (Bionic Beaver) AMI，發行序號為 20190722.1 或更新版本

2019.08.29 或之後發佈的 Amazon Linux 2 最低 AMI

設定 2019.08.29 或之後發佈的 Amazon Linux 2 最低 AMI 以支援休眠

1. 從儲存庫安裝 ec2-hibinit-agent 套件。

```
[ec2-user ~]$ sudo yum install ec2-hibinit-agent
```

2. 重新啟動服務。

```
[ec2-user ~]$ sudo systemctl start hibinit-agent
```

2019.08.29 前發佈的 Amazon Linux 2

將 2019.08.29 前發佈的 Amazon Linux 2 AMI 設定為支援休眠

1. 更新核心至 4.14.138-114.102 或更新版本。

```
[ec2-user ~]$ sudo yum update kernel
```

2. 從儲存庫安裝 ec2-hibinit-agent 套件。

```
[ec2-user ~]$ sudo yum install ec2-hibinit-agent
```

3. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

4. 確認核心版本已更新為 4.14.138-114.102 或更新版本。

```
[ec2-user ~]$ uname -a
```

5. 停止執行個體，並建立 AMI。如需詳細資訊，請參閱 [創建一個 Amazon EBS-backed AMI](#)。

## 2018.11.16 前發佈的 Amazon Linux

將 2018.11.16 前發佈的 Amazon Linux AMI 設定為支援休眠

1. 更新核心至 4.14.77-70.59 或更新版本。

```
[ec2-user ~]$ sudo yum update kernel
```

2. 從儲存庫安裝 ec2-hibinit-agent 套件。

```
[ec2-user ~]$ sudo yum install ec2-hibinit-agent
```

3. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

4. 確認核心版本已更新為 4.14.77-70.59 或更高版本。

```
[ec2-user ~]$ uname -a
```

5. 停止執行個體，並建立 AMI。如需詳細資訊，請參閱 [創建一個 Amazon EBS-backed AMI](#)。

## CentOS 版本 8 或更新版本

將 CentOS 版本 8 或更新版本的 AMI 設定為支援休眠

1. 更新核心至 4.18.0-305.7.1.el8\_4.x86\_64 或更新版本。

```
[ec2-user ~]$ sudo yum update kernel
```

2. 安裝 Fedora Extra Packages for Enterprise Linux (EPEL) 儲存庫。

```
[ec2-user ~]$ sudo yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
```

3. 從儲存庫安裝 ec2-hibinit-agent 套件。

```
[ec2-user ~]$ sudo yum install ec2-hibinit-agent
```

4. 啟用休眠代理程式以在開機時開始。

```
[ec2-user ~]$ sudo systemctl enable hibinit-agent.service
```

5. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

6. 確認核心版本已更新為 4.18.0-305.7.1.el8\_4.x86\_64 或更新版本。

```
[ec2-user ~]$ uname -a
```

## Fedora 版本 34 或更新版本

將 Fedora 版本 34 或更新版本的 AMI 設定為支援休眠

1. 更新核心至 5.12.10-300.fc34.x86\_64 或更新版本。

```
[ec2-user ~]$ sudo yum update kernel
```

2. 從儲存庫安裝 ec2-hibinit-agent 套件。

```
[ec2-user ~]$ sudo dnf install ec2-hibinit-agent
```

3. 啟用休眠代理程式以在開機時開始。

```
[ec2-user ~]$ sudo systemctl enable hibinit-agent.service
```

4. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

5. 確認核心版本已更新為 5.12.10-300.fc34.x86\_64 或更新版本。

```
[ec2-user ~]$ uname -a
```

## Red Hat Enterprise Linux 版本 8 或 9

### 設定 Red Hat Enterprise Linux 8 或 9 AMI 以支援休眠

1. 更新核心至 4.18.0-305.7.1.el8\_4.x86\_64 或更新版本。

```
[ec2-user ~]$ sudo yum update kernel
```

2. 安裝 Fedora Extra Packages for Enterprise Linux (EPEL) 儲存庫。

RHEL 版本 8 :

```
[ec2-user ~]$ sudo yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
```

RHEL 版本 9 :

```
[ec2-user ~]$ sudo yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-9.noarch.rpm
```

3. 從儲存庫安裝 ec2-hibinit-agent 套件。

```
[ec2-user ~]$ sudo yum install ec2-hibinit-agent
```

4. 啟用休眠代理程式以在開機時開始。

```
[ec2-user ~]$ sudo systemctl enable hibinit-agent.service
```

5. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

6. 確認核心版本已更新為 4.18.0-305.7.1.el8\_4.x86\_64 或更新版本。

```
[ec2-user ~]$ uname -a
```

## Ubuntu 20.04 LTS (Focal Fossa) 在序號 20210820 前發佈的版本

若要將序號 20210820 前發佈的 Ubuntu 20.04 LTS (Focal Fossa) AMI 設定為支援休眠

1. 將 grub2 更新 linux-aws-kernel 至 5.8.0-1038.40 或更新版本，並將 grub2 更新至 2.04-1ubuntu26.13 或更新版本。

```
[ec2-user ~]$ sudo apt update
[ec2-user ~]$ sudo apt dist-upgrade
```

2. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

3. 確認核心版本已更新為 5.8.0-1038.40 或更新版本。

```
[ec2-user ~]$ uname -a
```

4. 確認 grub2 版本已更新為 2.04-1ubuntu26.13 或更新版本。

```
[ec2-user ~]$ dpkg --get-selections | grep grub2-common
```

## Ubuntu 18.04 (Bionic Beaver) 在序號 20190722.1 前發佈的版本

將序號 20190722.1 前發佈的 Ubuntu 18.04 LTS AMI 設定為支援休眠

1. 更新核心至 4.15.0-1044 或更新版本。

```
[ec2-user ~]$ sudo apt update
[ec2-user ~]$ sudo apt dist-upgrade
```

2. 從儲存庫安裝 ec2-hibinit-agent 套件。

```
[ec2-user ~]$ sudo apt install ec2-hibinit-agent
```

3. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

4. 確認核心版本已更新為 4.15.0-1044 或更新版本。

```
[ec2-user ~]$ uname -a
```

## Ubuntu 16.04 (Xenial Xerus)

若要設定 Ubuntu 16.04 LTS 以支援休眠，您必須安裝 linux-aws-hwe 核心套件 4.15.0-1058 或更新版本，以及 ec2-劫持代理程式。

### Important

Canonical 支援 linux-aws-hwe 核心套件。對 Ubuntu 16.04 LTS 的標準支援已於 2021 年 4 月結束，且軟體套件不會再接收定期更新。但是，此套件會收到其他安全性更新，直到 2024 年結束 Extended Security Maintenance 支援為止。如需詳細資訊，請參閱 Canonical Ubuntu 部落格上的 [Amazon EC2 Hibernation for Ubuntu 16.04 LTS 現已推出](#)。

建議您升級至 Ubuntu 20.04 LTS (Focal Fossa) AMI 或 Ubuntu 18.04 LTS (Bionic Beaver) AMI。

將 Ubuntu 16.04 LTS AMI 設定為支援休眠

1. 更新核心至 4.15.0-1058-aws 或更新版本。

```
[ec2-user ~]$ sudo apt update
[ec2-user ~]$ sudo apt install linux-aws-hwe
```

2. 從儲存庫安裝 ec2-hibinit-agent 套件。

```
[ec2-user ~]$ sudo apt install ec2-hibinit-agent
```

3. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

4. 確認核心版本已更新為 4.15.0-1058-aws 或更新版本。

```
[ec2-user ~]$ uname -a
```

## 啟用 Amazon EC2 執行個體的休眠

若要讓執行個體休眠，必須在啟動執行個體時先為其啟用休眠。

### Important

啟動執行個體之後，就不能對它啟用或停用休眠。

### 主題

- [為隨需執行個體啟用休眠功能](#)
- [為 Spot 執行個體啟用休眠功能](#)
- [檢視執行個體是否已啟用休眠](#)

### 為隨需執行個體啟用休眠功能

使用下列任一種方法為隨需執行個體啟用休眠功能。

#### New console

##### 為隨需執行個體啟用休眠功能

1. 遵循程序[啟動執行個體](#)，但在完成下列步驟以啟用休眠之前，請勿啟動執行個體。
2. 若要啟用休眠功能，請在啟動執行個體精靈中設定下列欄位：
  - a. 在 Application and OS Images (Amazon Machine Image) (應用程式和作業系統映像 (Amazon Machine Image)) 中，選取支援休眠的 AMI。如需詳細資訊，請參閱 [AMI](#)。
  - b. 在 Instance type (執行個體類型) 中，選取支援的執行個體類型。如需詳細資訊，請參閱 [例證族群](#)。
  - c. 在 Configure storage (設定儲存) 中，選擇 Advanced (進階) (在右側)，並指定根磁碟區的下列資訊：
    - 針對 Size (GiB) (大小 (GiB))，輸入 EBS 根磁碟區大小。磁碟區必須夠大，才能存放 RAM 內容並容納您預期的使用量。
    - 針對 Volume type (磁碟區類型)，選取支援的 EBS 磁碟區類型：一般用途 SSD (gp2 和 gp3) 或佈建 IOPS SSD (io1 和 io2)。
    - 針對 Encrypted (加密)，選擇 Yes (是)。如果您在此 AWS 區域中預設啟用加密，則會選取 [是]。



- 針對 KMS key (KMS 金鑰)，選取磁碟區的加密金鑰。如果您在此 AWS 區域中預設啟用加密，則會選取預設加密金鑰。

如需根磁碟區必要條件的詳細資訊，請參閱[亞馬遜 EC2 執行個體休眠的先決條件](#)。

- d. 展開 Advanced details (進階詳細資訊)，針對 Stop - Hibernate behavior (停止 - 休眠行為)，選擇 Enable (啟用)。
3. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。如需詳細資訊，請參閱[使用新的啟動執行個體精靈啟動執行個體](#)。

## Old console

為隨需執行個體啟用休眠功能

1. 遵循[使用舊版啟動執行個體精靈啟動執行個體](#)程序。
2. 在 Choose an Amazon Machine Image (AMI) (選取 Amazon Machine Images (AMI)) 頁面上，選取支援休眠的 AMI。如需支援的 AMI 詳細資訊，請參閱[亞馬遜 EC2 執行個體休眠的先決條件](#)。
3. 在 Choose an Instance Type (選取執行個體類型) 頁面上，選取支援的執行個體類型，然後選取 Next: Configure Instance Details (下一步：設定執行個體詳細資訊)。如需支援的執行個體類型資訊，請參閱[亞馬遜 EC2 執行個體休眠的先決條件](#)。
4. 在 Configure Instance Details (設定執行個體詳細資訊) 頁面上，針對 Stop - Hibernate Behavior (停止 – 休眠行為)，選取 Enable hibernation as an additional stop behavior (啟用休眠做為額外的停止行為) 核取方塊。
5. 在 Add Storage (新增儲存) 頁面上，請為根磁碟區指定下列資訊：
  - 針對 Size (GiB) (大小 (GiB))，輸入 EBS 根磁碟區大小。磁碟區必須夠大，才能存放 RAM 內容並容納您預期的使用量。
  - 針對 Volume Type (磁碟區類型)，選取支援的 EBS 磁碟區類型，一般用途 SSD (gp2 和 gp3) 或佈建 IOPS SSD (io1 和 io2)。
  - 針對 Encryption (加密)，選取磁碟區的加密金鑰。如果您在此 AWS 區域中預設啟用加密，則會選取預設加密金鑰。

如需根磁碟區必要條件的詳細資訊，請參閱[亞馬遜 EC2 執行個體休眠的先決條件](#)。

- 依精靈提示繼續作業。檢閱完您在 Review Instance Launch (檢閱執行個體啟動) 頁面的選項後，請選擇 Launch (啟動)。如需詳細資訊，請參閱 [使用舊版啟動執行個體精靈啟動執行個體](#)。

## AWS CLI

為隨需執行個體啟用休眠功能

使用 [run-instances](#) 命令來啟動執行個體。使用 `--block-device-mappings file://mapping.json` 參數指定 EBS 根磁碟區參數，然後使用 `--hibernation-options Configured=true` 參數啟用休眠。

```
aws ec2 run-instances \
 --image-id ami-0abcdef1234567890 \
 --instance-type m5.large \
 --block-device-mappings file://mapping.json \
 --hibernation-options Configured=true \
 --count 1 \
 --key-name MyKeyPair
```

在 `mapping.json` 中指定下列內容。

```
[
 {
 "DeviceName": "/dev/xvda",
 "Ebs": {
 "VolumeSize": 30,
 "VolumeType": "gp2",
 "Encrypted": true
 }
 }
]
```

### Note

`DeviceName` 的值必須符合與 AMI 相關聯的根設備名稱。若要尋找根設備名稱，請使用 [describe-images](#) 命令。

```
aws ec2 describe-images --image-id ami-0abcdef1234567890
```

如果您在此 AWS 區域中預設啟用加密，則可以省略 "Encrypted": true。

## PowerShell

若要啟用隨需執行個體的休眠，請使用 AWS Tools for Windows PowerShell

使用指 [New-EC2Instance](#) 令啟動執行個體。先定義區塊型設備映射，然後使用 -BlockDeviceMappings 參數將其新增至命令，以指定 EBS 根磁碟區。使用 -HibernationOptions\_Configured \$true 參數啟用休眠。

```
PS C:\> $ebs_encrypt = New-Object Amazon.EC2.Model.BlockDeviceMapping
PS C:\> $ebs_encrypt.DeviceName = "/dev/xvda"
PS C:\> $ebs_encrypt.Ebs = New-Object Amazon.EC2.Model.EbsBlockDevice
PS C:\> $ebs_encrypt.Ebs.VolumeSize = 30
PS C:\> $ebs_encrypt.Ebs.VolumeType = "gp2"
PS C:\> $ebs_encrypt.Ebs.Encrypted = $true

PS C:\> New-EC2Instance `
 -ImageId ami-0abcdef1234567890 `
 -InstanceType m5.Large `
 -BlockDeviceMappings $ebs_encrypt `
 -HibernationOptions_Configured $true `
 -MinCount 1 `
 -MaxCount 1 `
 -KeyName MyKeyPair
```

### Note

DeviceName 的值必須符合與 AMI 相關聯的根設備名稱。若要尋找根裝置名稱，請使用 [Get-EC2Image](#) 指令。

```
Get-EC2Image -ImageId ami-0abcdef1234567890
```

如果您在此 AWS 區域中預設啟用加密功能，則可以 Encrypted = \$true 從區塊裝置對應中省略。

## 為 Spot 執行個體啟用休眠功能

使用下列任一種方法為 Spot 執行個體啟用休眠功能。如需中斷時讓 Spot 執行個體休眠的詳細資訊，請參閱 [Spot 執行個體中斷](#)。

### Console

您可以使用 Amazon EC2 主控台內的啟動執行個體精靈，為 Spot 執行個體啟用休眠功能。

#### 為 Spot 執行個體啟用休眠功能

1. 遵循程序 [使用啟動執行個體精靈要求 Spot 執行個體](#)，但在完成下列步驟以啟用休眠之前，請勿啟動執行個體。
2. 若要啟用休眠功能，請在啟動執行個體精靈中設定下列欄位：
  - a. 在 Application and OS Images (Amazon Machine Image) (應用程式和作業系統映像 (Amazon Machine Image)) 中，選取支援休眠的 AMI。如需詳細資訊，請參閱 [AMI](#)。
  - b. 在 Instance type (執行個體類型) 中，選取支援的執行個體類型。如需詳細資訊，請參閱 [例證族群](#)。
  - c. 在 Configure storage (設定儲存) 中，選擇 Advanced (進階) (在右側)，並指定根磁碟區的下列資訊：
    - 針對 Size (GiB) (大小 (GiB))，輸入 EBS 根磁碟區大小。磁碟區必須夠大，才能存放 RAM 內容並容納您預期的使用量。
    - 針對 Volume type (磁碟區類型)，選取支援的 EBS 磁碟區類型：一般用途 SSD (gp2 和 gp3) 或佈建 IOPS SSD (io1 和 io2)。
    - 針對 Encrypted (加密)，選擇 Yes (是)。如果您在此 AWS 區域中預設啟用加密，則會選取 [是]。
    - 針對 KMS key (KMS 金鑰)，選取磁碟區的加密金鑰。如果您在此 AWS 區域中預設啟用加密，則會選取預設加密金鑰。

如需根磁碟區必要條件的詳細資訊，請參閱 [亞馬遜 EC2 執行個體休眠的先決條件](#)。

- d. 展開進階詳細資訊，除了用於設定 Spot 執行個體的欄位之外，請執行下列操作：
  - i. 對於要求類型，請選擇持續。
  - ii. 對於中斷行為，請選擇休眠。或者，對於停止 - 休眠行為，請選擇啟用。這兩個欄位都能在 Spot 執行個體上啟用休眠。您只須設定其中之一。

3. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。如需詳細資訊，請參閱 [使用新的啟動執行個體精靈啟動執行個體](#)。

## AWS CLI

您可以使用 [run-instances](#) AWS CLI 命令，為 Spot 執行個體啟用休眠功能。

使用 **hibernation-options** 參數為 Spot 執行個體啟用休眠功能

使用 [run-instances](#) 命令要求 Spot 執行個體。使用 `--block-device-mappings file://mapping.json` 參數指定 EBS 根磁碟區參數，然後使用 `--hibernation-options Configured=true` 參數啟用休眠。Spot 要求類型 (SpotInstanceType) 必須是 `persistent`。

```
aws ec2 run-instances \
 --image-id ami-0abcdef1234567890 \
 --instance-type c4.xlarge \
 --block-device-mappings file://mapping.json \
 --hibernation-options Configured=true \
 --count 1 \
 --key-name MyKeyPair \
 --instance-market-options \
 { \
 "MarketType": "spot", \
 "SpotOptions": { \
 "MaxPrice": "1", \
 "SpotInstanceType": "persistent" \
 } \
 } \
 }
```

在 `mapping.json` 中指定 EBS 根磁碟區參數，如下所示。

```
[\
 { \
 "DeviceName": "/dev/xvda", \
 "Ebs": { \
 "VolumeSize": 30, \
 "VolumeType": "gp2", \
 "Encrypted": true \
 } \
 } \
]
```

]

**Note**

DeviceName 的值必須符合與 AMI 相關聯的根設備名稱。若要尋找根設備名稱，請使用 [describe-images](#) 命令。

```
aws ec2 describe-images --image-id ami-0abcdef1234567890
```

如果您在此 AWS 區域中預設啟用加密，則可以省略 "Encrypted": true。

## PowerShell

若要啟用 Spot 執行個體的休眠，請使用 AWS Tools for Windows PowerShell

使用命 [New-EC2Instance](#) 令要求 Spot 執行個體。先定義區塊型設備映射，然後使用 -BlockDeviceMappings 參數將其新增至命令，以指定 EBS 根磁碟區。使用 -HibernationOptions\_Configured \$true 參數啟用休眠。

```
PS C:\> $ebs_encrypt = New-Object Amazon.EC2.Model.BlockDeviceMapping
PS C:\> $ebs_encrypt.DeviceName = "/dev/xvda"
PS C:\> $ebs_encrypt.Ebs = New-Object Amazon.EC2.Model.EbsBlockDevice
PS C:\> $ebs_encrypt.Ebs.VolumeSize = 30
PS C:\> $ebs_encrypt.Ebs.VolumeType = "gp2"
PS C:\> $ebs_encrypt.Ebs.Encrypted = $true

PS C:\> New-EC2Instance `
 -ImageId ami-0abcdef1234567890 `
 -InstanceType m5.Large `
 -BlockDeviceMappings $ebs_encrypt `
 -HibernationOptions_Configured $true `
 -MinCount 1 `
 -MaxCount 1 `
 -KeyName MyKeyPair `
 -InstanceMarketOption @(
 MarketType = spot;
 SpotOptions @{
 MaxPrice = 1;
 SpotInstanceType = persistent}
)
```

**Note**

DeviceName 的值必須符合與 AMI 相關聯的根設備名稱。若要尋找根裝置名稱，請使用 [Get-EC2Image](#) 指令。

```
Get-EC2Image -ImageId ami-0abcdef1234567890
```

如果您在此 AWS 區域中預設啟用加密功能，則可以 Encrypted = \$true 從區塊裝置對應中省略。

您可以透過多種方式讓 Spot 執行個體休眠。如需詳細資訊，請參閱 [指定中斷行為](#)。

### 檢視執行個體是否已啟用休眠

透過下列說明來檢視執行個體是否已啟用休眠。

#### Console

##### 檢視執行個體是否已啟用休眠

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後在 Instance details (執行個體詳細資訊) 區段的 Details (詳細資料) 標籤上，檢查 Stop-hibernate behavior (停止休眠行為)。Enabled (啟用) 表示執行個體已啟用休眠。

#### AWS CLI

##### 檢視執行個體是否已啟用休眠

使用 [describe-instances](#) 命令，並指定 `--filters "Name=hibernation-options.configured,Values=true"` 參數來篩選已啟用休眠的執行個體。

```
aws ec2 describe-instances \
 --filters "Name=hibernation-options.configured,Values=true"
```

輸出中的下列欄位表示執行個體已啟用休眠。

```
"HibernationOptions": {
 "Configured": true
}
```

## PowerShell

使用 AWS Tools for Windows PowerShell 來檢視執行個體是否已啟用休眠

使用命 [Get-EC2Instance](#) 令並指定 `-Filter @{ Name="hibernation-options.configured"; Value="true"}` 參數來篩選啟用休眠的執行個體。

```
(Get-EC2Instance -Filter @{Name="hibernation-options.configured";
Value="true"}).Instances
```

輸出會列出已啟用休眠的 EC2 執行個體。

## 在執行個體上停用 KASLR (僅限 Ubuntu)

若要在配備 Ubuntu 16.04 LTS (Xenial Xerus)、Ubuntu 18.04 LTS (Bionic Beaver) (發行序號為 20190722.1 或更新版本) 或 Ubuntu 20.04 LTS (Focal Fossa) (發行序號為 20210820 或更新版本) 的新啟動執行個體上執行休眠，建議停用 KASLR (Kernel Address Space Layout Randomization)。在 Ubuntu 16.04 LTS、Ubuntu 18.04 LTS 或 Ubuntu 20.04 LTS 上，KASLR 預設為啟用。

KASLR 是標準的 Linux 核心安全功能，可以將核心的基底位址值隨機化，有助於儘量避免受到尚未發現的記憶體存取漏洞所拖累和影響。在啟用 KASLR 的情況下，執行個體在休眠之後可能不會恢復。

若要進一步了解 KASLR，請參閱 [Ubuntu 功能](#)。

在使用 Ubuntu 啟動的執行個體上停用 KASLR

1. 使用 SSH 連線到您的執行個體。如需詳細資訊，請參閱 [the section called “從 Linux 或 macOS 進行連線”](#)。
2. 在您選擇的編輯器中開啟 `/etc/default/grub.d/50-cloudimg-settings.cfg` 檔案。編輯 `GRUB_CMDLINE_LINUX_DEFAULT` 行，在行尾附加 `nokaslr` 選項，如下列範例所示。

```
GRUB_CMDLINE_LINUX_DEFAULT="console=tty1 console=ttyS0
nvme_core.io_timeout=4294967295 nokaslr"
```

3. 儲存檔案並結束您的編輯器。
4. 執行下列命令來重建 grub 組態。



```
[ec2-user ~]$ sudo update-grub
```

5. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

6. 執行以下命令來確認已新增 nokaslr。

```
[ec2-user ~]$ cat /proc/cmdline
```

命令的輸出應該會包含 nokaslr 選項。

## 休眠亞馬遜 EC2 實例

如果執行個體是 EBS 支援的執行個體、[啟用休眠功能](#)並符合[休眠必要條件](#)，您可以在隨需執行個體或 Spot 執行個體上啟動休眠。如果無法讓執行個體進入休眠，則會以一般方式關機。

### Console

#### 讓執行個體休眠

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選取 Instance state (執行個體狀態)、Hibernate instance (休眠執行個體)。如果停用了 Hibernate instance (休眠執行個體)，則已進入休眠或已停止的執行個體將無法進入休眠。如需詳細資訊，請參閱 [亞馬遜 EC2 執行個體休眠的先決條件](#)。
4. 出現確認提示時，請選擇 Hibernate (休眠)。讓執行個體進入休眠可能需要幾分鐘。執行個體狀態會先變更為 Stopping (正在停止)，然後在執行個體已進入休眠時變更為 Stopped (已停止)。

### AWS CLI

#### 讓 EBS 支援的執行個體休眠

使用 [stop-instances](#) 命令，並指定 `--hibernate` 參數。

```
aws ec2 stop-instances \
 --instance-ids i-1234567890abcdef0 \
 --hibernate
```

```
--hibernate
```

## PowerShell

若要使執行個體休眠，請使用 AWS Tools for Windows PowerShell

使用指 [Stop-EC2Instance](#) 命令並指定 `-Hibernate $true` 參數。

```
Stop-EC2Instance `
 -InstanceId i-1234567890abcdef0 `
 -Hibernate $true
```

## Console

檢視執行個體是否已啟動休眠

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體，然後在詳細資訊索引標籤的執行個體詳細資訊區段中，查看狀態轉換訊息的值。

用戶端。UserInitiated休眠：使用者啟動的休眠表示您已在隨需執行個體或 Spot 執行個體上啟動休眠。

## AWS CLI

檢視執行個體是否已啟動休眠

使用 [describe-instances](#) 命令並指定 `state-reason-code` 篩選條件，以查看已初始化休眠的執行個體。

```
aws ec2 describe-instances `
 --filters "Name=state-reason-code,Values=Client.UserInitiatedHibernate"
```

輸出中的下列欄位表示隨需執行個體或 Spot 執行個體已啟動休眠。

```
"StateReason": {
 "Code": "Client.UserInitiatedHibernate"
```

```
}
```

## PowerShell

使用 AWS Tools for Windows PowerShell 來檢視執行個體是否已初始化休眠

使用命 [Get-EC2Instance](#) 令並指定 `state-reason-code` 篩選器，以查看起始休眠的執行個體。

```
Get-EC2Instance `
 -Filter @{"Name"="state-reason-code";Value="Client.UserInitiatedHibernate"}
```

輸出會列出已初始化休眠的 EC2 執行個體。

## 啟動休眠狀態的 Amazon EC2 執行個體

透過與啟動已停止執行個體的相同方式，來啟動休眠的執行個體。

### Note

對於 Spot 執行個體，如果是 Amazon EC2 讓執行個體休眠，則只有 Amazon EC2 可以恢復執行個體。只有是「您」讓 Spot 執行個體休眠時，您才能恢復該執行個體。只有在容量可用，且 Spot 價格低於或等於您指定的最高價格時，才能恢復 Spot 執行個體。

## Console

### 啟動休眠的執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取休眠的執行個體，並選取 Instance state (執行個體狀態)、Start instance (啟動執行個體)。執行個體進入 running 狀態可能需要數分鐘。在此期間內，執行個體的 [狀態檢查](#) 會顯示執行個體處於失敗狀態，直到執行個體啟動。

## AWS CLI

### 啟動休眠的執行個體

使用 [start-instances](#) 命令。

```
aws ec2 start-instances \
 --instance-ids i-1234567890abcdef0
```

## PowerShell

使用啟動休眠的執行個體 AWS Tools for Windows PowerShell

使用 [Start-EC2Instance](#) 命令。

```
Start-EC2Instance \
 -InstanceId i-1234567890abcdef0
```

## 亞馬遜 EC2 執行個體休眠的疑難

使用此資訊來協助您診斷並修正讓執行個體進入休眠時可能遇到的問題。

### 休眠問題

- [無法在啟動後立即進入休眠](#)
- [從 stopping 轉換為 stopped 花費太長時間，且記憶體狀態未在啟動後復原](#)
- [執行個體「卡」在停止狀態](#)
- [休眠後無法立即啟動 Spot 執行個體](#)
- [繼續執行 Spot 執行個體失敗](#)

### 無法在啟動後立即進入休眠

如果您在啟動執行個體後太快嘗試讓執行個體進入休眠，則會出現錯誤。

Linux 執行個體必須等待大約兩分鐘，啟動後 Windows 執行個體必須等待約五分鐘，然後才能休眠。

### 從 **stopping** 轉換為 **stopped** 花費太長時間，且記憶體狀態未在啟動後復原

如果休眠的執行個體花費長時間才能從 stopping 狀態轉換為 stopped，且如果記憶體狀態未在啟動後復原，則可能表示休眠未正確設定。

### Linux 執行個體

請檢查執行個體系統日誌，尋找與休眠相關的訊息。若要存取系統日誌，請[連線](#)到執行個體，或使用 [get-console-output](#) 命令。從 hibinit-agent 尋找日誌行。如果日誌行指出失敗或日誌行不存在，則很可能在啟動時有錯誤的休眠設定。

例如，下列訊息表示執行個體根磁碟區不夠大：`hibinit-agent: Insufficient disk space. Cannot create setup for hibernation. Please allocate a larger root device.`

如果 `hibinit-agent` 最後的日誌行是 `hibinit-agent: Running: swapoff /swap`，則表示休眠已成功設定。

如果您沒有在程序中看到任何日誌，則您的 AMI 可能不支援休眠。如需支援的 AMI 資訊，請參閱[亞馬遜 EC2 執行個體休眠的先決條件](#)。如果您使用自己的 Linux AMI，請確定您遵循的指示[設定一個 AMI 以支援休眠](#)。

## Windows Server 2016 及更新版本

請檢查 EC2 啟動日誌，尋找與休眠相關的訊息。若要存取 EC2 啟動日誌，[連線](#)至執行個體並在文字編輯器中開啟 `C:\ProgramData\Amazon\EC2-Windows\Launch\Log\Ec2Launch.log` 檔案。如果使用 EC2Launch v2，請開啟 `C:\ProgramData\Amazon\EC2Launch\log\agent.log`。

### Note

根據預設，Windows 會將檔案和資料夾隱藏在 `C:\ProgramData` 下。若要檢視 EC2 啟動目錄和檔案，您必須在 Windows 檔案總管中輸入路徑或變更資料夾屬性，才可顯示隱藏的檔案和資料夾。

尋找休眠的日誌行。如果日誌行指出失敗或日誌行不存在，則很可能在啟動時有錯誤的休眠設定。

例如，下列訊息表示無法設定休眠：`Message: Failed to enable hibernation.`。如果錯誤訊息包含十進位 ASCII 值，您可以將 ASCII 值轉換為純文字，以便讀取完整的錯誤訊息。

如果日誌行包含 `HibernationEnabled: true`，則表示休眠已成功設定。

## Windows Server 2012 R2 和舊版

請檢查 EC2 設定日誌，尋找與休眠相關的訊息。若要存取 EC2 設定日誌，請[連線](#)至執行個體，接著在文字編輯器中開啟 `C:\Program Files\Amazon\Ec2ConfigService\Logs\Ec2ConfigLog.txt` 檔案。尋找 `SetHibernateOnSleep` 的日誌行。如果日誌行指出失敗或日誌行不存在，則很可能在啟動時有錯誤的休眠設定。

例如，下列訊息表示執行個體根磁碟區不夠大：`SetHibernateOnSleep: Failed to enable hibernation: Hibernation failed with the following error: There is not enough space on the disk.`

如果日誌行是 `SetHibernateOnSleep: HibernationEnabled: true`，則表示休眠已成功設定。

## 視窗實體大小

如果您使用的記憶體少於 1 GB 的 T3 或 T3A Windows 執行個體，請嘗試將執行個體的大小增加為至少具有 1 GB 記憶體的執行個體。

## 執行個體「卡」在停止狀態

如果您讓執行個體進入休眠，但該執行個體「卡」在 `stopping` 狀態，您可以強制將其停止。如需詳細資訊，請參閱 [停用執行個體時的疑難排解](#)。

## 休眠後無法立即啟動 Spot 執行個體

如果您嘗試在休眠後兩分鐘內啟動 Spot 執行個體，可能會出現下列錯誤：

```
You failed to start the Spot Instance because the associated Spot Instance request is not in an appropriate state to support start.
```

請等待 Linux 執行個體大約兩分鐘，Windows 執行個體約 5 分鐘，然後重試啟動執行個體。

## 繼續執行 Spot 執行個體失敗

如果您的 Spot 執行個體已成功休眠，但是無法繼續執行，而是會重新啟動 (未保留休眠狀態而直接重新啟動)，可能是因為使用者資料包含下列指令碼：

```
/usr/bin/enable-ec2-spot-hibernation
```

從啟動範本的使用者資料欄位中移除此指令碼，然後請求新的 Spot 執行個體。

請注意，即使執行個體無法繼續執行，在不保留休眠狀態的狀態下，仍可使用與從 `stopped` 狀態啟動的相同方式啟動執行個體。

## 重新啟動您的執行個體

重新啟動執行個體等同於重新啟動作業系統。在大多數情況下，將執行個體重新開機只需要幾分鐘的時間。

重新啟動執行個體時，其會保留下列項目：

- 公有 DNS 名稱 (IPv4)

- 私有 IPv4 地址
- 公有 IPv4 地址
- IPv6 地址 (如適用)
- 其執行個體儲存體磁碟區上的任何資料

有別於[停用和啟動](#)執行個體，將執行個體重新開機不會啟動新的執行個體計費期間 (最少收費一分鐘)。

我們可能會因必要的維護而排程重新開機執行個體 (例如，套用更新後需要重新開機)。您不需要採取任何動作；建議您於排程的時間範圍內等待其重新開機。如需詳細資訊，請參閱[您的執行個體的排程事件](#)。

建議您使用 Amazon EC2 主控台、命令列工具或 Amazon EC2 API 將執行個體重新開機，而不是從執行個體執行重新開機作業系統的命令。如果您使用 Amazon EC2 主控台、命令列工具或 Amazon EC2 API 將執行個體重新開機，執行個體若未於幾分鐘內完全關機，將執行硬式重新啟動。如果您使用 AWS CloudTrail，則使用 Amazon EC2 將執行個體重新開機也會建立 API 記錄，記錄執行個體重新開機的時間。

## Windows 執行個體

如果 Windows 正在執行個體上安裝更新，建議您不要在所有更新安裝完畢之前，使用 Amazon EC2 主控台或命令列來重新開機或關機執行個體。當您使用 Amazon EC2 主控台或命令列重新開機或關機執行個體時，會有硬式重新開機執行個體的風險。正在安裝更新時的硬式重新開機可能會讓執行個體進入不穩定狀態。

## Console

### 使用主控台重新開機執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體，然後選取 Instance state (執行個體狀態)、Reboot instance (將執行個體重新開機)。

或者，選取執行個體，然後選取 Actions (動作)、Manage instance state (管理執行個體狀態)。在開啟的畫面中，選擇 Reboot (重新開機)，然後選擇 Change state (變更狀態)。

4. 出現確認提示時，選擇重新開機。

執行個體會保持在 `running` 狀態。

## Command line

### 重新啟動執行個體

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱[存取 Amazon EC2](#)。

- [reboot-instances](#) (AWS CLI)
- [Restart-EC2Instance](#) (AWS Tools for Windows PowerShell)

### 若要執行受控的錯誤注入實驗

您可以用 AWS Fault Injection Service 來測試執行個體重新啟動時應用程式的回應方式。如需詳細資訊，請參閱 [AWS Fault Injection Service 使用者指南](#)。

## 終止 Amazon EC2 實例

當您不再需要執行個體時，可以將它刪除。這就是終止您的執行個體。執行個體的狀態變更為 shutting-down 或 terminated 時刻起，該執行個體便停止收取費用。

執行個體終止後，您將無法連接或啟動該執行個體。然而，您可使用相同 AMI 來啟動其他執行個體。如果您想要停止或休眠執行個體，請參閱[停止並啟動 Amazon EC2 執行個體](#)或[休眠您的 Amazon EC2 實例](#)。如需詳細資訊，請參閱 [重新啟動、停止、休眠和終止之間的差異](#)。

### 目錄

- [實例終止的工作原理](#)
- [終止執行個體](#)
- [對執行個體終止進行故障診斷](#)
- [啟用終止保護](#)
- [變更執行個體啟動的關機行為](#)
- [在執行個體終止時保留資料](#)

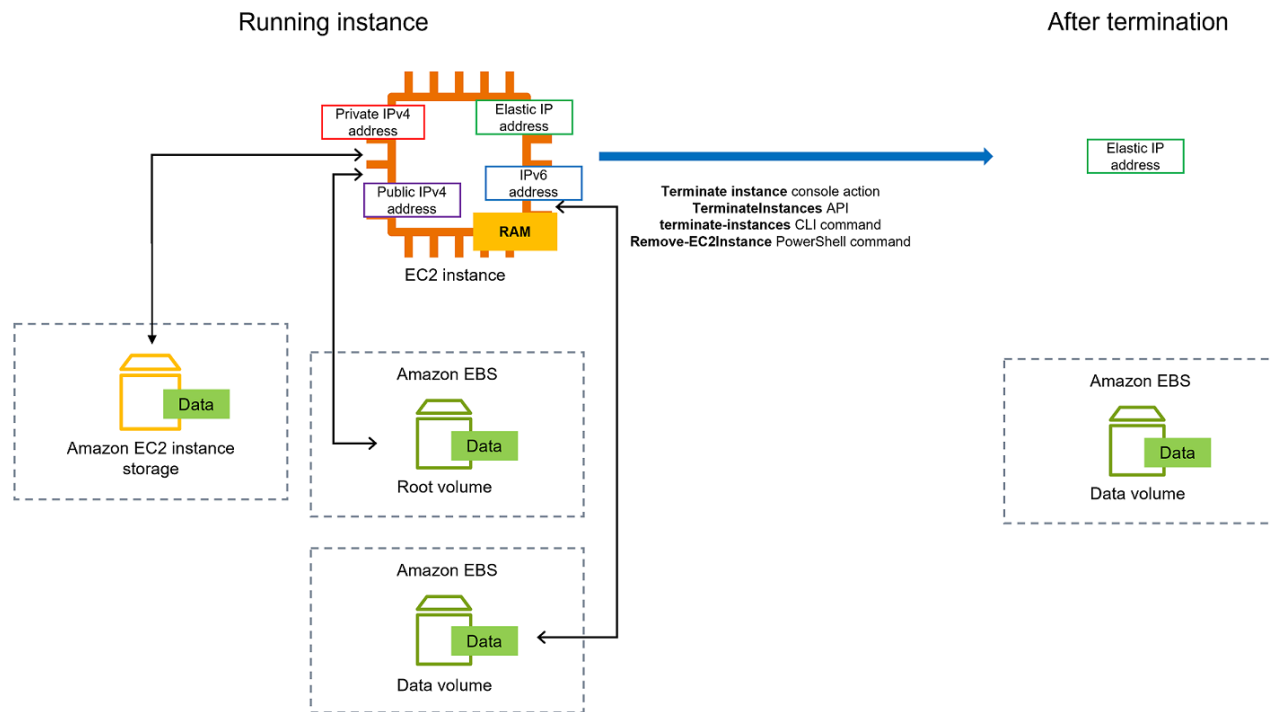
### 實例終止的工作原理

當您終止執行個體時，變更會在執行個體的作業系統層級註冊，部分資源會遺失，而且部分資源會持續存在。

下圖顯示 Amazon EC2 執行個體終止時遺失的內容，以及遺失的內容。執行個體終止時，會清除任何執行個體儲存磁碟區上的資料，以及儲存在執行個體 RAM 的資料。與執行個體相關聯的任何彈性 IP



位址都會中斷連線。對於 Amazon EBS 磁碟區和這些磁碟區上的資料，結果取決於磁碟區的終止時刪除設定。根據預設，會刪除根磁碟區並保留資料磁碟區。



## 考量事項

- 執行個體終止時，與該執行個體建立關聯的執行個體存放區磁碟區上之資料，將被刪除。
- 根據預設，執行個體終止時，Amazon EBS 根設備磁碟區將自動被刪除。然而，您於啟動時連接的其他 EBS 磁碟區或您連接至現有執行個體的 EBS 磁碟區都將保留，即使該執行個體已終止。如需詳細資訊，請參閱 [在執行個體終止時保留資料](#)。

### Note

執行個體終止時未刪除的任何磁碟區將繼續產生費用。

- 若要防止某個執行個體遭到他人意外終止，請[啟用終止保護](#)。
- 若要控制執行個體在執行個體啟動關機時是否停止或終止，請變更執行個體[起始的關機行為](#)。
- 如果您在執行個體終止時執行指令碼，則執行個體可能會發生異常終止，因為我們無法確保關閉指令碼執行。Amazon EC2 會嘗試完全關閉執行個體，並執行任何系統關閉指令碼；不過，某些事件 (例如硬體故障) 可能會使這些系統關閉指令碼無法執行。

## 若您終止執行個體會如何

### 在作業系統層級註冊的變更

- API 請求會將按鈕事件傳送給訪客。
- 由於按鈕事件而停止各種系統服務。系統的正常關機由systemd ( Linux ) 或系統進程 ( 視窗 ) 提供。順利關機由從虛擬化管理程序按下 ACPI 關機按鈕事件觸發。
- 啟動 ACPI 關機。
- 執行個體會在正常關機程序結束後關閉。OS 關機時間無法設定。執行個體仍會短暫顯示於主控台，之後該項目會自動刪除。

### 資源丟失

- 存放在執行個體儲存體磁碟區的資料。
- 如果 DeleteOnTermination 屬性設定為 true，則為存放在 Amazon EBS 根裝置磁碟區上的資料。

### 持續存在的資源

- 存放在其他 Amazon EBS 磁碟區上的資料，會在啟動執行個體當下或之後連接。

### 測試應用程式對執行個體終止的回應

您可以用 AWS Fault Injection Service 來測試執行個體終止時應用程式的回應方式。如需詳細資訊，請參閱 [AWS Fault Injection Service 使用者指南](#)。

## 終止執行個體

您可以隨時終止執行個體。

### Console

欲使用主控台來終止執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，並選取 Instance state (執行個體狀態)、Terminate instance (終止執行個體)。
4. 出現確認提示時，請選擇 Terminate (終止)。

5. 在您終止執行個體之後，它會在短時間內保持可見狀態，且狀態為 `terminated`。

如果終止失敗或終止的執行個體可見超過數小時，請參閱 [仍顯示已終止的執行個體](#)。

## Command line

欲使用命令列來終止執行個體

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [terminate-instances](#) (AWS CLI)
- [Remove-EC2Instance](#) (AWS Tools for Windows PowerShell)

## 對執行個體終止進行故障診斷

請求者必須具有撥打電話 `ec2:TerminateInstances` 的權限。如需詳細資訊，請參閱 [使用執行個體的範例原則](#)。

如果終止您的執行個體，並啟動另一個執行個體，很可能您已經透過像 EC2 Fleet 或 Amazon EC2 Auto Scaling 這樣的功能來設定自動擴展。如需詳細資訊，請參閱 [執行個體自動啟動或終止](#)。

如果終止保護已開啟，則無法終止執行個體。如需詳細資訊，請參閱 [終止保護](#)。

若您的執行個體處於 `shutting-down` 狀態的時間過長，應該會由 Amazon EC2 服務內的自動化程序清理 (終止)。如需詳細資訊，請參閱 [延遲的執行個體終止](#)。

## 啟用終止保護

若要避免意外終止執行個體，您可啟用執行個體的終止保護。`DisableApiTermination` 屬性控制是否可以使用 AWS Management Console、AWS Command Line Interface (AWS CLI) 或 API 終止執行個體。根據預設，您的執行個體會停用終止保護，這表示您的執行個體可以使用 AWS Management Console AWS CLI、或 API 終止。您可在執行個體啟動時、運作中或停止時，設定此屬性值 (適用於 Amazon EBS 支援的執行個體)。

若已設定 `InstanceInitiatedShutdownBehavior` 屬性，則 `DisableApiTermination` 屬性不會讓您無法從執行個體啟動關機 (使用作業系統命令進行系統關機) 來終止該執行個體。如需詳細資訊，請參閱 [變更執行個體啟動的關機行為](#)。

## 考量事項

- 啟用終止保護並不會防 AWS 止在有 [排程事件](#) 終止執行個體時終止執行個體。

- 啟用終止保護不會阻止 Amazon EC2 Auto Scaling 在執行個體狀態不良時或在縮減事件期間終止執行個體。您可以使用[執行個體停止保護功能](#)，控制 Auto Scaling 群組是否能在縮減時終止特定的執行個體。您可以[暫停 ReplaceUnhealthy 縮減程序](#)，以控制 Auto Scaling 群組是否能終止運作狀態不佳的執行個體。
- 您無法為 競價型執行個體啟用終止保護。

### 欲在執行個體啟動時啟用終止保護

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在儀表板中，選擇 Launch Instance (啟動執行個體)，然後依據精靈的說明執行。
3. 在 Configure Instance Details (設定執行個體詳細資訊) 頁面選取 Enable termination protection (啟用終止保護) 核取方塊。

### 欲在執行個體運作中或停止時啟用終止保護

1. 選取執行個體，然後選取 Actions (動作)、Instance Settings (執行個體設定)、Change Termination Protection (變更終止保護)。
2. 選擇 Yes, Enable (是，啟用)。

### 欲在執行個體運作中或停止時停用終止保護

1. 選取執行個體，然後選取 Actions (動作)、Instance Settings (執行個體設定)、Change Termination Protection (變更終止保護)。
2. 選擇 Yes, Disable (是，停用)。

### 欲使用命令列啟用或停用終止保護

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱[存取 Amazon EC2](#)。

- [modify-instance-attribute](#) (AWS CLI)
- [Edit-EC2InstanceAttribute](#) (AWS Tools for Windows PowerShell)

### 終止具有終止保護的多個實例

如果您在同一要求中終止跨多個可用區域的多個執行個體，並且啟用了終止保護的一或多個指定執行個體，則要求會失敗，並產生下列結果：

- 不會終止與受保護執行個體位於相同可用區域中的指定執行個體。
- 位於不同可用區域中的指定執行個體 (其中沒有其他指定執行個體受到保護) 會成功終止。

## 範例

假設您在兩個可用區域中有以下四個執行個體。

| 執行個體   | 可用區域  | 終止保護     |
|--------|-------|----------|
| 執行個體 1 | AZ 一個 | Disabled |
| 執行個體 2 |       | Disabled |
| 執行個體 3 | AZ B  | Enabled  |
| 執行個體 4 |       | Disabled |

如果您嘗試在同一個請求中終止所有這些執行個體，請求會報告失敗，結果如下：

- 執行個體 1 和執行個體 2 已成功終止，因為兩個執行個體均未啟用終止保護。
- 執行個體 3 和執行個體 4 無法終止，因為執行個體 3 已啟用終止保護。

## 變更執行個體啟動的關機行為

根據預設，從 Amazon EBS 後端執行個體起始關閉動作時 (使用 `shutdown` 或 `poweroff` 等命令)，該執行個體會停止。您可變更此行為，來讓執行個體終止，而無需變更執行個體的 `InstanceInitiatedShutdownBehavior` 屬性。您可在執行個體運作中或停止時，變更此屬性。

`halt` 命令不會啟動關機。如果使用，則執行個體不會終止；而是會將 CPU 放入 HLT，而且執行個體仍會繼續執行。

### Note

`InstanceInitiatedShutdownBehavior` 屬性僅適用從執行個體的作業系統執行關機，不適用使用 `StopInstances` API 或 Amazon EC2 主控台來停止執行個體。

您可使用 Amazon EC2 主控台或命令列來變更 `InstanceInitiatedShutdownBehavior` 屬性。

## Console

### 變更執行個體啟動的關機行為

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取實例。
4. 依序選擇 Actions (動作)、Instance settings (執行個體設定)、Change shutdown behavior (變更關機行為)。

關機行為會顯示目前的行為。

5. 若要變更此行為，對於關機行為，選擇停止或終止。
6. 選擇儲存。

## Command line

### 變更執行個體啟動的關機行為

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱[存取 Amazon EC2](#)。

- [modify-instance-attribute](#) (AWS CLI)
- [Edit-EC2InstanceAttribute](#) (AWS Tools for Windows PowerShell)

## 在執行個體終止時保留資料

根據您的使用案例，當 Amazon EC2 執行個體終止時，您可能會想要保留執行個體儲存體磁碟區或 Amazon EBS 磁碟區上的資料。在終止執行個體時，執行個體儲存體磁碟區上的資料就不會再持續存在。如果您需要將儲存在執行個體儲存體磁碟區上的資料保留超過執行個體的存留期，必須手動將該資料複製到更持久的儲存體，例如 Amazon EBS 磁碟區、Amazon S3 儲存貯體或 Amazon EFS 檔案系統。如需詳細資訊，請參閱 [Amazon EC2 執行個體的儲存選項](#)。

對於 Amazon EBS 磁碟區上的資料，Amazon EC2 會針對每一連接的 Amazon EBS 磁碟區，使用 `DeleteOnTermination` 屬性的值，以判定是否保留或刪除該磁碟區。

`DeleteOnTermination` 屬性的預設值會根據磁碟區是執行個體的根磁碟區，還是連接至執行個體的非根磁碟區而有所不同。

## 根磁碟區

根據預設，當您啟動執行個體時，執行個體根磁碟區的DeleteOnTermination屬性會設定為true。因此，執行個體終止時，預設會刪除執行個體的根磁碟區。

## 非根磁碟區

依預設，當您將非根 EBS 磁碟區連接至執行個體時，其DeleteOnTermination屬性會設定為false。因此，預設會保留這些磁碟區。

### Note

執行個體終止後，您可拍攝已保留磁碟區的快照，並將其連接到另一個執行個體。您必須刪除磁碟區以免日後產生費用。

AMI 建立者及執行個體的啟動者可以設定 DeleteOnTermination 屬性。當 AMI 建立者或執行個體的啟動者建立此屬性時，新設定會覆寫原始 AMI 預設設定。使用 AMI 啟動執行個體後，建議您驗證 DeleteOnTermination 屬性的預設設定。

若要確認 Amazon EBS 磁碟區是否會在執行個體終止時遭到刪除，請在執行個體的詳細資訊窗格中檢視磁碟區的詳細資訊。在 Storage (儲存) 索引標籤上的 Block devices (區塊型儲存設備) 之下，向右捲動以檢視磁碟區的 Delete on termination (在終止時刪除) 設定。

- 如果為是，磁碟區會在執行個體終止時遭到刪除。
- 如果為否，磁碟區不會在執行個體終止時遭到刪除。執行個體終止時未刪除的任何磁碟區將繼續產生費用。

## 將根磁碟區變更為在啟動時保留

啟動執行個體時，您可以使用主控台變更 DeleteOnTermination 屬性。若要變更運作中執行個體的這項屬性，您必須使用命令列。

使用下列其中一種方法，將根磁碟區變更為在啟動時予以保留。

### Console

欲使用主控台在啟動時將執行個體的根磁碟區變更為保留

1. 遵循程序[啟動執行個體](#)，但在完成下列步驟以將根磁碟區變更為保留之前，請勿啟動執行個體。

2. 在儲存 (磁碟區) 下方，展開根磁碟區下的資訊。
3. 針對在終止時刪除，選擇否
4. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。如需詳細資訊，請參閱 [使用新的啟動執行個體精靈啟動執行個體](#)。

## Command line

欲使用命令列在啟動時將執行個體的根磁碟區變更為保留

啟動 EBS 後端執行個體時，您可使用下列命令之一將根設備磁碟區變更為保留。如需有關這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [run-instances](#) (AWS CLI)
- [New-EC2Instance](#) (AWS Tools for Windows PowerShell)

在您要保留的磁碟區的區塊型裝置映射中，包括 `--DeleteOnTermination`，並指定 `false`。

例如，若要保留磁碟區，將下列選項新增至 `run-instances` 命令：

```
--block-device-mappings file://mapping.json
```

在 `mapping.json` 中，指定裝置名稱，例如 `/dev/sda1` 或者 `/dev/xvda`，針對 `--DeleteOnTermination`，指定 `false`。

```
[
 {
 "DeviceName": "device_name",
 "Ebs": {
 "DeleteOnTermination": false
 }
 }
]
```

將執行中執行個體的根磁碟區變更為持續存在

您可使用下列命令之一，將運作中 EBS 後端執行個體的根設備磁碟區變更為保留。如需有關這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。



- [modify-instance-attribute](#) (AWS CLI)
- [Edit-EC2InstanceAttribute](#) (AWS Tools for Windows PowerShell)

例如，使用下列命令：

```
aws ec2 modify-instance-attribute --instance-id i-1234567890abcdef0 --block-device-mappings file://mapping.json
```

在 mapping.json 中，指定裝置名稱，例如 /dev/sda1 或者 /dev/xvda，針對 --DeleteOnTermination，指定 false。

```
[
 {
 "DeviceName": "device_name",
 "Ebs": {
 "DeleteOnTermination": false
 }
 }
]
```

## 執行個體淘汰

當 AWS 偵測到託管執行個體的基礎硬體無法修復的故障時，執行個體就會排定為淘汰。執行個體根裝置會決定執行個體淘汰的行為：

- 如果您的執行個體根設備是 Amazon EBS 磁碟區，執行個體會停止，且您可以隨時再次將其啟動。啟動停止的執行個體會將其遷移至新硬體。
- 如果您的執行個體根裝置是執行個體存放區磁碟區，則執行個體會終止，且無法再次使用。

如需執行個體事件類型的詳細資訊，請參閱[您的執行個體的排程事件](#)。

### 內容

- [識別排程淘汰的執行個體](#)
- [針對已排程淘汰的 EBS 後端執行個體採取的動作](#)
- [針對排程淘汰的執行個體存放區支援執行個體採取的動作](#)

## 識別排程淘汰的執行個體

如果已排程淘汰您的執行個體，則您會在事件執行之前收到一封內含執行個體 ID 及淘汰日期的電子郵件。您也可以使用 Amazon EC2 主控台或命令列來檢查排程為淘汰的執行個體。

### Important

如果執行個體已排程停用，建議您儘快採取動作，因為執行個體可能無法連線。(您收到的電子郵件通知說明如下：「由於此降級，您的執行個體可能已經無法連線。」) 如需您應採取之建議動作的詳細資訊，請參閱[Check if your instance is reachable](#)。

### 識別排程淘汰的執行個體的方法

- [電子郵件通知](#)
- [主控台識別](#)

#### 電子郵件通知

如果已排程淘汰您的執行個體，則您會在事件執行之前收到一封內含執行個體 ID 及淘汰日期的電子郵件。

電子郵件會傳送給主要帳戶持有人和營運聯絡人。如需詳細資訊，請參閱 AWS Billing 使用者指南中的[新增、變更或移除替代聯絡人](#)。

#### 主控台識別

如果您使用未定期查看執行個體淘汰通知的電子郵件帳戶，則可以使用 Amazon EC2 主控台或命令列來判斷是否已排程淘汰任何執行個體。

#### 使用主控台來識別排定淘汰的執行個體

1. 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 EC2 Dashboard (EC2 儀表板)。在排程事件下，您可以看到與 Amazon EC2 執行個體和磁碟區相關聯的事件，依區域組織。

## Scheduled events

### US East (N. Virginia)

- 7 instance(s) have scheduled events
- 1 volume(s) are impaired

3. 如果您有一個已列出排程事件的執行個體，請在區域名稱下方選取其連結以前往 Events (事件) 頁面。
4. 事件頁面列出所有資源，以及與其相關聯的事件。若要檢視已排定淘汰的執行個體，請從第一個篩選條件清單中選取 Instance resources (執行個體資源)，然後從第二個篩選條件清單中選取 Instance stop or retirement (停止或淘汰執行個體)。
5. 如果篩選條件結果顯示執行個體已排定淘汰，請將其選取，然後記下詳細資訊窗格中 Start time (開始時間) 欄位中的日期和時間。這是您的執行個體淘汰日期。

### 使用命令列來識別排定淘汰的執行個體

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱[存取 Amazon EC2](#)。

- [describe-instance-status](#) (AWS CLI)
- [Get-EC2InstanceStatus](#) (AWS Tools for Windows PowerShell)

### 針對已排程淘汰的 EBS 後端執行個體採取的動作

若要保留已淘汰執行個體上的資料，您可以執行下列其中一個動作。在執行個體淘汰日期之前採取此動作非常重要，以防止不可預期的停機時間和資料遺失。

對於 Linux 執行個體，如果您不確定執行個體是由 EBS 或執行個體存放區支援，請參閱[判斷 Linux 執行個體的根裝置類型](#)。

### 檢查您的執行個體是否可存取

當您收到執行個體已排程停用的通知時，我們建議您儘快採取下列動作：

- 透過[連線](#)或 ping 您的執行個體來檢查您的執行個體是否可以存取。
- 如果可取得您的執行個體，您應該計劃在排程的停止日期之前的適當時間停止/啟動執行個體，這影響最小。如需停止和啟動執行個體的詳細資訊，以及執行個體停止時會發生何事，例如與該執行個體相關聯的公有、私有和彈性 IP 地址的影響，請參閱[停止並啟動 Amazon EC2 執行個體](#)。請注意，當您停止並啟動執行個體時，執行個體存放區磁碟區上的資料會遺失。
- 如果您的執行個體無法連線，您應立即採取動作並執行[停止/啟動](#)來恢復您的執行個體。
- 或者，如果您想要[終止](#)執行個體，請計劃盡快執行個體，以便停止對執行個體產生費用。

## 為您的執行個體建立備份

從您的執行個體建立 EBS 後端 AMI，以便有備份。若要確保資料完整性，請在建立 AMI 之前停止執行個體。您可等候預定的淘汰日期 – 當執行個體停止時 – 或在淘汰日期之前自行停止執行個體。您可以隨時再次啟動該執行個體。如需詳細資訊，請參閱 [創建一個 Amazon EBS-backed AMI](#)。

## 啟動替代執行個體

從執行個體建立 AMI 之後，您就可以使用 AMI 來啟動替代用的執行個體。從 Amazon EC2 主控台中，選取新 AMI，然後選取動作、啟動。遵循精靈以啟動您的執行個體。如需有關精靈中每個步驟的詳細資訊，請參閱[使用新的啟動執行個體精靈啟動執行個體](#)。

## 針對排程淘汰的執行個體存放區支援執行個體採取的動作

若要保留已淘汰執行個體上的資料，您可以執行下列其中一個動作。在執行個體淘汰日期之前採取此動作非常重要，以防止不可預期的停機時間和資料遺失。

### Warning

如果您的執行個體後端執行個體超其淘汰日期，則將會終止，且您無法復原該執行個體或儲存在其中的任何資料。無論執行個體的根設備是哪種，執行個體存放區磁碟區上的資料在執行個體淘汰時都會遺失，即使磁碟區已連接至 EBS 後端執行個體。

## 檢查您的執行個體是否可存取

當您收到執行個體已排程停用的通知時，我們建議您儘快採取下列動作：

- 透過[連線](#)或 ping 您的執行個體來檢查您的執行個體是否可以存取。

- 如果您的執行個體無法連線，則復原執行個體的可能性很少。如需詳細資訊，請參閱[無法連線故障診斷執行個體](#)。AWS 將在排定的淘汰日期終止您的執行個體，因此，對於無法連線的執行個體，您可以立即自行[終止](#)執行個體。

## 啟動替代執行個體

使用 AMI 工具從您的執行個體建立執行個體後端 AMI，如[建立執行個體存放區後端 Linux AMI](#)所述。從 Amazon EC2 主控台中，選取新 AMI，然後選取動作、啟動。遵循精靈以啟動您的執行個體。如需有關精靈中每個步驟的詳細資訊，請參閱[使用新的啟動執行個體精靈啟動執行個體](#)。

## 將您的執行個體轉換為 EBS 支援的執行個體

將您的資料傳輸到 EBS 磁碟區、建立磁碟區快照，然後從該快照建立 AMI。您可以從新的 AMI 啟動替代用的執行個體。如需詳細資訊，請參閱[將執行個體後端 AMI 轉換為 Amazon EBS 後端的 AMI](#)。

## 執行個體恢復

### Important

下列資訊適用於在狀態良好的執行個體上設定復原相關功能。如果您目前在存取執行個體時遇到困難，請參閱[疑難排解 EC2 執行個體](#)。

如果 AWS 判斷執行個體因為基礎硬體問題而無法使用，您可以為執行個體彈性設定兩種機制，以恢復可用性：簡化的自動復原和以 Amazon CloudWatch 動作為基礎的復原。這個程序稱為執行個體復原。

至少必須事先設定或啟用一個機制，並使用支援的資源，才能執行執行個體復原程序。根據預設，啟動支援的執行個體時，會啟用簡化的自動復原功能。

### 主題

- [執行個體復原概觀](#)
- [執行個體復原選](#)
- [設定 CloudWatch 動作型復原](#)
- [設定簡化的自動復原](#)

## 執行個體復原概觀

以下是可能需要執行個體復原的基礎硬體問題範例：

- 網路連線中斷
- 系統電力中斷
- 實體主機的軟體問題
- 實體主機上會影響網路連線的硬體問題

復原的執行個體與原始執行個體完全相同，包括：

- 執行個體 ID
- 公用、私有和彈性 IP 位址
- 執行個體中繼資料
- 置放群組
- 附加的 EBS 磁碟區
- 可用區域

執行個體復原成功將會顯示為非計劃的重新啟動。換句話說，存儲在易失性存儲器中的內容將丟失，實際存儲數據將被刪除，並且操作系統的正常運行時間將從零開始。

為了協助防止資料遺失，我們建議您定期建立有價值資料的備份。如需 Amazon EC2 執行個體備份和復原最佳實務的詳細資訊，請參閱 [Amazon EC2 的最佳實務](#)。

## 執行個體復原選

下列執行個體復原的替代方案符合執行個體的使用案例時，可以考慮這些替代方法。

### Auto Scaling 群組

您可以使用 Auto Scaling 群組，將執行個體集合分組在一起，以達到擴展和可用性的目的。如果 Auto Scaling 群組中的某個執行個體無法使用，Auto Scaling 群組將自動取代 (不會復原) 該執行個體。如需詳細資訊，請參閱 [什麼是 Amazon EC2 Auto Scaling ?](#) 在 Amazon EC2 Auto Scaling 用戶指南中。

### Amazon EBS Multi-Attach

您可以為執行個體設定 Amazon EBS 多重連接，以允許多個執行個體連接到相同的 EBS 磁碟區。與適當的軟體結合使用時，可啟用高可用性叢集。如需 Linux 執行個體的設定範例，請參閱 [儲存部落格上已啟用 Amazon EBS 多重連接的磁碟區上的簡化叢集儲存 : GFS2](#)。AWS

## 設定 CloudWatch 動作型復原

### Important

- 下列資訊適用於在狀態良好的執行個體上設定復原相關功能。如果您目前在存取執行個體時遇到困難，請參閱[疑難排解 EC2 執行個體](#)。
- 為了讓您的工作負載在執行個體復原成功後正常運作，您的執行個體必須啟動並接受流量，而不需要手動介入。

您可以設定 Amazon CloudWatch 動作型復原，以將復原動作新增至 Amazon CloudWatch 警示。CloudWatch 以動作為基礎的復原可搭配 `StatusCheckFailed_System` 量度使用。CloudWatch 以動作為基礎的 to-the-minute 復原可提供復原回應時間粒度，並提供復原動作和結果的 Amazon 簡單通知服務 (Amazon SNS) 通知。與簡化的自動復原相較之下，這些組態選項可以更精細地控制系統狀態檢查失敗事件回應，進而加快復原嘗試的速度。如需有關可用 CloudWatch 選項的詳細資訊，請參閱[執行個體的狀態檢查](#)。

Amazon CloudWatch 動作型復原不會在 AWS Health Dashboard。如需詳細資訊，請參閱 [the section called “疑難排解 CloudWatch 動作型復原失敗”](#)。

### 主題

- [以 CloudWatch 動作為基礎的復原需求與限制](#)
- [設定 CloudWatch 動作型復原](#)
- [疑難排解 CloudWatch 動作型復原失敗](#)

以 CloudWatch 動作為基礎的復原需求與限制

CloudWatch 以動作為基礎的復原可在下列情況下嘗試復原執行個體：

- 處於該 `running` 狀態。如需詳細資訊，請參閱 [the section called “執行個體生命週期”](#)。
- 使用 `default` (隨選) 或 `dedicated` 執行個體租用。如需詳細資訊，請參閱 [the section called “執行個體購買選項”](#)。
- 屬於 Amazon EC2 具有可用容量的執行個體類型。在某些情況下，例如嚴重中斷，可用的容量不足，並且某些復原嘗試可能會失敗。
- 不使用 `dedicated` 執行個體租用。對於 Amazon EC2 專用執行個體，您可以使用 [專用執行個體自動復原](#) 來自動復原運作狀態不良的執行個體。

- 不使用彈性織物適配器。
- 不是「Auto Scaling」群組的成員。
- 目前尚未進行排定的維護事件。
- 使用下列其中一種執行個體類型：
  - 一般用途：A1 | M3 | M4 | M5 | M5a | M5n | M5zn | M6a | M6g | M6i | M6in | M7a | M7g | M7i | M7i-flex | T1 | T2 | T3 | T3a | T4g
  - 計算優化:C3 | C4 | C5 | C5 安培 | C5 安 | C6 g | C6 g | 氯化碳 | C6 英寸 | C7 安培 | C7 克 | 氯化鎂 | 氯化碳
  - 內存優化:R3 | R4 | R5 | R5 | R5B | R5n | 六甲 | 六克 | 六分之一 | 六英寸 | 六英寸 | R7 克 | R7 克 | 六 TB | 烏 7 英寸-二四 TB | 三十二 TB | X1
  - 加速運算：G3 | G3s | G5g | Inf1 | P2 | P3 | VT1
  - 高效能運算：HPC6a | HPC7a
  - 金屬例證：上述任何具有金屬例證大小的類型。
- 具有執行個體儲存磁碟區，並使用下列其中一種執行個體類型：M3 | C3 | X1 | X1

#### Warning

- 如果執行個體停止，執行個體儲存磁碟區上的資料將會遺失。如需停止執行個體的詳細資訊，請參閱[the section called “執行個體停止和啟動”](#)。
- 如果系統狀態檢查失敗，執行個體儲存區和區塊裝置對應的資料可能會遺失。對於這些執行個體類型，您可以考慮使用[the section called “啟用終止保護”](#)。

我們建議您定期建立有價值資料的備份。如需 Amazon EC2 備份和復原最佳實務的相關資訊，請參閱 [Amazon EC2 的最佳實務](#)。

您也可以使用 AWS Management Console 或 AWS CLI 來檢視支援 CloudWatch 動作型復原的執行個體類型。

#### Console

若要檢視支援 Amazon CloudWatch 動作型復原的執行個體類型

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。



2. 在左側導覽窗格中選擇 Instance Types (執行個體類型)。
3. 在篩選條件列中，輸入 Auto Recovery support: true (支援自動復原程序：是)。或者，當您輸入字元且篩選條件名稱顯示時，您可以選取它。

「執行個體類型」表格會顯示支援 Amazon CloudWatch 動作型復原的所有執行個體類型。

## AWS CLI

若要檢視支援 Amazon CloudWatch 動作型復原的執行個體類型

使用 [describe-instance-types](#) 命令。

```
aws ec2 describe-instance-types --filters Name=auto-recovery-supported,Values=true
--query "InstanceTypes[*].[InstanceType]" --output text | sort
```

## 設定 CloudWatch 動作型復原

CloudWatch 以動作為基礎的復原可搭配 `StatusCheckFailed_System` 量度使用。CloudWatch 透過 CloudWatch 主控台設定以動作為基礎的復原。若要設定以 CloudWatch 動作為基礎的復原，請參閱 [Amazon CloudWatch 使用者指南中的將復原動作新增至 CloudWatch 警示](#)。

## 疑難排解 CloudWatch 動作型復原失敗

下列問題可能會造成透過動作型復原功能的執行 CloudWatch 行個體復原失敗：

- CloudWatch 在中的服務事件期間，以動作為基礎的復原不會運作 AWS Health Dashboard。您可能不會收到此類事件的復原故障通知。如需最新的服務使用狀態資訊，請參閱 [服務健全狀況狀態頁面](#)。
- 替換硬體的容量暫時不足。
- 執行個體已達到復原嘗試的每日限額上限。如果自動復原失敗，而原始系統狀態檢查未通過的原因判定為硬體劣化，則執行個體可能隨後會遭到淘汰。

如果即使嘗試多次復原，執行個體的系統狀態檢查失敗仍然存在，請參閱 [使用失敗狀態檢查進行執行個體疑難排解](#)，

## 設定簡化的自動復原

### ⚠ Important

- 下列資訊適用於在狀態良好的執行個體上設定復原相關功能。如果您目前在存取執行個體時遇到困難，請參閱[疑難排解 EC2 執行個體](#)。
- 為了讓您的工作負載在執行個體復原成功後正常運作，您的執行個體必須啟動並接受流量，而不需要手動介入。

依預設，簡化的自動復原會監控所有支援的執行個體。在偵測到系統狀態檢查失敗的情況下，簡化了自動復原嘗試將執行個體修復為健康狀態。簡化的自動復原在中的服務事件期間無法運作 AWS Health Dashboard。如需詳細資訊，請參閱 [the section called “簡化的自動復原失敗疑難”](#)。

當簡化的自動恢復事件發生時，您將收到一個 AWS Health Dashboard 事件。若要設定這些事件的通知，請參閱《[使AWS 使用者通知 用指南](#)》AWS 使用者通知中的《[入門使用](#)》。您也可以使用 Amazon EventBridge 規則，使用下列事件代碼監控簡化的自動復原事件：

- AWS\_EC2\_SIMPLIFIED\_AUTO\_RECOVERY\_SUCCESS - 成功的事件
- AWS\_EC2\_SIMPLIFIED\_AUTO\_RECOVERY\_FAILURE - 失敗的事件

如需詳細資訊，請參閱 [Amazon EventBridge 規則](#)。

### 主題

- [簡化自動復原的需求與限制](#)
- [設定簡化的自動復原](#)
- [簡化的自動復原失敗疑難](#)

### 簡化自動復原的需求與限制

在下列情況下，簡化的自動復原將嘗試復原執行個體：

- 處於該running狀態。如需詳細資訊，請參閱 [the section called “執行個體生命週期”](#)。
- 使用 default (隨選) 或dedicated執行個體租用。如需詳細資訊，請參閱 [the section called “執行個體購買選項”](#)。

- 屬於 Amazon EC2 具有可用容量的執行個體類型。在某些情況下，例如嚴重中斷，可用的容量不足，並且某些復原嘗試可能會失敗。
- 不使用dedicated執行個體租用。對於 Amazon EC2 專用執行個體，您可以使用[專用執行個體自動復原](#)來自動復原運作狀態不良的執行個體。
- 不使用彈性織物適配器。
- 不是metal執行個體大小。
- 不是「Auto Scaling」群組的成員。
- 目前尚未進行排定的維護事件。
- 沒有實例存儲卷。
- 使用下列其中一種執行個體類型：
  - 一般用途：A1 | M3 | M4 | M5 | M5a | M5n | M5zn | M6a | M6g | M6i | M6in | M7a | M7g | M7i | M7i-flex | T1 | T2 | T3 | T3a | T4g
  - 計算優化:C3 | C4 | C5 | C5 安培 | C5 安 | C6 g | C6 g | 氯化碳 | C6 英寸 | C7 安培 | C7 克 | 氯化鎂 | 氯化碳
  - 內存優化:R3 | R4 | R5 | R5 | R5B | R5n | 六甲 | 六克 | 六分之一 | 六英寸 | 六英寸 | R7 克 | R7 克 | 六 TB | 烏 7 英寸-二四 TB | 三十二 TB | X1
  - 加速運算：G3 | G3s | G5g | Inf1 | P2 | P3 | VT1
  - 高效能運算：HPC6a | HPC7a

#### Warning

- 如果執行個體停止，執行個體儲存磁碟區上的資料將會遺失。如需停止執行個體的詳細資訊，請參閱[the section called “執行個體停止和啟動”](#)。
- 如果系統狀態檢查失敗，執行個體儲存區和區塊裝置對應的資料可能會遺失。對於這些執行個體類型，您可以考慮使用[the section called “啟用終止保護”](#)。

我們建議您定期建立有價值資料的備份。如需 Amazon EC2 備份和復原最佳實務的相關資訊，請參閱 [Amazon EC2 的最佳實務](#)。

## 設定簡化的自動復原

當您啟動支援的執行個體時，預設會啟用簡化的自動復原功能。您可以將自動復原行為設定為啟動執行個體disabled期間或之後。此default設定不會針對不支援的執行個體類型啟用簡化的自動復原功能。

### Console

#### 啟動執行個體時停用簡化的自動復原程序

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)，然後選擇 Launch Instance (啟動執行個體)。
3. 在 Advanced details (進階詳細資訊) 區段，針對 Instance auto-recovery (執行個體自動復原)，選取 Disabled (已停用)。
4. 根據需要進行剩餘的執行個體啟動設定，然後啟動執行個體。

#### 針對執作中或已停止的執行個體停用簡化的自動復原程序

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體，然後選取 Actions (動作)、Instance Settings (執行個體設定)、Change Shutdown Behavior (變更關機行為)。
4. 選擇 Off (關閉)，然後選擇 Save (儲存)。

#### 針對執作中或已停用的執行個體將自動復原行為設定為 **default**

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體，然後選取 Actions (動作)、Instance Settings (執行個體設定)、Change Shutdown Behavior (變更關機行為)。
4. 選擇預設，然後選擇儲存。

### AWS CLI

#### 啟動時停用簡化的自動復原程序

使用 [run-instances](#) 命令。

```
aws ec2 run-instances \
--image-id ami-1a2b3c4d \
--instance-type t2.micro \
--key-name MyKeyPair \
--maintenance-options AutoRecovery=Disabled \
[...]
```

針對執行中或已停止的執行個體停用簡化的自動復原程序

使用 [modify-instance-maintenance-options](#) 命令。

```
aws ec2 modify-instance-maintenance-options \
--instance-id i-0abcdef1234567890 \
--auto-recovery disabled
```

針對執行中或已停用的執行個體將自動復原行為設定為 **default**

使用 [modify-instance-maintenance-options](#) 命令。

```
aws ec2 modify-instance-maintenance-options \
--instance-id i-0abcdef1234567890 \
--auto-recovery default
```

## 簡化的自動復原失敗疑難

下列問題可能會導致透過簡化的自動復原執行個體復原失敗：

- 在中的服務事件期間，簡化的自動復原無法運作 AWS Health Dashboard。您可能不會收到此類事件的復原故障通知。如需最新的服務使用狀態資訊，請參閱[服務健全狀況狀態頁面](#)。
- 替換硬體的容量暫時不足。
- 執行個體已達到復原嘗試的每日限額上限。如果自動復原失敗，而原始系統狀態檢查未通過的原因判定為硬體劣化，則執行個體可能隨後會遭到淘汰。

如果即使嘗試多次復原，執行個體的系統狀態檢查失敗仍然存在，請參閱[使用失敗狀態檢查進行執行個體疑難排解](#)，

# 使用執行個體中繼資料

執行個體中繼資料 是關於您執行個體的資料，您可以用來設定或管理執行中的執行個體。執行個體中繼資料會分為幾個類別，例如主機名稱、事件及安全群組。

您也可以使用執行個體中繼資料，存取您在啟動執行個體時指定的使用者資料。例如，您可以指定設定您執行個體的參數，或是包含簡單的指令碼。您也可以建置一般 AMI，並使用使用者資料來修改啟動時提供的組態檔案。例如，如果您為各種小型企業執行 Web 伺服器，它們都可以使用相同的通用 AMI，並從您在啟動時在使用者資料中指定的 Amazon S3 儲存貯體擷取其內容。若要隨時新增客戶，為客戶建立儲存貯體、新增其內容，以及利用使用者資料中提供給程式碼的唯一儲存貯體名稱來啟動 AMI。如果您使用相同 RunInstances 呼叫啟動多個執行個體，則該保留項目中的所有執行個體都可以使用該使用者資料。屬於相同保留項目的每個執行個體都有唯一的編 ami-launch-index 號，因此您可以撰寫程式碼來控制執行個體的功能。例如，第一個主機可能會選取本身做為叢集中的原始節點。如需 AMI 啟動範例的詳細資訊，請參閱 [Linux 實例：AMI 啟動指數值](#)。

EC2 執行個體同時也會包含 動態資料，例如執行個體啟動時產生的執行個體身分文件。如需詳細資訊，請參閱 [動態資料分類](#)。

## Important

雖然您只能在執行個體內部存取執行個體中繼資料和使用者資料，資料並未受到驗證或密碼編譯法保護。可直接存取執行個體的任何人，以及可能在該執行個體上執行的任何軟體，都能檢視其中繼資料。因此，您不應該將敏感性資料 (例如密碼或長期加密金鑰) 儲存為使用者資料。

## 目錄

- [使用 IMDSv2](#)
- [設定執行個體中繼資料選項](#)
- [擷取執行個體中繼資料](#)
- [使用執行個體使用者資料](#)
- [擷取動態資料](#)
- [執行個體中繼資料分類](#)
- [Linux 實例：AMI 啟動指數值](#)
- [執行個體身分文件](#)
- [執行個體身分角色](#)

## 使用 IMDSv2

您可以使用以下其中一個方法，從執行中的執行個體存取執行個體中繼資料：

- 執行個體中繼資料服務第 1 版 (IMDSv1) – 請求/回應方法
- 執行個體中繼資料服務第 2 版 (IMDSv2) – 工作階段導向方法

依預設，您可以使用 IMDSv1 或 IMDSv2，或兩者。

您可以在每個執行個體上設定執行個體中繼資料服務 (IMDS)，此類本機程式碼或使用者必須使用 IMDSv2。當您指定必須使用該 IMDSv2 時，IMDSv1 則無法繼續運作。如需如何設定執行個體以使用 IMDSv2 的資訊，請參閱 [設定執行個體中繼資料選項](#)。

PUT 或 GET 標頭對於 IMDSv2 來說是唯一的。如果這些標頭存在於請求中，則該請求適用於 IMDSv2。如果不存在相應標頭，則假設該請求適用於 IMDSv1。

如需深入了解 IMDSv2，請參閱 [透過 EC2 執行個體中繼資料服務的增強功能，提高開放式防火牆、反向代理伺服器及 SSRF \(伺服器端請求偽造\) 弱點的防禦能力](#)。

若要擷取執行個體中繼資料，請參閱 [擷取執行個體中繼資料](#)。

### 主題

- [執行個體中繼資料服務第 2 版的運作方式](#)
- [轉換為使用執行個體中繼資料服務第 2 版](#)
- [使用支援的 AWS 開發套件](#)

## 執行個體中繼資料服務第 2 版的運作方式

IMDSv2 會使用工作階段導向請求。使用工作階段導向請求，您就能建立定義工作階段持續期間的工作階段字符，此期間最短 1 秒，最長可達 6 小時。在指定持續期間，您可以將相同的工作階段字符用於後續請求。在指定持續期間到期之後，您必須建立新的工作階段字符，才能使用未來請求。

### Note

本節中的範例使用執行個體中繼資料服務 (IMDS) 的 IPv4 地址：169.254.169.254。如果您要透過 IPv6 地址擷取 EC2 執行個體的中繼資料，請確定您啟用並改用 IPv6 地址：[fd00:ec2::254]。IMDS 的 IPv6 地址與 IMDSv2 命令相容。IPv6 地址只能在 [AWS Nitro 系統上建置的執行個體](#)和[支援 IPv6 的子網路](#)中存取 (僅限雙堆疊或 IPv6)。

下列範例使用殼層指令碼和 ImDSv2 來擷取頂層執行個體中繼資料項目。每個例子：

- 使用 PUT 請求，建立持續 6 小時 (21,600 秒) 的工作階段字符
- 將工作階段權杖標頭儲存在名為 TOKEN (Linux 執行個體) 或 token (Windows 執行個體) 的變數中
- 使用字符請求上層中繼資料項目

## Linux 範例

您可以執行兩個單獨的命令，或將它們合併。

### 單獨命令

首先，使用以下命令產生字符。

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"``
```

然後，使用字符產生使用下列命令的頂層中繼資料項目。

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/
```

### 組合命令

您可以存放字符並組合命令。下面的範例合併了上述兩個命令，並將工作階段字符標頭存放在一個名為 TOKEN 的變數中。

#### Note

如果在建立字符時發生錯誤，變數中會存放錯誤消息而非有效字符，命令也不會發揮作用。

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"`` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/
```

建立字符之後，您可以重複使用直到到期為止。在下列範例命令中，其中會取得用於啟動執行個體的 AMI ID，會重複使用前一個範例中 \$TOKEN 中儲存的字符。



```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/ami-id
```

## Windows 範例

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/meta-data/
```

建立字符之後，您可以重複使用直到到期為止。在下列範例命令中，其中會取得用於啟動執行個體的 AMI ID，會重複使用前一個範例中 `$token` 中儲存的字符。

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -uri http://169.254.169.254/latest/meta-data/ami-id
```

當您使用 IMDSv2 請求執行個體中繼資料時，請求必須包含下列項目：

1. 使用 PUT 請求，在執行個體中繼資料服務中起始工作階段。PUT 請求會傳回一個字符，其必須包含在執行個體中繼資料服務的後續 GET 請求中。字符必須使用 IMDSv2 存取中繼資料。
2. 將字符包含在 IMDS 的所有 GET 請求中。字符使用方式設定為 `required` 時，不包含有效字符或字符已到期的請求會收到 401 - Unauthorized HTTP 錯誤碼。
  - 字符是執行個體特定金鑰。字符在其他 EC2 執行個體上無效，而且如果您嘗試在產生字符的執行個體外部使用該字符，則會遭到拒絕。
  - PUT 請求必須包含指定字符存留時間 (TTL)，時間會以秒數表示且最長可達 6 小時 (21,600 秒)。字符會代表邏輯工作階段。TTL 會指定字符有效的時間長度，也就是工作階段的持續期間。
  - 在字符到期之後，若要繼續存取執行個體中繼資料，您必須使用另一個 PUT 建立新的工作階段。
  - 您可以選擇重複使用字符或使用每個請求來建立新字符。對於少量請求，您每次需要存取 IMDS 時，就能更輕鬆地產生和立即使用字符。但為了提升效率，您可以為該字符指定時間更長的持續期間，然後再重複使用該字符，而不需要在每次要請求執行個體中繼資料時寫入 PUT。並行字符數量沒有實際限額，每個都代表它自己的工作階段。但 IMDSv2 仍會受到正常 IMDS 連線和調節限額的限制。如需詳細資訊，請參閱 [查詢調節](#)。

HTTP GET 及 HEAD 方法可在 IMDSv2 執行個體中繼資料請求中使用。如果 PUT 請求包含 `X-Forwarded-For` 標頭，則會遭到拒絕。

依預設，PUT 請求在 IP 通訊協定等級的 1 回應跳轉限額 (存留時間)。如果您需要更大的躍點限制，可以使用 [修改實例](#)- AWS CLI 元數據選項命令對其進行調整。例如，您可以需要更大的跳轉限制，才能與執行個體上執行之容器服務回溯相容。如需詳細資訊，請參閱 [修改現有執行個體的執行個體中繼資料選項](#)。

## 轉換為使用 執行個體中繼資料服務第 2 版

移轉至 IMDSv2 時，我們建議您使用下列工具和轉換路徑。

### 主題

- [協助轉換至 IMDSv2 的工具](#)
- [需要 IMDSv2 的建議路徑](#)

### 協助轉換至 IMDSv2 的工具

如果您的軟體使用 IMDSv1，請使用下列工作，以協助將您的軟體重新設定為使用 IMDSv2。

### AWS 軟體

最新版本的 AWS CLI 和 AWS 開發套件支援 IMDSv2。若要使用 IMDSv2，請確定 EC2 執行個體有最新版本的 CLI 及開發套件。如需有關更新 CLI 的資訊，請參閱 AWS Command Line Interface 使用者指南中的 [安裝、更新和解除安裝 AWS CLI](#)。

所有 Amazon Linux 2 和 Amazon 2023 軟體套件都支援 IMDSv2。在 Amazon 2023 中，默認情況下停用 IMDSv1。

如需支援 IMDSv2 的最低 AWS SDK 版本，請參閱 [使用支援的 AWS 開發套件](#)

### IMDS Packet Analyzer

IMDS Packet Analyzer 是開放原始碼工具，可識別並記錄執行個體開機階段的 IMDSv1 呼叫。這有助於識別在 EC2 執行個體上發出 IMDSv1 呼叫的軟體，讓您準確找出需要更新的項目，讓您的執行個體準備好只使用 IMDSv2。您可以從命令列執行 IMDS Packet Analyzer，或將其安裝為服務。如需詳細資訊，請參閱上的 [IMDS 封包分析器GitHub](#)。

### CloudWatch

IMDSv2 會使用權杖後端工作階段，而 IMDSv1 不會。此 MetadataNoToken CloudWatch 量度會追蹤使用 IMDSv1 之「執行處理中繼資料服務」(IMDS) 的呼叫次數。透過追蹤此指標至零，您可以決定是否要升級所有軟體以使用 IMDSv2 及升級時間。

停用 IMDSv1 之後，您可以使用 `MetadataNoTokenRejected` CloudWatch 量度追蹤 IMDSv1 呼叫嘗試和拒絕的次數。透過追蹤此指標，您可以確定是否需要更新軟體以使用 IMDSv2。

如需詳細資訊，請參閱 [執行個體指標](#)。

## EC2 API 和 CLI 的更新

對於新執行個體，您可以使用 `RunInstances` API 啟動需要使用 IMDSv2 的新執行個體。如需詳細資訊，請參閱 [設定新執行個體的執行個體中繼資料選項](#)。

對於現有的執行個體，您可以使用 `ModifyInstanceMetadataOptions` API 來要求使用 IMDSv2。如需詳細資訊，請參閱 [修改現有執行個體的執行個體中繼資料選項](#)。

如要求 Auto Scaling 群組啟動的所有新執行個體都能在 IMDSv2 上使用，您的 Auto Scaling 群組可以使用啟動範本或啟動組態。當您 [建立啟動範本](#) 或 [建立啟動組態](#) 時，必須設定 `MetadataOptions` 參數以要求使用 IMDSv2。Auto Scaling 群組會使用新啟動範本或啟動組態來啟動新執行個體，但現有的執行個體不受影響。對於 Auto Scaling 群組中的現有執行個體，您可以使用 `ModifyInstanceMetadataOptions` API 來要求在現有執行個體上使用 IMDSv2，或終止執行個體，Auto Scaling 群組會使用新的啟動範本或啟動設定中定義的執行個體中繼資料選項設定來啟動新的取代執行個體。

## 使用預設情況下設定 IMDSv2 的 AMI

當您啟動執行個體時，您可以自動將其設定為預設使用 IMDSv2 (`HttpTokens` 參數設為 `required`)，方法是透過 AMI 來加以啟動，該 AMI 的 `ImdsSupport` 參數設定為 `v2.0`。當您使用 `register-image` CLI 命令註冊 AMI 時，可將 `ImdsSupport` 參數設為 `v2.0`，或者可使用 `modify-image-attribute` CLI 命令修改現有 AMI。如需詳細資訊，請參閱 [設定 AMI](#)。

## IAM 政策和 SCP

您可以使用 IAM 政策或 AWS Organizations 服務控制政策 (SCP) 來控制使用者，如下所示：

- 除非執行個體設定為使用 `ImDSv2`，否則無法使用 `RunInstances` API 啟動執行個體。
- 無法使用 `ModifyInstanceMetadataOptions` API 修改執行中的執行個體，以重新啟用 IMDSv1。

IAM 政策或 SCP 必須包含下列 IAM 條件索引鍵：

- `ec2:MetadataHttpEndpoint`
- `ec2:MetadataHttpPutResponseHopLimit`
- `ec2:MetadataHttpTokens`

如果 API 或 CLI 呼叫中的參數與包含條件索引鍵之政策中指定的狀態不相符，則 API 或 CLI 呼叫會失敗並顯示 `UnauthorizedOperation` 回應。

此外，您可以選擇另外一個保護層，以強制執行從 IMDSv1 變更為 IMDSv2 的作業。在存取管理層相對於透過 EC2 角色登入資料呼叫的 API，您可以在 IAM 政策或 AWS Organizations 服務控制政策 (SCP) 中使用新的條件金鑰。具體而言，在 IAM 政策中使用數值為 2.0 的條件索引鍵 `ec2:RoleDelivery`，從 IMDSv1 取得之 EC2 角色憑證產生的 API 呼叫都會收到 `UnauthorizedOperation` 回應。也能利用 SCP 要求的條件進行更廣泛範圍的作業。這可確保透過 IMDSv1 提供的登入資料無法確實用於呼叫 API，因為與指定條件不相符的任何 API 呼叫，將會收到 `UnauthorizedOperation` 錯誤。

如需 IAM 政策的範例，請參閱[使用執行個體中繼資料](#)。如需 SCP 的詳細資訊，請參閱《AWS Organizations 使用者指南》中的[服務控制政策](#)。

## 需要 IMDSv2 的建議路徑

使用上述工具時，我們建議您遵循此路徑來轉換至 IMDSv2。

### 步驟 1：開始時

將開發套件、CLI 以及在其 EC2 執行個體上使用角色憑證的軟體更新為 IMDSv2 相容版本。如需有關更新 CLI 的資訊，請參閱 AWS Command Line Interface 使用者指南中的[升級至最新版本的 AWS CLI](#)。

之後，使用 IMDSv2 請求變更可直接存取執行個體中繼資料的軟體 (換言之，就是未使用軟體開發套件的軟體)。您可以使用 [IMDS Packet Analyzer](#)，識別需要變更以使用 IMDSv2 請求的軟體。

### 步驟 2：追蹤轉換進度

使用指標追蹤您的轉換進度 `CloudWatch 度MetadataNoToken`。此指標會顯示對執行個體上 IMDS 的 IMDSv1 呼叫次數。如需詳細資訊，請參閱 [執行個體指標](#)。

### 步驟 3：當 IMDSv1 使用量為零時

當 `CloudWatch 指標MetadataNoToken` 記錄為零 IMDSv1 用量時，您的執行個體就可以完全轉換為使用 IMDSv2。在這個階段，您可以執行下列操作：

- 帳戶預設

您可以將 IMDSv2 設定為必要的帳戶預設值。啟動執行個體時，執行個體組態會自動設定為帳戶預設值。

若要設定帳戶預設值，請執行下列動作：

- Amazon EC2 主控台：在 EC2 儀表板的帳戶屬性、資料保護和安全下，對於 IMDS 預設值，將執行個體中繼資料服務設定為啟用，將中繼資料版本設定為僅 V2 (需要令牌)。如需詳細資訊，請參閱 [將 IMDSv2 設定為帳戶的預設值](#)。
- AWS CLI：使用 [修改執行個體中繼資料預設 CLI 命令](#)，並指定 `--http-tokens required --http-put-response-hop-limit 2`
- 新執行個體

啟動新執行個體時，您可以執行下列操作：

- Amazon EC2 主控台：在啟動執行個體精靈中，將 Metadata accessible (可存取中繼資料) 設定為 Enabled (已啟用)，並將 Metadata version (中繼資料版本) 設定為 V2 only (token required) (僅 V2 (需要權杖))。如需詳細資訊，請參閱 [啟動時設定執行個體](#)。
- AWS CLI：使用 [執行個體 CLI 命令](#)，並指定 IMDSv2 為必要項目。
- 現有執行個體

針對現有執行個體，可以執行以下操作：

- Amazon EC2 主控台：在執行個體頁面，選取您的執行個體，選擇動作、執行個體設定、修改執行個體中繼資料選項，針對 IMDSv2，選擇必要。如需詳細資訊，請參閱 [需要使用 IMDSv2](#)。
- AWS CLI：使用 [modify-instance-metadata-options](#) CLI 命令，指定僅使用 IMDSv2。

您可以在執行中的執行個體上修改執行個體中繼資料選項，不需要在修改執行個體中繼資料選項後重新啟動執行個體。

#### 步驟 4：檢查執行個體是否已轉換至 IMDSv2

您可以檢查是否有任何執行個體尚未設定為需要使用 IMDSv2，換句話說，IMDSv2 仍設定為 optional。如果任何執行個體仍被設定為 optional，您可以重複上述 [步驟 3](#)，修改執行個體中繼資料選項以使得 IMDSv2 變為 required。

篩選執行個體：

- Amazon EC2 主控台：在執行個體頁面中，使用 IMDSv2 = 選用 篩選條件來篩選執行個體。如需有關篩選的詳細資訊，請參閱 [使用主控台篩選資源](#)。您也可以檢視 IMDSv2 對於每個執行個體是必要還是選用：在偏好設定視窗中，打開 IMDSv2，將 IMDSv2 資料欄新增至執行個體表格。
- AWS CLI：使用 [describe-instances](#) CLI 命令並依據 `metadata-options.http-tokens = optional` 進行篩選，如下所示：

```
aws ec2 describe-instances --filters "Name=metadata-options.http-tokens,Values=optional" --query "Reservations[*].Instances[*].[InstanceId]" --output text
```

### 步驟 5：所有執行個體都已轉換至 IMDSv2 時

ec2:MetadataHttpTokensec2:MetadataHttpPutResponseHopLimit、和 ec2:MetadataHttpEndpoint IAM 條件金鑰可用來控制和 [ModifyInstanceMetadataOptionsAPI](#) [RunInstances](#)和對應 CLIs 的使用。如果已建立政策，而且 API 呼叫中的參數與使用條件金鑰之政策中指定的狀態不相符，則 API 或 CLI 會失敗並顯示 UnauthorizedOperation 回應。如需 IAM 政策的範例，請參閱[使用執行個體中繼資料](#)。

此外，停用 IMDSv1 之後，您可以使用MetadataNoTokenRejected CloudWatch 量度追蹤 IMDSv1 呼叫嘗試和拒絕的次數。如果停用 IMDSv1 之後，您的軟體無法正常運作，且MetadataNoTokenRejected測量結果記錄了 IMDSv1 呼叫，很可能需要更新此軟體才能使用 IMDSv2。

### 使用支援的 AWS 開發套件

若要使用 IMDSv2，您的 EC2 執行個體必須使用支援使用 IMDSv2 的 AWS 開發套件版本。所有 AWS 開發套件的最新版本都支援使用 IMDSv2。

#### Important

建議您保持最新的 SDK 版本，以確保使用最新功能、安全性更新和基礎相依性。不建議您繼續使用不受支援的開發套件版本，由您自行決定。如需詳細資訊，請參閱《AWS 開發套件和工具參考指南》中的 [AWS 開發套件及工具維護政策](#)。

以下是支援使用 IMDSv2 的最低版本：

- [AWS CLI](#)— 1.16.289
- [AWS Tools for Windows PowerShell](#) – 4.0.1.0
- [AWS SDK for .NET](#) – 3.3.634.1
- [AWS SDK for C++](#) – 1.7.229
- [AWS SDK for Go](#) – 1.25.38

- [AWS 適用於轉到 V2 的開發套件](#) — 0.19.0
- [AWS SDK for Java](#) – 1.11.678
- [AWS SDK for Java 2.x](#) – 2.10.21
- [AWS Node.js JavaScript 中適用的軟體開發套件](#)
- [AWS SDK for PHP](#) – 3.147.7
- [AWS 開發套件為 Python \(肉核\)](#) — 1.13.25
- [AWS SDK for Python \(Boto3\)](#) – 1.12.6
- [AWS SDK for Ruby](#) – 3.79.0

## 設定執行個體中繼資料選項

執行個體中繼資料服務 (IMDS) 會在每個 EC2 執行個體本機執行。執行個體中繼資料選項是指控制 EC2 執行個體上 IMDS 可存取性和行為的一組組態。

您可以為每個執行個體設定下列執行個體中繼資料選項：

執行個體中繼資料服務 (IMDS) : | enabled disabled

您可以在執行個體上啟用或停用 IMDS。停用時，您或任何程式碼都無法存取執行個體上的執行個體中繼資料。

IMDS 在執行個體上有兩個端點：IPv4 (169.254.169.254) 和 IPv6 ([fd00:ec2::254])。當您啟用 IMDS 時，會自動啟用 IPv4 端點。如果您想要啟用 IPv6 端點，您需要明確地這樣做。

IPv6 端點 : enableddisabled

您可以在執行個體上明確啟用 IPv6 IMDS 端點。啟用 IPv6 端點時，IPv4 端點會保持啟用狀態。IPv6 端點僅在 [AWS Nitro 系統上建置的執行個體](#) 和支援 [IPv6 的子網路 \(僅限雙堆疊或 IPv6\) 中支援](#)。

中繼資料版本:IMDSv1 or IMDSv2 (token optional)| IMDSv2 only (token required)

要求執行個體中繼資料時，IMDSv2 呼叫需要使用權杖。IMDSv1 呼叫不需要使用權杖。您可以將執行個體設定為允許 IMDSv1 或 IMDSv2 呼叫 (其中記號是選用的)，或僅允許 IMDSv2 呼叫 (其中需要權杖)。

中繼資料回應躍點限制：1— 64

躍點限制是允許 PUT 回應進行的網路躍點數目。您可以將躍點限制設定為最小值1和最大值64。在容器環境中，建議將躍點限制設定為2。如需詳細資訊，請參閱 [考量事項](#)。

存取執行個體中繼資料中的標籤：enabled| disabled

您可以從執行個體的中繼資料啟用或停用執行個體標籤的存取權。如需詳細資訊，請參閱 [使用執行個體中繼資料中的執行個體標籤](#)。

## 執行個體中繼資料選項的設定

執行個體中繼資料選項可以在不同層級設定，如下所示：

- 帳戶 — 您可以在帳戶層級為每個執行個體中繼資料選項設定預設值 AWS 區域。啟動執行個體時，執行個體中繼資料選項會自動設定為帳戶層級值。您可以在啟動時變更這些值。帳戶層級的預設值不會影響現有例項。
- AMI — 您可以在註冊或修改 AMI v2.0 時將 `imds-support` 參數設置為。使用此 AMI 啟動執行個體時，執行個體中繼資料版本會自動設定為 IMDSv2，且躍點限制設定為 2。
- 執行個體 — 您可以在啟動時變更執行個體的所有執行個體中繼資料選項，覆寫預設設定。您也可以執行中或已停止的執行個體上啟動後，變更執行個體中繼資料選項。請注意，變更可能受到 IAM 或 SCP 政策的限制。

如需詳細資訊，請參閱 [設定新執行個體的執行個體中繼資料選項](#) 及 [修改現有執行個體的執行個體中繼資料選項](#)。

## 執行個體中繼資料選項的優先順序

每個執行個體中繼資料選項的值都會在執行個體啟動時決定，並遵循階層優先順序。最上方優先順序最高的階層如下所示：

- 優先順序 1：啟動時的執行個體組態 — 您可以在啟動範本或執行個體組態中指定值。此處指定的任何值都會覆寫在帳戶層級或 AMI 中指定的值。
- 優先順序 2：帳戶設定 — 如果在執行個體啟動時未指定值，則由帳戶層級設定決定 (針對每個 AWS 區域設定)。帳戶層級設定可以包含每個中繼資料選項的值，或完全不指定任何偏好設定。
- 優先順序 3：AMI 組態 — 如果未在執行個體啟動或帳戶層級指定值，則由 AMI 組態決定。這僅會套用到 `HttpTokens` 和 `HttpPutResponseHopLimit`。

每個中繼資料選項都會分別評估。執行個體可以混合使用直接執行個體設定、帳戶層級預設值，以及 AMI 的組態設定。



您可以在執行中或停止的執行個體上啟動後變更任何中繼資料選項的值，除非變更受 IAM 或 SCP 政策的限制。

### 決定中繼資料選項的值-範例 1

在此範例中，EC2 執行個體會啟動至帳戶層級設定 `HttpPutResponseHopLimit` 為 1 的區域。指定的 AMI 已 `ImdsSupport` 設定為 `v2.0`。啟動時不會直接在執行個體上指定中繼資料選項。執行個體會使用下列中繼資料選項啟動：

```
"MetadataOptions": {
 ...
 "HttpTokens": "required",
 "HttpPutResponseHopLimit": 1,
 ...
}
```

這些值的確定方式如下：

- 啟動時未指定中繼資料選項：在執行個體啟動期間，執行個體啟動參數或啟動範本中未提供中繼資料選項的特定值。
- 科目設定採用下一個優先順序：如果在啟動時沒有指定的特定值，則會優先考慮「區域」內帳戶層次的設定。這表示會套用在帳戶層級設定的預設值。在此情況下，會將設定 `HttpPutResponseHopLimit` 為 1。
- AMI 設定最後優先順序：如果在啟動時或帳戶層級 `HttpTokens` (執行個體中繼資料版本) 沒有指定的特定值，則會套用 AMI 設定。在這種情況下，AMI 設置 `ImdsSupport: v2.0` 確定 `HttpTokens` 該設置為 `required`。請注意，雖然 AMI 設定 `ImdsSupport: v2.0` 是設定的 `HttpPutResponseHopLimit: 2`，但帳戶層級設定已覆寫 `HttpPutResponseHopLimit: 1`，其優先順序較高。

### 決定中繼資料選項的值-範例 2

在此範例中，EC2 執行個體的啟動設定與上一個範例 1 相同，但在啟動時將 `HttpTokens` 設定為 `optional` 直接在執行個體上。執行個體會使用下列中繼資料選項啟動：

```
"MetadataOptions": {
 ...
 "HttpTokens": "optional",
 "HttpPutResponseHopLimit": 1,
 ...
}
```

的值 `HttpPutResponseHopLimit` 是以與範例 1 中相同的方式來決定。不過，的值決定如下：啟動時在執行個體上設定的中繼資料選項優先。`HttpTokens` 即使 AMI 設定為 `ImdsSupport: v2.0` (換句話說，`HttpTokens` 設定為 `required`)，但啟動時 (`HttpTokens` 設為 `optional`) 在執行個體上指定的值優先。

## 設定執行個體元資料版本

啟動執行個體時，執行個體中繼資料版本的值為 `IMDSv1 or IMDSv2 (token optional)` 或 `IMDSv2 only (token required)`。

執行個體啟動時，您可以手動指定中繼資料版本的值，或使用預設值。如果您手動指定值，它會取代任何預設值。如果您選擇不手動指定值，它將由預設設定組合決定，如下表所述。

此表格顯示啟動時執行處理的中繼資料版本 (由欄 4 中的「結果執行處理組態」指示) 如何由不同組態層級的設定決定。優先順序是從左到右，其中第一欄的優先順序最高，如下所示：

- 欄 1：啟動參數 — 代表您在啟動時手動指定之執行處理的設定。
- 欄 2：帳戶層級預設值 — 代表帳戶的設定。
- 第 3 欄：AMI 預設值 — 代表 AMI 上的設定。

| 啟動參數           | 帳戶層級預設    | AMI 預設值 | 產生的實例配置   |
|----------------|-----------|---------|-----------|
| 僅 V2 (需要令牌)    | 無偏好       | 僅限 V2   | 僅限 V2     |
| 僅 V2 (需要令牌)    | 僅限 V2     | 僅限 V2   | 僅限 V2     |
| 僅 V2 (需要令牌)    | V1 或第 2 版 | 僅限 V2   | 僅限 V2     |
| V1 或 V2 (令牌可選) | 無偏好       | 僅限 V2   | V1 或第 2 版 |
| V1 或 V2 (令牌可選) | 僅限 V2     | 僅限 V2   | V1 或第 2 版 |
| V1 或 V2 (令牌可選) | V1 或第 2 版 | 僅限 V2   | V1 或第 2 版 |
| 未設定            | 無偏好       | 僅限 V2   | 僅限 V2     |

| 啟動參數             | 帳戶層級預設    | AMI 預設值 | 產生的實例配置   |
|------------------|-----------|---------|-----------|
| 未設定              | 僅限 V2     | 僅限 V2   | 僅限 V2     |
| 未設定              | V1 或第 2 版 | 僅限 V2   | V1 或第 2 版 |
| 僅 V2 ( 需要令牌 )    | 無偏好       | null    | 僅限 V2     |
| 僅 V2 ( 需要令牌 )    | 僅限 V2     | null    | 僅限 V2     |
| 僅 V2 ( 需要令牌 )    | V1 或第 2 版 | null    | 僅限 V2     |
| V1 或 V2 ( 令牌可選 ) | 無偏好       | null    | V1 或第 2 版 |
| V1 或 V2 ( 令牌可選 ) | 僅限 V2     | null    | V1 或第 2 版 |
| V1 或 V2 ( 令牌可選 ) | V1 或第 2 版 | null    | V1 或第 2 版 |
| 未設定              | 無偏好       | null    | V1 或第 2 版 |
| 未設定              | 僅限 V2     | null    | 僅限 V2     |
| 未設定              | V1 或第 2 版 | null    | V1 或第 2 版 |

## 使用 IAM 條件金鑰限制執行個體中繼資料選項

您可以在 IAM 政策或 SCP 中使用 IAM 條件金鑰，如下所示：

- 只有在執行個體設定為需要使用 IMDSv2 時才允許啟動該執行個體
- 限制允許的轉跳數目
- 關閉對執行個體中繼資料的存取

### 任務

- [設定新執行個體的執行個體中繼資料選項](#)
- [修改現有執行個體的執行個體中繼資料選項](#)

**Note**

您應該謹慎繼續執行和仔細進行測試，然後再進行任何變更。請記得下列事項：

- 如果您強制使用 IMDSv2，使用 IMDSv1 來存取執行個體中繼資料的應用程式或代理程式便會中斷。
- 如果您關閉對執行個體中繼資料的所有存取，則需要存取執行個體中繼資料才能正常運作的應用程式或代理程式便會中斷。
- 對於 IMDSv2，在檢索字符時必須使用 `/latest/api/token`。
- (僅限視窗) 如果您的 PowerShell 版本早於 4.0，您必須[更新至視窗管理架構 4.0](#)，才能要求使用 IMDSv2。

## 設定新執行個體的執行個體中繼資料選項

您可以為新執行個體設定下列執行個體中繼資料選項。

### 選項

- [需要使用 IMDSv2](#)
- [啟動 IPv4 和 IPv6 端點](#)
- [關閉對執行個體中繼資料的存取](#)

### 需要使用 IMDSv2

您可以使用下列方法要求在新執行個體上使用 IMDSv2。

### 需要 IMDSv2

- [將 IMDSv2 設定為帳戶的預設值](#)
- [啟動時設定執行個體](#)
- [設定 AMI](#)
- [使用 IAM 政策](#)

### 將 IMDSv2 設定為帳戶的預設值

您可以在帳戶層級為每 AWS 區域個執行個體中繼資料服務 (IMDS) 設定預設版本。這表示當您啟動新執行個體時，執行個體中繼資料版本會自動設定為帳戶層級的預設值。不過，您可以在啟動時或啟

動後手動覆寫值。如需帳戶層級設定和手動覆寫如何影響執行個體的詳細資訊，請參閱。[執行個體中繼資料選項的優先順序](#)

#### Note

設定帳戶層級預設值並不會重設現有執行個體。例如，如果您將帳戶層級預設值設定為 IMDSv2，則任何設定為 IMDSv1 的現有執行個體都不會受到影響。如果要變更現有例證上的值，您必須手動變更例證本身的值。

您可以將執行個體中繼資料版本的帳戶預設值設定為 IMDSv2，以便帳戶中的所有新執行個體都必須使用 IMDSv2 啟動，而且會停用 IMDSv1。使用此帳戶的預設值，當您啟動執行個體時，執行個體的預設值如下：

- 主控台：中繼資料版本僅設為 V2 (需要 Token)，而中繼資料回應躍點限制設定為 2。
- AWS CLI: HttpTokens 設定為 required 且 HttpPutResponseHopLimit 設定為 2。

#### Note

將帳戶預設值設定為 IMDSv2 之前，請確定您的執行個體不依賴於 IMDSv1。如需詳細資訊，請參閱 [需要 IMDSv2 的建議路徑](#)。

## Console

若要將 IMDSv2 設定為指定區域帳戶的預設值

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 若要變更 AWS 區域，請使用頁面右上角的「地區」選取器。
3. 在導覽窗格中，選擇 EC2 Dashboard (EC2 儀表板)。
4. 在帳戶屬性下，選擇資料保護和安全性。
5. 在「IMDS 預設值」旁邊，選擇「管理」。
6. 在 [管理 IMDS 預設值] 頁面上，執行下列動作：
  - a. 針對執行個體中繼資料服務，選擇 [啟
  - b. 針對 Metadata version (中繼資料版本)，選擇 V2 only (token required) (僅限 V2 (需要權杖))。

- c. 對於中繼資料回應躍點限制，如果執行個體將託管容器，請指定 2。否則，請選擇無偏好設定。如果未指定偏好設定，則在啟動時，如果 AMI 需要 IMDSv2，則預設值為 2；否則預設值為 1。
- d. 選擇更新。

## AWS CLI

若要將 IMDSv2 設定為指定區域帳戶的預設值

使用 [修改執行個體-中繼資料預設指令](#)，並指定要在其中修改 [IMDS](#) 帳戶層級設定的區域。2 如果您的執行個體將託管容器，則 `--http-put-response-hop-limit` 「包含 `--http-tokens`」設定為 `required` 和設定為。否則，請指定 `-1` 以指示沒有偏好。如果指定 `-1` (未指定偏好設定)，則在啟動時，2 如果 AMI 需要 IMDSv2，則預設值為。1

```
aws ec2 modify-instance-metadata-defaults \
 --region us-east-1 \
 --http-tokens required \
 --http-put-response-hop-limit 2
```

預期的輸出結果

```
{
 "Return": true
}
```

若要檢視指定區域的執行個體中繼資料選項的預設帳戶設定

使用指 [get-instance-metadata-defaults](#) 令並指定「區域」。

```
aws ec2 get-instance-metadata-defaults --region us-east-1
```

範例輸出

```
{
 "AccountLevel": {
 "HttpTokens": "required",
 "HttpPutResponseHopLimit": 2
 }
}
```

## 啟動時設定執行個體

[啟動執行個體](#)時，您可以設定下列欄位，將執行個體設定為需要使用 IMDSv2：

- Amazon EC2 主控台：將 Metadata version (中繼資料版本) 設為 V2 only (token required) (僅 V2 (需要權杖))。
- AWS CLI：將 HttpTokens 設為 required。

當您指定需要 IMDSv2 時，也必須透過將可存取中繼資料設定為已啟用 (主控台) 或將 HttpEndpoint 設為 enabled (AWS CLI) 來啟用執行個體中繼資料服務 (IMDS) 端點。

在容器環境中，當需要 IMDSv2 時，建議將躍點限制設定為 2。如需詳細資訊，請參閱 [考量事項](#)。

### New console

要求在新執行個體上使用 IMDSv2

- 在 Amazon EC2 主控台中啟動新執行個體時，展開 Advanced details (進階詳細資訊)，並執行下列動作：
  - 針對 Metadata accessible (可存取中繼資料)，選擇 Enabled (已啟用)。
  - 針對 Metadata version (中繼資料版本)，選擇 V2 only (token required) (僅限 V2 (需要權杖))。
  - (容器環境) 對於中繼資料回應躍點限制，請選擇 2。

如需詳細資訊，請參閱 [進階詳細資訊](#)。

### Old console

要求在新執行個體上使用 IMDSv2

- 在 Amazon EC2 主控台中啟動新執行個體時，請在 Configure Instance Details (設定執行個體詳細資訊) 頁面上選取下列選項：
  - 在 Advanced Details (進階詳細資訊) 下，針對 (可存取中繼資料)，選取 Enabled (已啟用)。
  - 針對 Metadata version (中繼資料版本)，請選取 V2 (token required) (V2 (需要字符))。

如需詳細資訊，請參閱 [步驟 3：設定執行個體詳細資訊](#)。

## AWS CLI

要求在新執行個體上使用 IMDSv2

下列 [run-instances](#) 範例會啟動 `c6i.large` 設定為 `--metadata-options` 的 `HttpTokens=required` 執行個體。當您為 `HttpTokens` 指定值時，您也必須將 `HttpEndpoint` 設定為 `enabled`。由於中繼資料擷取請求的安全權杖標頭設定為 `required`，因此在要求執行個體中繼資料時需要執行個體使用 IMDSv2。

在容器環境中，當需要 IMDSv2 時，建議將躍點限制設定為 2 與 `HttpPutResponseHopLimit=2`

```
aws ec2 run-instances \
 --image-id ami-0abcdef1234567890 \
 --instance-type c6i.large \
 ...
 --metadata-options
 "HttpEndpoint=enabled,HttpTokens=required,HttpPutResponseHopLimit=2"
```

## PowerShell

要求在新執行個體上使用 IMDSv2

下列 [New-EC2Instance](#) 指令程式範例會啟動 `MetadataOptions_HttpEndpoint` 設定為 `enabled` 且 `MetadataOptions_HttpTokens` 參數為 `required` 的 `c6i.large` 執行個體。當您為 `HttpTokens` 指定值時，您也必須將 `HttpEndpoint` 設定為 `enabled`。由於中繼資料擷取請求的安全權杖標頭設定為 `required`，因此在要求執行個體中繼資料時需要執行個體使用 IMDSv2。

```
New-EC2Instance \
 -ImageId ami-0abcdef1234567890 \
 -InstanceType c6i.large \
 -MetadataOptions_HttpEndpoint enabled \
 -MetadataOptions_HttpTokens required
```

## AWS CloudFormation

若要使用指定例證的詮釋資料選項 AWS CloudFormation，請參閱《使用指南》中 AWS CloudFormation 的 [AWS::EC2::LaunchTemplate MetadataOptions](#) 性質。



## 設定 AMI

註冊新 AMI 或修改現有 AMI 時，可以將 `imds-support` 參數設定為 `v2.0`。從此 AMI 啟動的執行個體會將 Metadata version (中繼資料版本) 設定為 V2 only (token required) (僅 V2 (需要權杖)) (主控台) 或將 `HttpTokens` 設為 `required` (AWS CLI)。透過這些設定，執行個體要求在請求執行個體中繼資料時使用 IMDSv2。

請注意，當您將 `imds-support` 設為 `v2.0` 時，從這個 AMI 啟動的執行個體也將有 Metadata response hop limit (中繼資料回應跳轉限制) (主控台) 或將 `http-put-response-hop-limit` (AWS CLI) 設為 2。

### Important

除非 AMI 軟體支援 IMDSv2，否則請勿使用此參數。將值設為 `v2.0` 後，將無法復原。「重設」AMI 的唯一方法是從基礎快照中建立一個新的 AMI。

## 設定適用於 IMDSv2 的新 AMI

使用下列其中一種方法來設定 ImDSv2 的新 AMI。

### AWS CLI

以下 [register-image](#) 範例使用指定的 EBS 根磁碟區快照作為裝置 `/dev/xvda` 來註冊 AMI。為 `imds-support` 參數指定 `v2.0`，以便從此 AMI 啟動的執行個體在請求執行個體中繼資料時要求使用 IMDSv2。

```
aws ec2 register-image \
 --name my-image \
 --root-device-name /dev/xvda \
 --block-device-mappings DeviceName=/dev/
xvda,Ebs={SnapshotId=snap-0123456789example} \
 --architecture x86_64 \
 --imds-support v2.0
```

### PowerShell

下列指 [Register-EC2Image](#) 令程式範例會使用指定的 EBS 根磁碟區快照作為裝置來註冊 AMI。`/dev/xvda` 為 `ImdsSupport` 參數指定 `v2.0`，以便從此 AMI 啟動的執行個體在請求執行個體中繼資料時要求使用 IMDSv2。

```

Import-Module AWS.Tools.EC2 # Required for Amazon.EC2.Model object creation.
Register-EC2Image `
 -Name 'my-image' `
 -RootDeviceName /dev/xvda `
 -BlockDeviceMapping (
 New-Object `
 -TypeName Amazon.EC2.Model.BlockDeviceMapping `
 -Property @{
 DeviceName = '/dev/xvda';
 EBS = (New-Object -TypeName Amazon.EC2.Model.EbsBlockDevice -Property
@{
 SnapshotId = 'snap-0123456789example';
 VolumeType = 'gp3'
 })
 }) `
 -Architecture X86_64 `
 -ImdsSupport v2.0

```

## 設定適用於 IMDSv2 的現有 AMI

使用下列其中一種方法來為 IMDSv2 設定現有的 AMI。

### AWS CLI

下列 [modify-image-attribute](#) 範例只會修改 IMDSv2 的現有 AMI。為 `imds-support` 參數指定 `v2.0`，以便從此 AMI 啟動的執行個體在請求執行個體中繼資料時要求使用 IMDSv2。

```

aws ec2 modify-image-attribute \
 --image-id ami-0123456789example \
 --imds-support v2.0

```

### PowerShell

下列 [Edit-EC2ImageAttribute](#) 指令程式範例只會修改 IMDSv2 的現有 AMI。為 `imds-support` 參數指定 `v2.0`，以便從此 AMI 啟動的執行個體在請求執行個體中繼資料時要求使用 IMDSv2。

```

Edit-EC2ImageAttribute `
 -ImageId ami-0abcdef1234567890 `
 -ImdsSupport 'v2.0'

```

## 使用 IAM 政策

您也可以建立 IAM 政策，以防止使用者啟動新執行個體，除非其在新執行個體上需要 IMDSv2。

透過使用 IAM 政策，在所有新執行個體上強制使用 IMDSv2

若要在請求執行個體中繼資料時，確保使用者僅能啟動需要使用 IMDSv2 的執行個體，您可以指定必須符合需要 IMDSv2 的條件，才能啟動執行個體。如需 IAM 政策範例，請參閱[使用執行個體中繼資料](#)。

## 啟動 IPv4 和 IPv6 端點

IMDS 在執行個體上有兩個端點：IPv4 (169.254.169.254) 和 IPv6 ([fd00:ec2::254])。當您啟用 IMDS 時，會自動啟用 IPv4 端點。即使您將執行個體啟動至僅 IPv4 子網路，IPv6 端點仍會保持停用狀態。若要啟用 IPv6 端點，您需要明確執行此作業。當您啟用 IPv6 端點時，IPv4 端點會保持啟用狀態。

您可以在執行個體啟動時或之後啟用 IPv6 端點。

## 啟用 IPv6 端點的需求

- 選取的執行個體類型建立在 [AWS Nitro 系統](#) 上。
- 選取的子網路支援 IPv6，其中子網路可以是 [雙堆疊或僅 IPv6](#)。

使用下列任一方法來啟動已啟用 IMDS IPv6 端點的執行個體。

### New console

若要在執行個體啟動時啟用 IMDS IPv6 端點

- 在 Amazon EC2 主控台中[啟動執行個體](#)，並在 Advanced details (進階詳細資訊) 下指定以下內容：
  - 對於中繼資料 IPv6 端點，請選擇已啟用。

如需詳細資訊，請參閱 [進階詳細資訊](#)。

### AWS CLI

若要在執行個體啟動時啟用 IMDS IPv6 端點

以下 [run-instances](#) 範例會啟動已啟用執行個體中繼資料服務 IPv6 端點的 `c6i.large` 執行個體。若要啟用 IPv6 端點，對於 `--metadata-options` 參數，請指定 `HttpProtocolIpv6=enabled`。當您為 `HttpProtocolIpv6` 指定值時，您也必須將 `HttpEndpoint` 設定為 `enabled`。

```
aws ec2 run-instances \
 --image-id ami-0abcdef1234567890 \
 --instance-type c6i.large \
 ...
 --metadata-options "HttpEndpoint=enabled,HttpProtocolIpv6=enabled"
```

## PowerShell

若要在執行個體啟動時啟用 IMDS IPv6 端點

下列 [New-EC2Instance](#) 指令程式範例會啟動已為 IMDS 啟用 IPv6 端點的 `c6i.large` 執行個體。若要啟用 IPv6 端點，請將 `MetadataOptions_HttpProtocolIpv6` 指定為 `enabled`。當您為 `MetadataOptions_HttpProtocolIpv6` 指定值時，您也必須將 `MetadataOptions_HttpEndpoint` 設定為 `enabled`。

```
New-EC2Instance `\
 -ImageId ami-0abcdef1234567890 `\
 -InstanceType c6i.large `\
 -MetadataOptions_HttpEndpoint enabled `\
 -MetadataOptions_HttpProtocolIpv6 enabled
```

## 關閉對執行個體中繼資料的存取

可以在啟動執行個體時停用 IMDS 來關閉對執行個體中繼資料的存取。稍後透過重新啟用 IMDS 可以開啟存取。如需詳細資訊，請參閱 [開啟對執行個體中繼資料的存取](#)。

### Important

可以選擇在啟動時或啟動後停用 IMDS。如果在啟動時停用 IMDS，下列項目可能無法運作：

- 您可能沒有執行個體的 SSH 存取權。`public-keys/0/openssh-key` 是執行個體的公有 SSH 金鑰，它將不可存取，因為通常會從 EC2 執行個體中繼資料中提供並存取該金鑰。
- EC2 使用者資料將無法使用，且不會在執行個體啟動時執行。EC2 使用者資料託管於 IMDS。如果停用 IMDS，可以有效地關閉對使用者資料的存取。

若要存取此功能，可以在啟動後重新啟用 IMDS。

## New console

啟動時關閉對執行個體中繼資料的存取

- 在 Amazon EC2 主控台中[啟動執行個體](#)，並在 Advanced details (進階詳細資訊) 下指定以下內容：
  - 針對 Metadata accessible (可存取中繼資料)，選擇 Disabled (已停用)。

如需詳細資訊，請參閱 [進階詳細資訊](#)。

## Old console

啟動時關閉對執行個體中繼資料的存取

- 在 Amazon EC2 主控台中啟動執行個體時，請在 Configure Instance Details (設定執行個體詳細資訊) 頁面上選取下列選項：
  - 在 Advanced Details (進階詳細資訊) 下，針對 (可存取中繼資料)，選取 Disabled (已停用)。

如需詳細資訊，請參閱 [步驟 3：設定執行個體詳細資訊](#)。

## AWS CLI

啟動時關閉對執行個體中繼資料的存取

將 `--metadata-options` 設為 `HttpEndpoint=disabled` 來啟動執行個體。

```
aws ec2 run-instances \
 --image-id ami-0abcdef1234567890 \
 --instance-type c6i.large \
 ...
 --metadata-options "HttpEndpoint=disabled"
```

## PowerShell

啟動時關閉對執行個體中繼資料的存取

下列 [New-EC2Instance](#) 指令程式範例會啟動 `MetadataOptions_HttpEndpoint` 設定為 `disabled` 的執行個體。

```
New-EC2Instance `
 -ImageId ami-0abcdef1234567890 `
 -InstanceType c6i.large `
 -MetadataOptions_HttpEndpoint disabled
```

## AWS CloudFormation

若要使用指定例證的詮釋資料選項 AWS CloudFormation，請參閱《使用者指南》中 AWS CloudFormation 的 [AWS::EC2::LaunchTemplate MetadataOptions](#) 性質。

## 修改現有執行個體的執行個體中繼資料選項

您可以修改現有執行個體的執行個體中繼資料選項。

您也可以建立 IAM 政策，以防止使用者修改現有執行個體上的執行個體中繼資料選項。若要控制哪些使用者可以修改執行個體中繼資料選項，請指定政策，防止具有指定角色之使用者以外的所有使用者使用 [ModifyInstanceMetadataOptionsAPI](#)。如需 IAM 政策範例，請參閱 [使用執行個體中繼資料](#)。

## 查詢現有執行個體的執行個體中繼資料選項

您可以使用下列其中一種方法，查詢現有執行個體的執行個體中繼資料選項。

### Console

若要查詢現有執行個體的執行個體中繼資料選項，請使用主控台

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體。
4. 選擇動作、執行個體設定、修改執行個體中繼資料選項。
5. 在「修改執行個體中繼資料選項」對話方塊中，檢閱目前執行個體中繼資料選項。

### AWS CLI

若要查詢現有執行個體的執行個體中繼資料選項，請使用 AWS CLI

使用 [describe-instances](#) CLI 命令。

```
aws ec2 describe-instances \
 --instance-id i-1234567898abcdef0 \
 --query 'Reservations[].Instances[].MetadataOptions'
```

## PowerShell

若要使用工具查詢現有執行個體的執行個體中繼資料選項 PowerShell

使用指 [Get-EC2Instance](#) 令程式。

```
(Get-EC2Instance \
 -InstanceId i-1234567898abcdef0).Instances.MetadataOptions
```

## 需要使用 IMDSv2

使用下列其中一種方法來修改現有執行個體上的執行個體中繼資料選項，以便在請求執行個體中繼資料時要求 IMDSv2。當要求 IMDSv2 時，就無法使用 IMDSv1。

### Note

在要求使用 IMDSv2 之前，請確定執行個體未進行 IMDSv1 呼叫。此 `MetadataNoTokenCloudWatch` 量度會追蹤 IMDSv1 呼叫。當執行個體的 IMDSv1 用量 `MetadataNoToken` 記錄為零時，該執行個體就會準備好要求 IMDSv2。

## Console

要求在現有執行個體上使用 IMDSv2

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體。
4. 選擇動作、執行個體設定、修改執行個體中繼資料選項。
5. 在修改執行個體中繼資料選項對話方塊中，執行下列動作：
  - a. 針對執行個體中繼資料服務，請選取啟用。
  - b. 針對 IMDSv2，請選擇必要。
  - c. 選擇儲存。

## AWS CLI

要求在現有執行個體上使用 IMDSv2

使用 [modify-instance-metadata-options](#) CLI 命令，並將 `http-tokens` 參數設定為 `required`。當您為 `http-tokens` 指定值時，您也必須將 `http-endpoint` 設定為 `enabled`。

```
aws ec2 modify-instance-metadata-options \
 --instance-id i-1234567898abcdef0 \
 --http-tokens required \
 --http-endpoint enabled
```

## PowerShell

要求在現有執行個體上使用 IMDSv2

使用指 [Edit-EC2InstanceMetadataOption](#) 令程式，並將 `HttpTokens` 參數設定為 `required`。當您為 `HttpTokens` 指定值時，您也必須將 `HttpEndpoint` 設定為 `enabled`。

```
(Edit-EC2InstanceMetadataOption \
 -InstanceId i-1234567898abcdef0 \
 -HttpTokens required \
 -HttpEndpoint enabled).InstanceMetadataOptions
```

## 還原使用 IMDSv1

當要求 IMDSv2 時，在請求執行個體中繼資料時，IMDSv1 將無法運作。當 IMDSv2 為選用時，IMDSv2 和 IMDSv1 都可以運作。因此，若要還原 IMDSv1，請使用下列其中一種方法，將 IMDSv2 設為選用。

## Console

在執行個體上還原使用 IMDSv1

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體。
4. 選擇動作、執行個體設定、修改執行個體中繼資料選項。
5. 在修改執行個體中繼資料選項對話方塊中，執行下列動作：



- a. 針對執行個體中繼資料服務，請確定已選取啟用。
- b. 針對 IMDSv2，請選擇選用。
- c. 選擇儲存。

## AWS CLI

在執行個體上還原使用 IMDSv1

您可使用 [modify-instance-metadata-options](#) CLI 命令並將 `http-tokens` 設定為 `optional`，以在要求執行個體中繼資料時還原使用 IMDSv1。

```
aws ec2 modify-instance-metadata-options \
 --instance-id i-1234567898abcdef0 \
 --http-tokens optional \
 --http-endpoint enabled
```

## PowerShell

在執行個體上還原使用 IMDSv1

您可以使用 `HttpTokens` 設定為 `optional` 的 [Edit-EC2InstanceMetadataOption](#) 指令程式，在要求執行個體中繼資料時還原 IMDSv1 的使用。

```
(Edit-EC2InstanceMetadataOption \
 -InstanceId i-1234567898abcdef0 \
 -HttpTokens optional \
 -HttpEndpoint enabled).InstanceMetadataOptions
```

## 變更 PUT 回應跳轉限制

對於現有執行個體，您可以變更 PUT 回應跳轉限制的設定。

目前只有 AWS CLI 和 AWS SDK 支援變更 PUT 回應躍點限制。

## AWS CLI

如何變更 PUT 回應跳轉限制

使用 [modify-instance-metadata-options](#) CLI 命令，並將 `http-put-response-hop-limit` 參數設定為必要的跳轉數目。在下列範例中，跳轉限制設定為 3。請注意，為 `http-put-response-hop-limit` 指定數值時，您還必須將 `http-endpoint` 設定為 `enabled`。

```
aws ec2 modify-instance-metadata-options \
 --instance-id i-1234567898abcdef0 \
 --http-put-response-hop-limit 3 \
 --http-endpoint enabled
```

## PowerShell

### 如何變更 PUT 回應跳轉限制

使用 C [Edit-EC2InstanceMetadataOption](#)mdlet 並將 `HttpPutResponseHopLimit` 參數設定為所需的跳點數目。在下列範例中，跳轉限制設定為 3。請注意，為 `HttpPutResponseHopLimit` 指定數值時，您還必須將 `HttpEndpoint` 設定為 `enabled`。

```
(Edit-EC2InstanceMetadataOption \
 -InstanceId i-1234567898abcdef0 \
 -HttpPutResponseHopLimit 3 \
 -HttpEndpoint enabled).InstanceMetadataOptions
```

## 啟動 IPv4 和 IPv6 端點

IMDS 在執行個體上有兩個端點：IPv4 (169.254.169.254) 和 IPv6 ([fd00:ec2::254])。當您啟用 IMDS 時，會自動啟用 IPv4 端點。即使您將執行個體啟動至僅 IPv4 子網路，IPv6 端點仍會保持停用狀態。若要啟用 IPv6 端點，您需要明確執行此作業。當您啟用 IPv6 端點時，IPv4 端點會保持啟用狀態。

您可以在執行個體啟動時或之後啟用 IPv6 端點。

### 啟用 IPv6 端點的需求

- 選取的執行個體類型建立在 [AWS Nitro 系統](#)上。
- 選取的子網路支援 IPv6，其中子網路可以是 [雙堆疊或僅 IPv6](#)。

目前只有 AWS CLI 和 AWS SDK 支援在執行個體啟動後啟用 IMDS IPv6 端點。

## AWS CLI

為您的執行個體啟用 IMDS IPv6 端點

使用 [modify-instance-metadata-options](#) CLI 命令，並將 `http-protocol-ipv6` 參數設定為 `enabled`。請注意，為 `http-protocol-ipv6` 指定數值時，您還必須將 `http-endpoint` 設定為 `enabled`。

```
aws ec2 modify-instance-metadata-options \
 --instance-id i-1234567898abcdef0 \
 --http-protocol-ipv6 enabled \
 --http-endpoint enabled
```

## PowerShell

為您的執行個體啟用 IMDS IPv6 端點

使用指 [Edit-EC2InstanceMetadataOption](#) 令程式，並將 `HttpProtocolIpv6` 參數設定為 `enabled`。請注意，為 `HttpProtocolIpv6` 指定數值時，您還必須將 `HttpEndpoint` 設定為 `enabled`。

```
(Edit-EC2InstanceMetadataOption \
 -InstanceId i-1234567898abcdef0 \
 -HttpProtocolIpv6 enabled \
 -HttpEndpoint enabled).InstanceMetadataOptions
```

## 開啟對執行個體中繼資料的存取

無論使用哪個版本的 IMDS，都可以啟用執行個體上 IMDS 的 HTTP 端點，來開啟對執行個體中繼資料的存取。您可以停用 HTTP 端點，以隨時反轉這項變更。

使用下列其中一種方法來開啟對執行個體上執行個體中繼資料的存取。

## Console

開啟對執行個體中繼資料的存取

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體。

4. 選擇動作、執行個體設定、修改執行個體中繼資料選項。
5. 在修改執行個體中繼資料選項對話方塊中，執行下列動作：
  - a. 針對執行個體中繼資料服務，請選取啟用。
  - b. 選擇儲存。

## AWS CLI

開啟對執行個體中繼資料的存取

使用 [modify-instance-metadata-options](#) CLI 命令，並將 `http-endpoint` 參數設定為 `enabled`。

```
aws ec2 modify-instance-metadata-options \
 --instance-id i-1234567898abcdef0 \
 --http-endpoint enabled
```

## PowerShell

開啟對執行個體中繼資料的存取

使用指 [Edit-EC2InstanceMetadataOption](#) 令程式，並將 `HttpEndpoint` 參數設定為 `enabled`。

```
(Edit-EC2InstanceMetadataOption \
 -InstanceId i-1234567898abcdef0 \
 -HttpEndpoint enabled).InstanceMetadataOptions
```

關閉對執行個體中繼資料的存取

無論使用哪個版本的 IMDS，都可以停用執行個體上 IMDS 的 HTTP 端點，來關閉對執行個體中繼資料的存取。您可以啟用 HTTP 端點，以隨時反轉這項變更。

使用下列其中一種方法來關閉對執行個體上執行個體中繼資料的存取。

## Console

關閉對執行個體中繼資料的存取

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。

3. 選取執行個體。
4. 選擇動作、執行個體設定、修改執行個體中繼資料選項。
5. 在修改執行個體中繼資料選項對話方塊中，執行下列動作：
  - a. 針對執行個體中繼資料服務，清除啟用。
  - b. 選擇儲存。

## AWS CLI

關閉對執行個體中繼資料的存取

使用 [modify-instance-metadata-options](#) CLI 命令，並將 `http-endpoint` 參數設定為 `disabled`。

```
aws ec2 modify-instance-metadata-options \
 --instance-id i-1234567898abcdef0 \
 --http-endpoint disabled
```

## PowerShell

關閉對執行個體中繼資料的存取

使用指 [Edit-EC2InstanceMetadataOption](#) 令程式，並將 `HttpEndpoint` 參數設定為 `disabled`。

```
(Edit-EC2InstanceMetadataOption \
 -InstanceId i-1234567898abcdef0 \
 -HttpEndpoint disabled).InstanceMetadataOptions
```

## 擷取執行個體中繼資料

因為您的執行個體中繼資料可從您執行中的執行個體取得，您無須使用 Amazon EC2 主控台或 AWS CLI。若您正在撰寫要從您的執行個體執行的指令碼，這將會很有幫助。例如，您可以存取從執行個體中繼資料存取您執行個體的本機 IP 地址，管理與外部應用程式的連線。

執行個體中繼資料分為數種分類。對於每個執行個體中繼資料類別的說明，請參閱 [執行個體中繼資料分類](#)。

若要從執行中的執行個體檢視所有分類的執行個體中繼資料，請使用下列 IPv4 或 IPv6 URI。

## IPv4

```
http://169.254.169.254/latest/meta-data/
```

## IPv6

```
http://[fd00:ec2::254]/latest/meta-data/
```

IP 地址是 link-local 地址且僅在執行個體中有效。如需詳細資訊，請參閱本指南中的 [Link-local 地址](#)，以及 Wikipedia 上的 [Link-local address](#)。

### Note

本節中的範例使用 IMDS 的 IPv4 地址：169.254.169.254。如果您要透過 IPv6 地址擷取 EC2 執行個體的執行個體中繼資料，請確定您啟用並改用 IPv6 地址：[fd00:ec2::254]。IMDS 的 IPv6 地址與 IMDSv2 命令相容。IPv6 地址只能在 [AWS Nitro 系統上建置的執行個體](#) 和 [支援 IPv6 的子網路中存取 \(僅限雙堆疊或 IPv6\)](#)。

根據您使用 IMDSv1 或 IMDSv2，命令格式會有所不同。依預設，您可以同時使用兩種 IMDS 版本。欲要求使用 IMDSv2，請參閱 [使用 IMDSv2](#)。

若要擷取 Linux 執行個體上的執行個體

您可以使用 cURL 等工具，如下列範圍所示。

## IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/
```

## IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/
```

若要擷取 Windows 執行個體上的執行個體

您可以使用 PowerShell 指令程式擷取 URI。例如，如果您執行的是 3.0 版或更新版本 PowerShell，請使用下列指令程式。

## IMDSv2

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/meta-data/
```

## IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/
```

如果您不想使用 PowerShell，則可以安裝第三方工具，例如 GNU Wget 或 cURL。

### Important

若您在 Windows 執行個體上安裝第三方工具，請確認您已仔細閱讀過隨附的文件，因為呼叫 HTTP 的方法和輸出格式可能會與此處記載的不同。

## 成本

您無須為用來擷取執行個體中繼資料和使用者資料的 HTTP 請求支付費用。

## 考量事項

若要避免執行個體中繼資料擷取的問題，請考慮下列事項：

- 在容器環境中，建議將躍點限制設定為 2。

依預設，AWS 開發套件會使用 IMDSv2 呼叫。如果 IMDSv2 呼叫沒有收到回應，開發套件會重試呼叫，如果仍然不成功，則會使用 IMDSv1。這可能會導致延遲，特別是在容器環境中。在容器環境中，如果跳轉限制為 1，則 IMDSv2 回應不會傳回，因為前往容器會被視為額外的網路跳轉。若要避免降回至 IMDSv1 的程序和產生的延遲，在容器環境中，我們建議您將跳轉限制設定為 2。如需詳細資訊，請參閱 [設定執行個體中繼資料選項](#)。

- (僅限視窗) 使用視窗系統建立自訂 AMI。

為了確保 IMDS 在您從自訂 Windows AMI 啟動執行個體時可以運作，AMI 必須是使用 Windows Sysprep 建立的標準化映像檔。否則，IMDS 將無法運作。表單詳細資訊，請參閱[使用視窗系統建立 AMI](#)。

- 對於 IMDSv2，在檢索字符時必須使用 `/latest/api/token`。

發出 PUT 請求到任何版本特定的路徑 (例如 `/2021-03-23/api/token`) 會導致中繼資料服務傳回 403 Forbidden 錯誤。這是預期行為。

- 如果需要 IMDSv2，則 IMDSv1 將無法運作。

您可以檢查執行個體是否需要 IMDSv2，如下所示：選取執行個體以檢視其詳細資訊，然後檢查 IMDSv2 的值。該值為必要 (只能使用 IMDSv2) 或選用 (可以使用 IMDSv2 和 IMDSv1)。

## 回應及錯誤訊息

所有執行個體中繼資料都會以文字傳回 (HTTP 內容類型 `text/plain`)。

特定中繼資料資源的請求會傳回適當的值。若資源無法使用，則會傳回 404 - Not Found HTTP 錯誤碼。

一般中繼資料資源的請求 (URI 的結尾為 `/`) 會傳回可用資源的清單，或是 404 - Not Found HTTP 錯誤碼 (若沒有該資源的話)。清單項目會位於不同列，以換行字元 (ASCII 10) 終止。

對於使用 執行個體中繼資料服務第 2 版 提出的請求，可傳回下列 HTTP 錯誤碼：

- 400 - Missing or Invalid Parameters –PUT 請求無效。
- 401 - Unauthorized –GET 請求使用的字符無效。建議動作會產生新字符。
- 403 - Forbidden –不允許請求，或 IMDS 已關閉。

## 擷取執行個體中繼資料的範例

下列範例提供您可以在 Amazon EC2 執行個體上使用的命令。Linux 和 Windows 執行個體的指令格式不同。

### 範例

- [取得執行個體中繼資料的可用版本](#)
- [取得上層中繼資料項目](#)



- [取得中繼資料項目的值](#)
- [取得可用公有金鑰清單](#)
- [顯示可使用公有金鑰 0 的格式](#)
- [取得公有金鑰 0 \(格式為 OpenSSH 金鑰格式\)](#)
- [取得執行個體的子網 ID](#)
- [取得執行個體的執行個體標籤](#)

## 取得執行個體中繼資料的可用版本

此範例會取得執行個體中繼資料的可用版本。每個版本會參照在發佈新執行個體中繼資料類別時的執行個體中繼資料建置。執行個體中繼資料建置版本與 Amazon EC2 API 版本無關。若您有依存於先前版本中結構和資訊的指令碼，您也可以取得先前版本。

### Note

為避免每次 Amazon EC2 發佈新的執行個體中繼資料建置時都必須更新您的程式碼，我們建議您在路徑中使用 `latest`，而不是版本編號。例如，以如下方式使用 `latest`：

```
curl http://169.254.169.254/latest/meta-data/ami-id
```

## Linux

### IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/
1.0
2007-01-19
2007-03-01
2007-08-29
2007-10-10
2007-12-15
2008-02-01
2008-09-01
2009-04-04
2011-01-01
2011-05-01
```

```
2012-01-12
2014-02-25
2014-11-05
2015-10-20
2016-04-19
...
latest
```

## IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/
1.0
2007-01-19
2007-03-01
2007-08-29
2007-10-10
2007-12-15
2008-02-01
2008-09-01
2009-04-04
2011-01-01
2011-05-01
2012-01-12
2014-02-25
2014-11-05
2015-10-20
2016-04-19
...
latest
```

## Windows

### IMDSv2

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/
1.0
2007-01-19
```

```
2007-03-01
2007-08-29
2007-10-10
2007-12-15
2008-02-01
2008-09-01
2009-04-04
2011-01-01
2011-05-01
2012-01-12
2014-02-25
2014-11-05
2015-10-20
2016-04-19
...
latest
```

## IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/
1.0
2007-01-19
2007-03-01
2007-08-29
2007-10-10
2007-12-15
2008-02-01
2008-09-01
2009-04-04
2011-01-01
2011-05-01
2012-01-12
2014-02-25
2014-11-05
2015-10-20
2016-04-19
...
latest
```

### 取得上層中繼資料項目

此範例會取得最上層的中繼資料項目。有關響應中物件的更多內容，敬請參閱[執行個體中繼資料分類](#)。

## Linux

### IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-
data/
ami-id
ami-launch-index
ami-manifest-path
block-device-mapping/
events/
hostname
iam/
instance-action
instance-id
instance-life-cycle
instance-type
local-hostname
local-ipv4
mac
metrics/
network/
placement/
profile
public-hostname
public-ipv4
public-keys/
reservation-id
security-groups
services/
```

### IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/
ami-id
ami-launch-index
ami-manifest-path
block-device-mapping/
events/
hostname
iam/
```

```
instance-action
instance-id
instance-type
local-hostname
local-ipv4
mac
metrics/
network/
placement/
profile
public-hostname
public-ipv4
public-keys/
reservation-id
security-groups
services/
```

## Windows

### IMDSv2

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/meta-data/
ami-id
ami-launch-index
ami-manifest-path
block-device-mapping/
hostname
iam/
instance-action
instance-id
instance-life-cycle
instance-type
local-hostname
local-ipv4
mac
metrics/
network/
placement/
```

```
profile
public-hostname
public-ipv4
public-keys/
reservation-id
security-groups
services/
```

## IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/
ami-id
ami-launch-index
ami-manifest-path
block-device-mapping/
hostname
iam/
instance-action
instance-id
instance-type
local-hostname
local-ipv4
mac
metrics/
network/
placement/
profile
public-hostname
public-ipv4
public-keys/
reservation-id
security-groups
services/
```

### 取得中繼資料項目的值

這些範例會取得上述範例中取得的某些頂層中繼資料項目的值。IMDSv2 請求會使用在之前範例命令中建立的儲存字符，前提是字符並未過期。

## Linux

### IMDSv2

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/
latest/meta-data/ami-id
ami-0abcdef1234567890
```

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/
latest/meta-data/reservation-id
r-0efghijk987654321
```

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/
latest/meta-data/local-hostname
ip-10-251-50-12.ec2.internal
```

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/
latest/meta-data/public-hostname
ec2-203-0-113-25.compute-1.amazonaws.com
```

### IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/ami-id
ami-0abcdef1234567890
```

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/reservation-id
r-0efghijk987654321
```

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/local-hostname
ip-10-251-50-12.ec2.internal
```

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/public-hostname
ec2-203-0-113-25.compute-1.amazonaws.com
```

## Windows

### IMDSv2

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
GET -Uri http://169.254.169.254/latest/meta-data/ami-id
ami-0abcdef1234567890
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
GET -Uri http://169.254.169.254/latest/meta-data/reservation-id
r-0efghijk987654321
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
GET -Uri http://169.254.169.254/latest/meta-data/local-hostname
ip-10-251-50-12.ec2.internal
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
GET -Uri http://169.254.169.254/latest/meta-data/public-hostname
ec2-203-0-113-25.compute-1.amazonaws.com
```

### IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/ami-id
ami-0abcdef1234567890
```

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/reservation-
id
r-0efghijk987654321
```

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/local-
hostname
ip-10-251-50-12.ec2.internal
```

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/public-
hostname
ec2-203-0-113-25.compute-1.amazonaws.com
```



## 取得可用公有金鑰清單

此範例會取得可用公有金鑰的清單。

### Linux

#### IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/public-keys/
0=my-public-key
```

#### IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/public-keys/
0=my-public-key
```

### Windows

#### IMDSv2

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/meta-data/public-keys/
0=my-public-key
```

#### IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/public-keys/ 0=my-public-key
```

## 顯示可使用公有金鑰 0 的格式

此範例會顯示可使用公有金鑰 0 的格式。

## Linux

### IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/public-keys/0/
openssh-key
```

### IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/public-keys/0/
openssh-key
```

## Windows

### IMDSv2

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
openssh-key
```

### IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
openssh-key
```

取得公有金鑰 0 (格式為 OpenSSH 金鑰格式)

此範例會取得公有金鑰 0 (格式為 OpenSSH 金鑰格式)。

## Linux

### IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
ssh-rsa MIICiTCcAafICCCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAKGA1UEBhMCMC
VVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBA5TC0lBTSBDb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWx1eHAd
BgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAKGA1UEBhMCMCVMxCzAJBgNVBAGTAldBMRAwDgYD
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBA5TC0lBTSBDb25z
b2x1MRIwEAYDVQQDEw1UZXR0Q21sYWx1eHAdBgkqhkiG9w0BCQEWEG5vb251QGft
YXpvbi5jb20wZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySwTc2XADZ4nB+BLygVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZnczvQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
FFbjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvjx79LjStB
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE my-public-key
```

### IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
ssh-rsa MIICiTCcAafICCCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAKGA1UEBhMCMC
VVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBA5TC0lBTSBDb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWx1eHAd
BgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAKGA1UEBhMCMCVMxCzAJBgNVBAGTAldBMRAwDgYD
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBA5TC0lBTSBDb25z
b2x1MRIwEAYDVQQDEw1UZXR0Q21sYWx1eHAdBgkqhkiG9w0BCQEWEG5vb251QGft
YXpvbi5jb20wZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySwTc2XADZ4nB+BLygVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZnczvQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
FFbjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvjx79LjStB
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE my-public-key
```

## Windows

### IMDSv2

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key-ssh-rsa MIICiTCcAaFICCCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAKGA1UEBhMCMVVMxCzAJBgNVBAGTAldBMRAwDgYDVQQLhEwdTZWF0dGx1MQ8wDQYDVQKKEwZBbWF6b24xFDASBgNVBAsTC01BTSBDb25zb2x1MRIwEAYDVQQDEwLUZXN0Q21sYWxhZAdBgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcNMTIwNDI0MjA0NTIxWjCBiDELMAKGA1UEBhMCMVVMxCzAJBgNVBAGTAldBMRAwDgYDVQQLhEwdTZWF0dGx1MQ8wDQYDVQKKEwZBbWF6b24xFDASBgNVBAsTC01BTSBDb25zb2x1MRIwEAYDVQQDEwLUZXN0Q21sYWxhZAdBgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wZGZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ21uUSfwfEvySWtC2XADZ4nB+BLyGVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9TrDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE Ibb30hjZnczvQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0FkbFFBjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvjx79LjStBNYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE my-public-key
```

### IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key-ssh-rsa MIICiTCcAaFICCCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAKGA1UEBhMCMVVMxCzAJBgNVBAGTAldBMRAwDgYDVQQLhEwdTZWF0dGx1MQ8wDQYDVQKKEwZBbWF6b24xFDASBgNVBAsTC01BTSBDb25zb2x1MRIwEAYDVQQDEwLUZXN0Q21sYWxhZAdBgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcNMTIwNDI0MjA0NTIxWjCBiDELMAKGA1UEBhMCMVVMxCzAJBgNVBAGTAldBMRAwDgYDVQQLhEwdTZWF0dGx1MQ8wDQYDVQKKEwZBbWF6b24xFDASBgNVBAsTC01BTSBDb25zb2x1MRIwEAYDVQQDEwLUZXN0Q21sYWxhZAdBgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wZGZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ21uUSfwfEvySWtC2XADZ4nB+BLyGVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9TrDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE Ibb30hjZnczvQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0FkbFFBjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvjx79LjStBNYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE my-public-key
```

## 取得執行個體的子網 ID

此範例會取得執行個體的子網 ID。

### Linux

#### IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:29:96:8f:6a:2d/subnet-id
subnet-be9b61d7
```

#### IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:29:96:8f:6a:2d/subnet-id
subnet-be9b61d7
```

### Windows

#### IMDSv2

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:29:96:8f:6a:2d/subnet-id
subnet-be9b61d7
```

#### IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:29:96:8f:6a:2d/subnet-id
subnet-be9b61d7
```

## 取得執行個體的執行個體標籤

在以下範例中，範例執行個體具有[已啟用執行個體中繼資料上的標籤](#)以及執行個體標籤 Name=MyInstance 和 Environment=Dev。

此範例會取得執行個體的所有執行個體標籤索引鍵。

Linux

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/tags/instance
Name
Environment
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/tags/instance
Name
Environment
```

下列範例會取得在前面的範例中獲得的 Name 索引鍵的值。IMDSv2 請求使用在之前範例命令中建立的儲存字符，前提是字符並未過期。

IMDSv2

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/tags/instance/Name
MyInstance
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/tags/instance/Name
MyInstance
```

## Windows

### IMDSv2

```
PS C:\> $token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/meta-data/tags/instance
Name
Environment
```

### IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/tags/instance
Name
Environment
```

下列範例會取得在前面的範例中獲得的 Name 索引鍵的值。IMDSv2 請求使用在之前範例命令中建立的儲存字符，前提是字符並未過期。

### IMDSv2

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/meta-data/tags/instance/Name
MyInstance
```

### IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/tags/instance/Name
MyInstance
```

## 查詢調節

我們會根據每個執行個體來調整對 IMDS 的查詢，並且我們會限制從執行個體到 IMDS 的同時連線數。

如果您使用 IMDS 擷取 AWS 安全認證，請避免在每次交易期間查詢認證，或從大量執行緒或處理序同時查詢認證，因為這可能會導致限制。相反的，我們建議您快取登入資料，直到其接近到期時間為止。如需有關 IAM 角色以及與該角色相關聯的安全憑證的詳細資訊，請參閱 [從執行個體中繼資料中擷取安全登入資料](#)。

若您在存取 IMDS 時遭到限流，請使用指數退避策略重試您的查詢。

## 限制 IMDS 存取

您可以考慮使用本機防火牆規則，以停用從一些或所有程式對於 IMDS 的存取。

### Note

對於 [AWS Nitro System 上建置的執行個體](#)，當 VPC 中的網路設備 (例如虛擬路由器) 將封包轉送至 IMDS 位址，並停用執行個體的預設[來源/目標](#)檢查時，就可以從您自己的網路連線 IMDS。若要防止來源從您的 VPC 外部到達 IMDS，建議您修改網路應用裝置的組態，以丟棄具有 IMDS 目標 IPv4 位址的封包，以 169.254.169.254 及 (如果啟用了 IPv6 端點) IMDS 的 IPv6 位址。[fd00:ec2::254]

## Linux

### 使用 iptables 限制存取

下列範例會使用 Linux iptables 及其 owner 模組以防止 Apache Web 伺服器 (依據 apache 的預設安裝使用者 ID) 存取 169.254.169.254。其使用拒絕規則以拒絕所有執行個體中繼資料請求 (無論是 IMDSv1 或 IMDSv2) 以該使用者的身分執行任何程序。

```
$ sudo iptables --append OUTPUT --proto tcp --destination 169.254.169.254 --match owner --uid-owner apache --jump REJECT
```

或者，您可以使用允許規則，考慮只允許存取特定使用者或群組。就安全觀點而言，允許規則可能更加簡單，因為它們會要求您做出關於那些軟體需要存取執行個體中繼資料的決策。如果您使用允許規則，即使您稍後變更執行個體上的軟體或組態，您也不太可能意外地允許軟體存取中繼資料服務 (亦即您並非有意存取)。您也可以使用允許規則來組合群組使用方式，如此您就能新增並從允許的群組移除使用者，而不需要變更防火牆規則。

下列範例可防止所有程序 (在使用者帳戶 `trustworthy-user` 中執行的程序除外) 存取 IMDS。



```
$ sudo iptables --append OUTPUT --proto tcp --destination 169.254.169.254 --match owner ! --uid-owner trustworthy-user --jump REJECT
```

### Note

- 若要使用本機防火牆規則，您需要採用之前的範例命令，才能滿足您的需求。
- 依預設，iptables 規則在系統重新啟動時不會持續存在。但可使用 OS 功能將之設定為持續存在，但此處並不會敘述此內容。
- 如果此群組是指定本機使用者的主要群組，iptables owner 模組僅會比對群組成員資格。而不會比對其他群組。

## 使用 PF 或 IPFW 限制存取

如果您使用 FreeBSD 或 OpenBSD，您也可以考慮使用 PF 或 IPFW。下列範例將 IMDS 的存取權僅限於根使用者。

### PF

```
$ block out inet proto tcp from any to 169.254.169.254
```

```
$ pass out inet proto tcp from any to 169.254.169.254 user root
```

### IPFW

```
$ allow tcp from any to 169.254.169.254 uid root
```

```
$ deny tcp from any to 169.254.169.254
```

### Note

PF 及 IPFW 命令的順序非常重要。PF 預設會試最後一個比對規則，而 IPFW 預設是第一個比對規則。

## Windows

### 使用 Windows 防火牆以限制存取

下列 PowerShell 範例會使用內建的 Windows 防火牆來防止網際網路資訊伺服器網路伺服器 (根據其預設安裝使用者識別碼 NT AUTHORITY\IUSR) 存取 169.254.169.254。其使用拒絕規則以拒絕所有執行個體中繼資料請求 (無論是 IMDSv1 或 IMDSv2) 以該使用者的身分執行任何程序。

```
PS C:\> $blockPrincipal = New-Object -TypeName System.Security.Principal.NTAccount ("NT AUTHORITY\IUSR")
PS C:\> $BlockPrincipalSID =
 $blockPrincipal.Translate([System.Security.Principal.SecurityIdentifier]).Value
PS C:\> $BlockPrincipalSDDL = "D:(A;;CC;;; $BlockPrincipalSID)"
PS C:\> New-NetFirewallRule -DisplayName "Block metadata service from IIS" -Action
 block -Direction out `
-Protocol TCP -RemoteAddress 169.254.169.254 -LocalUser $BlockPrincipalSDDL
```

或者，您可以使用允許規則，考慮只允許存取特定使用者或群組。就安全觀點而言，允許規則可能更加簡單，因為它們會要求您做出關於那些軟體需要存取執行個體中繼資料的決策。如果您使用允許規則，即使您稍後變更執行個體上的軟體或組態，您也不太可能意外地允許軟體存取中繼資料服務 (亦即您並非有意存取)。您也可以使用允許規則來組合群組使用方式，如此您就能新增並從允許的群組移除使用者，而不需要變更防火牆規則。

下列範例會防止以變數 `blockPrincipal` 中指定 OS 群組身分執行的所有程序存取執行個體中繼資料 (在此範例中，Windows 群組 Everyone)，但 `exceptionPrincipal` 中指定的程序除外 (在此範例中，是名為 `trustworthy-users` 的群組)。您必須指定拒絕和允許原則，因為 Windows 防火牆 (與 Linux iptables 中的 `! --uid-owner trustworthy-user` 規則不同) 不會提供快速鍵機制，以透過拒絕所有其他規則而僅允許特定原則 (使用者或群組)。

```
PS C:\> $blockPrincipal = New-Object -TypeName System.Security.Principal.NTAccount
 ("Everyone")
PS C:\> $BlockPrincipalSID =
 $blockPrincipal.Translate([System.Security.Principal.SecurityIdentifier]).Value
PS C:\> $exceptionPrincipal = New-Object -TypeName System.Security.Principal.NTAccount
 ("trustworthy-users")
PS C:\> $ExceptionPrincipalSID =
 $exceptionPrincipal.Translate([System.Security.Principal.SecurityIdentifier]).Value
PS C:\> $PrincipalSDDL = "O:LSD:(D;;CC;;; $ExceptionPrincipalSID)(A;;CC;;;
 $BlockPrincipalSID)"
PS C:\> New-NetFirewallRule -DisplayName "Block metadata service for
 $($blockPrincipal.Value), exception: $($exceptionPrincipal.Value)" -Action block -
 Direction out `
```

```
-Protocol TCP -RemoteAddress 169.254.169.254 -LocalUser $PrincipalSDDL
```

### Note

若要使用本機防火牆規則，您需要採用之前的範例命令，才能滿足您的需求。

## 使用 netsh 規則限制存取

您可以使用 netsh 而考慮封鎖所有軟體，但那些規則較不彈性。

```
C:\> netsh advfirewall firewall add rule name="Block metadata service altogether"
dir=out protocol=TCP remoteip=169.254.169.254 action=block
```

### Note

- 若要使用本機防火牆規則，您需要採用之前的範例命令，才能滿足您的需求。
- 必須從提升權限的命令提示設定 netsh，但不可設定為拒絕或允許特定原則。

## 使用執行個體使用者資料

您可以使用執行個體使用者資料來自訂執行個體。啟動執行個體時，您可以將參數或指令碼儲存為使用者資料。啟動執行個體時，使用者資料中的所有指令碼都會執行。您可以將使用者資料做為執行個體屬性來檢視。您也可以透過執行個體中繼資料服務 (IMDS) 檢視執行個體的使用者資料。

### 考量事項

- 使用者資料或會做為不透明資料處理：您給予的即為您取回的。是否要解譯則取決於執行個體。
- 使用者資料必須為 base64 編碼形式。Amazon EC2 主控台可為您執行 base64 編碼，或是接受 base64 編碼輸入。
- 使用者資料在 base64 編碼之前限制為 16 KB (原始形式)。長度為  $n$  的字串在 base64 編碼之後的大小為  $\text{ceil}(n/3)*4$ 。
- 使用者資料在擷取時必須以 base64 解碼。如果您使用執行個體中繼資料或主控台擷取資料，將會自動為您解碼資料。

- 若您停止執行個體，請修改其使用者資料，然後啟動執行個體，則更新後的使用者資料便不會在您啟動執行個體時自動執行。使用 Windows 執行個體時，您可以設定設定，以便在啟動執行個體時執行一次更新的使用者資料指令碼，或是每次重新開機或啟動執行個體時執行。
- 使用者資料是執行個體屬性。如果您從執行個體建立 AMI，執行個體使用者資料不會包含在 AMI 中。

## 在啟動時指定執行個體使用者資料

您可以在啟動執行個體時指定使用者資料。如需主控台操作指示，請參閱[在啟動時指定執行個體使用者資料](#)。如需使用的 Linux 範例 AWS CLI，請參閱[the section called “使用者資料和 AWS CLI”](#)。如需使用視窗工具的視窗範例 PowerShell，請參閱[the section called “使用者資料和視窗工具 PowerShell”](#)。

## 修改執行個體使用者資料

您可以修改具有 EBS 根磁碟區之執行個體的使用者資料。執行個體必須處於停止狀態。如需主控台操作指示，請參閱[檢視及更新執行個體使用者資料](#)。如需使用的 Linux 範例 AWS CLI，請參閱[修改執行個體屬性](#)。如需使用視窗工具的視窗範例 PowerShell，請參閱[the section called “使用者資料和視窗工具 PowerShell”](#)。

## 從執行個體擷取執行個體使用者資料

### Note

本節中的範例使用 IMDS 的 IPv4 地址：169.254.169.254。如果您要透過 IPv6 地址擷取 EC2 執行個體中繼資料，請確定您啟用並改用 IPv6 地址：[fd00:ec2::254]。IMDS 的 IPv6 地址與 IMDSv2 命令相容。IPv6 位址只能在 [AWS Nitro 系統上建置的執行個體和支援 IPv6 的子網路中存取 \(僅限雙堆疊或 IPv6\)](#)。

若要從執行個體擷取使用者資料，請使用下列 URI。

```
http://169.254.169.254/latest/user-data
```

使用者資料的請求會直接傳回資料 (內容類型 application/octet-stream)。如果執行個體沒有任何使用者資料，則請求會傳回 404 - Not Found。

此範例會傳回以逗點分隔文字提供的使用者資料。

## Linux

### IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/user-data
1234,john,reboot,true | 4512,richard, | 173,,,
```

### IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/user-data
1234,john,reboot,true | 4512,richard, | 173,,,
```

## Windows

### IMDSv2

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/user-data
1234,john,reboot,true | 4512,richard, | 173,,,
```

### IMDSv1

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} `
-Method PUT -Uri http://169.254.169.254/latest/api/token} -Method GET -uri
http://169.254.169.254/latest/user-data
1234,john,reboot,true | 4512,richard, | 173,,,
```

此範例會傳回以指令碼提供的使用者資料。

## Linux

### IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/user-data
#!/bin/bash
yum update -y
service httpd start
chkconfig httpd on
```

### IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/user-data
#!/bin/bash
yum update -y
service httpd start
chkconfig httpd on
```

## Windows

### IMDSv2

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/user-data
<powershell>
$file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
New-Item $file -ItemType file
</powershell>
<persist>>true</persist>
```

### IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/user-data
<powershell>
$file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
```

```
New-Item $file -ItemType file
</powershell>
<persist>>true</persist>
```

## 從電腦中擷取執行個體使用者資料

您可以從自己的電腦擷取執行個體的使用者資料。如需主控台操作指示，請參閱：[檢視及更新執行個體使用者資料](#)。如需使用的範例 AWS CLI，請參閱[使用者資料和 AWS CLI](#)。如需使用 Windows 工具的範例 PowerShell，請參閱[使用者資料和視窗工具 PowerShell](#)。

## 擷取動態資料

若要從執行中的執行個體擷取動態資料，請使用下列 URI。

```
http://169.254.169.254/latest/dynamic/
```

### Note

本節中的範例使用 IMDS 的 IPv4 地址：169.254.169.254。如果您要透過 IPv6 地址擷取 EC2 執行個體的執行個體中繼資料，請確定您啟用並改用 IPv6 地址：[fd00:ec2::254]。IMDS 的 IPv6 地址與 IMDSv2 命令相容。IPv6 位址只能在 [AWS Nitro 系統上建置的執行個體和支援 IPv6 的子網路中存取 \(僅限雙堆疊或 IPv6\)](#)。

此範例會示範如何擷取高層級的執行個體身分分類。

### Linux

#### IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/dynamic/instance-identity/rsa2048
pkcs7
document
signature
dsa2048
```

## IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/dynamic/instance-identity/
rsa2048
pkcs7
document
signature
dsa2048
```

## Windows

### IMDSv2

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-
seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
GET -Uri http://169.254.169.254/latest/dynamic/instance-identity/
document
rsa2048
pkcs7
signature
```

### IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/dynamic/instance-
identity/
document
rsa2048
pkcs7
signature
```

如需動態資料的詳細資訊及如何擷取它的範例，請參閱[執行個體身分文件](#)。

## 執行個體中繼資料分類

執行個體中繼資料分為數種分類。若要擷取執行個體中繼資料，請在請求中指定類別，回應中即會傳回中繼資料。



發佈新類別時，將會使用新版本編號建立新的執行個體中繼資料建置。在下表中，Version when category was released (類別發佈時的版本) 資料欄指定了發佈執行個體中繼資料類別時的建置版本。為避免每次 Amazon EC2 發佈新的執行個體中繼資料建置時都必須更新您的程式碼，請在中繼資料請求中使用 latest，而不是版本編號。如需詳細資訊，請參閱 [取得執行個體中繼資料的可用版本](#)。

Amazon EC2 發行新的執行個體中繼資料類別時，新類別的執行個體中繼資料可能無法用於現有的執行個體。使用建置在 [Nitro 系統](#) 中的執行個體，您只能擷取啟動時可用類別的執行個體中繼資料。對於使用 Xen hypervisor 的執行個體，您可以 [停止然後啟動](#) 執行個體，以更新執行個體可用的類別。

下表列出執行個體中繼資料的分類。下表中的某些類別名稱包含對您的執行個體而言唯一的資料預留位置。例如，*mac* 代表網路介面的 MAC 地址。擷取執行個體中繼資料時，必須使用實際的值取代預留位置。

| 類別                                 | 描述                                                                                                           | 類別發佈時的版本   |
|------------------------------------|--------------------------------------------------------------------------------------------------------------|------------|
| ami-id                             | 用於啟動執行個體的 AMI ID。                                                                                            | 1.0        |
| ami-launch-index                   | 如果您使用相同的 RunInstances 呼叫啟動多個執行個體，此值會指出每個執行個體的啟動順序。第一個啟動的執行個體值為 0。如果您使用 Auto Scaling 或 EC2 叢集啟動執行個體，則此值永遠為 0。 | 1.0        |
| ami-manifest-path                  | 指向 Amazon S3 中 AMI 資訊清單檔案的路徑。若您使用 Amazon EBS 後端 AMI 啟動執行個體，傳回的結果將會是 unknown。                                 | 1.0        |
| ancestor-ami-ids                   | 任何為了建立此 AMI 重新組合之執行個體的 AMI ID。此值只有在 AMI 資訊清單檔案包含 ancestor-amis 鍵時才會存在。                                       | 2007-10-10 |
| autoscaling/target-lifecycle-state | 顯示 Auto Scaling 執行個體正在轉換到的目標 Auto Scaling 生命週期狀態的值。當執行個體在 2022 年 3 月 10 日之後轉換                                | 2021-07-15 |

| 類別                                  | 描述                                                                                                                                                                                                                           | 類別發佈時的版本   |
|-------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
|                                     | <p>到目標生命週期狀態之一時存在。可能的值如下：Detached   InService   Standby   Terminated   Warmed:Hibernated   Warmed:Running   Warmed:Stopped   Warmed:Terminated。請參閱《Amazon EC2 Auto Scaling 使用者指南》中的<a href="#">透過執行個體中繼資料擷取目標生命週期狀態</a>。</p> |            |
| block-device-mapping/ami            | 包含根/開機檔案系統的虛擬裝置。                                                                                                                                                                                                             | 2007-12-15 |
| block-device-mapping/ebs<br>N       | 與任何 Amazon EBS 磁碟區相關聯的虛擬裝置。Amazon EBS 磁碟區只在啟動時或在執行個體上次啟動時存在，才能用於中繼資料。N 表示 Amazon EBS 磁碟區的索引 (例如 ebs1 或 ebs2)。                                                                                                                | 2007-12-15 |
| block-device-mapping/ephemeral<br>N | 用於任何非 NVMe 執行個體存放區磁碟區的虛擬裝置。N 表示每個磁碟區的索引。區塊型設備映射內的執行個體存放區磁碟區數量，可能與執行個體的執行個體存放區磁碟區實際數量不同。執行個體類型決定了執行個體存放區磁碟區可用執行個體的數量。如果區塊型設備映射內執行個體存放區磁碟區的數量，超過可用執行個體，請忽略額外的執行個體存放區磁碟區。                                                       | 2007-12-15 |

| 類別                                                   | 描述                                                                                          | 類別發佈時的版本   |
|------------------------------------------------------|---------------------------------------------------------------------------------------------|------------|
| block-device-mapping/<br>root                        | 與根設備相關聯的虛擬裝置或分割區，或虛擬裝置上的分割區，其中根 (/ 或 C:) 檔案系統與指定的執行個體相關聯。                                   | 2007-12-15 |
| block-device-mapping/<br>swap                        | 與 swap 相關聯的虛擬裝置。並非總是存在。                                                                     | 2007-12-15 |
| elastic-gpus/assoc<br>iations/ <i>elastic-gpu-id</i> | 若有任何連接到執行個體的 Elastic GPU，則包含帶有 Elastic GPU 相關資訊 (包含其 ID 和連線資訊) 的 JSON 字串。                   | 2016-11-30 |
| elastic-inference/<br>associations/ <i>eia-id</i>    | 若有任何連接到執行個體的 Elastic Inference 加速器，則包含帶有 Elastic Inference 加速器的相關資訊 (包含其 ID 和類型) 的 JSON 字串。 | 2018-11-29 |
| events/maintenance/<br>history                       | 如果執行個體有已完成或已取消的維護事件，請讓 JSON 字串包含事件的相關資訊。如需詳細資訊，請參閱 <a href="#">檢視已完成或已取消事件的事件歷史記錄</a> 。     | 2018-08-17 |
| events/maintenance/<br>scheduled                     | 如果執行個體有使用中的維護事件，請讓 JSON 字串包含事件的相關資訊。如需詳細資訊，請參閱 <a href="#">檢視排程事件</a> 。                     | 2018-08-17 |

| 類別                                 | 描述                                                                                                                                                                                            | 類別發佈時的版本   |
|------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| events/recommendations/rebalance   | 針對執行個體發出 EC2 執行個體重新平衡建議通知的大約時間 (UTC)。以下是此類別中繼資料的範例：{"noticeTime": "2020-11-05T08:22:00Z"}。此類別只有在發出通知後才可用。如需詳細資訊，請參閱 <a href="#">EC2 執行個體重新平衡建議</a> 。                                          | 2020-10-27 |
| hostname                           | 如果 EC2 執行個體使用的是 IP 型命名 (IPBN)，則這是執行個體的私有 IPv4 DNS 主機名稱。如果 EC2 執行個體使用的是資源型命名 (RBN)，則這是 RBN。若有多個網路介面，則其代表 eth0 裝置 (裝置編號為 0 的裝置)。如需 IPBN 和 RBN 的詳細資訊，請參閱 <a href="#">Amazon EC2 執行個體主機名稱類型</a> 。 | 1.0        |
| iam/info                           | 如果存在與執行個體相關聯的 IAM 角色，則會包含執行個體設定檔上次更新時間的相關資訊，包括執行個體的 LastUpdated 日期和 InstanceProfile ID。InstanceProfileArn 否則不會存在。                                                                             | 2012-01-12 |
| iam/security-credentials/role-name | 若有和執行個體相關聯的 IAM 角色， <i>role-name</i> 即為角色的名稱，且 <i>role-name</i> 包含與角色相關聯的暫時安全登入資料 (如需詳細資訊，請參閱 <a href="#">從執行個體中繼資料中擷取安全登入資料</a> )。否則不會存在。                                                    | 2012-01-12 |

| 類別                                                         | 描述                                                                                                                                                                              | 類別發佈時的版本   |
|------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| identity-credentials/ec2/info                              | identity-credentials/ec2/security-credentials/ec2-instance 中有關憑證的資訊。                                                                                                            | 2018-05-23 |
| identity-credentials/ec2/security-credentials/ec2-instance | 執行個體身分角色的登入資料，可讓執行個體內部軟體自行識別，AWS 以支援 EC2 執行個體 Connect 和 AWS Systems Manager 預設主機管理組態等功能。這些認證沒有附加政策，因此除了向 AWS 功能識別執行個體之外，它們沒有其他 AWS API 權限。如需詳細資訊，請參閱 <a href="#">執行個體身分角色</a> 。 | 2018-05-23 |
| instance-action                                            | 通知執行個體應重新開機以準備綁定。有效值：none   shutdown   bundle-pending 。                                                                                                                         | 2008-09-01 |
| instance-id                                                | 此執行個體的 ID。                                                                                                                                                                      | 1.0        |
| instance-life-cycle                                        | 此執行個體的購買選項。如需詳細資訊，請參閱 <a href="#">執行個體購買選項</a> 。                                                                                                                                | 2019-10-01 |
| instance-type                                              | 執行個體的類型。如需詳細資訊，請參閱 <a href="#">Amazon EC2 執行個體類型</a> 。                                                                                                                          | 2007-08-29 |
| ipv6                                                       | 執行個體的 IPv6 地址。若有多個網路介面，則其代表 eth0 裝置 (裝置編號為 0 的裝置) 網路介面，以及第一個指派的 IPv6 地址。如果網路介面 [0] 上沒有 IPv6 地址，則不會設定此項目，並會產生 HTTP 404 回應。                                                       | 2021-01-03 |

| 類別                                        | 描述                                                                                                                                                                                                      | 類別發佈時的版本   |
|-------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| kernel-id                                 | 使用此執行個體啟動的核心 ID (若適用的話)。                                                                                                                                                                                | 2008-02-01 |
| local-hostname                            | 若有多個網路介面，則其代表 eth0 裝置 (裝置編號為 0 的裝置)。如果 EC2 執行個體使用的是 IP 型命名 (IPBN)，則這是執行個體的私有 IPv4 DNS 主機名稱。如果 EC2 執行個體使用的是資源型命名 (RBN)，則這是 RBN。如需 IPBN、RBN 和 EC2 執行個體命名的詳細資訊，請參閱 <a href="#">Amazon EC2 執行個體主機名稱類型</a> 。 | 2007-01-19 |
| local-ipv4                                | 執行個體的私有 IPv4 地址。若有多個網路介面，則其代表 eth0 裝置 (裝置編號為 0 的裝置)。如果這是僅限 IPv6 執行個體，則不會設定此項目，並會產生 HTTP 404 回應。                                                                                                         | 1.0        |
| mac                                       | 執行個體的媒體存取控制 (MAC) 地址。若有多個網路介面，則其代表 eth0 裝置 (裝置編號為 0 的裝置)。                                                                                                                                               | 2011-01-01 |
| metrics/vhostmd                           | 不再提供使用。                                                                                                                                                                                                 | 2011-05-01 |
| network/interfaces/macs/mac/device-number | 與該執行個體相關聯的唯一裝置編號。裝置編號對應至裝置名稱；例如，device-number 為 2 即為 eth2 裝置。此類別對應到 Amazon EC2 API 和 AWS CLI 的 EC2 命令使用的 DeviceIndex 和 device-index 欄位。                                                                 | 2011-01-01 |
| network/interfaces/macs/mac/interface-id  | 網路介面的 ID。                                                                                                                                                                                               | 2011-01-01 |

| 類別                                                          | 描述                                                                                                                                                  | 類別發佈時的版本   |
|-------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| network/interfaces/<br>macs/mac/ipv4-associations/public-ip | 與每個公有 IP 地址相關聯並指派給該介面的私有 IPv4 地址。                                                                                                                   | 2011-01-01 |
| network/interfaces/<br>macs/mac/ipv6s                       | 指派給介面的 IPv6 位址。                                                                                                                                     | 2016-06-30 |
| network/interfaces/<br>macs/mac/ipv6-prefix                 | 指派給網路介面的 IPv6 前置詞。                                                                                                                                  |            |
| network/interfaces/<br>macs/mac/local-hostname              | 執行個體的私有 IPv4 DNS 主機名稱。若有多個網路介面，則其代表 eth0 裝置 (裝置編號為 0 的裝置)。如果這是僅限 IPv6 執行個體，則這是資源型名稱。如需 IPBN 和 RBN 的詳細資訊，請參閱 <a href="#">Amazon EC2 執行個體主機名稱類型</a> 。 | 2007-01-19 |
| network/interfaces/<br>macs/mac/local-ipv4s                 | 與介面相關聯的私有 IPv4 地址。如果這是僅限 IPv6 網路介面，則不會設定此項目，並會產生 HTTP 404 回應。                                                                                       | 2011-01-01 |
| network/interfaces/<br>macs/mac/mac                         | 執行個體的 MAC 地址。                                                                                                                                       | 2011-01-01 |
| network/interfaces/<br>macs/ <i>mac</i> /network-card       | 網路卡的索引。某些執行個體類型支援多個網路卡。                                                                                                                             | 2020-11-01 |
| network/interfaces/<br>macs/mac/owner-id                    | 網路介面擁有者的 ID。在多介面環境中，介面僅能由第三方 (例如 Elastic Load Balancing) 連接。介面上的流量一律由介面的擁有者付費。                                                                      | 2011-01-01 |

| 類別                                                          | 描述                                                                                                                                                                               | 類別發佈時的版本   |
|-------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| network/interfaces/<br>macs/mac/public-<br>hostname         | 介面的公有 DNS (IPv4)。只在 enableDnsHostnames 屬性設為 true 時才會傳回此類別。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的 <a href="#">VPC 的 DNS 屬性</a> 。如果執行個體只有公有 IPv6 地址且沒有公有 IPv4 地址，則不會設定此項目，並會產生 HTTP 404 回應。 | 2011-01-01 |
| network/interfaces/<br>macs/mac/public-ipv4s                | 與界面相關聯的公有 IP 地址與彈性 IP 地址。執行個體上可能會有多個 IPv4 地址。                                                                                                                                    | 2011-01-01 |
| network/interfaces/<br>macs/mac/security-<br>groups         | 網路介面所屬的安全群組。                                                                                                                                                                     | 2011-01-01 |
| network/interfaces/<br>macs/mac/security-<br>group-ids      | 網路介面所屬的安全群組 ID。                                                                                                                                                                  | 2011-01-01 |
| network/interfaces/<br>macs/mac/subnet-id                   | 介面所在的子網 ID。                                                                                                                                                                      | 2011-01-01 |
| network/interfaces/<br>macs/mac/subnet-ipv4-<br>cidr-block  | 介面所在子網的 IPv4 CIDR 區塊。                                                                                                                                                            | 2011-01-01 |
| network/interfaces/<br>macs/mac/subnet-ipv6-<br>cidr-blocks | 介面所在子網的 IPv6 CIDR 區塊。                                                                                                                                                            | 2016-06-30 |
| network/interfaces/<br>macs/mac/vpc-id                      | 介面所在的 VPC ID。                                                                                                                                                                    | 2011-01-01 |



| 類別                                               | 描述                                                          | 類別發佈時的版本   |
|--------------------------------------------------|-------------------------------------------------------------|------------|
| network/interfaces/macs/mac/vpc-ipv4-cidr-block  | VPC 的主要 IPv4 CIDR 區塊。                                       | 2011-01-01 |
| network/interfaces/macs/mac/vpc-ipv4-cidr-blocks | VPC 的 IPv4 CIDR 區塊。                                         | 2016-06-30 |
| network/interfaces/macs/mac/vpc-ipv6-cidr-blocks | 介面所在 VPC 的 IPv6 CIDR 區塊。                                    | 2016-06-30 |
| placement/availability-zone                      | 啟動執行個體的可用區域。                                                | 2008-02-01 |
| placement/availability-zone-id                   | 啟動的執行個體所在的靜態可用區域 ID。可用區域 ID 會在各帳戶間保持一致。但是，它可能與可用區域不同，視帳戶而異。 | 2019-10-01 |
| placement/group-name                             | 在啟動執行個體的位置放置群組名稱。                                           | 2020-08-24 |
| placement/host-id                                | 啟動執行個體位置的主機 ID。僅適用於 專用執行個體。                                 | 2020-08-24 |
| placement/partition-number                       | 啟動執行個體位置的分割區編號。                                             | 2020-08-24 |
| placement/region                                 | 啟動執行處理所在的 AWS 區域。                                           | 2020-08-24 |
| product-codes                                    | AWS Marketplace 與例證相關聯的產品代碼 (如果有的話)。                        | 2007-03-01 |

| 類別                        | 描述                                                                                                                                                                                 | 類別發佈時的版本   |
|---------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| public-hostname           | 執行個體的公有 DNS (IPv4)。只在 enableDnsHostnames 屬性設為 true 時才會傳回此類別。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的 <a href="#">VPC 的 DNS 屬性</a> 。如果執行個體只有公有 IPv6 地址且沒有公有 IPv4 地址，則不會設定此項目，並會產生 HTTP 404 回應。 | 2007-01-19 |
| public-ipv4               | 公有 IPv4 地址。若彈性 IP 地址與執行個體相關聯，傳回的值即為彈性 IP 地址。                                                                                                                                       | 2007-01-19 |
| public-keys/0/openssh-key | 公有金鑰。只有在執行個體啟動階段提供時才可用。                                                                                                                                                            | 1.0        |
| ramdisk-id                | 啟動階段指定之 RAM 磁碟的 ID (若適用的話)。                                                                                                                                                        | 2007-10-10 |
| reservation-id            | 保留 ID。                                                                                                                                                                             | 1.0        |
| security-groups           | 套用至執行個體的安全群組名稱。<br><br>在啟動之後，您可以變更執行個體的安全群組。這類變更除了會在此反映，也會在 network/interfaces/mac/mac/security-groups 中反映。                                                                        | 1.0        |
| services/domain           | 區域資 AWS 源的網域。                                                                                                                                                                      | 2014-02-25 |

| 類別                    | 描述                                                                                                                                                                                                    | 類別發佈時的版本   |
|-----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| services/partition    | 資源所在的分割區。對於標準 AWS 區域，分割區為aws。如果您有其他分割區的資源，則該分割區為 <i>aws-partitionname</i> 。例如，中國 (北京) 區域的資源分割區，即為 aws-cn。                                                                                             | 2015-10-20 |
| spot/instance-action  | 動作 (休眠、停止或終止) 和動作發生的約略時間 (UTC)。只有在 Spot 執行個體已標記為休眠、停止或終止時，才會顯示此項目。如需詳細資訊，請參閱 <a href="#">instance-action</a> 。                                                                                        | 2016-11-15 |
| spot/termination-time | Spot 執行個體的作業系統接收關機訊號的近似時間 (UTC)。此項目只有在 Spot 執行個體已由 Amazon EC2 標記為終止時，才會存在且包含時間值 (例如，2015-01-05T18:02:00Z)。若您自行終止 Spot 執行個體，則 termination-time 項目便不會設為時間。如需詳細資訊，請參閱 <a href="#">termination-time</a> 。 | 2014-11-05 |
| tags/instance         | 與執行個體相關聯的執行個體標籤。僅當您明確允許存取執行個體中繼資料中的標籤時才可用。如需詳細資訊，請參閱 <a href="#">允許存取執行個體中繼資料中的標籤</a> 。                                                                                                               | 2021-03-23 |

## 動態資料分類

下表列出動態資料的分類。

| 類別                          | 描述                                                                     | 類別發佈時的版本   |
|-----------------------------|------------------------------------------------------------------------|------------|
| fws/instance-monitoring     | 顯示客戶是否已在中啟用詳細一分鐘監控的 CloudWatch 值。有效值：enabled   disabled                | 2009-04-04 |
| instance-identity/document  | 包含執行個體屬性的 JSON，例如 instance-id、私有 IP 地址等。請參閱 <a href="#">執行個體身分文件</a> 。 | 2009-04-04 |
| instance-identity/pkcs7     | 用來使用簽章確認文件的真確性和其內容。請參閱 <a href="#">執行個體身分文件</a> 。                      | 2009-04-04 |
| instance-identity/signature | 可由其他廠商用來確認其來源和真確性的資料。請參閱 <a href="#">執行個體身分文件</a> 。                    | 2009-04-04 |

## Linux 實例：AMI 啟動指數值

此範例示範如何使用使用者資料和執行個體中繼資料來設定 Linux 執行個體。

### Note

本節中的範例使用 IMDS 的 IPv4 地址：169.254.169.254。如果您要透過 IPv6 地址擷取 EC2 執行個體的執行個體中繼資料，請確定您啟用並改用 IPv6 地址：[fd00:ec2::254]。IMDS 的 IPv6 地址與 IMDSv2 命令相容。IPv6 位址只能在 [AWS Nitro 系統上建置的執行個體](#)和[支援 IPv6 的子網路中存取 \(僅限雙堆疊或 IPv6\)](#)。

Alice 想要啟動四個她最愛的資料庫 AMI 的執行個體，其中第一個為原始執行個體，其餘三個則為複本。當她啟動這些執行個體時，她想要為每個複本新增關於複寫策略的使用者資料。她知道資料將會在四個執行個體上可用，因此她需要使用允許每個執行個體識別其適用部分的方式架構使用者資料。她可以使用 ami-launch-index 執行個體中繼資料值執行此作業，每一個執行個體的該值都是唯一的。如果您同時啟動一個以上的執行個體，ami-launch-index 表示執行個體的啟動順序。第一個啟動的執行個體的值為 0。

下列是 Alice 建構的使用者資料。

```
replicate-every=1min | replicate-every=5min | replicate-every=10min
```

`replicate-every=1min` 資料定義第一個複本的組態，`replicate-every=5min` 定義第二個複本的組態，以此類推。Alice 決定以 ASCII 字串提供此資料，其中以直立線符號 (|) 分隔不同執行個體的資料。

Alice 使用 [run-instances](#) 命令啟動四個執行個體，指定使用者資料。

```
aws ec2 run-instances \
 --image-id ami-0abcdef1234567890 \
 --count 4 \
 --instance-type t2.micro \
 --user-data "replicate-every=1min | replicate-every=5min | replicate-every=10min"
```

在啟動它們之後，所有執行個體都會擁有使用者資料的複本，並且常見的中繼資料會顯示於此：

- AMI ID : ami-0abcdef1234567890
- 保留 ID : r-1234567890abcabc0
- 公有金鑰 : 無
- 安全群組名稱 : 預設
- 執行個體類型 : t2.micro

但是，每個執行個體都擁有特定的唯一中繼資料。

#### 執行個體 1

| 中繼資料             | Value                                    |
|------------------|------------------------------------------|
| instance-id      | i-1234567890abcdef0                      |
| ami-launch-index | 0                                        |
| public-hostname  | ec2-203-0-113-25.compute-1.amazonaws.com |
| public-ipv4      | 67.202.51.223                            |
| local-hostname   | ip-10-251-50-12.ec2.internal             |
| local-ipv4       | 10.251.50.35                             |

## 執行個體 2

| 中繼資料             | Value                                     |
|------------------|-------------------------------------------|
| instance-id      | i-0598c7d356eba48d7                       |
| ami-launch-index | 1                                         |
| public-hostname  | ec2-67-202-51-224.compute-1.amazonaws.com |
| public-ipv4      | 67.202.51.224                             |
| local-hostname   | ip-10-251-50-36.ec2.internal              |
| local-ipv4       | 10.251.50.36                              |

## 執行個體 3

| 中繼資料             | Value                                     |
|------------------|-------------------------------------------|
| instance-id      | i-0ee992212549ce0e7                       |
| ami-launch-index | 2                                         |
| public-hostname  | ec2-67-202-51-225.compute-1.amazonaws.com |
| public-ipv4      | 67.202.51.225                             |
| local-hostname   | ip-10-251-50-37.ec2.internal              |
| local-ipv4       | 10.251.50.37                              |

## 執行個體 4

| 中繼資料             | Value                                     |
|------------------|-------------------------------------------|
| instance-id      | i-1234567890abcdef0                       |
| ami-launch-index | 3                                         |
| public-hostname  | ec2-67-202-51-226.compute-1.amazonaws.com |

| 中繼資料           | Value                        |
|----------------|------------------------------|
| public-ipv4    | 67.202.51.226                |
| local-hostname | ip-10-251-50-38.ec2.internal |
| local-ipv4     | 10.251.50.38                 |

Alice 可以使用 `ami-launch-index` 值判斷使用者資料中的哪一個部分適用於特定執行個體。

1. 她連線到其中一個執行個體，擷取該執行個體的 `ami-launch-index`，確保它是其中一個複本：

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/meta-data/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/ami-launch-index
2
```

對於後續步驟，IMDSv2 請求會從之前的 IMDSv2 命令來使用儲存的字符，但前提是字符並未過期。

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/ami-launch-index
2
```

2. 她將 `ami-launch-index` 儲存為變數。

IMDSv2

```
[ec2-user ~]$ ami_launch_index=`curl -H "X-aws-ec2-metadata-token: $TOKEN" \
http://169.254.169.254/latest/meta-data/ami-launch-index`
```

IMDSv1

```
[ec2-user ~]$ ami_launch_index=`curl http://169.254.169.254/latest/meta-data/ami-launch-index`
```

3. 她將使用者資料儲存為變數。

## IMDSv2

```
[ec2-user ~]$ user_data=`curl -H "X-aws-ec2-metadata-token: $TOKEN"
http://169.254.169.254/latest/user-data`
```

## IMDSv1

```
[ec2-user ~]$ user_data=`curl http://169.254.169.254/latest/user-data`
```

4. 最後，Alice 使用 `cut` 命令擷取適用於該執行個體的使用者資料部分。

## IMDSv2

```
[ec2-user ~]$ echo $user_data | cut -d"|" -f"$ami_launch_index"
replicate-every=5min
```

## IMDSv1

```
[ec2-user ~]$ echo $user_data | cut -d"|" -f"$ami_launch_index"
replicate-every=5min
```

## 執行個體身分文件

您啟動的每個執行個體都有一個提供執行個體本身相關資訊的執行個體身分文件。您可以使用執行個體身分文件來驗證執行個體的屬性。

執行個體停止、開始、重新開始或啟動時，會產生執行個體身分文件。執行個體身分文件會透過執行個體中繼資料服務 (IMDS) 公開 (以純文字 JSON 格式)。IPv4 地址 `169.254.169.254` 是 link-local 地址且僅在執行個體中有效。如需詳細資訊，請參閱維基百科上的 [Link-local address](#)。IPv6 地址 `[fd00:ec2::254]` 是唯一本地地址且僅在執行個體中有效。如需詳細資訊，請參閱維基百科上的 [Unique local address](#) (唯一本地地址)。

### Note

本節中的範例使用 IMDS 的 IPv4 地址：`169.254.169.254`。如果您要透過 IPv6 地址擷取 EC2 執行個體的執行個體中繼資料，請確定您啟用並改用 IPv6 地址：`[fd00:ec2::254]`。IMDS 的 IPv6 地址與 IMDSv2 命令相容。IPv6 位址只能在 [AWS Nitro 系統上建置的執行個體](#) 和 [支援 IPv6 的子網路中存取 \(僅限雙堆疊或 IPv6\)](#)。



您可以隨時從執行中的執行個體擷取 執行個體身分文件。執行個體身分文件 包含了下列資訊：

| 資料                      | 描述                                         |
|-------------------------|--------------------------------------------|
| accountId               | 啟動執行個體的 AWS 帳戶 ID。                         |
| architecture            | 用於啟動執行個體 (i386   x86_64   arm64) 的 AMI 架構。 |
| availabilityZone        | 執行個體執行的可用區域。                               |
| billingProducts         | 執行個體的計費產品。                                 |
| devpayProductCodes      | 已廢除。                                       |
| imageId                 | 用來啟動執行個體的 AMI ID。                          |
| instanceId              | 執行個體的 ID。                                  |
| instanceType            | 執行個體的執行個體類型。                               |
| kernelId                | 與執行個體相關聯的核心 ID (如適用)。                      |
| marketplaceProductCodes | 用來啟動執行個體的 AMI AWS Marketplace 產品代碼。        |
| pendingTime             | 執行個體啟動的日期和時間。                              |
| privateIp               | 執行個體的私有 IPv4 地址。                           |
| ramdiskId               | 與此執行個體相關聯的 RAM 磁碟 ID (如適用)。                |
| region                  | 執行個體執行時所在區域。                               |
| version                 | 執行個體身分文件 格式的版本。                            |

## 擷取純文字 執行個體身分文件

### 要擷取純文字 執行個體身分文件

連接至執行個體並執行下列命令。

## Linux

### IMDSv2

```
$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-
metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/dynamic/
instance-identity/document
```

### IMDSv1

```
$ curl http://169.254.169.254/latest/dynamic/instance-identity/document
```

## Windows

### IMDSv2

```
PS C:\> [string]$token = (Invoke-WebRequest -Method Put -Headers @{'X-aws-
ec2-metadata-token-ttl-seconds' = '21600'} http://169.254.169.254/latest/api/
token).Content
```

```
PS C:\> (Invoke-WebRequest -Headers @{'X-aws-ec2-metadata-token' = $Token}
http://169.254.169.254/latest/dynamic/instance-identity/document).Content
```

### IMDSv1

```
PS C:\> (Invoke-WebRequest http://169.254.169.254/latest/dynamic/instance-identity/
document).Content
```

下列為範例輸出。

```
{
 "devpayProductCodes" : null,
 "marketplaceProductCodes" : ["1abc2defghijklm3nopqrs4tu"],
 "availabilityZone" : "us-west-2b",
 "privateIp" : "10.158.112.84",
```

```
"version" : "2017-09-30",
"instanceId" : "i-1234567890abcdef0",
"billingProducts" : null,
"instanceType" : "t2.micro",
"accountId" : "123456789012",
"imageId" : "ami-5fb8c835",
"pendingTime" : "2016-11-19T16:32:11Z",
"architecture" : "x86_64",
"kernelId" : null,
"ramdiskId" : null,
"region" : "us-west-2"
}
```

## 驗證 執行個體身分文件

如果您打算將 執行個體身分文件 的內容用於重要目的，您應該在使用它之前驗證其內容與真實性。

純文字 執行個體身分文件 會伴隨著三個雜湊和加密的簽章。您可以使用這些簽章來驗證 執行個體身分文件 的來源和真實性及其包含的資訊。提供的簽章如下：

- Base64 編碼的簽章 – 此為使用 RSA 金鑰對加密之執行個體身分文件的 base64 編碼 SHA256 雜湊。
- PKCS7 簽章 – 此為使用 DSA 金鑰對加密的 執行個體身分文件 SHA1 雜湊。
- RSA-2048 簽章 – 此為使用 RSA-2048 金鑰對加密的執行個體身分文件 SHA256 雜湊。

每個簽章都可以在執行個體中繼資料的不同端點上使用。您可以根據您的雜湊和加密需求使用任何一個簽章。若要驗證簽章，您必須使用對應的 AWS 公用憑證。

下列主題提供使用每個簽章來驗證 執行個體身分文件 的詳細步驟。

- [使用 PKCS7 簽章驗證 執行個體身分文件](#)
- [使用 Base64 編碼簽章驗證 執行個體身分文件](#)
- [使用 RSA-2048 簽章驗證 執行個體身分文件](#)

## 使用 PKCS7 簽章驗證 執行個體身分文件

本主題說明如何使用 PKCS7 簽章和 AWS DSA 公用憑證來驗證執行個體身分識別文件。

## Linux 執行個體

### 使用 PKCS7 簽名和 AWS DSA 公用憑證驗證執行個體身分證明文件

1. 連線到執行個體。
2. 從執行個體中繼資料擷取 PKCS7 簽章，並將其與所需的標頭和頁尾新增至名為 `pkcs7` 的新檔案。根據執行個體所使用的 IMDS 版本，使用下列其中一個命令。

#### IMDSv2

```
$ echo "-----BEGIN PKCS7-----" >> pkcs7 \
&& TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-
metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
dynamic/instance-identity/pkcs7 >> pkcs7 \
&& echo "" >> pkcs7 \
&& echo "-----END PKCS7-----" >> pkcs7
```

#### IMDSv1

```
$ echo "-----BEGIN PKCS7-----" >> pkcs7 \
&& curl -s http://169.254.169.254/latest/dynamic/instance-identity/pkcs7
>> pkcs7 \
&& echo "" >> pkcs7 \
&& echo "-----END PKCS7-----" >> pkcs7
```

3. 在 [AWS 公開憑證](#) 尋找您所在區域的 DSA 公有憑證，並將內容新增至名為 `certificate` 的新檔案。
4. 使用 OpenSSL `smime` 命令以驗證簽章。包含指出需要驗證簽章的 `-verify` 選項，以及指出憑證不需要驗證的 `-noverify` 選項。

```
$ openssl smime -verify -in pkcs7 -inform PEM -certfile certificate -noverify | tee
document
```

如果簽章有效，則會出現 `Verification successful` 訊息。

此命令也會將執行個體身分文件內容寫入名為 `document` 的新檔案。您可以使用下列命令，將來自執行個體中繼資料的執行個體身分文件內容與此檔案內容比較。

```
$ openssl dgst -sha256 < document
```

```
$ curl -s -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/dynamic/instance-identity/document | openssl dgst -sha256
```

如果無法驗證簽章，請聯絡 AWS Support。

## Windows 執行個體

### 必要條件

此程序需要 System.Security Microsoft .NET Core 類別。若要將類別新增至您的 PowerShell 工作階段，請執行下列命令。

```
PS C:\> Add-Type -AssemblyName System.Security
```

#### Note

該命令僅將類添加到當前 PowerShell 會話中。如果您啟動新的工作階段，您必須再次執行命令。

## 使用 PKCS7 簽名和 AWS DSA 公用憑證驗證執行個體身分證明文件

1. 連線到執行個體。
2. 從執行個體中繼資料擷取 PKCS7 簽章，將其轉換為位元組陣列，然後再新增至名為 `$Signature` 的變數。根據執行個體所使用的 IMDS 版本，使用下列其中一個命令。

### IMDSv2

```
PS C:\> [string]$token = (Invoke-WebRequest -Method Put -Headers @{'X-aws-ec2-metadata-token-ttl-seconds' = '21600'} http://169.254.169.254/latest/api/token).Content
```

```
PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest -Headers @{'X-aws-ec2-metadata-token' = $Token} http://169.254.169.254/latest/dynamic/instance-identity/pkcs7).Content)
```

## IMDSv1

```
PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest
http://169.254.169.254/latest/dynamic/instance-identity/pkcs7).Content)
```

3. 從執行個體中繼資料擷取純文字執行個體身分文件，將其轉換為位元組陣列，然後新增至名為 `$Document` 的變數。根據執行個體所使用的 IMDS 版本，使用下列其中一個命令。

## IMDSv2

```
PS C:\> $Document = [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest -Headers
@{'X-aws-ec2-metadata-token' = $Token} http://169.254.169.254/latest/dynamic/
instance-identity/document).Content)
```

## IMDSv1

```
PS C:\> $Document = [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest
http://169.254.169.254/latest/dynamic/instance-identity/document).Content)
```

4. 在 [AWS 公開憑證](#) 尋找您所在區域的 DSA 公有憑證，並將內容新增至名為 `certificate.pem` 的新檔案。
5. 從憑證檔案擷取憑證，並將其存放在名為 `$Store` 的變數中。

```
PS C:\> $Store =
[Security.Cryptography.X509Certificates.X509Certificate2Collection]::new([Security.Cryptography.CertificatePath
certificate.pem]))
```

6. 驗證簽章。

```
PS C:\> $SignatureDocument = [Security.Cryptography.Pkcs.SignedCms]::new()
```

```
PS C:\> $SignatureDocument.Decode($Signature)
```

```
PS C:\> $SignatureDocument.CheckSignature($Store, $true)
```

如果簽章有效，則命令不會傳回任何輸出。如果無法驗證簽章，則命令會傳回 `Exception calling "CheckSignature" with "2" argument(s): "Cannot find the original signer.` 如果無法驗證您的簽章，請聯絡 AWS Support。

## 7. 驗證執行個體身分文件的內容。

```
PS C:
\> [Linq.Enumerable]::SequenceEqual($SignatureDocument.ContentInfo.Content, $Document)
```

如果執行個體身分文件的內容有效，則命令會傳回 True。如果無法驗證執行個體身分文件，請聯絡 AWS Support。

## 使用 Base64 編碼簽章驗證 執行個體身分文件

本主題說明如何使用 base64 編碼的簽章和 AWS RSA 公用憑證來驗證執行個體身分識別文件。

### Linux 執行個體

使用 base64 編碼的簽名和 AWS RSA 公用憑證驗證執行個體身分識別文件

1. 連線到執行個體。
2. 從執行個體中繼資料擷取 base64 編碼的簽章，將其轉換為二進位，然後新增至名為 signature 的檔案。根據執行個體所使用的 IMDS 版本，使用下列其中一個命令。

#### IMDSv2

```
$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/dynamic/instance-identity/signature | base64 -d >> signature
```

#### IMDSv1

```
$ curl -s http://169.254.169.254/latest/dynamic/instance-identity/signature |
base64 -d >> signature
```

3. 從執行個體中繼資料擷取純文字 執行個體身分文件，並新增至名為 document 的檔案。根據執行個體所使用的 IMDS 版本，使用下列其中一個命令。

#### IMDSv2

```
$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
```

```
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/dynamic/instance-identity/document >> document
```

## IMDSv1

```
$ curl -s http://169.254.169.254/latest/dynamic/instance-identity/document >> document
```

4. 在 [AWS 公開憑證](#) 尋找您所在區域的 RSA 公有憑證，並將內容新增至名為 `certificate` 的新檔案。
5. 從 AWS RSA 公開憑證擷取公開金鑰，並將其儲存到名為 `key` 的檔案中。

```
$ openssl x509 -pubkey -noout -in certificate >> key
```

6. 使用 OpenSSL `dgst` 命令來驗證 執行個體身分文件。

```
$ openssl dgst -sha256 -verify key -signature signature document
```

如果簽章有效，則會出現 `Verification successful` 訊息。

此命令也會將執行個體身分文件內容寫入名為 `document` 的新檔案。您可以使用下列命令，將來自執行個體中繼資料的執行個體身分文件內容與此檔案內容比較。

```
$ openssl dgst -sha256 < document
```

```
$ curl -s -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/dynamic/instance-identity/document | openssl dgst -sha256
```

如果無法驗證簽章，請聯絡 AWS Support。

## Windows 執行個體

使用 base64 編碼的簽名和 AWS RSA 公用憑證驗證執行個體身分識別文件

1. 連線到執行個體。
2. 從執行個體中繼資料擷取 base64 編碼的簽章，將其轉換為位元組陣列，然後新增至名為 `$Signature` 的變數。根據執行個體所使用的 IMDS 版本，使用下列其中一個命令。



## IMDSv2

```
PS C:\> [string]$token = (Invoke-WebRequest -Method Put -Headers @{'X-aws-ec2-metadata-token-ttl-seconds' = '21600'} http://169.254.169.254/latest/api/token).Content
```

```
PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest -Headers @{'X-aws-ec2-metadata-token' = $Token} http://169.254.169.254/latest/dynamic/instance-identity/signature).Content)
```

## IMDSv1

```
PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest http://169.254.169.254/latest/dynamic/instance-identity/signature).Content)
```

3. 從執行個體中繼資料擷取純文字執行個體身分文件，將其轉換為位元組陣列，然後新增至名為 `$Document` 的變數。根據執行個體所使用的 IMDS 版本，使用下列其中一個命令。

## IMDSv2

```
PS C:\> $Document = [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest -Headers @{'X-aws-ec2-metadata-token' = $Token} http://169.254.169.254/latest/dynamic/instance-identity/document).Content)
```

## IMDSv1

```
PS C:\> $Document = [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest http://169.254.169.254/latest/dynamic/instance-identity/document).Content)
```

4. 在 [AWS 公開憑證](#) 尋找您所在區域的 RSA 公有憑證，並將內容新增至名為 `certificate.pem` 的新檔案。
5. 驗證執行個體身分文件。

```
PS C:\> [Security.Cryptography.X509Certificates.X509Certificate2]::new((Resolve-Path certificate.pem)).PublicKey.Key.VerifyData($Document, 'SHA256', $Signature)
```

如果簽章有效，則命令會傳回 `True`。如果無法驗證簽章，請聯絡 AWS Support。

## 使用 RSA-2048 簽章驗證 執行個體身分文件

本主題說明如何使用 RSA-2048 簽名和 AWS RSA-2048 公用憑證來驗證執行個體身分識別文件。

### Linux 執行個體

使用 RSA-2048 簽名和 AWS RSA-2048 公開憑證來驗證執行個體身份證明文件

1. 連線到執行個體。
2. 從執行個體中繼資料擷取 RSA-2048 簽章，並將其與所需的標頭和頁尾新增至名為 `rsa2048` 的檔案。根據執行個體所使用的 IMDS 版本，使用下列其中一個命令。

#### IMDSv2

```
$ echo "-----BEGIN PKCS7-----" >> rsa2048 \
&& TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-
metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
dynamic/instance-identity/rsa2048 >> rsa2048 \
&& echo "" >> rsa2048 \
&& echo "-----END PKCS7-----" >> rsa2048
```

#### IMDSv1

```
$ echo "-----BEGIN PKCS7-----" >> rsa2048 \
&& curl -s http://169.254.169.254/latest/dynamic/instance-identity/rsa2048
>> rsa2048 \
&& echo "" >> rsa2048 \
&& echo "-----END PKCS7-----" >> rsa2048
```

3. 在 [AWS 公開憑證](#) 尋找您所在區域的 RSA-2048 公有憑證，並將內容新增至名為 `certificate` 的新檔案。
4. 使用 OpenSSL `smime` 命令以驗證簽章。包含指出需要驗證簽章的 `-verify` 選項，以及指出憑證不需要驗證的 `-noverify` 選項。

```
$ openssl smime -verify -in rsa2048 -inform PEM -certfile certificate -noverify |
tee document
```

如果簽章有效，則會出現 `Verification successful` 訊息。如果無法驗證簽章，請聯絡 AWS Support。

## Windows 執行個體

### 必要條件

此程序需要 System.Security Microsoft .NET Core 類別。若要將類別新增至您的 PowerShell 工作階段，請執行下列命令。

```
PS C:\> Add-Type -AssemblyName System.Security
```

#### Note

該命令僅將類添加到當前 PowerShell 會話中。如果您啟動新的工作階段，您必須再次執行命令。

使用 RSA-2048 簽名和 AWS RSA-2048 公開憑證來驗證執行個體身份證明文件

1. 連線到執行個體。
2. 從執行個體中繼資料擷取 RSA-2048 簽章，將其轉換為位元組陣列，然後再新增至名為 `$Signature` 的變數。根據執行個體所使用的 IMDS 版本，使用下列其中一個命令。

#### IMDSv2

```
PS C:\> [string]$token = (Invoke-WebRequest -Method Put -Headers @{'X-aws-ec2-metadata-token-ttl-seconds' = '21600'} http://169.254.169.254/latest/api/token).Content
```

```
PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest -Headers @{'X-aws-ec2-metadata-token' = $Token} http://169.254.169.254/latest/dynamic/instance-identity/rsa2048).Content)
```

#### IMDSv1

```
PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest http://169.254.169.254/latest/dynamic/instance-identity/rsa2048).Content)
```

3. 從執行個體中繼資料擷取純文字執行個體身分文件，將其轉換為位元組陣列，然後新增至名為 `$Document` 的變數。根據執行個體所使用的 IMDS 版本，使用下列其中一個命令。

## IMDSv2

```
PS C:\> $Document = [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest -Headers
@{'X-aws-ec2-metadata-token' = $Token} http://169.254.169.254/latest/dynamic/
instance-identity/document).Content)
```

## IMDSv1

```
PS C:\> $Document = [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest
http://169.254.169.254/latest/dynamic/instance-identity/document).Content)
```

4. 在 [AWS 公開憑證](#) 尋找您所在區域的 RSA-2048 公有憑證，並將內容新增至名為 `certificate.pem` 的新檔案。
5. 從憑證檔案擷取憑證，並將其存放在名為 `$Store` 的變數中。

```
PS C:\> $Store =
[Security.Cryptography.X509Certificates.X509Certificate2Collection]::new([Security.Cryptog
Path certificate.pem]))
```

6. 驗證簽章。

```
PS C:\> $SignatureDocument = [Security.Cryptography.Pkcs.SignedCms]::new()
```

```
PS C:\> $SignatureDocument.Decode($Signature)
```

```
PS C:\> $SignatureDocument.CheckSignature($Store, $true)
```

如果簽章有效，則命令不會傳回任何輸出。如果無法驗證簽章，則命令會傳回 `Exception calling "CheckSignature" with "2" argument(s): "Cannot find the original signer.` 如果無法驗證您的簽章，請聯絡 AWS Support。

7. 驗證執行個體身分文件的內容。

```
PS C:
> [Linq.Enumerable]::SequenceEqual($SignatureDocument.ContentInfo.Content, $Document)
```

如果執行個體身分文件的內容有效，則命令會傳回 `True`。如果無法驗證執行個體身分文件，請聯絡 AWS Support。

## AWS 公開憑證

下列 AWS 公用憑證可用來驗證執行個體身分識別文件的內容，如下列主題所述：

- [使用 PKCS7 簽章進行驗證](#)
- [使用 Base64 編碼簽章進行驗證](#)
- [使用 RSA-2048 簽章進行驗證](#)

請依據您的所在區域以及您正在使用的驗證程序，確定您使用正確的憑證。如果您正在驗證 PKCS7 簽章，請使用 DSA 憑證。如果您正在驗證 base6 編碼簽章，請使用 RSA 憑證。如果您正在驗證 RSA-2048 簽章，請使用 RSA-2048 憑證。

展開下方的每個區域，以檢視特定區域的憑證。

美國東部 (俄亥俄) — us-east-2

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgqhkJ00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEyBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQKKEudBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJAMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEyBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQKKEudBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbcwggEsBgcqhkJ00AQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5006kK/n1Lz11r7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nvwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MbcJl/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buyCU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKml1qx41lHW
MXrs3IgiB6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLeAwLAIUWXB1k40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAOqgAwIBAgIUUJTc+h0U+8Gk3J1qsX438Dk5c58wDQYJKoZIhvcNAQEL
```

```

BQAwxDELMAkGA1UEBhMCMVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBxZWlGU2Vydm1jZXMgTEdD
MB4XDTE0MDQyOTE3MTE0VowXDTI5MDQyODE3MTE0VowXDELMAkGA1UEBhMCMVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBxZWlGU2Vydm1jZXMgTEdDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJ0+eIB
UqPfQG09kZ1wpWpmy08bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXk3HEnf0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUuizvtUF2
UTgwZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWdwUuizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTA1VTMRkwFwYDVQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQHEwdT
ZWf0dGx1MSAwHgYDVQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUvJTc+h0U
+8Gk3J1qsX438Dk5c58wEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQAYwJQaVNWJqW0R0T0xV0SoN1GLk9x9kKEuN67RN9CLin4dA97qa7Mr5W4P
FZ6vnh5Cj0hQBRXV9xJUeYSdqVItNAUfK/fEzDdjf1nUfP1Q30J49u6CV01NoJ9m
usvY9kWcV46dqn2bk2MyfTTgvmepP8fiMRPxxnVRkSz1ldP5Fg==
-----END CERTIFICATE-----

```

## RSA-2048

```

-----BEGIN CERTIFICATE-----
MIIIEjCCAvqgAwIBAgIJAM07oeX4xevdMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQHEwdTZWf0
dGx1MSAwHgYDVQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNjA2MTAx
MjU4MThaGA8yMTk1MTEExNDEyNTgxOFowXDELMAkGA1UEBhMCMVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWlGU2Vydm1jZXMgTEdMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA6v6kGMnRmFDLxBEqXzP4npl65000kmQ7w8YXQygSdmNIoScGSU5wfh9
mZdcvCxCdXgALFsFqPvH8fqiE9ttI0fEfuZvH0s8wUsIdKr0Zz0MjSx3cik4tKET
ch0EKfMnzK0gDBavraCDeX1rUDU0Rg7HFqNA0ry3uqDmnqtK0XC9GenS3z/7ebJ
fIBEPAAm5oYMFvPx6M6St77WdNE8wEU8SuerQughiMVx9kMB07imeVHBiELbMQ0N
lwSWRL/61fA02keGSTfSp/0m3u+lesf2VwVFhqIJs+JbsEscPx0kIRlzy8mGd/JV
ONb/DQpTedzUKLgXbw7Kt03HTG9iXQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU2CTGYE5fTjx7gQXzdZSGPEWAJY4wgY4GA1UdIwSBhjCBg4AU2CTG
YE5fTjx7gQXzdZSGPEWAJY6hYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQIEExB
XNoaW5ndG9uIFN0YXR1MRAwDgYDVQHEwdTZWf0dGx1MSAwHgYDVQKExdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQ4IJAM07oeX4xevdMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBANDqkIpVypr2PveqUsAKke1wKC0Suw1UmH9k
xX1/VRoHbrI/UznrXtPQOPMmHA2LKSTedwsJuorUn3cFH6qNs8ixBDrl8pZwFK0Y
IBJcTFBbI1xBEFkZo03wczzo5+8vPQ60RVqAaYb+iCa1HFJpccC30vajfa4GRdNb
n6FYnluIcDbmpcQePoVQwX7W3o0YLB1QLN7fE6H1j4TBI5Fd030uKzmaifQ1wLYt
DVxVCNDabp0r6Uozd5ASm4ihPPoEoKo7I1p0f0T6fZ41U2xWA4+HF/89UoygZSo7
K+cQ90xGxJ+gm1YbLFR5rbJ0LfjrgDAb2ogbFy8LzHo2ZtSe60M=

```

```
-----END CERTIFICATE-----
```

## 美國東部 (維吉尼亞) — us-east-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAqQCCQCWukjZ5V4aZzAJBgqhkJ00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQKKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQKKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbcwggEsBgcqhkJ00AQBMIIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nvwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJ1/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buyCU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
1Ra2v1ntMX3carVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKml1qx41lHW
MXrs3IgIb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAKGBYqGSM44BAMDLwAwLAIUWXB1k40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUE1y2NIKC+Rg4uu4u32koG9QEYIwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1N1YXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBZXWVzU2VydmljZXMgTEEx
MB4XDTE0MDQyOjE3MzQzQmVwOXM0DTE0MDQyOjE3MzQzQmVwOXM0XDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1N1YXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBZXWVzU2VydmljZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCjvRjF/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJO+eIB
UqPfQG09kZ1wpWpmy08bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHGDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUuizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWdwUuizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdT
ZWF0dGx1MSAwHgYDVQKKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUE1y2NIKC
U+Rg4uu4u32koG9QEYIwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
```

```
AA0BgQA1xSmwcWnhT4uAeSinJuz+1BTcKhVSWb5jT8pYjQb8ZoZkXXRGb09mvYeU
Neq0Br27rvRAnaQ/9LUQf72+SahDFuS4CMI8nwowytqbmwquqFr4dxA/SDADyRiF
ea1UoMuNHTY49J/1vPomqsVn7mugTp+TbjqCf0JTpu0temHcFA==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIIEejCCAvqgAwIBAgIJALFpzEAVWaQZMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQKIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQKHEwdTZWF0
dGx1MSAwHgYDVQKKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNTA4MTQw
ODU5MTJaGA8yMTk1MDEExNzA4NTkxMlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAGT
EFdhc2hpbmd0b24gU3RhZGUxEDA0BgNVBAClBTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWlgaU2Vydm1jZXMgTEExBjIjANBgkqhkiG9w0BAQEFAAOCQA8AMIIB
CgKCAQEAjS2vqZu9mE0h0q+0bRpAbCuiapbZMFNQqRg7kT1r7Cf+gDqXKpHPjsng
SfNz+JHQd8WPI+pmNs+q0Z2aTe23klmf2U52KH9/j1k8R1Ibap/yFibFTSedmegX
E5r447GbJRSHumuIIIfZTZ/or1puII05/Vz7S0j22tdkdY2ADp7caZkNxpSP915fk
2jJMTBU0zyXUS2rBU/u1NHbTTeePjcEkvzVYPahD30TeQ+/A+uWUu89bHSQ0JR8h
Um4cFApzZgN3aD5j2LrSMu2pctkQwf9CaWyVznqrsGYjY0Y66LuFzSCXwqSnFBfv
fFBAFsJcGy24G2DoMyYkF3MyZ1u+rwIDAQABo4HUMIHRMASGA1UdDwQEAwIHGDAd
BgNVHQ4EFgQUrynSPp4uqSECwy+Pi04qyJ8TWSkwyY4GA1UdIwSBhjCBg4AUryns
Pp4uqSECwy+Pi04qyJ8TWSmhYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQKIEExBx
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQKHEwdTZWF0dGx1MSAwHgYDVQKKEXdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQzAJALFpzEAVWaQZMBlGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBADW/s81XijwdP6NkEoH1m9XLrvK4YTqkNfR6
er/uRRgTx2QjFcmNrx+g87gAm11lz+D0crAZ5LbEhDMs+JtZyR3ty0HkDk6SJM85
haoJNAFF7EQ/zCp1EJRikLLsC7bcDL/Eriv1swt78/BB4RnC9W9kSp/sxd5svJMg
N9a6FAP1pNRsWAnbP8JB1AP93oJzb1X2LQXgykTghMkQ07NaY5hg/H5o4dMPc1TK
1YGq1FUCH6A2vdrxmpKDLmTn5//5pujdD2MN0df6sZwtxwZ0os1jV4rDjm9Q3VpA
NWIsDEcp3GUB4pro0R+C7PNkY+VG0DitB0w09qBGosCBstwyEqY=
-----END CERTIFICATE-----
```

## 美國西部 (加利佛尼亞北部) — us-west-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgqhkiG9w0BAQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQKIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQKHEwdTZWF0dGx1MSAwHgYD
VQKKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQKIEExBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQKHEwdTZWF0dGx1MSAwHgYDVQKKEXdBbWF6b24gV2ViIFN1
```



```
cnZpY2VzIExMQzCCAbcwggEsBgcqhkJ00AQBMIIbHwKBgQCjkvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nvwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0Ca0/buycU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmeve5f8LIE/Gf
MNMp9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKml1qx41lHW
MXrs3IgIb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAKGBYqGSM44BAMDlwAwLAIUwXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----
```

## RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUK2zmY9PUSTR7rc1k20wPYu4+g7wwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBACTB1N1YXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWlU2Vydm1jZXMgTEEx
MB4XDTI0MDQyOTE3MDI0M1oXDTI0MDQyOTE3MDI0M1owXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACTB1N1YXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWlU2Vydm1jZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvrjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJO+eIB
UqPfQG09kZlwpWpmy08bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUuizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWdwUuizvtUF2UTihYKReMFwCzAJ
BgNVBAYTA1VTMRkwFwYDVoQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVoQHEwdT
ZWFOdGx1MSAwHgYDVoQKExdBbWF6b24gV2ViIFN1cnZpY2VzIExMQzIUK2zmY9PU
STR7rc1k20wPYu4+g7wwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQA1Ng4QmN4n7iPh5CnadS0c0ZfM7by0dBePwZJyGvOHdaw6P6E/vEk76KsC
Q8p+akuzVzVPkU4kBK/TRqLp19wEwoVwhhTaxHjQ1tTRHqXIVlRkw4JrtFbeNM21
GlkSLonuzmNZdivn9WuQYeGe7nUD4w3q9GgiF3CPorJe+UxtbA==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJANNPkIpcyEtIMA0GCSqGSIb3DQEBgwUAMFwCzAJBgNV
BAYTA1VTMRkwFwYDVoQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVoQHEwdTZWF0
dGx1MSAwHgYDVoQKExdBbWF6b24gV2ViIFN1cnZpY2VzIExMQzAgFw0xNTEwMjkw
OTAzMDdaGA8yMTk1MDQwMzA5MDMwN1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
```

```

YXpvgbiBXZWigU2Vydm1jZXMgTEExDMIIIBjANBgkqhkiG9w0BAQEFAAOCQAQ8AMIIB
CgKCAQEApHQGvHvq3SVCzDrC7575BW7GWLzCj8CLqYcL3YY7Jffupz70jcft057Z
4fo5Pj0CaS8DtPzh8+8vdwUSMbiJ6cDd3ooio3MnCc6DwzmsY+pY7CiI3UVG7KcH
4TriDqr1Iii7nB5MiPJ8wTeAqX89T3SYaf6Vo+4Gcb3LCDGvncZ9TrGcz2CHkJsJ
AIGwgopFpwhIjVYm7obmuIxSIUv+oNH0wXgDL029Zd98SnIYQd/njiqkzE+lvXgk
4h4Tu17xZIKBgFcTtWPky+POGu81DYFqiWVEyR2JJKm2/iR1dL1YsT39kbNg47xY
aR129sS4nB5Vw3TRQA2jL0ToTIxzhQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHGDAd
BgNVHQ4EFgQUgepyi0Ns8j+q67dmcWu+mKKDa+gwgY4GA1UdIwSBhjCBg4AUgepy
i0Ns8j+q67dmcWu+mKKDa+ihYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQKIEExBX
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQKHEwdTZWF0dGx1MSAwHgYDVQKQExdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQ4IJANNPkIpcyEtIMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAGLFWyutf1u0xcAc+kmnMPqtc/Q6b79VIX0E
tNoKMI2KR81cV8ZE1XDb0NC6v8UeLpe1WBKjAwQtEjL1ifKg9hdY9Rj4RXIDSK7
33qCQ8juF4vep2U5TTBd6hfWxt1Izi88xudjixmbpUU4YKr8UPbmixldYR+BEx0u
B1KJi9l11xvuc/Igy/xeH0AZEjAXzVvHp8Bne33VvWmiMxWECZCiJxE4I7+Y6fqJ
pLLSFFJKbNaFyX1DiJ3kXyPEZSc1xiWeyRB2ZbTi5eu7vMG4i3AYWuFVLthaBgu
lPfhafJpj/JDcqt2vKUKfur5edQ6j1CGdxqqjawn0TEqcN8m7us=
-----END CERTIFICATE-----

```

## 美國西部 (奧勒岡) — us-west-2

### DSA

```

-----BEGIN CERTIFICATE-----
MIIC7TCCAqQCCQCWukjZ5V4aZzAJBgqhkiG9w0BAQEFAAOCQAQ8AMIIB
CgKCAQEApHQGvHvq3SVCzDrC7575BW7GWLzCj8CLqYcL3YY7Jffupz70jcft057Z
4fo5Pj0CaS8DtPzh8+8vdwUSMbiJ6cDd3ooio3MnCc6DwzmsY+pY7CiI3UVG7KcH
4TriDqr1Iii7nB5MiPJ8wTeAqX89T3SYaf6Vo+4Gcb3LCDGvncZ9TrGcz2CHkJsJ
AIGwgopFpwhIjVYm7obmuIxSIUv+oNH0wXgDL029Zd98SnIYQd/njiqkzE+lvXgk
4h4Tu17xZIKBgFcTtWPky+POGu81DYFqiWVEyR2JJKm2/iR1dL1YsT39kbNg47xY
aR129sS4nB5Vw3TRQA2jL0ToTIxzhQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHGDAd
BgNVHQ4EFgQUgepyi0Ns8j+q67dmcWu+mKKDa+gwgY4GA1UdIwSBhjCBg4AUgepy
i0Ns8j+q67dmcWu+mKKDa+ihYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQKIEExBX
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQKHEwdTZWF0dGx1MSAwHgYDVQKQExdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQ4IJANNPkIpcyEtIMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAGLFWyutf1u0xcAc+kmnMPqtc/Q6b79VIX0E
tNoKMI2KR81cV8ZE1XDb0NC6v8UeLpe1WBKjAwQtEjL1ifKg9hdY9Rj4RXIDSK7
33qCQ8juF4vep2U5TTBd6hfWxt1Izi88xudjixmbpUU4YKr8UPbmixldYR+BEx0u
B1KJi9l11xvuc/Igy/xeH0AZEjAXzVvHp8Bne33VvWmiMxWECZCiJxE4I7+Y6fqJ
pLLSFFJKbNaFyX1DiJ3kXyPEZSc1xiWeyRB2ZbTi5eu7vMG4i3AYWuFVLthaBgu
lPfhafJpj/JDcqt2vKUKfur5edQ6j1CGdxqqjawn0TEqcN8m7us=
-----END CERTIFICATE-----

```

## RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUFX8PxCkbHwpD31b0yCtyz3GclbgwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBhZG90bG93dG93dG93dG93
MT0VowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBhZG90bG93dG93dG93dG93
A4GNADCBiQKBgQCHvrjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJ0+eIB
UqPfQG09kZlwpWpmy08bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnf0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHGDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUuizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWdwUuizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTA1VTMRkwFwYDVoQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVoQHEwdT
ZWFOdGx1MSAwHgYDVoQKExdBbWF6b24gV2ViIFNlcnZpY2VzIEEMQ4IUFx8PxCkb
HwpD31b0yCtyz3GclbgwEgYDVR0TAQH/BAGwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQbzO1+9Xy1+UsbUBI95H09mbbdnux+aMJXgG9uFZNjgNEbMcvx+h8P9IMko
z7PzFdheQQ1NLjshH9mSR1SyC4m9ja6BsejH5nLBWyCdjfdP3muZM405+r7vUa10
dWU+hP/T7DUrPAIVM0E7mpYa+WPWJrN6B1RwQkKQ7twm9kDa1A==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJJALZL31rQCSTMMMA0GCSqGSIb3DQEBwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVoQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVoQHEwdTZWF0
dGx1MSAwHgYDVoQKExdBbWF6b24gV2ViIFNlcnZpY2VzIEEMQzAgFw0xNTA4MTQw
OTAxMzJaGA8yMTk1MDEzNzA5MDEzMlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBhZG90bG93dG93dG93dG93dG93dG93dG93dG93dG93dG93dG93dG93dG93
YXpvbiBhZG90bG93dG93dG93dG93dG93dG93dG93dG93dG93dG93dG93dG93dG93
CgKCAQEA02Y59qtAA0a6uzo7nEQcnJ260KF+LRPwZfixBH+EbEN/Fx0gYy1jppjCP
s5+VRNg6/WbfqAsV6X2VSjUKN59ZMnMY9ALA/Ipz0n00Huxj38EBZmX/NdNqKm7C
qWu1q5kmIvYjKGiadfboU8wLwLcHo8ywwvfgI6FiGGsE09VMC56E/hL6Cohko11LW
dizyvRcvG/IidazVkJQCn/4zC9PU0VyKdhw33jXy8BTg/QH927QuNk+ZzD7HH//y
tIYxDhR6TIZsSnRjz3b0cEHxt1nsidc65mY0ejQTy4hy7ioSiapw316mdbtE+RTN
fch9FPiFKQNBpiqfAW5Ebp3La13/+wIDAQABo4HUMIHRMAsGA1UdDwQEAwIHGDAd
BgNVHQ4EFgQU7coQx8Qnd75qA9XotSWT3IhVJmowgY4GA1UdIwSBhjCBg4AU7coQ
x8Qnd75qA9XotSWT3IhVJmqhYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVoQIEExB
XNoaW5ndG9uIFN0YXR1MRAwDgYDVoQHEwdTZWF0dGx1MSAwHgYDVoQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIEEMQ4IJALZL31rQCSTMMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAFZ1e2MnzRaXCALwEC1pW/f0oRG8nHr1PZ9W
OYZEWbh+QanRgaikBNDtVTwARQcZm3z+HWSkaIx3cyb6vM0DSkZuiwzm1LJ9rDPc
```

```
aBm03SEt5v8mcc7sXWvgFjCnUpzozsmky6JheCD401Cf8k0o1Z93FQnTrbg620K0h
83mGCDvEKU3hLH97FYUq+3N/IliWFDhvibAYYKFJydZLhIdlCiiB99AM6Sg53rm
oukS3csyUxZyTU2hQfdjyo1nqW9yhvFAKjnnggiwxNKTPZzstKW8+cnYwiiTwJN
QpVoZdt0SfbuNnmwRUMi+QbuccXweav29QeQ3ADqjgB0CZdSRkk=
-----END CERTIFICATE-----
```

## 非洲 (開普敦) — af-south-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7DCCAqCCQCncbCtQbjuyzAJBgqhkJ00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEExBXlYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQKQExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xOTA2MDQxMjQ4MDVaFw00
NTA2MDQxMjQ4MDVaFwCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBXlYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbYwggErBgqhkJ00AQBMIIIBHgKBgQC12Nr1gMrHcFSZ7S/A
pQBSCMHWmn2qeoQTMVWqe50fnTd0zGFxDdIjKxUK58/8zjWG5uR4TXRzmZpGpmXB
bSufAR6BGqud2LnT/HIWGJAsnX2u0tSyNfCoJigqwha5w+CqZ6I7iBDdnB4TtTw
q06TlnExHFVj8LMkyLZgiaE1CQIVAIhdobse4K0QnbAhCL6R2euQzloXAoGAV/21
WUUmZ/79Ga0JvQcz1FNy1sT0pU9rU4TenqLQIt5iccn/7EIfNtvV05TZKu1IKq7J
gXZr0x/KIT8zsNweetL0aGehPIYRMPX0vunMMR7hN7qA7W17WZv/76adywIsnDKq
ekfe15jinaX8MsKUdyDK7Y+ifCG4PVhoM4+W2XwDgYQAaGAIx0KbVgwLxbn6Pi2
6hB0ihFv16jKxAQI0hHzXJLV0Vyv9QwnqjJJRf0Cy3dB0zicLXiIxeIdYfvqJr+u
h1N8rGxEZYjYjEUKMGvsc0DW85jonXz0bNfcP0aaKH01KKVjL+OZi5n2kn9wgd05
F3CVnM18BUra8A1Tr2yrrE6TVZ4wCQYHkoZiZjgEAwMvADAsAhQfa7MCJZ+/TEY5
AUr0J4wm8VzjoAIUSYZVu2NdRJ/ERPmDfhW5EsjH1CA=
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIICNjCCAZ+gAwIBAgIJAKumfZiRrNvHMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXlYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xOTEwMjQ4
NzE0MDVaGA8yMTk5MDUwMjA3MTQwNVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAGT
EFdhc2hpbmd0b24gU3RhZGUxEDA0BgNVBAcTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWJgU2Vydm1jZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKB
gQDFd571nUzVtke3rPyRkYfvs3jh0C0EMzzG72boyUNjnfW1+m0TeFraTLKb9T6F
7TuB/ZEN+vm1Yqr2+5Va8U8qLbPF0bRH+FdaKjhgWZdYXxGzQzU3ioy5W5ZM1VyB
7iUsxEA1xSybC3ziPYaHI42UiTkQNaHmoroNeqVyHNnBpQIDAQABMA0GCSqGSIb3
DQEBCwUAA4GBAAJLy1WyE1Eg0pW4B1XPyRVD4pAds8Guw2+krqgkY0HxLCdjosuH
```

```
RytGDGN+q75aAoXzW5a7SGpxLxk6Hfv0xp3RjDHsoeP0i1d8MD3hAC5ezxS4oukK
s5gbP0nokhKTMpXbTdRn5ZifCbWlx+bYN/mTYKvxho7b5SVg2o1La9aK
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIID0zCCAi0gAwIBAgIJAIIFI+05A6/ZIMA0GCSqGSIb3DQEBCwUAMFwx CzAJBgNV
BAYTA1VTMRkwFwYDVoQIEExBXyXNoaW5ndG9uIFN0YXR1MRAwDgYDVoQHEwdTZWF0
dGx1MSAwHgYDVoQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xOTA2MDQx
MjQ4MDRaGA8yMTk4MTEwNzEyNDgwNFowXDELMAkGA1UEBhMCMVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhRGUxEDA0BgNVBACTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWVlU2Vydm1jZXMgTEExMTEuIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAY7/WHBBH0rk+20aumT07g8rxrSM0UXgki3eYgKauPCG4Xx//vwQbuZwI
oeVmR9nqnhfij2w0cQdbLandh0EGtbxerete3IoXzd1KXJb11Pvmzrzyu5SPBPuP
iCeV4qdjjkXo2YWM6t9YQ911hcG96YSp89TBXFYU3KLxfqAdTVhuC0NRGhXpyii
j/czo9njofHhghTr7UEyPun8NVs2QwctLQ86N5zWR3Q0GRoVqqMrJs0cowHTrVw2
9Qr7QBjjB0VbyYmtYxm/DtiKprYV/e6bCAVok015X1sZDd3oC0QNoG1v5XbHJe2o
JFD8GRRy2rkW0/1NwVFDcweC6zC3QwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQCE
goqzjpCpmMgCpsZFhwvRaSmbSPkTK7wNImUjrsB0fBjsfFu1yg1Zgn2nDCK7kQhx
jMjMniVxbps3yMqQ2cHUKkCkF5t+WldfeT4Vv1Rz6HSA8sd0kgVcIesIaoy2aaXU
VEB/oQziRGyKdN1d4TGYVZXG44CkrzSDvlbmfITq5tL+kAieznVF3bzHgPZW6hKP
EXC3G/IXrXicFEe6YyE1Rak162VncYSXiGe/i2XvsiNH3Qlmnx5XS7W0SCN0oAxW
EH9twibauv82DVg1W0kQu8EwFw8hFde9X0Rkiu0qVcuU81JgFEvPWMDFU5sGB6ZM
gkEKTzMv1ZpPbBhg99Jl
-----END CERTIFICATE-----
```

## 亞太區域 (香港) — ap-east-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq4CCQC07MJe5Y3VLjAJBgqhkiG9w0AQMDFwx CzAJBgNVBAYTA1VTMRkw
FwYDVoQIEExBXyXNoaW5ndG9uIFN0YXR1MRAwDgYDVoQHEwdTZWF0dGx1MSAwHgYD
VoQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xOTAyMDMwMjIxMjFaFw00
NTAyMDMwMjIxMjFaMFwx CzAJBgNVBAYTA1VTMRkwFwYDVoQIEExBXyXNoaW5ndG9u
IFN0YXR1MRAwDgYDVoQHEwdTZWF0dGx1MSAwHgYDVoQKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbgwggEsBgqhkiG9w0AQBMIIBHwKBgQDvQ9RzVvf4MAwGbbqfX
b1CvCoVb99570kLGN/04CowHXJ+vTBR7eyIa6AoX1tsQXB0mrJswToFKKxT4gbuw
jK7s9QXX4CmTRwCEg02RXtZSVj0hsUQMh+yf7Ht40VL97LWnNfGsX2cwjcRWHYgI
71vnuBNBzLQHdSEwMNq0Bk76PwIVAMan6XIEEPnwr4e6u/RNnWBGkd9FAoGBA0CG
eSNmXPw4QFu4pI1Aykm6EnTZKKHT87gdXkAkfoC5fAf0xxhnE2HezZH9Ap2tMV5
```

```
8bWNvoPHvoKCQqwfM+OUB1AxC/3vqoVkkL2mG1KgUH9+hrtPMTkw03RREnKe7I50
x9qDimJp0ihrl4I0dYvy9xU0oz+DzFAW8+yLWVYpA4GFAAKBgQDbnBAKSxWr9QH
6Dt+EFdGz61AZLedeBKpaP53Z1DT034J0C55YbJTwbTFGqPtOLxnUVD1GiD6GbmC
80f3jvogPR1mSmGsydbNbZnbUEVWrRhe+y5zJ3g9qs/DWmDW0deEFvkhWVnLJkFJ
9pd0u/ibRPH11E2nz6pK7Gb0QtLyHTAJBgqhkj00AQDAzAAMC0CFQCoJlwGtJQC
cLoM4p/jtVF0j26xbgIUUS4pDKyHaG/eaygLtTfPqzWHc=
-----END CERTIFICATE-----
```

## RSA

```
-----BEGIN CERTIFICATE-----
MIICSzCCAbQCCQDtQvkVxRvK9TANBgkqhkiG9w0BAQsFADBqMQswCQYDVQQGEwJV
UzETMBEGA1UECBMKV2FzaGluZ3RvbjEQMA4GA1UEBxMHU2VhdHRsZTEYMBYGA1UE
ChMPQW1hem9uLmNvbSBjb250bWRowGAYDVQQDExF1YzIuYW1hem9uYXdzLmNvbTAe
Fw0xOTAyMDMwMzAwMDZaFw0yOTAyMDIwMzAwMDZaMGoxCzAJBgNVBAYTA1VTMRMw
EQYDVQQIEWpXYXNoaW5ndG9uMRAwDgYDVQQHEwdTZWF0dGx1MRgwFgYDVQQKEw9B
bWF6b24uY29tIEluYy4xGjAYBgNVBAMTEWVjMi5hbWF6b25hd3MuY29tMIGfMA0G
CSqGSIb3DQEBAQUAA4GNADCBiQKBgQC1kkHXyTfc7gY5Q55JJhjTieHAgacaQkiR
Pity9QPDE3b+NXDh4UdP1xdIw73JcIIG3sG9RhWiXVCHh6KkuCTqJfPUknIKk8vs
M3RXf1UpBe8Pf+P92pxqPMCz1Fr2NehS3JhhpkCZVGxxwLC5gaG0Lr4rFORubjYY
Rh84dK98VwIDAQABMA0GCSqGSIb3DQEBCwUAA4GBAA6xV9f0HMqXjPHuGILDyaNN
dKcvp1NFwDTydg32MNubAGnecoEBtUPtxBsLoVYXC0b+b5/ZMDubPF9tU/vSXuo
TpYM5Bq57gJzDRaB0ntQbX9bgHiUxw6XZwaTS/6xjRJDT5p3S1E0mPI31P/eJv4o
Ezk5zb3eIf10/sqt4756
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIID0zCCAiOgAwIBAgIJAMoxixvs3YsSMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEwBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKEXdBbWF6b24uY29tIEluYy4xIFN1cnZpY2VzIExMQzAgFw0xODA3MjAw
ODQ0NDRAgA8yMTk3MTIyMzA4NDQ0NFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24uY3RhdGUxEDA0BgNVBAcTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWlGU2Vydm1jZXMgTEExMIIBIjANBgkqhkiG9w0BAQEFAAOCQA8AMIIB
CgKCAQEA4T1PNs0g0FDrG1WePoHe0Sm0JTA3HCry5LSbYD33GFU2eBr0IxoU/+SM
rInKu3GghAMfH7WxPW3etIAZiyTDDU5RLcUq2Qwdr/ZpXAWpYocNc/CEmBFtfbx
Fz4uwBIN3/dRM0RSbe/wP9EcgMNUGQMMZWeAji8sMtwp0b1NWAP9BniUG0F1cz6Dp
uPovwDTLdAYT3TyhzlohKL3f6048TR5yTaV+3Ran2SGRhyJjfh3FRpP4VC+z5LnT
WPQHN74Kdq35UgrUxNhJraMGczzno1UuoR/tFMwR93401GsM9fVA7SW3jjCGF81z
PSzjy+ArKyQqIpLW1YGWDFk3sf08FQIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQDK
2/+C3nPMgty0FX/I3Cyk+Pui44IgwCsIdNGwuJysdq5VIIfnjegEu2zIMWJSKGO
1MzoQXjffkVZ97J7RNDW06oB7kj3WVE8a7U4WE0fn0/CbMUF/x99CckNDwpjgW+
```

```
K8V8SzAsQDvYZs2KaE+18GFfLVF1TGUYK2rPSZMHyX+v/TI1c/qUceBycrIQ/kke
jDFsihUMLqgm0V2hXKUpIsmiWMGrFQV4AeV0iXP8L/ZhcepLf1t5SbsGdUA3AUY1
3If8s81uTheiQjwY5t9nM0SY/1Th/tL3+RaEI79VNEVfG1FQ8mgqCK0ar4m0oZJ1
tmmEJM7xeURdpBBx36Di
-----END CERTIFICATE-----
```

## 亞太區域 (海德拉巴) — ap-south-2

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC8DCCArCgAwIBAgIGAXjrQ4+XMAkGByqGSM44BAMwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAGMEFdhc2hpbmd0b24g
U4EddRIpUt9KnC7s50f2EbdSP09EAMMeP4C2USZpRV1AI1H7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdrrmVC1pJ+f6AR7ECLCT7up1/63xhv401fnxqimFQ8E
+4P208UewwI1VBNAFpEy9nXzrith1yrv8iIDGZ3RSAHHAhUA12BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v60uqC+VdMCz0HgmdRWVeOutRZT
+ZxBxCBgLRJFneJ6EwoFh03zwyjMim4TwWeotUfI0o4K0uHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/
hWuWfBpKLZ16Ae1ULZAFM0/7PSSoDgYUAAoGBAJCKGBBoxIUxqBk94JHhwZZbgvbP0DA0oHENQWxp/981I7/
Y0fYJ0VMJS22aCnHDurofmo5rvNIkgXi7Rztbhu
+1ko9rK6DgmpUwBU0WZtf34aZ2IWNBwHaVhHvWAQf9/46u18dMa2YucK1Wi+Vc+M
+K1drvXgmhym6ErN1zhJyMAkGByqGSM44BAMDLwAwLAIUaaPKxa0HoYvwz709xXpsQueIq+UCFFa/
GpzoD0Sok11057NU/2hnsiW4
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIICmzCCAZygAwIBAgIGAXjwLj9CMA0GCSqGSIb3DQEBBQUAMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIDBBXYXNoaW50
+sFcobrjvcAYm0PNRD8f4R1jAzvoLt2+qGe0TAY01Httj6cmsYN3AP1hN5iYuppFiYs12eNPa/
CD0Vg0BAfDF1V5rzjpA0j7TJabVh4kj7JvtD+xYMi6wEQA4x6SP0NY40eZ2+8o/
HS8nucpWDVdPR06ciWU1MhjmDmwIDAQABMA0GCSqGSIb3DQEBBQUAA4GBAAy6sgTdRkTqELHBeWj69q60xHyUmsWqHAQ
TGgbYP0yP2qfM10cCIImzRI5W0gn8gogderVfeT7nH5ih0TWEy/QDwfkQ601L4erm4yh4YQq8vcqAPSkf04N
-----END CERTIFICATE-----
```

### RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAIvWfPw/X82fMA0GCSqGSIb3DQEBQwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXXYXNoaW50dG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKEXdBbWf6b24gV2ViIFN1cnZpY2VzIExMQzAgFw0yMjA3MDQx
NDMwMjhaGA8yMjAxMTIwODE0MzAyOFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAGT
```

```

EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAClTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWlgaU2VydmJjZXMgTExDMiIBIjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIB
CgKCAQEAg29QEFriG+qFEjYw/v62nN701MJY/Hevx5TtmU/VIYBPQa3HUGTBabbI
2Tmy8UMpa8kZeaYeI3RAfiQWt0Ws7wUrBu02Pdp518WDPaJUH7RWEuu1BDDkyZRW
NAMNPCn3ph70d243IFcLGku7HVeke15poqRpSfojrMasjlf+CvixUeAJbmFoxUHK
kh5unzG2sZy04wHXcJPQkRf5a8zSTPe9YZP1kXPPEv4p/jTSggaYPxXyS6QVaT1V
zLeLFZ0fesLPMeil3KYQtV7IKLQiEA2F6dxWnxNWQ1yMHtdq6PucfEmVx17i/Xza
yNBRo0azY8WUNVKErRhp/pU8Nh3GQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHGDAd
BgNVHQ4EFgQU9A01aZk9RLXk2ZvRVoUxYvQy9uwwgY4GA1UdIwSBhjCBg4AU9A01
aZk9RLXk2ZvRVoUxYvQy9uyhYKReMFwCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBX
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6
b24gV2ViIFNlcnZpY2VzIEExMQ4IJAIVWfPw/X82fMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBADexluMRQRftqViahCnauEWGdMvLCBr8A+Yr
6hJq0guoxEk/lahxR137DnFMPuSbi1Rx5QKo7oBrWfG/zsgQUnF2IwHTzwD+i/2m
XCane6FiS5RpK31GdILq8ZmlhQk+6iI8yoZLr0LCfTh+CLgIKH0knfR51FzgzAiF
SI8/Q9mm+uvYtSTZECI6Zh57QZPoETAG/y1+9ji0y21Aelqa/k1i+Qo8gMf0c+Pm
dwY7o6fV+oucgr1sdey6VM45LeyILQqv0RXtVzjuowanzmCCFMjgqi09oZAWu40h
+F3unijELo01vZJs8s2N3KGlo3/jtUFTX6RTKShZ1APLwBi5GMI=
-----END CERTIFICATE-----

```

## 亞太區域 (雅加達) — ap-southeast-3

### DSA

```

-----BEGIN CERTIFICATE-----
MIIC8DCCArCgAwIBAgIGAXbVDEikMAKGBYqGSM44BAMwXDELMAGKA1UEBhMVCVVMxGTAXBgNVBAGMEFdhc2hpbmd0b24g
U4EddRIpUt9KnC7s50f2EbdSP09EAMMeP4C2USZpRV1AI1H7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdirmVClpJ+f6AR7ECLCT7up1/63xhv401fnxqimFQ8E
+4P208UewwI1VBNAFpEy9nXzriith1yrv8iIDGZ3RSAHHAUA12BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v60uqC+VdMCz0HgmdRWVeOutRZT
+ZxBxCBGLRjFneJ6EwoFh03zwyjMim4TwWeotUfI0o4K0uHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/
hWuWfBpKLZ16Ae1U1ZAFM0/7PSSoDgYUAAoGBAPjuieX05N3JQ6cVwntJie67D80uNo4jGRn
+crEtL7Y00jSVB9zGE1ga
+UgRPIaYETL293S8rTJTvgXAqdpBwfaHC6NUzre8U8iJ8FMNn1P9Gw1oUIlgQBjORyynVJexoB31TDZM
+/52g90/bpq1QqNyKbeIgyBB1c1dAtr1QLnsMAKGBYqGSM44BAMDlwAwLAIUK8E6RDIRtwK+9qnaTOBhv0/
njuQCFFocyT10xK+UDR888oNsdgtif2Sf
-----END CERTIFICATE-----

```

### RSA

```

-----BEGIN CERTIFICATE-----

```



```

MIICMzCCAzygAwIBAgIGAXbVDG2yMA0GCSqGSIb3DQEBBQUAMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIDBBXYXNoaW50
Vbt0gQ1ebWcur2hS07PnJifE40PxQ7RgSAlc4/spJp1sDP+ZrS0L01ZJfKhXf1R9S3AUwLnsC7b
+IuVXdY5LK9RKqu64nyXP5dx170z0L81oEyCSuRR2fs+04i2QsWBVP+KFNA7P5L1EHRjkgT08kjNKvwiRV
+0kP9ab5wIDAQABMA0GCSqGSIb3DQEBBQUAA4GBAII4WUy6+DKh0JDSzQEZNyBgNLSuSuC2owtMxCwGB6nBfzzfcekWvs
+87w/g91NwUnUt0ZHYyh2tuBG6hVJuUEwDJ/z3wDd6wQviL0TF3MITawt9P8siR1hXqLJNxpjRQFZrgHqi
-----END CERTIFICATE-----

```

## RSA-2048

```

-----BEGIN CERTIFICATE-----
MIIIEejCCAvqgAwIBAgIJAMtdyRcH51j9MA0GCSqGSIb3DQEBQwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0yMjA0MDgX
MjM5MTZaGA8yMjAxMDkxMjE5MzcxN1owXDELMAkGA1UEBhMCMVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIGU2Vydm1jZXMgTEExDMIIIBjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAvUsKCxoh6KXRYJLeYTWAQfaBQeCwhJaR56mfUeFHJE4g8aFjWkiN4uc1
Tv0yYNnIZKTHWmzmulmdinWNbwP0GiR0Hb/i7ro0HhvnptyycGt8ag8affiIbx5X
7ohdwSN2KJ6G0IKf1Ix7f2NEI0oAMM/9k+T1eVF+MVWzpZoiDp8frLNkqp8+RAgz
ScZsRfWv3u/if5xJAVdg2nckIWDMSHEVPoz01Jo7v0ZuDtWwSL1LHnL5ozvsKEk
+ZJyEi23r+U1hIT1NTBdp4yoigNQexedtwCSr7q36o0dDwvZpqY1kLi3uxZ4ta+a
01pz0STwMLgQZSbKWQrpMvsIAPrxoQIDAQAB04HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU1GgnGdNpbnL31LF30Jomg7Ji9hYwgY4GA1UdIwSBhjCBg4AU1Ggn
GdNpbnL31LF30Jomg7Ji9hahYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBX
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQ4IJAMtdyRcH51j9MBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBACV100qQlatBKVeiWMrhpczsJroxDxLZT0ba
6wTMzk7c3akb6XM0SZFbGaiFkebPZqTHEhD1rC1M2j9AI1YcCx6YCrTf4cuhn2mD
gcJN33143e0WSaeRY3ee4j+V9ne98y3k02wLz95VrRgc1PFR8po2iWGzGhwUi+FG
q8dXeCH3N0DZgQsSgQWwmdNQXZZej6RHLU/8In5trHKLY0ppnLBjn/UZQbeTyW5q
RJB3GaveXjfgFUWj2q0cDuRGaikdS+dYaLsi5z9cA3FolHzWxx9M0s8io8vKqQzV
XU1rLTNwuhZy88c01qGPxnoRbw7TmifwPw/cunNrsjUU0gs6ZTk=
-----END CERTIFICATE-----

```

## 亞太區域 (墨爾本) — ap-southeast-4

### DSA

```

-----BEGIN CERTIFICATE-----
MIIC7zCCAq
+gAwIBAgIGAXjWF7P2MAkGByqGSM44BAMwXDELMAkGA1UEBhMCMVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24gU3RhdGUx
U4EddRIpUt9KnC7s50f2EbdSP09EAMMeP4C2USZpRV1AI1H7WT2NWPq/

```

```

xfW6MPbLm1Vs14E7gB00b/JmYldrmVClpJ+f6AR7ECLCT7up1/63xhv401fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzriith1yrv8iIDGZ3RSAHHAhUAL2BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v60uqC+VdMCz0HgmdRWVe0utRZT
+ZxBxCBgLRJFnEj6EwoFh03zwykjMim4TwWeotUfI0o4K0uHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/
hWuWfBpKLZL16Ae1ULZAFM0/7PSSoDgYQAAoGAPRxsQP9E3dw8QXK1rgBgEVCprLHdk/bbrMas0XMu1Eh0D
+q
+0PcTr8+iwbtoXLY5MCeatWIpl1GrXQjVqsF8vQqx1EuRuYKbR3nq4mWwaeGlx9AG5EjQHRa3GQ44wWH0dof0M3NRI1M
-----END CERTIFICATE-----

```

RSA

```

-----BEGIN CERTIFICATE-----
MIICMzCCAzygAwIBAgIGAXjSh40SMA0GCSqGSIb3DQEBBQUAMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIDBBXYXNoaW50
+qWTGAbGsPeMX4hBMjAJUKys2NIRcRZaLM/BCew2FIPVjNt1aj6Gwn9ipU4M1z3zIwAMWi1AvGMSreppt
+wV6MRtf0jh0Dvj/veJe88aEzJMozNgkJFRS
+WFwSckQeL56tf6kY6QTlNo8V/0CsQIDAQAABMA0GCSqGSIb3DQEBBQUAA4GBAF7vpPghH0FRo5gu49EArRNPrIvW1egM
wcgkqIwwuXYj+1rh1L+/
iMpqWjdVGEqIZSeXn5fLmdx50eegFCwND837r9e8XYTiQS143Sxt9+Yi6BZ7U7YD8kK9NBWoJxqFqUeHdpRCs007C0jT3
-----END CERTIFICATE-----

```

RSA-2048

```

-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAN4GTQ64zVs8MA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXyXNoaW50dG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0yMjA3MTMx
MzZmMjM0b24gU3RhdGUxIDA0BgNVBAcTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWlgaU2Vydm1jZXMgTEExDQYIKoZIhjANBgkqhkiG9w0BAQEFAAOCQA8AMIIB
CgKCAQEA2BYgeCr+Rk/jIAED0HS7wJq162vc83QEwjuzk0q0FEReIZz1N1fBRNXK
g0T178Kd3gLYcE59wEFbTe/X5y0A1Lo95x1anSAo7R+Cisf9C2HQuJp+gVb+zx71
lniPF7gHzigpm0M8DdAU/IW+wkZwGbp4z7Hq9+bJ0P21tvPJ5yxSgkFuDsI9VBHa
CLoprHsCHh2VdP8KcMgQQMmHe1NmBpyTk0u1/aLmQkCQEX6ZIRG0eq228fwlh/t+
Ho+jv87duihVKic6MrL32S1D+maX0LSDUydWda0LLTGkh7oV7+bFuH6msrXUu+Uz
ZEP1r/MidCWMhfgRfzeTBz0HA97qxQIDAQABo4HUMIHRMASGA1UdDwQEAWIHGDAd
BgNVHQ4EFgQUChMd1cHqzmsQ5hpUK3EMLhHdsi4wgY4GA1UdIwSBhjCBg4AUChMd
1cHqzmsQ5hpUK3EMLhHdsi6hYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBX
YXNoaW50dG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQ4IJAN4GTQ64zVs8MBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAI4PFyVN+7EGS0bioiPnv0LL0f70SSzUZJ8p
X090d4rWea7jIbgZ2AKb+ErynkU9xVg7XQ05k6KDWgp/4jYFL2dqnt/YAY4PS0un
RSrYE1awxLT0BcLn4rcSDC79vQe1xGC5//wDdV6b399C0AHRAK6axWYy5w32u9PL

```

```
uw0cIp3Ch8JoNwcgTHKRRGzePmBeR4PNqhHTArG4/dJk6/aU04OpX0WzI6L67CGY
6Nex3dau+gkLCK93dTEkrXtyXHu4wB0J9zd1w+iQ0SEa9eKc78/NjEsF/FZdGrWC
t571IM00XJhQ1kRgSwNeZdQWV1dRakv06sfcvVYkfj1wAvZvvAw=
-----END CERTIFICATE-----
```

## 亞太區域 (孟買) — ap-south-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgqhkJ00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEExBXlYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQKKExdBbWF6b24gV2ViIFN1cnZpY2VzIEEMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBXlYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQKKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEEMQzCCAbcwggEsBgcqhkJ00AQDMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MbcJl/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buyCU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmveve5f8LIE/Gf
MNmP9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKm11qx411HW
MXrs3IgIb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGBYqGSM44BAMDlwAwLAIUwXB1k40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUdLA+x6tTAP3LRT1r0z6n0xfsozdMwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhZGUxEDA0
BgNVBAcTB1N1YXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWlGU2Vydm1jZXMgTEEx
MB4XDTE0MDQyOTE0MTMwMVowXDTI1MDQyODE0MTMwMVowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhZGUxEDA0BgNVBAcTB1N1YXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWlGU2Vydm1jZXMgTEExDMIGFMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvrjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJ0+eIB
UqPfQG09kZ1wpWpmy08bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXk3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUuizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWdwUuizvtUF2UTihYKReMFwxCzAJ
-----END CERTIFICATE-----
```

```
BgNVBAYTA1VTMRkwFwYDVQIQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQHQEwdT
ZWF0dGx1MSAwHgYDVQKQExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUDLA+x6tT
AP3LRTIz0z6n0xfsozdMwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQAZ7rYKoAwwiiH1M5GJbrT/BEk3002VrEPw8ZxgppQ/EK1zML0s/0Cyimp7
UYyUgYFQe5nq37Z94r0USemgv/WRxaMwrL1LqD78cuF9DSkXaZIX/kECtVaUnjk8
BZx0QhoIH0pQocJUS1m/dLeMuE0+0A3HNR6JVktGsUdv9u1mKw==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIID0zCCAiOgAwIBAgIJAPRYyD8TtmC0MA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQIQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQHQEwdTZWF0
dGx1MSAwHgYDVQKQExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNjAzMDcx
MDQ1MDFaGA8yMTk1MDgxMTEwNDUwMVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWVzIGU2Vydm1jZXMgTExDMIIIBjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIB
CgKCAQEALSS5I/eCT2PM0+qusorBx67QL26BIWQHd/yF6ARtHBb/1DdFLRqE5Dj
07Xw7eENC+T79m0x0AbeWg91Ka0D0zw6i9I/2/HpK0+NDEdD6sPKDA1d45jRra+v
CqAjI+nV9Vw91wv7HjMk3RcjWGziM8/hw+3YNIutt7aQzZRwIW1Bpcqx3/AFd8Eu
2UsRMSHgkGUW6UzUF+h/U8218XfrauKNGmNKDYUhtmyBrHT+k6J0hQ4pN7fe6h+Z
w9RVHm24BGh1LxLHLms0IxvbrF277uX9Dxu1HfKfu5D2kimTY7xSZDNLr2dt+kNY
/+iWdIeEFpPT0PLSILt52wP6stF+3QIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQBIE
6w+WWC2gCfoJ06c9HMyGLMFEpqZmz1n5IcQt1h9iy07Vkm1wkJiZsMhXpk73zXf
TPxuXEacTX3S0Ea070IMCFwkus05f61e0yFTynHCzBgZ3U0UkRVZA3WcpbNB6Dwy
h7ysV1qyT9WZd7E0Ym5j5oue2G2xdei+6etgn5UjyWm6liZGrc0F6WPTdmzqa6WG
ApEqanpkQd/HM+hUYex/ZS6zEhd4CCDLgYkIjlrFbFb3pJ10VLztIfSN5J40o1pu
JVCfIq5u1NkpzL7ys/Ub8eYipbzI6P+yxXiUSuF0v9b98ymczMYjrSQXIf1e8In3
0P2Cc1Choz8XDQcvvKAh
-----END CERTIFICATE-----
```

## 亞太區域 (大阪) — ap-northeast-3

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgqhkiG9w0BAQ0DMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQIQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQHQEwdTZWF0dGx1MSAwHgYD
VQKQExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xMjAzMDQ1MDFaGA8yMTk1
MDQ1MDFaGA8yMTk1MDgxMTEwNDUwMVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
ODAxMDUxMjAzMDQ1MDFaGA8yMTk1MDgxMTEwNDUwMVowXDELMAkGA1UEBhMCVVMx
IFN0YXR1MRAwDgYDVQHQEwdTZWF0dGx1MSAwHgYDVQKQExdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbcwggEsBgqhkiG9w0BAQ0BMIIIBHwKBGQCjkvcS2bb1VQ4yt/5e
```

```
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLcLnd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MbcJl/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buyCU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKm11qx411HW
MXrs3IgIb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDlwAwLAIUwXB1k40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----
```

## RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUHTRhxHhBZF0GvTFKxHoy9+f5H18wDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBACeTB1N1YXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBxZWVlU2Vydm1jZXMgTEExD
MB4XDTE0MDQyOTE2NTQwN1oXDTE1MDQyODE2NTQwN1owXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACeTB1N1YXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBxZWVlU2Vydm1jZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCChvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJ0+eIB
UqPfQG09kZ1wpWpmy08bGB2RwqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEf0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUuizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWdwUuizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdT
ZWV0dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUHTRhxHhB
ZF0GvTFKxHoy9+f5H18wEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBGQAUZX7DcYbhWNTD4BNghr5beruT20UoGHH9J73UKxwdqeb9bH1LIWhIZ00X
/1mjn3bWBgCwfoS8gjZwsVB6fZbNBRy8urdBZJ87xF/4JPbjt7S9oGx/zthDUYrC
yK0Y0v4G0PgiS81CvYLg09LpmYhLSJbXEN1kC04v5yxdKxZxyg==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIID0zCCAiOgAwIBAgIJAMn1yPk22ditMA0GCSqGSIb3DQEBGwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWV0
dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNzA3MTkx
MTEyNThaGA8yMTk2MTIyMjExMTI1FowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACeTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWVlU2Vydm1jZXMgTEExDMIIIBjANBgkqhkiG9w0BAQEFAAOCQA8AMIIB
```

```
CgKCAQEArznEYef8IjhrJoazI0QGZkmlmHm/4rEbyQbMnifxjsDE8YwtHNwaM91z
zmyK6Sk/tK1Wxcn13g31iq305ziyFPEewe5Qbwf1iz2cMsvfNBcTh/E6u+mBPH3J
gvGanqUJt6c4IbipdEouIjjnyVwd4D6erL1/ENijeR10xVpaqSW5SBK7jms49E
pw3wtbchEl3qsE42Ip4IYmWxqjgaxB7vps91n4kfyzAjUmklcqTfMfPckzmJCRgp
Vh1C79vRQhmriVKD6BXwfZ8tG3a7mijeDn7kTsQzg007Z2SAE63PI048JK8Hc0bH
tXORUQ/XF1jzi/SIaUJZT7kq3kwl8wIDAQABMA0GCSqGSIb3DQEBwUAA4IBAQBj
Tht09dLvU2QmKuXAhxXjsIdlQgGG3ZGh/Vke4If1ymgLx95v2Vj9Moxk+gJuUSRL
BzFte3TT6b3jPolbECgmAorjj8NxjC17N8QAAI1d0S0gI8kqkG7V8iRyPIFekv+M
pcai1+cIv5IV5qAz8Q0MGYfGdYkcoBjsgiyvMJU/2N2UbZJNGWvcEGkdjGJUYY00
NaspCAFm+6HA/K7BD9zXB1IKsprLgqhiIUgEaW3UFEbThJT+z8UfHG9fQjzzfN/J
nT6vuY/0RRU1xAZPyh2gr5okN/s6rnmh2zmBHU1n8cbCc64MVfXe2g3EZ9G1q/9n
izPrI09hMypJDP04ugQc
-----END CERTIFICATE-----
```

## 亞太區域 (首爾) — ap-northeast-2

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgqhkJ00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEyBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQKQExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEyBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQKQExdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbcwggEsBgcqhkJ00AQBMIIbHwKBgQCjkvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nvwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJ1/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buyCU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKml1qx41lHW
MXrs3IgIb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLeAwLAIUWXB1k40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUbsn2UI06vYk4iNwV0RPxJJtH1gwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhZGUxEDA0
```

```
BgNVBAcTB1N1YXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBxZWlgaU2Vydm1jZXMgTExD
MB4XDTI0MDQy0TEzZmZg0NloXDTI0MDQy0DEzZmZg0NlowXDELMAkGA1UEBhMCMVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1N1YXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBxZWlgaU2Vydm1jZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCChvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJO+eIB
UqPfQG09kZ1wpWpmy08bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnf0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHGDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdT
ZWF0dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUbBSn2UIO
6vYk4iNwV0RPxJJtHlgwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQAmjTja1G8MGLqWTC2uYqEM8nzI3px1eo0ArvFRsyqQ3fgmWcQpxExqUqRy
l3+2134Kv8dFab04Gut5wlfRtc20wPKKicmv/IXGN+9bKFnQFjTqif08NIzrDZch
aFT/uvxrIiM+oN2YsHq66GUh02+xVRXDxVxM/V0bFgPERbJpyA==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIID0zCCAi0gAwIBAgIJANuCGcCht0JhMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNTA5MTQx
NTU3NDRaGA8yMTk1MDIxNzE1NTc0NFowXDELMAkGA1UEBhMCMVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWlgaU2Vydm1jZXMgTExDMIIIBiJANBgkqhkiG9w0BAQEFAA0CAQ8AMIIB
CgKCAQEa66iNv6pJPmGM20W8HbVYJS1KcAg2vUGx8xeAbzZIQdpGfKabVcUHGB6m
Gy59VXDMD1rJckDDk6dxU0hmcX9z785TtVZURq1fua9QosdbTzX4kAgHGdp4xQEs
m06QZqg5qKjBP6xr3+PshfQ1rB8Bmwg0gXEm22CC7o77+7N7Mu2sWzWbiUR7vi14
9FjWS8XmMNwFT1Shp4l1TDTevDWW/uYmC30RThM9S4QPvTZ0rAS18hHVam8BCTxa
LHaVCH/Yy52rsz0hM/F1ghnSnK105ZKj+b+KIp3adBL80MCjgc/Pxi0+j3HQLdYE
32+FaXWU84D2iP2gDT28evnstzuYTQIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQC1
mA4q+12pxy7By6g3nBk1s34PmWikNRJBw0qhF8ucGRv8aiNhRRye9lokXomwo8r
KHbbqvtK8510xUZp/Cx4sm4aTgcMvfJP29jGLc1DzeqADIVkWEJ4+xncxSYV1S9x
+78TvF/+8h9U2LnS164PXaKdxHy2IsHIVRN4GtoaP2Xhpa1S0M328Jykq/571nfN
1WRD1c/fQf1edgzRjhQ4whcAhv7WRRF+qTbfQJ/vDxy81ki0svU9XzUaZ0fZSfXX
wXxZamQb0NvFcXVHY/0PSiM8nQoUmkkBQuK1eDwRWvkoJKYKy1r3jvXK7HIWtM1r04
jmXe0aMy3thyK6g5sJVg
-----END CERTIFICATE-----
```

## 亞太區域 (新加坡) — ap-southeast-1

## DSA

```

-----BEGIN CERTIFICATE-----
MIIC7TCCAqQCCQCWukjZ5V4aZzAJBgqhkJ00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEiExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQKKExdBbWF6b24gV2ViIFN1cnZpY2VzIEEMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEiExBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQKKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEEMQzCCAbcwggEsBgcqhkJ00AQBMIIbHwKBgQCjKvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buycU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmeve5f8LIE/Gf
MNMp9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKm11qx411HW
MXrs3IgIb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDlwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----

```

## RSA

```

-----BEGIN CERTIFICATE-----
MIIDITCCAOqgAwIBAgIU5SqP6ih+++5KF07NXnggrWf26mhSUwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAClTB1N1YXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBxZWVzIGU2Vydm1jZXMgTEEx
MB4XDTE0MDQyOTU0MzAxNFoXDTI1MDQyODE0MzAxNFowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAClTB1N1YXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBxZWVzIGU2Vydm1jZXMgTEExDMIGFMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJ0+eIB
UqPfQG09kZ1wpWpmy08bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUuizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWdwUuizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTA1VTMRkwFwYDVQQIEiExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdT
ZWF0dGx1MSAwHgYDVQKKExdBbWF6b24gV2ViIFN1cnZpY2VzIEEMQz4IUSqP6ih++
+5KF07NXnggrWf26mhSUwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQAw13BxW11U/JL58j//Fmk7qqtrZTqXmaz1qm2W1IpJpW750M0cP4ux1uPy
eM0RdVZ4jHSMv5gtLAv/PjExBfw9n6vNck+5GZG4Xec5DoapBZHxmfMo93sjxBFP
4x9rWn0GuwAV09ukjYPevq2Rerilrq5VvppHtbATVNY2qecXDA==
-----END CERTIFICATE-----

```



```
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAJVMGw5SHkcvMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXlYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNTEwMjkw
ODUzMTlaGA8yMTk1MDQwMzA4NTcxOVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAGT
EFdhc2hpbmd0b24gU3RhZGUxEDA0BgNVBAClB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWlgaU2Vydm1jZXMgTEExDMIIIBIjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIB
CgKCAQEAlaSSLfB170gmikjLReHuNhVuvM20dCsVzptUyRbut+KmIEEc24wd/xVy
2RMIrydGedk4tUjkUy0yfET50AyT43jTzDPHZTkRSVkyjBdcYbe9o/0Q4P7IVS3
X1vwrUu0qo9nSID0mxMn0oF1l8KAqnn10tQ0W+1NSTkasW7QVzcb+3okPEVhPA0q
Mn1Y3vkMQGI8zX4i0KbEcSVIzf6wuIffXMGHVC/JjwihJ2USQ8fq6oy686g54P4w
R0g415kLYcodjqThmGJPNUPAZ7M0c5Z4pymFuCHgNAZNvjhZDA8420jecqm62zcm
Tzh/pNMNeGCRYq2EQX0aQtY0Ij7b0QIDAQABo4HUMIHRMAsGA1UdDwQEAwIHGDAd
BgNVHQ4EFgQU6SSB+3qALorPMVNjToM1Bj3oJMswgY4GA1UdIwSBhjCBg4AU6SSB
+3qALorPMVNjToM1Bj3oJMuhYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBX
lYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQ4IJAJVMGw5SHkcvMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAF/0dWqkIEZKg5rca8o0P0VS+to1JJE/FRZO
atH0eaQbWzyac6NEwjYeeV2kY63skJ+QPuYbSuIBLM8p/uTRIVYM4LZYImLGuvo0
IdtJ8mAzq8CZ3ipdMs1hrRqF5GRp8lg4w2QpX+PfhW47iI0BiqSAUkIr3Y3BDaDn
EjeXF6qS4iPIvBaQQ0cvdddNh/pE33/ceghbkZNTYkrwMyBkQ1RTTVKXFN7pCRUV
+L9FuQ9y8mP0BYZa5e1sdkwebyU+eqVzsil98ntkhpjvRkaJ5+Drs8TjGaJWlRw
5Wu0r8unKj7YxdL1bv7//RtVYVVvi2961doRUYv4SCvJF11z00dQ=
-----END CERTIFICATE-----
```

## 亞太區域 (雪梨) — ap-southeast-2

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgqhkiG9w0AQMDFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEExBXlYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xMjAxMDUxMjU2MTJhFw0z
ODAxMDUxMjU2MTJhMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBXlYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbcwggEsBgqhkiG9w0AQMIIIBHwKBQCjkvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
-----END CERTIFICATE-----
```

```

hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nvwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MbcJl/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buycU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmeve5f8LIE/Gf
MNMp9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKm11qx411HW
MXrs3IgIb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDlwAwLAIUwXB1k40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----

```

## RSA

```

-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUFxWyAdk4oiXI0C9PxcgjYYh71mwwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBACeTB1N1YXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWlgaU2Vydm1jZXMgTE
xDMGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCCHvRjf/0kStpJ248khtIaN8qk
DN3tkw4VjvA9nvP12anJ0+eIBUqPfQG09kZ1wpWpmy08bGB2RWqWxCwB/dcnIob6w420
k9WY5C0IIGtDRNauN3kuvGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0Ic
L6NLA+H94/QIDAQABo4HfMIHcMAsGA1UdDwQEAwIHGdAdBgNVHQ4EFgQUJdbMCB
XKtvCcWdwUUizvtUF2UTgwgZkGA1UdIwSBKTCBjoAUJdbMCBXXKtvCcWdwUUizvt
UF2UTihYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVRQQIEwBXYXNoaW5ndG9uIFN0
YXR1MRAwDgYDVRQHEwdTZWF0dGx1MSAwHgYDVRQQEwdBbWF6b24gV2ViIFN1cnZp
Y2VzIEExMQ4IUFxWyAdk4oiXI0C9PxcgjYYh71mwwEgYDVR0TAQH/BAgwBgEB/wI
BADANBgkqhkiG9w0BAQsFAA0BgQByjeQe61r7fiIhoGdjBXyzDfkX01GGvMIhRh57
G1bbceQfaYdZd7Pt0j1bpycKGaTvhUdkpM0iV2Hi9d00YawkdhyJDstmDNKu6P9+b6
Kak8He5z3NU1tUR2YuTwcZ7Ye8N1dx//ws3raErfTI7D6s9m630X8cAJ/f8bNgikwpw
==
-----END CERTIFICATE-----

```

## RSA-2048

```

-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAL2b0gb+dq9rMA0GCSqGSIb3DQEBwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVRQQIEwBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVRQHEwdTZWF0
dGx1MSAwHgYDVRQQEwdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUFxMQzAgFw0x
NTEwMjkwOTAwNTdaGA8yMTk1MDQwMzA5MDA1N1owXDELMAkGA1UEBhMCVVMxGTAXBg
NVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACeTB1N1YXR0bGUxIDAeBgNVBA
oTF0FtYXpvbiBXZWlgaU2Vydm1jZXMgTEwMIIBiANBgkqhkiG9w0BAQEFAAOCAQ8A
MIIB CgKCAQEAmRcyLWraysQS8yDC1b5Abs3TUaJabjqWu7d5gHik5Icd6dK18EYpQ
SeSvz6pLhkg04xBbCRGlge8LS/0ijcZ5HwdrxBiKbicR1YvIPaIyEQQvF5sX6UWkGYw

```

```

Ma5IRGj4YbRmJkBybw+AAV9Icb5LJNOMWPi340WM+2tMh+8L234v/JA6ogpdPuDr
sM6YFHMZ0NWo58MQ0FnEj2D7H58Ti//vFP10TaaPWaAIRF85zBiJtKcFJ6vPidqK
f2/SDuAvZmyHC8ZBHg1moX9bR5FsU3QazfbW+c+JzAQWHj2AaQrGSCITxCM1S9sJ
L51DeoZBjnx8cnRe+HCaC4YoRBiqIQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHGDAd
BgNVHQ4EFgQU/wHIo+r5U31VIsPoWoRVsNXGxowwgY4GA1UdIwSBhjCBg4AU/wHI
o+r5U31VIsPoWoRVsNXGxoyhYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBX
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6
b24gV2ViIFNlcnZpY2VzIEExMQ4IJAL2b0gb+dq9rMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBACobLv8Ix1QyORTz/9q7/VJL509/p4HAeve
92riHp6+Moi0/dSEYpEFTgdWB9W3YCNc34Ss9TJq2D7t/zLGG1bI4wYXU6VJjL0S
hCjWeIyBXUZ0ZKFCb0DSJeUElsTRSXSfuVrZ9EAwjLvHni3BaC9Ve34iP71ifr75
8Tpk6PEj0+JwiiJFH8E4GhcV5chB0/iooU6ioQqJrMwFYnwo1cVZJD5v6D0mu9bS
TMIJLJKv4QQQqPsNdiB7G9bfbk6trP8fUVYLHLsV1Iy51Gx+tgwFEYkG1N8I00/
2LCawwaWm8FYAFd3IZL04RImNs/IMG7VmH1bf4swH0BHgCN1uYo=
-----END CERTIFICATE-----

```

## 亞太區域 (東京) — ap-northeast-1

### DSA

```

-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgqhkJ00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEExBXyXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQQKEXdBbWF6b24gV2ViIFNlcnZpY2VzIEExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBXyXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFNl
cnZpY2VzIEExMQzCCAbcwggEsBgqhkJ00AQBMIIbHwKBgQCjkvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MbcJl/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buycU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKml1qx41lHW
MXrs3IgiB6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----

```

### RSA

```

-----BEGIN CERTIFICATE-----

```

```

MIIDITCCAoqgAwIBAgIULgwDh77TiDrPPBJwscqDwiBHkEFQwDQYJKoZIhvcNAQEL
BQAwxDELMAKGA1UEBhMVCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBxZWVjZmUyZmVjZmVjZmVj
MB4XDTE0MDQyOTYyMjM0MDQyOTYyMjM0MDQyOTYyMjM0MDQyOTYyMjM0MDQyOTYy
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBxZWVjZmUyZmVjZmVjZmVjZmVjZmVjZmVjZmVjZmVjZmVj
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJO+eIB
UqPfQG09kZlwpWpmy08bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHGDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUUizvtUF2
UTgwZkGA1UdIwSBkTCBj0AUJdbMCBXXtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdT
ZWF0dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IULgwDh77Ti
DrPPBJwscqDwiBHkEFQwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQBtjAg1Bde1t4F9EHCZ0j4qnY6Gigy070u54i+1R77MhbpzE8V28Li9l+YT
QMIn6SzJqU3/fIycIro10VY11HmaKYgPGSEZxBenSBHfzwDLRmC9oRp4QMe0Bj0C
gepj11UoiN70A6PtA+ycn1sP0oJvdBjhvayLiuM3tUfLTrgHbw==
-----END CERTIFICATE-----

```

## RSA-2048

```

-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAL9KIB7Fgvg/MA0GCSqGSIb3DQEBQwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IULgwDh77Ti
OTAwMjVhGA8yMTk1MDExNzA5MDAyNVowXDELMAKGA1UEBhMVCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWVjZmUyZmVjZmVjZmVjZmVjZmVjZmVjZmVjZmVjZmVjZmVjZmVjZmVj
CgKCAQEAz0djWUcmRW85C5CiCKPFiTIv6y20uopFxE5d3Wtab10bm06vnXVKXu
tz3AndG+Dg0zIL0gM1U+QmrSR0PH2Pfv9iejfLak9iwdm1WbwRrCEAj5VxPe0Q+I
Kezn0txzqQ5Wo5NLE9bA61sziUAFNVsTFUzphEwRohcekYyd3bBC4v/RuAjCXHVx
40z6AiksnA0GN2VABM1TeMnvPItK0CIerL111SqXX1gbtL1gxSW40JWdF3WPB68E
e+/1U3F70Er7XqmN0D0L6yh92QqZ8fHjG+af0L9Y2Hc4g+P1nk4w4iohQ0PABqzb
MPjK7B2Rze0f90Ec51GBQu13kxkWwQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHGDAd
BgNVHQ4EFgQU5D55IFdU/QwYbikgtWvkU3fDwRgwY4GA1UdIwSBhjCBg4AU5D55
IFdU/QwYbikgtWvkU3fDwRihYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIEExB
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQ4IULgwDh77TiDrPPBJwscqDwiBHkEFQwEgYD
VRRsdg/eumFft0rL0Ko54pJ+Kim2cngCWNhkcZctRHBV567AJNt4+ZDG5
hDgV0IxW01+eaLE4qzqWP/9Vr0+p3reuumgFZLvpVpwXBBBeBFUf2drUR14aWfI2
L/6VGINXys7uP8v/2VBS7r6XZRnPBuY/R4hv5efYXnjwA9gq8+a3stC2ur8m5yS1
faKSWE4H320yAyaZWH4gpwUdbU1YgPHtm/ohRtiWPrN7KEG5Wq/REzMIjZCnx0fS

```

```
6KR6PNj1hxBsImQhmBvz6j5PLQx0xBZIpDoiK278e/1Wqm9LrBc=
-----END CERTIFICATE-----
```

## 加拿大 (中部) — ca-central-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAqQ0CCQCWukjZ5V4aZzAJBgqhkJ00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbcwggEsBgcqhkJ00AQBMIIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLcLnd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nvwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MbcJl/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buycU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmveve5f8LIE/Gf
MNmP9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKml1qx411HW
MXrs3IgiB6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAKGBYqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAOqgAwIBAgIUlrLgixJJ5C4G8z6pZ5rB0JU2aQwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAClB1N1YXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBZXWV2VydmljZXMgTEEx
MB4XDTE0MDQy0TE1MzU0M1oXDTE1MDQy0DE1MzU0M1owXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAClB1N1YXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBZXWV2VydmljZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJ0+eIB
UqPfQG09kZ1wpWpmy08bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdT
ZWF0dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUlrLgixJJ
-----END CERTIFICATE-----
```

```
B5C4G8z6pZ5rB0JU2aQwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQBHiQJmzyFAaSYs8SpiRijIDZW2RIo7qBkb/pI3rqK6y0WDlPuMr6yNI81D
IrkGGftg4Z+2KETyU4x76HSf0s//vfH3QA57qFaAwddhKYy4BhteFQ1/Wex3xTLX
LiwI07kwJvJy3mS6UfQ4HcvZy219tY+0iy0Wrz/jVxwq7T0kCw==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIID0zCCAi0gAwIBAgIJAJNKhJhaJ0uMMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWw6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNjA3Mjkx
MTM3MTdaGA8yMTk2MDEwMjExMzcxN1owXDELMAKGA1UEBhMVCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhZGUxEDA0BgNVBAClB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWVzIGU2VydmIjZXMgTEExMjE1IjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIB
CgKCAQEAEhDuh6j1ACSt057nSxAcwMaGr8Ez87VA2RW2HyY819XoHndnxmP50Cqld
+26AJt1t1qHpI1YdtnZ60rVgVhXcVtbtvte0lZ3ldEzC3PMvmISBhHs6A3SWhA9ln
InHbToLX/SWqBHL0X78HkPRaG2k0C0HpRy+fG9gvz8HCiQaXCbWNFDHZev90ToNI
xhXBVzIa3AgUnGma1CYZuh5AfVRCEeALG60kxMMC8IoAN7+HG+pMdqAhJxGUcM00
LBvmTGGewhi04MUZwf0kwn9JjQZuyLg6B10D4Y6s0LB2P1MovmSJKGY4JcF8Qu3z
xxUb17Bh9pvzFR5gJN1pjM2n3gJEPwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQAj
UNKM+gIIHNk0G0tzv6vZBT+o/vt+tIp8lEozwPQh1121iw/I7ZvhMLAigx7eyvf
IxUt9/nf8pxWaeGzi98RbSmbap+uxYRynqe1p5rifTam0sguuPrhVp1120gRWLcT
rjg/K60UMXRsmg2w/cxV45pUBcyVb5h60p5uEVAVq+CVns13ExiQL6kk3guG4+Yq
LvP1p4DZfeC33a2Rfre2IHLsJH5D4SdWcYqBsftPf3FQThH0l0KoacGrXtsedsxs
9aRd70zuSEJ+mBxmzXsJswM840oh78DjkdpQgv967p3d+8NiSLt3/n7MgnUy6WwB
KtDujDnB+ttEHwRRngX7
-----END CERTIFICATE-----
```

## 加拿大西部 (卡加利) — ca-west-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq
+gAwIBAgIGAYPouptUMAKGByqGSM44BAMwXDELMAKGA1UEBhMVCVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24gU3RhZGUxEDA0BgNVBAClB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
U4EddRIpUt9Knc7s50f2EbdSP09EAMMeP4C2USZpRV1AI1H7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdirmVClpJ+f6AR7ECLCT7up1/63xhv401fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzrith1yrv8iIDGZ3RSAHHAhUA12BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmU1r7v60uqC+VdMCz0HgmdRWVeOutRZT
+ZxBxBcBgLRJFnEj6EwoFh03zwyjMim4TwWeotUfI0o4K0uHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/hWuWfBpKLZ16Ae1U1ZAFM0/7PSSoDgYQAAoGAMITzTJUa6cBsIfdHN69zW/
-----END CERTIFICATE-----
```

```
aHjUB4r1ZfKb1FMhIp9EZtEf5n+06oXjUG2+dKRS1FQeEK333ehNZsPd6uqey6TYKtHpFb5XRLS8BpqB
+7gnbAd0CBZM5o4NWesSQ1GLnTdQcGZkYG/
QESkbadoCXQTifCujJE682hTDLIVt1d4ewwCQYHKoZiZjgEAWmVADAsAhRJc4gRS/HWTkCR2MESaQEe/
jOMNQIUNoTwLvuPmGPupPlGiHe0veZi08=
-----END CERTIFICATE-----
```

## RSA

```
-----BEGIN CERTIFICATE-----
MIICMzCCAzygAwIBAgIGAYPou9weMA0GCSqGSIb3DQEBBQUAMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIDBBXYXNoaW50
v4XBVH13ZCMgq1RHMqV8AWI5i06gFn2A9sN3AZXTMqwtZeIddebq3k6Wt7ieYvpXTg0qvgsjQIovRZwaBDBJy9x8C2hw
+w9lMQjFhkJ7Jy/
PHCJ69EzebQIDAQABMA0GCSqGSIb3DQEBBQUAA4GBAGe9Snkz1A6rHBH6/5kDtYvtPYwhx2sXNxztbhkXErfk40Nw514
gvDVtWG7qyb6fAqgoisyAbk8K9LzxSim2S1nmT9vD84B/t/VvwQBylc
+ej8kRmMH7fquZLp7IXfmtBzyUqu6Dpbne+chG2
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJALyTn5IHrIZjMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0yMzEyMDcx
NTM3MDFaGA8yMjAzMDUxMzE1MzcwMVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAGT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWJgU2VydmJjZXMgTEExMjE1MjE1MjE1MjE1MjE1MjE1MjE1MjE1MjE1
CgKCAQEA1GP5os424BjMGPCK0Sg0c1P71zUiB85du03M4hfjzS0szsBpmBGFDLz1
owYHtIx1q3+Vi1Lt5Q1x3id/ov1QyaBPFwXVek1HVXy9vieCcI3TdjGjT11W/8MM
m3X26QPcsnHM/Kk2wJ7s186MrqmdSsp3SCPpxv4vEG2Q9yR2bXY41hpc2rW1W8qU
D0JGXlUvmmAdFnto20l1XWZ6xFen1h60DRugek/ufCbN+lJky0xLqPoavH0Ybjsb
UpsAsBs7phaoN+X/5hIERfbp5Lfvnqq54pNG5Knu4Kynfw9+kA/WS4cJ6FTTN5t+
y0P1HvcL+BL2RuDy6T2bB21xw5WqtQIDAQABo4HUMIHRMASGA1UdDwQEAwIHGDAd
BgNVHQ4EFgQURTVu/Dd4zDnmS5G5CfVlnmUBN0swgY4GA1UdIwSBhjCBg4AURTvu
/Dd4zDnmS5G5CfVlnmUBN0uhYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBX
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQzAgIjALyTn5IHrIZjMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAFt523A3Aug6/F8xxyITgA8gkU0btFh1XNSP
U4U20Q9n0tWI9WqnKNWH3KBxwY5EPitU6b3LM4xc9lDWpz7h2Pto+WhxP9LVKe6f
r8r7teTLCVZ7cfYZHzHg+f1ZjVpAgzE5BVfR1j3QKpv0hYT3J1wMtI++Vorq5Nf
aPjzedehJLhmZVALwnfqfLrgv6/gmraP9Vmoa8U4D6AljNiQGYaLwyoPoRm3bUs2
v1Mh9GkEQ1b9+1pFXcqqzJJTGRuiPCyPbECI79FAnx5JM/CkGJV8H10mjIW1qkK1
Y2qT7wzErrKLJyB53Pw15BdIM1onbDAQreZb0yZQLdoEl/tx7Uk=
-----END CERTIFICATE-----
```

## 歐洲 (法蘭克福) — eu-central-1

## DSA

```

-----BEGIN CERTIFICATE-----
MIIC7TCCAqQCCQCWukjZ5V4aZzAJBgqhkJ00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEiExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQKKExdBbWF6b24gV2ViIFN1cnZpY2VzIEEMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEiExBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQKKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEEMQzCCAbcwggEsBgcqhkJ00AQBMIIIBHwKBGQCjKvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nvwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MbcJl/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buycU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmve5f8LIE/Gf
MNMp9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKm11qx411HW
MXrs3IgIb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDlwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----

```

## RSA

```

-----BEGIN CERTIFICATE-----
MIIDITCCAOqgAwIBAgIUFD5GsmkxRuecttwsCG763m3u63UwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMVCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBAClB1N1YXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBxZWVzIGU2Vydm1jZXMgTEEx
MB4XDTE0MDQyOTE1NTUyOVVoXDTI1MDQyODE1NTUyOVowXDELMAkGA1UEBhMVCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAClB1N1YXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBxZWVzIGU2Vydm1jZXMgTEExDMIGFMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJ0+eIB
UqPfQG09kZ1wpWpmy08bGB2RWqWxCwUB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUuizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWdwUuizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTA1VTMRkwFwYDVQQIEiExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdT
ZWF0dGx1MSAwHgYDVQKKExdBbWF6b24gV2ViIFN1cnZpY2VzIEEMQ4IUF5Gsmkx
RuecttwsCG763m3u63UwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQBbH0WaX1BsW56Hqk588MmJxs0rvcKfDjF57RgEDgnGnQaJcStCVWD09UY0
JX2tdsPw+E7AjDqjsuxYaotLn3Mr3mK0sN0Xq9BljBnWD4pARg89KZnZI8FN35HQ
0/LY0VHCknuPL123VmVRNs51qQA9hkPjvw21UzpdLxaUxt9Z/w==
-----END CERTIFICATE-----

```



```
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAKD+v6LeR/WrMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXlYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNTA4MTQw
OTA4MTlaGA8yMTk1MDExNzA5MDgxOVowXDELMAKGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhZGUxEDA0BgNVBAClB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBhZG9uU2VydmljZXMgTEExDMIIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAKa8FLhxs1cSJKG+Q+q/vTf8zVnDAPZ3U6oqpp0W/cupCtpwMAQcky8DY
Yb62GF7+C6usniaq/9W6xPn/3o//wti0cNt6MLsiUeHqN15H/4U/Q/fr+GA8pJ+L
npqZDG2tFi1WMvvGhGgIbScrjR4V03TuKy+rZXYvMRk1RXZ9gPhk6evFnviwHsE
jV5AEjxLz3duD+u/SjPp1v1oxe2KuWnyC+EKInnka909s14ZAUh+qIYfZK85DAjm
GJP4W036E9wTJQF2hZJrzsiB1MGyC1WI9veRISd30izZZL6VVXLXUtHwVHnVASrS
zZDVpzj+3yD5hRXsvFigGhY0FCVFnwIDAQABo4HUMIHRMAsGA1UdDwQEAwIHGDAd
BgNVHQ4EFgQUxC2l6pvJaRf1gu3MudN6zTuP6YehYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBX
lYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4I
JAKD+v6LeR/WrMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAIK+DtbUPppJXFqQMv1f2Gky5/82ZwgbbfXa
HBeGSii55b3tsyC3ZW5Z1MJ7Dtnr3vUkiWbV1EUaZGOUlndUFtXUMABCb/coDndw
CAr53XTv7UwGVNe/AF0/6pQDdPxXn3xBhF0mTKPr0GdvYmjZUtQMSVb91bMWCFFs
w+SwDLnm5NF4yZchIcTs2fdpoyZp0HDXy0xgx01gWhKTnYbaZ0xkJvEvccckxVAwJ
obF8NyJ1a0/pWdjh1HafEXEN81xyTTY0a0BGTuY0BD2cTYynauVKY4fqHUKr3v
Z6fboaHEd4RFamShM8uvSu6eEFD+qRmvq1codbpsS0huGNLzh0Q=
-----END CERTIFICATE-----
```

## 歐洲 (愛爾蘭) — eu-west-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgqhkiG9w0BAQ0DMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEExBXlYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xMjAxMDUxMjU2MTJhFw0z
ODAxMDUxMjU2MTJhMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBXlYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbcwggEsBgqhkiG9w0BAQ0BMIIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
-----END CERTIFICATE-----
```

```

hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nvwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MbcJl/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buycU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmeve5f8LIE/Gf
MNMp9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKm11qx411HW
MXrs3IgIb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDlwAwLAIUwXB1k40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----

```

## RSA

```

-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUakDaQ1Zqy87Hy9ESXA1pFC116HkwDQYJKoZIhvcNAQEL
BQAwXDELMAKGA1UEBhMVCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBACeTB1N1YXR0bGUxIDAEBgNVBAQoTF0FtYXpvbiBZXWlG2Vydm1jZXMgTEEx
MB4XDTE0MDQyOTE2MTg0MjE1MDQyODE2MTg0MjE1MDQyODE2MTg0MjE1MDQyODE2MTg0
MjE1MDQyODE2MTg0MjE1MDQyODE2MTg0MjE1MDQyODE2MTg0MjE1MDQyODE2MTg0MjE1
BgNVBAQoTF0FtYXpvbiBZXWlG2Vydm1jZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJ0+eIB
UqPfQG09kZ1wpWpmy08bGB2RWqWxCwB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HENf0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHGDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUuizvtUF2
UTgwgZkGA1UdIwSBKTCBjoAUJdbMCBXXtvCcWdwUuizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTA1VTMRkwFwYDQoQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDQoQHEwdT
ZWf0dGx1MSAwHgYDQoQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUakDaQ1Zq
y87Hy9ESXA1pFC116HkwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQADIKn/MqALGPuK5+prZZ50x4bBZLPtre02C7r0ppqU2kPM21VPyYYydkvP0
lgSmmsErGu/oL9JNztDe2oCA+kNy17ehcsf8cw0uP861czNFKCeU8b7FgBbL+sIm
qi33rAq6owWGi/5uEcfCR+JP7W+oSYYvir5r/yDmWzx+BvH5S/g==
-----END CERTIFICATE-----

```

## RSA-2048

```

-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJA0rmqHuaUt0vMA0GCSqGSIb3DQEBwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDQoQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDQoQHEwdTZWf0
dGx1MSAwHgYDQoQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUAQABgkqhkiG9w0
OTA2MTlaGA8yMTk1MDQwMzA5MDYxOjVowXDELMAKGA1UEBhMVCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACeTB1N1YXR0bGUxIDAEBgNVBAQoTF0Ft
YXpvbiBZXWlG2Vydm1jZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUAA4GCAQEAjE7nVu+aHL
tzp9FYV25Qs1mvJ1JXD7J0iQ1Gs/RirW9a5ZECctc4ssnf
zQHq2JRVr0GRchvDrbm1HaP/avtFQR/Thvf1twu9AR0VT22dU0TvERdkNzveoFCy

```

```

hf52Rqf0DMrLXG8ZmQPPXPDFAv+sVMWCDftcChxRYZ6mP90+TpgYNT1krD5PdvJU
7HcXrkNHDYqbsg8A+Mu2hzl0QkvUET83CsglibeK54HP9w+FsD6F5W+6ZSHGJ881
FI+qYKs7xsjJQYgXWfEt6bbckWs1kZIaIOyMzYdPF6ClYzEec/UhIe/uJyUUNfpT
VIsI50ltBbcPF4c7Y20j0IwwI2Sg0QIDAQABo4HUMIHRMAsGA1UdDwQEAwIHGDAd
BgNVHQ4EFgQUF2DgPUZivKQR/Zl8mB/MxIkjZDUwgY4GA1UdIwSBhjCBg4AUF2Dg
PUZivKQR/Zl8mB/MxIkjZDWhYKReMFwCzAJBgNVBAYTA1VTMRkwFwYDVQQIEsBX
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6
b24gV2ViIFNlcnZpY2VzIEExMQ4IJA0rmqHuaUt0vMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAgm6+57W5brzJ3+T8/XsIdLTuiBSe5ALgSqI
qn05usUKAeQsa+kZIJPyEri5i8LEodh46DAF1R1XTMYgXXx10YggX88XPmPtok17
14hib/D9/lu4IaFIyLzYNSzsETyWkWoGve7ZFz60MTRTwY2u8YgJ5dec7gQgPSGj
avB0vTIgoW41G58sfw5b+wjXCsh0nR0on79RcQFFhGnvup0MZ+JbljyhZUYFzCli
31jPziKzqWa87xh2DbAyvj2KZrZtTe2LQ48Z4G8wWytJzxEeZdREe4NoETf+Mu5G
4CqoaPR05KwkdNudGNwXewydb3+agdCgfTs+uAjeXKNdSpbhMYg=
-----END CERTIFICATE-----

```

## 歐洲 (倫敦) — eu-west-2

### DSA

```

-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgqhkJ00AQDMFwCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEsBXN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6
b24gV2ViIFNlcnZpY2VzIEExMQ4IJA0rmqHuaUt0vMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAgm6+57W5brzJ3+T8/XsIdLTuiBSe5ALgSqI
qn05usUKAeQsa+kZIJPyEri5i8LEodh46DAF1R1XTMYgXXx10YggX88XPmPtok17
14hib/D9/lu4IaFIyLzYNSzsETyWkWoGve7ZFz60MTRTwY2u8YgJ5dec7gQgPSGj
avB0vTIgoW41G58sfw5b+wjXCsh0nR0on79RcQFFhGnvup0MZ+JbljyhZUYFzCli
31jPziKzqWa87xh2DbAyvj2KZrZtTe2LQ48Z4G8wWytJzxEeZdREe4NoETf+Mu5G
4CqoaPR05KwkdNudGNwXewydb3+agdCgfTs+uAjeXKNdSpbhMYg=
-----END CERTIFICATE-----

```

### RSA

```

-----BEGIN CERTIFICATE-----

```

```

MIIDITCCAoqgAwIBAgIUCgCV/DPxYNND/swDgEKGiC5I+EwwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBxZWVlU2Vydm1jZXMgTExD
MB4XDTE0MDQyOTE2MjKxNFoXDTE1MDQyODE2MjKxNFowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBxZWVlU2Vydm1jZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCChvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJO+eIB
UqPfQG09kZ1wpWpmy08bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHGDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdT
ZWVlU2h1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUCgCV/DPx
YNND/swDgEKGiC5I+EwwEgYDVR0TAAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQATPu/sOE2esNa4+XPEGK1EJSgqzyBSQLQc+Vwo6FAJhGG9fp7D97jhHeLC
5vwfmtTAFnGBxadfa0T3ASKxn0ZhXtnRna460LtnNHm7ArCVgXKJo7uBn6ViXtFh
uEEw4y6p9YaLQna+VC8Xtgw6WKq2JXuKzuhuNKSFAgGw9vRcHg==
-----END CERTIFICATE-----

```

### RSA-2048

```

-----BEGIN CERTIFICATE-----
MIID0zCCAiOgAwIBAgIJANBx0E2b0CEPMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWVl
dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNjA4MTEw
NDU2NDJaGA8yMTk2MDExNTE0NTY0M1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWVlU2Vydm1jZXMgTExDMIIIBiJANBgkqhkiG9w0BAQEFAA0CAQ8AMIIB
CgKCAQEArYS3mJLGAmrh2DmiPLbqr4Z+xWXTzBWCj0wpsuHE9H6dWUuy12Bgnu+Z
d8QvW306Y1eec45M4F2RA3J4hWhtShzsm10JVRt+YulGeTf90CPr26QmIFfs5nD4
fgsJQEry2MBSGA9Fqx3Cw6qkWcr0PsCR+bH0U0XykdK10MnIbpBf0kTfciAupQEA
dEHnM2J1L2iI0NTLBgKxy5PXLH9weX20BFauNmHH9/J070pwL20SN5f8TxcM9+pj
Lbk8h1V4KdIwVQpdWkbDL9BCG1YjyadQJxSxz1J343NzrnDM0M4h4HtVaK0S7bQo
Bqt2ruopLRCYgcuFHck/1348iAmbRQIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQBG
wujwU10tpi3iBgmhjmC1gZyMMn0aQIxMigoFNqXMUNx1Mq/e/Tx+SNa0EAu0n2FF
aiYjvY0/hX0x75ewzZvM7/zJWIdLdsgewpUq0BH4DXFhbSk2TxggSPb0WRqTBxq5
Ed7F7+7GRIeBbRzdLqmISDnfyey8ufW0ks51XcQNomDIRG5s9XZ5KHviDCar8FgL
HngBCdFI04CMagM+pwT09XN1Ivt+NzUj208ca3oP1IwEAd5KhIhPLcihBQA5/Lpi
h1s3170z1JQ1HZbDrH1pgp+8hSI0DwwDvb3IIH8kPR/J0Qn+hv012H0paUg2Ly0E
pt1RCZe+W7/dF4zsbqwk
-----END CERTIFICATE-----

```

## 歐洲 (米蘭) — eu-south-1

## DSA

```

-----BEGIN CERTIFICATE-----
MIIC7TCCAqwCCQCME1HPdwG37jAJBgqhkhj00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEExBXlYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xOTA0MjkyMDM1MjJaFw00
NTA0MjkyMDM1MjJaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBXlYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbYwggErBgcqhkhj00AQBMIIIBHgKBgQDAkoL4YfdMI/MrQ0oL
NPfeEk94eiCQA5xN0nU7+2eVQtEqjFbDADFENh1p3sh9Q90oheLFH8qpSfNDWn/0
ktCS909ApTY6Esx1ExjGSeQq/U+SC2JSuuTT4WFMKJ63a/czMtFkEPPnVIjJJmT
HJSKSsVUgpdDIRvJXuyB0zdB+wIVALQ30LaVGd1PMNfS1nD/Yyn+32wnAoGAPBQ3
7XHg5NL0S4326eFRUT+4ornQFjJjP6dp3p0BEzpImNmZTtkCNNUKE4Go9hv5T41h
R0p0DvWv0CBupMAZVBP90bp1XPCyEIZtuDqVa7ukP0UpQNgQhLLAqkigTyXV0Smt
ECBj9tu5WNP/x3iTZTHJ+g0rhIqpgh012UwJpKADgYQAAoGAV10EQPYUg5/M3xf
6vE7jKTxxyFWEyjKfJK7PZCz0IGrE/swgACy4PYQW+AwcUweSlK/Hx20aZVUKzWo
wDUbeu65DcRdw2rSwCbBTU342sitFo/iGCV/Gjf+BaiAJtxniZze7J1ob8v0BeLv
uaMQmg0YeZ5e0f104GtqP1+lhcQwCQYHkoZIZjgEAWmWADAtAhQdoeWlRkm0K49+
AeBK+j6m2h9SKQIVAIBNhS2a8cQVABDCQXVXrc0t0m08
-----END CERTIFICATE-----

```

## RSA

```

-----BEGIN CERTIFICATE-----
MIICNjCCAZ+gAwIBAgIJA0Z3GEIaDcugMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXlYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xOTEwMjky
NTE5MDlaGA8yMTk5MMDMyOTE1MTkwOVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhhdGUxEDA0BgNVBAClTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWlgaU2Vydm1jZXMgTEExDMIGFMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKB
gQCjiPgW3vsXRj4JoA16WQDyoPc/eh3QBARaApJEC4nPIGoUo1pAXcjFhWp1o20+
ivgfCsc4AU90pYdAPha3spLey/bhHPri1JZHRNqSckP0hzcCNmKhfnZTIEQCFvsp
DRp4zr91/WS06/f1JFBYJ6JHhp0KwM81XQG591V6kkow7QIDAQABMA0GCSqGSIb3
DQEBCwUAA4GBAGLLrY3P+HH6C57dYgtJkuGZGT2+rMkk2n81/abzTJvsqRqGRrWv
XRKRX1KdM/dfiuYGokDGxiC0Mg6TYy6wvsR2qRhtXW10tZkiHwQCn0ttz+8vpew
wx8JGMvowtuKB1iMsbwyRqZkFYLcvH+Opfb/Aayi20/ChQLdI6M2R5VU
-----END CERTIFICATE-----

```

## RSA-2048

```

-----BEGIN CERTIFICATE-----

```

```

MIID0zCCAiOgAwIBAgIJA0/+DgYF78KwMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEyBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xOTA0Mjky
MDM1MjJaGA8yMTk4MTAwMjIwMzUyMlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAGT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACyB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWVudmVjZXMgTEExDMIIIBIjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIB
CgKCAQEAv1ZLV+Z/P6INq+R1qLkzETBg7sFGKPiwHekbpuB61rRxKhhj8V9vaReM
lnv1Ur5LAPpMPYDsuJ4WoUbPYAqVqyMAo7ikJHCCM1cXgZJefgN6z9bpS+uA3YVh
V/0ipHh/X2hc2S9wvxKWiSHu6Aq9GVpqL035tJQD+NJuqFd+nXrtcw4yGtmvA6w1
5Bjn8WdsP3x0TKjrByYY1BhXpP/f1ohU9jE9dstsRXL+XTgTPWcWdCS2oRTWPGR
c5Aeh47nnDsyQfP9gLxHeYeQItV/BD9kU/2Hn6mnRg/B9/TYH8qz1RTzLapXp4/5
iNwusrTNexG18BgvAPrfhjDpdgYuTwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQB7
5ya11K/hKgvaRTvZwV8G1VZt0CGPtNv0i4AR/UN6Tmm51BzUB5nurB4z0R2MoYO
Uts9sLGVsFALJ4otoB77hyNpH3drttU1CVVwal/yK/RQLSon/IoUkaGEbqalu+mH
nYad5IG4tEbmeP456XXc058MKmnczNbPyw3FRzUZQtI/sf94qBwJ1Xo6XbzPKMy
xjL57LHZCcssD+XPifXay690FlsCIgLim11HgPkRIHE0XLSf3dsW9r+4CjoZqB/Z
jj/P4TLCxbYCLkvg1waMjgEWF40Img0fhx7yT2X92MiSrs3oncv/IqfdVTiN80Xq
jgnq1bf+EZEZKvb6UCQV
-----END CERTIFICATE-----

```

## 歐洲 (巴黎) — eu-west-3

### DSA

```

-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgqhkiG00AQMDFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEyBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xMjAxMDUxMjU2MTJhFw0z
ODAxMDUxMjU2MTJhMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEyBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbcwggEsBgqhkiG00AQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJ1/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buyCU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
1Ra2v1ntMX3caRVDdbtPEWmdxSCySfYFDk4mZr0LBA4GEAAKBgEbmeve5f8LIE/Gf
MNMp9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKm11qx411HW
MXrs3IgIb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAKGBYqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----

```

## RSA

```

-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUaC9fX57UDr6u1vBvsCsECKBZQyIwDQYJKoZIhvcNAQEL
BQAwXDELMakGA1UEBhMCMVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBZXWlGUmU2Vydm1jZXMgTE
MB4XDTE0MDQyOTE2MzczOFoXDTI5MDQyODE2MzczOFowXDELMakGA1UEBhMCMVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBZXWlGUmU2Vydm1jZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJ0+eIB
UqPfQG09kZ1wpWpmy08bGB2RWqWxCwB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnf0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHGDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUuizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWdwUuizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTA1VTMRkwFwYDVoQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVoQHEwdT
ZWF0dGx1MSAwHgYDVoQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUaC9fX57U
Dr6u1vBvsCsECKBZQyIwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQCARv1bQEDaMEzYI0nPlu8GHcMXgmgA94HyrXhMMcaIlQwocGBs6VILGVhM
TXP2r3JfFaPEpmXSQNQHvGA13c1KwAZbni8wtzv6qXb4L4muF34iQRHF0nYrEDoK7
mMPR8+oXKKuP0/mv/XKo6XAV5DDERdSYHX5kkA2R9wtvyZjPnQ==
-----END CERTIFICATE-----

```

## RSA-2048

```

-----BEGIN CERTIFICATE-----
MIID0zCCAiOgAwIBAgIJALWSfgHuT/ARMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVoQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVoQHEwdTZWF0
dGx1MSAwHgYDVoQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNzA1MzEx
MTE4MTZaGA8yMTk2MTEwMzExMTgXN1owXDELMakGA1UEBhMCMVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBZXWlGUmU2Vydm1jZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUAAQ8AMIIB
CgKCAQEAY5V7KDqnEvF3DrSProFcgU/oL+QYD62b1U+Naq8aPuljJe127Sm9WnWA
EBd0SASk0aQ9fzjCPoG5SGgWkxYoZjsevHpmzjVv9+Ci+F57bSuMbjgUbbvRIFUB
bxQojVoXQPHgK5v4330DxkQ4sjRyUbf4YV1AFdfU7zabC698YgPV0ExGhXP1Tvco
8mlc631ubw2g52j01zaozUkHPSbknTomhQIv06kUfX0e0TDMH4jLDG2ZIrUB1L4r
0WKG4KetduFrRZyDHF6ILZu+s6ywiMicUd+2U11DFC6oas+a8D11hm0/rpWU/ieV
jj4rWAFrsebnp+Nhgy96iiVUGS2LuQIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQDE
iYv6FQ6knXCg+svlcaQG9q59xUC5z8HvJZ1+SxzPKK4PKQdKvIIfE8GxVXq1ZG1
c15WKTfDMapnzb9RV/DTaVzWx3cMYT77vm1H11XGjhx611CGcENH1egI310TILsa
+KfopuJEQ09TDMAIkGjha+KieU/U5Ctv9fdej6d0GC60EuwKkTNzPWue6UMq8d4H
2xqJboWsE1t4nybEosvZfQJcZ8jyIYcYBnsG13vCLM+ixjuU5MVVQNMV/gBJzqJB
V+U00QiGiuT5cYgY/QihxdHt99zwGaE0ZBC7213NKr1NuLSrghDI2NLU8NsExq0Fy
OmY0v/xVmQUQ126jJXaM
-----END CERTIFICATE-----

```

```
-----END CERTIFICATE-----
```

## 歐洲 (西班牙) — eu-south-2

### DSA

```
-----BEGIN CERTIFICATE-----
```

```
MIIC8DCCAq
+gAwIBAgIGAXjwLk46MAkGByqGSM44BAMwXDELMakGA1UEBhMCMVVMxGTAXBgNVBAGMEFdhc2hpbmd0b24gU3RhdGUxED
U4EddRIpUt9KnC7s50f2EbdSP09EAMMeP4C2USZpRV1AI1H7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdirmVC1pJ+f6AR7ECLCT7up1/63xhv401fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzrith1yrv8iIDGZ3RSAHHAhUA12BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v60uqC+VdMCz0HgmdRWVe0utRZT
+ZxBxCBgLRJFnEj6EwoFh03zwyjMim4TwWeotUfI0o4K0uHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/hWuWfBpKLZ16Ae1U1ZAFM0/7PSSoDgYQAaOgAGG2m8EKmaf5qQqj3Z
+rzSaTaXE3B/R/4A2VuGqRYR7M1jPtwdmU6/3CPjCACcZmTic0AKbFiDHqadQgBZXfzGpzw8Zo
+eYmmk5fXycgnj57PYH1dIWU6I7mCbAah5MZMcmHaTmIsomGrhcnWB8d8q0U7oZ0UWK41biAQs1MihoUwCQYHKoZiZjg
WmbaU7YM5GwCFCvIJ0es05hZ8PHC52dAR8WWC6oe
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
```

```
MIICmzCCAzygAwIBAgIGAXjwLkiaMA0GCSqGSIb3DQEBBQUAMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIDDBBYXNoaW5m
VvR1+45Aey5zn3vPk6xBm5o9grSDL6D2iAuprQnfVXn8CIbSdbWFhA3fi5ippjKkh3s18VyCvCOUXKd0aNrYBrPRkrdH
+3m/
rxIUZ2IK1fD1C6sWAjddf6sBrV2w2a78H0H8EwuwiSggtURBjwJ7KPPJCqaqrQIDAQABMA0GCSqGSIb3DQEBBQUAA4GB
+FzqQDzun/
iMMzcFucmL15BxEb1rFX0z7IIu0eiGkndmrquEDCykztLku45s7hxdNy41tTuVAaE5aNBdw5J8U1mRvsKvHLY2Th6h
+hBgiphYp84DubWVYeP8YqLEJSqscKscWC
-----END CERTIFICATE-----
```

### RSA-2048

```
-----BEGIN CERTIFICATE-----
```

```
MIIEEjCCAvqgAwIBAgIJALWsm06DvSpQMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIExMQzAgFw0yMjA3MTgx
MzU4NDNaGA8yMjAxMTIyMjEzNTg0M1owXDELMakGA1UEBhMCMVVMxGTAXBgNVBAGT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBZXWV2Vydm1jZXMgTEExMjEiANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
```



```

CgKCAQEAAhuSpsHC00/fD2zN1BDpNLRndi9qbHsNeuz3WqN7Samj2aSrM2hS+i
hUxx0BspZj0tZC0sbpPZ+i74N0EQtFeqQoEGvKhB1nJiF4y5I81HDhs5qHvoIivm
7rbbik3zgm1PqS/DmDjVQaXPcD31Rd9ILwBmWEwJqHigyNV1xYtCzTQcrlBrvNZM
dnNgCDAAdX/HBEFxx9012xeu0bSt0s+PJWZ1RTbYrNe7LIH6ntUqHxP/ziQ5trXEZ
uqy7aWk1L8uK4jmyNph01baqBa3Y6pYmU1nC27UE4i3fnPB0LSiAr+SrwVvX1g4z
i1o8kr+tbIF+JmcgYLBv08Jwp+EUqQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQUwvGzKJL9A5LReJ4Fxo5K6I20xcowgY4GA1UdIwSBhjCBg4AUwvGz
KJL9A5LReJ4Fxo5K6I20xcqhqYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDQVQIEExBX
YXNoaW5ndG9uIFN0YXR1MRAwDgYDQVQHEwdTZWF0dGx1MSAwHgYDQVQKExdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQ4IJALWSm06DvSpQMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAJAZd31jyoTGLawAD2+v/vQsaB9vZIx5EImi
G8YGkd61uFwEhAmtrwyE/i6FDSIphDrMHBkw/D3BsqK+Ev/JOK/VYuaYDx/8fp
H4cwp9jC57CXzdIDREWNf6M9PsHFg2WA9XNNtC10ZL5WJiJwel8eDSg+sqJUxEO1
MW+QChq/20F6niyaRK4bXrZq14as7h+F9u3A9xHE0VP7Zk9C2ehrBXZCMLSdt3GV
fEuMea2RxMhoz34Hkdb6j18qoCfygubulovRNQjKw/cEmgPR16KfZPP5caILVt
9qkYPvePmbiVswZDee73cDymJYxLqILp0ZwyXvUH8StiH42FHZQ=
-----END CERTIFICATE-----

```

## 歐洲 (斯德哥爾摩) — eu-north-1

### DSA

```

-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgqhkJ00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDQVQIEExBXNoaW5ndG9uIFN0YXR1MRAwDgYDQVQHEwdTZWF0dGx1MSAwHgYD
VQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xMjAxMDUxMjU2MTJhFw0z
ODAxMDUxMjU2MTJhMFwxCzAJBgNVBAYTA1VTMRkwFwYDQVQIEExBXNoaW5ndG9u
IFN0YXR1MRAwDgYDQVQHEwdTZWF0dGx1MSAwHgYDQVQKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbcwggEsBgqhkJ00AQBMIIbHwKBGQCjKvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nvwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buyCU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmeve5f8LIE/Gf
MNMp9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKm11qx411HW
MXrs3IgIb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDlwAwLAIUWXB1k40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----

```

## RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAOqgAwIBAgIUN1c9U6U/xiVDFgJcYKZB4NkH1QEwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBZXWIGU2Vydm1jZXMgTExD
MB4XDTI0MDQyOTE2MDYwM1owXDTI0MDQyOTE2MDYwM1owXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBZXWIGU2Vydm1jZXMgTExDMIGfMA0GCsGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJO+eIB
UqPfQG09kZ1wpWpmy08bGB2RWqWxCwB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnf0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIhCMA5GA1UdDwQEAwIHGDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUuizvtUF2
UTgwZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWdwUuizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTA1VTMRkwFwYDQVQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDQVQHEwdT
ZWf0dGx1MSAwHgYDQVQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUN1c9U6U/
xiVDFgJcYKZB4NkH1QEwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQBTIQdoFSDRHkqpNPUBz9WXR205v/9bpmHojMYZb3Hw46wsaRso7STiGGX/
tRqjIkPUIXsdhZ3+7S/RmhFznmZc8e0bjU4n5vi9CJtQSt+1u4E17+V2bF+D3h/7
wcfE013414Q8JaTDtEf/aF3F0uyBvr4MDmD7mFvAMmDmBPS1A==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIID0zCCAiOgAwIBAgIJALc/uRngx++EnMA0GCsGSIb3DQEBcUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDQVQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDQVQHEwdTZWf0
dGx1MSAwHgYDQVQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xODA0MTAx
NDAwMTFaGA8yMTk3MDkxMzE0MDAxMVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBZXWIGU2Vydm1jZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIB
CgKCAQEazwCGJEJIxqtr2PD2a1mA6LhRzKhTBa1AZsg3eYfpETXIVlrpojMfvVoN
qHvGshWLGrrGTT6os/3gsaADheSaJKavxwX3X6tJA8fvEGqr3a1C1MffH9hBwBQqC
LbfUTAbkwis4GdTUw0wPjT1Cm3u9R/VzilCNwkj7iQ65AFaI8Enmsw3UGldEsop4
yChKB3KW3WI0FTh0+gD0YtjrqqYJxpG0YBpJp5vwd3fZ4t1vidmDMs7liv4f9Bx
p0oSmUobU4GULFhBchK1DukICVQdn0VzdMonYm7s+HtpFbVHR8yf6QoixBKGdSal
mBf7+y0ixjCn0pnC0VLVooGo4mi17QIDAQABMA0GCsGSIb3DQEBcUAA4IBAQDG
40NZiixgk2sjJctwbyD5WKLTH6+mxYcDw+3y/F0fwz561YORhP2FNnPOmEkf0S1/
Jqk4svzJbCbQeMzRoyaya/46d7UioXMHRZam5IaGBh0dQbi97R4VsQjwQj0RmQsq
yDueDyuKTWwLK9KvI+ZA6e6bRkdNGf1K4N8GGKQ+fBhPwVELkbT9f160JkezeeN
S+F/gDADGJgmPXFjogICb4Kvshq0H5Lm/xZ1DULF2g/cYhyNY6E0I/eS5m1I7R8p
D/m6WoyZdpInxJfxW6160MkxQMRVsruLTNGtby3u1g6ScjmpFtvAMhYeJBSdzKG4
FEyxIdEjoe01jhTsck3R
```

```
-----END CERTIFICATE-----
```

## 歐洲 (蘇黎世) — eu-central-2

### DSA

```
-----BEGIN CERTIFICATE-----
```

```
MIIC7zCCAq
+gAwIBAgIGAXjXiKJnMAKGBYqGSM44BAMwXDELMakGA1UEBhMCMVVMxGTAXBgNVBAGMEFdhc2hpbmd0b24gU3RhdGUxED
U4EddRIpUt9KnC7s50f2EbdSP09EAMMeP4C2USZpRV1AI1H7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdirmVClpJ+f6AR7ECLCT7up1/63xhv401fnxqimFQ8E
+4P208UewwI1VBNAfEy9nXzriith1yrv8iIDGZ3RSAHHAhUA12BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v60uqC+VdMCz0HgmdRWVeOutRZT
+ZxBxCBgLRJfEnEj6EwoFh03zwyjMim4TwWeotUfI0o4K0uHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/hWuWfBpKLZ16Ae1U1ZAFM0/7PSSoDgYQAAoGAYNjaCNg/
cfqQ011BUj5C1Uu1qwZ9Q+SfDzPZH9D2C0VbiRANiZoxrV8RdgmzzC5T7VcriVwjvvt2Ch//
b+sZ86E5h0XWwR+BeEjD9cu3eDj12XB5sWEbNHNx49p5Tmtu5r2LDt1L8X/
Rpfalu2Z20JgjFJWGf7hRwx456n
+lowCQYHkoZizjgEAwMvADAsAhRChsLcj4U5CVb2cp5M0RE1XbXmhAIUeGSnH+aiUQIWmPEFja+itWDufIk=
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
```

```
MIICmzCCAZygAwIBAgIGAXjSGFGiMA0GCSqGSIb3DQEBBQUAMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIDDBBXNoaW5m
opKZAUusJx2hpgU3pUhh1p9ATh/VeVD582jTd9IY
+8t5MDa6Z3fGliByEiXz0LEHdi8MBacLREu1TwIDAQABMA0GCSqGSIb3DQEBBQUAA4GBAILlpoE3k9o7KdALAxSfJNi
+g3RMzdbiFM+7MA63Nv5fsf+0xgcjSNBE1vPCDKFvTJ14QQhToy0561105GvdS9RK
+H8xrP2mrqngApoKTApv93vHBixgFSn5KrczR00YSm30jkqbydU7DF1mkXXR7GYE+5jbHvQHYiT1J5sMu
-----END CERTIFICATE-----
```

### RSA-2048

```
-----BEGIN CERTIFICATE-----
```

```
MIIEEjCCAvqgAwIBAgIJALvT012pxTxNMA0GCSqGSIb3DQEBwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKEXdBbW6b24gV2ViIFN1cnZpY2VzIExMQzAgFw0yMjA3MTgx
NTEyMDdaGA8yMjAxMTIwN1owXDELMakGA1UEBhMCMVVMxGTAXBgNVBAGT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWlgaU2Vydm1jZXMgTEExIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAYn+Lsnq1ykrfY1Zkk6aAAYNRend9Iw8AUwCBkg0r2eBiBBepYxHwU85N
++moQ+j0EV2VaahBeTLShGZZS1HsyK8+cYT2QzpgHioamcYhrPXyIxlWiRQ1aqSg
```

```

OFiE9bsqL3rCF5Vz+t0iTe5W/7ojf0Fls6++g7ZpobwJlpMbuJepqyeHMPyjv05A
age81lJewc4bxo2ntaW0HCqNksqfYB78j6X6kn3PFpX7FaYAwZA+Xx6C7UCY7rNi
UdQzfAo8htfJi4chz7frpUdQ9k13I0QrsLshBB5fFUj109NiFipCGBwi+8ZMeSn1
5qwBI01BWXPFg7WX60wyjhmh6JtE1wIDAQABo4HUMIHRMASGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU8HN4vvJrsZgPQeksMBgJb9xR1yYwgY4GA1UdIwSBhjCBg4AU8HN4
vvJrsZgPQeksMBgJb9xR1yahYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEsBX
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQ4IJALvT012pxTxNMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAG1HYDtchPfbvdHx9HQE8HgNugJUPdEqxun
t9U33p8VFrs+uLPtr0d9HDJEGvvs5h84EUie/oGJxRt7V1Vlid1PvHf6cRmpjgqY
YdggAVkZtY/PnFVmzf2bMV1SQPrqCl7U0zaw2Kvnj4zgX0rZyCetgrRZSUSxotyp
978WY9ccXwVSeYG/YAr5rJpS6ZH7ERQvUY0IzwFNea0Pg0TEVpcjW1V6+MQEvsEx
W85q+s6AVr49eppEx8SLJs10C23yB+L+t32tAveQImRwTJmpzZ5cxh/sYgDVeOC0
85H1NK/7H9fAzT1cPu1oHSnB0xYzzHG0AmXmusMfwUk8fL1RQkE=
-----END CERTIFICATE-----

```

## 以色列 (特拉維夫) — il-central-1

### DSA

```

-----BEGIN CERTIFICATE-----
MIIC7zCCAq+gAwIBAgIGAX0QPi
+9MAkGBYqGSM44BAMwXDELMakGA1UEBhMCMVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACMB1N1YX
U4EddRIpUt9KnC7s50f2EbdSP09EAMMeP4C2USZpRV1AI1H7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdirmVC1pJ+f6AR7ECLCT7up1/63xhv401fnxqimFQ8E
+4P208UewwI1VBNAFpEy9nXzriith1yrv8iIDGZ3RSAHHAhUA12BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v60uqC+VdMCz0HgmdRWVe0utRZT
+ZxBxCBgLRJFnEj6EwoFh03zwyjMim4TwWeotUfI0o4K0uHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/
hWuWfBpKLZ16Ae1ULZAFM0/7PSSoDgYQAAoGAbazCL5XXyPmcw3+oMYQUF5/9YogW6D0FZbYuyPgj0oUwWdl6fj1zWca
pq+11ezuK2DF0zNTEyPEwwCQYHKoZIZjgEAWMvADAsAhRt1jKpXsvrS
+XTo2M9h2s2uLAhEQIU0Z2FcnTSrshF2EIdixZZwtNv66Q=
-----END CERTIFICATE-----

```

### RSA

```

-----BEGIN CERTIFICATE-----
MIICmzCCAZygAwIBAgIGAX0QQGVLMA0GCSqGSIb3DQEBBQUAMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIDDBBXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IJALvT012pxTxNMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAG1HYDtchPfbvdHx9HQE8HgNugJUPdEqxunt9U33p8VFrs+uLPtr0d9HDJEGvvs5h84EUie/oGJxRt7V1Vlid1PvHf6cRmpjgqY
YdggAVkZtY/PnFVmzf2bMV1SQPrqCl7U0zaw2Kvnj4zgX0rZyCetgrRZSUSxotyp
978WY9ccXwVSeYG/YAr5rJpS6ZH7ERQvUY0IzwFNea0Pg0TEVpcjW1V6+MQEvsEx
W85q+s6AVr49eppEx8SLJs10C23yB+L+t32tAveQImRwTJmpzZ5cxh/sYgDVeOC0
85H1NK/7H9fAzT1cPu1oHSnB0xYzzHG0AmXmusMfwUk8fL1RQkE=
-----END CERTIFICATE-----

```

```
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJA0Vp1h2I9wW7MA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXyXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0yMjA3MTUx
MjQ0MTJhGA8yMjAxMTIxOTExNDQxMlowXDELMAKGA1UEBhMVCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAClB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWlU2VydmljZXMgTEExDMIIIBjANBgkqhkiG9w0BAQEFAAOCQAQ8AMIIB
CgKCAQEA13PkyWv161iV/SYf01UF076UpDfPm2SF/Rz/o33cm699X++EYPxTnoEc
vmWeS0I7eDXc40CuiToG0sEx0k1E0CX1Z1tK6qJ+ZgWQLZ9SZEC9H0NsSA6LhrHu
Nq0dzeK3LjhdFcX46/4GqdiptpTuM4m/h0Q5yx4JMQ/n1sdpv4M5VLRWwWW9Lem
ufb79Id709SispXgRsz1KXIjp7N9S4BY7itSXz97uSyzTqEjWZ6mDUhTu3t21GKC
6f1ALGTTTrG2yghEhz53rkvLsvwzjPSS1T6LIff0mrRPzHaf+EdaKoasE1E1SHh+ZH
9mI81HywpE+HZ+W+5hBCvjYp90Y1fwIDAQABo4HUMIHRMAsGA1UdDwQEAwIHGDAd
BgNVHQ4EFgQU58tN2J0+yEGq5JbIXxGi4vRVPyIwgY4GA1UdIwSBhjCBg4AU58tN
2J0+yEGq5JbIXxGi4vRVPyKhYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExB
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQ4IJA0Vp1h2I9wW7MBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBANBN0e1EqNy4+IU2yQzMJ+Wy5ZIOtTP6GSBR
7muVY1bDeAwtNTE0pwgrZV1C7/xq5Q0LC1y0Z70hHXEf8au7qStaAoUtxzvHTAZI
NC01woFU56UFw4N0vZII17iqEfoqRC4PpI30xqEJHFy0VL1vAzJoKB4QLLqDAYVA
LXCi0LoVT+y9tRYSxw5My00Bi6fxQIIAD12bE9xkunTN1Jkkwqo3LxNy/ryz4QWR
8K7jHUItifv4h/hxBKpHEquN8CkdvM9oeG17I8PFrSFEpGr1euDX0euZzzYiDBV
m6GpTJgzpVsEuIX52dPcPemwQncoIfZyhWDW85MJUnby2WTEcFo=
-----END CERTIFICATE-----
```

## 中東 (巴林) — me-south-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7jCCAq4CCQCVWigSmP8RhTAJBgcqhkiG00AQMDFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEExBXyXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xOTAyMDUxMzA2MjFaFw00
NTAyMDUxMzA2MjFaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBXyXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbgwgEsBgcqhkiG00AQBMIIBHwKBgQDcwojQfgWdV1Q1i00B
8n6cLZ38VE7ZmrjZ90QV//Gst6S1h7euhC23YppKXi1zovefSDwFU54zi3/oJ++q
PH1P1WGL8IZ34BUgRTtG4TVo1vp0smjkMvyRu5hIdKtzjV93Ccx15gVgyk+o1IEG
-----END CERTIFICATE-----
```

```
fZ2Kbw/Dd8JfoPS7KaSCmJKxXQIVAIZbIaDFRGa2qcMkw2HWASyND17bAoGBANTz
Idhfmq+l2I5iofY2oj3HI21Kj3LtZrWEg3W+/4rVhL31Tm0Nne1r19yGujrjQwy5
Zp9V4A/w9w2010Lx4K6hj34Eefy/aQnZwNdNhv/FQP7Az0fju+Y16L1300HQrL0z
Q+9cF7zEosekEnBQx3v6psNknKgD3Shgx+G0/LpCA4GFAAKBgQCVS7m77nuNALZ8
wvUqcooxXMPkxJF154NxAsAu19KP9KN4svm003Zrb7t2F0tXRM8zU3TqMpryq1o5
mpMPsZDg6RXo9BF7Hn0DoZ6PJTamkFA6md+NyTJWJKvXC7iJ8fGDBJqTciUHuCKr
12AztQ8bFwSrTgTzPE3p6U5ckcgV1TAJBgcqhkj00AQDAy8AMCwCFB2NZGwm5ED1
86ayV3c1PEDukgQIAhQow38rQkN/VwHVeSW9DqEshXHjuQ==
-----END CERTIFICATE-----
```

## RSA

```
-----BEGIN CERTIFICATE-----
MIIDPDCCAqWgAwIBAgIJAM16uIV/zqJFMA0GCSqGSIb3DQEBCwUAMHIXCzAJBgNV
BAYTA1VTMRMwEQYDVQQIDApXYXNoaW5ndG9uMRAwDgYDVQQHDAdTZWF0dGx1MSAw
HgYDVQQKDBdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzEaMBGGA1UEAwwRZWMyLmFt
YXpvbmF3cy5jb20wIBcNMTEkNDI2MTQzMjQ3WhgPMjE5ODAwMjE5ODAwNDdaMHIX
CzAJBgNVBAYTA1VTMRMwEQYDVQQIDApXYXNoaW5ndG9uMRAwDgYDVQQHDAdTZWF0
dGx1MSAwHgYDVQQKDBdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzEaMBGGA1UEAwwR
ZWMyLmFtYXpvbmF3cy5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBALVN
CDTZEEnIeoX1SEYqq6k1BV0Z1pY5y3Kno0reCAE589TwS4MX5+8Fzd6AmAcMugeBP
Qk7Hm6b2+g/d4tWycyxLaQ1cq81DB1GmXehRkZRgGeRge1ePwD1TUA0I8P/QBT7S
gUePm/kANSFU+P7s7u1NN1+vynyi0wUUr7/wIZTAgMBAAGjgdwgdQwHQYDVR00
BBYEFILtMd+T4YgH1cgc+hVsV0V+480FMIGkBgNVHSMGZwwgZmAFILtMd+T4YgH
1cgc+hVsV0V+480FoXakdDBYMQswCQYDVQQGEwJVUzETMBEGA1UECAwKV2FzaGlu
Z3RvbjEQMA4GA1UEBwwHU2VhdHRsZTEgMB4GA1UECgwXQW1hem9uIFd1YiBTZXJ2
aWN1cyBMTEMxGjAYBgNVBAMTEWVjMi5hbWF6b25hd3MuY29tggkAyXq4hX/OokUw
DAYDVR0TBAAUwAwEB/zANBgkqhkiG9w0BAQsFAA0BgQBhKNTBIFgWfd+ZhC/LhRUY
40jEiykmbEp6hlzQ79T0Tfbn5A4NYDI2icBP0+hmf6qSnIhwJF6typyd1yPK5Fqt
NTpxxcXmUKquX+pHmIkK1LKD08rNE84jqxrxRsfdi6by82fjVYf2pgjJW8R1FAw+
mL5WQRFexbfB5aXhcMo0AA==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIID0zCCAiOgAwIBAgIJANZkF1QR2rKqMA0GCSqGSIb3DQEBCwUAMFwxGjAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXyXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzEaMBGGA1UEAwwR
MzA2MjBaGA8yMTk4MDcxMTEzMDYyMFowXDELMAKGA1UEBhMVCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAClB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbWVjZXJ2eUdlMjZXMgTEExMTEyMTEyMTEyMTEyMTEyMTEyMTEyMTEyMTEyMTEy
CgKCAQEAy4Vnit2eBpEjKgOKBmyupJzJAI4fr74tuGJNwwa+Is2vH12jMzn9I11
```

```
UpvvEUYTIboIgISpf6SJ5LmV5rCv4jT4a1Wm0kjfNbiI1kUi8SxZrPypcw24m6ke
BVuxQZrZDs+xDUYIZifTmdgD50u5YE+TLg+YmXKnVgxBU6WZjbuK2INohi71aPBw
2zWUR7Gr/ggIpf635JLU3KIBLNEmrkXCVSnDF1sK4eeCrB7+UNak+4BwgpuykSGG
Op9+2vsuNqFeU119daQeG9roHR+4rIWSPa0opmMxv5nctgyp0rE6zKXx2dNXQldd
VULv+WH7s6Vm4+yBeG8ctPYH5G0o+QIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQB5
ZcViiZdFdpcXESZP/KmZNDxB/kkt1IEIhsQ+MnN29jayE5oLmtGjHj5dtA3XNK1r
f6PVygVTKbtQLQqunRT83e8+7iCZMKI5ev7pITUQVvTUwI+Fc01JkYZxRF1VBuFA
WGZ0+98kxCS4n6tTwVt+nSuJr9BJRVC17apfHBgSS8c50Wna0VU/Cc9ka4eAfQR4
7pYSDU3wSRE01cs30q341XZ629IyFirSJ5TTOIc0osNL7vwMQYj8H0n40BYqxKy8
ZJyvfxsIPh0Na76PaBIs6ZlqA0f1LrjGzxBPiwRM/XrGmF8ze4KzoUqJEnK1306A
KHKgfiigQZ1+gv5F1yXH
-----END CERTIFICATE-----
```

## 中東 (阿拉伯聯合大公國) — me-central-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq
+gAwIBAgIGAXjXhqnnMAkGByqGSM44BAMwXDELMakGA1UEBhMCMVVMxGTAXBgNVBAGMEFdhc2hpbmd0b24gU3RhdGUxED
U4EddRIpUt9KnC7s50f2EbdSP09EAMMeP4C2USZpRV1AI1H7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdirmVClpJ+f6AR7ECLCT7up1/63xhv401fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzrith1yrv8iIDGZ3RSAHHAhUA12BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtCnRhXuXmUr7v60uqC+VdMCz0HgmdRWVeOutRZT
+ZxBxCBgLRJFEnEj6EwoFh03zwyjMim4TwWeotUfI0o4K0uHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/hWuWfBpKLZ16Ae1U1ZAFM0/7PSSoDgYQAAGAW+csuHsWp/7/
pv8CTKFwxsYudxuR6rbWaHCykIeAydXL9AWnphK6yp10DEMBF168Xq8Hp23s0WYf8mo0hqCom9+0+ovuUFdpvCie86bp
TOZU568Ty1ff3dDWbdRzeNQRHodRG+XEQSizMkAreeWt4kBa+PUwCQYHKOZIZjgEAWMvADAsAhQD3Z
+XGmzKmgALgGcVX/Qf1+Tn4QIUH1cgksBSVKbwj81tovBMJeKgdYo=
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIICMzCCAzygAwIBAgIGAXjRrnDjMA0GCSqGSIb3DQEBBQUAMFwxZAJBgNVBAYTA1VTMRkwFwYDVQQIDDBBYXNoaW5m
KyA6zyruJQrYy00a6wqLA7eeUzk3bMiTkLsTeDQfrkaZMFBAjGaa0ymRo1C3qzE4rIenmahvUp1u9ZmLwL1idwXMRX2R
+d2SeoK0KQWoc2U0FZMHYxDue7zkyk1CIRaBukTeY13/
RIr1c6X61zJ5BBtZX1HwayjQIDAQABMA0GCSqGSIb3DQEBBQUAA4GBABTqTy3R6RXKPW45FA+cgo7YZEj/
Cnz5YaoUivRRdX2A83BHUBTvJE2+WX00FTEj4hRVjameE1nEno08Z7fUVl0AFD1Do69fhkJeSvn51D1WRrPnoWGgEfr1
B+Wqm3kVEz/QNcz6npmA6
-----END CERTIFICATE-----
```

## RSA-2048

```

-----BEGIN CERTIFICATE-----
MIIIEjCCAvqgAwIBAgIJAM4h7b1CVhqqMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0yMjA0MTEw
MDE1MDNaGA8yMjAxMDkxNTEwMTUwM1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhZGUxEDA0BgNVBACTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWlgaU2Vydm1jZXMgTEExDMIIIBjANBgkqhkiG9w0BAQEFAAOCQA8AMIIB
CgKCAQEApYbTWFm0hSoMpqPo72eqAmnn1dXGZM+G8EoZXzwHwT/+IHEXNB4q5N6k
tudYLre1bJxuzEw+iProSHjmb9bB9YscRTofjVhBlT35Fc+i8BaMeH94SR/eE8Q0
m1l8gnLNW3d62lyuhzuyv1e5wV1RqzYw+X2zRH4/wRD0C0pzjKoHIgyPKsMgsw5
aTZhNMsGxZN9dbkf0iCGeQLDytwU/JTh/HqvSr3VfU0apTJJiyAxCtZWgp1/7wC
Rv0CSMRJobpUqxZgl/VsttwnkikSFz1wGkcYeSQvk+odbnYQckA8tdddoVI56eD4
qtREQvfpMAX5v7fcqLex15d5vH8uZQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU0adrBTs+0hzwoAgUJ7RqQNdWufkwyY4GA1UdIwSBhjCBg4AU0adr
bTs+0hzwoAgUJ7RqQNdWufmYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExB
XYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQ4IJAM4h7b1CVhqqMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAICTdA0GEOnII8HaGCpCB8us/hGFaLptJaAf
D5SJAyVy66/mdfjGzE1BkkKxnbxemEVUIzbRid0nyilB+pKwN3edAjTZtWdpVA0V
R/G/qQPmcV1jtycBz4VC6Su0UYf1GzLH1GZ6GJWbuDtFzw8r7HGdRN1wrEPe3UF2
sMpuVezqnRUdVVRoVQP4jFgNsE7kNvtN2NiPhb/CtrpcwIQ7r6YeoHcBSheuV1Z
xZDHynC3KUprQGx1+Z9QqPrDf180MaoqAlTl4+W6Pr2NJYrVUFGS/ivYshMg574l
CPU6r4wWZSKwEUXq4BInYX6z6iclp/p/J5QnJp2mAwyi6M+I13Y=
-----END CERTIFICATE-----

```

## 南美洲 (聖保羅) — sa-east-1

## DSA

```

-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgqhkiG9w0BAQ0DMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xMjA0MTEwMDE1MDNaGA8y
MjAxMDkxNTEwMTUwM1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTODAxMDUxMjA0
MTEwMDE1MDNaGA8yMjAxMDkxNTEwMTUwM1owXDELMAkGA1UEBhMCVVMxGTAXBgNV
BACTB1N1YXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBxZWlgaU2Vydm1jZXMgTEExDMII
IBjANBgkqhkiG9w0BAQEFAAOCQA8AMIIBBgkqhkiG9w0BAQ0DCCABcwggESBgqhkiG
9w0BAQ0DCCABMIIBHwKBgQCjkvcS2bb1VQ4yt/5eih5006kK/n1Lz1l1r7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U

```



```

hhy1KHVpCG19fueQ2s6IL0Ca0/buycU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmeve5f8LIE/Gf
MNMp9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKm11qx411HW
MXrs3IgIb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDlwAwLAIUWXB1k40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----

```

## RSA

```

-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUX4Bh4MQ86Roh37VDRRX1MNOB3TcwDQYJKoZIhvcNAQEL
BQAwXDELMAKGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBAClTB1N1YXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBZXWlGU2Vydm1jZXMgTEEx
MB4XDTI0MDQyOTE2NDYwOVVoXDTI0MDQyODE2NDYwOVowXDELMAKGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAClTB1N1YXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBZXWlGU2Vydm1jZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCChvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJ0+eIB
UqPfQG09kZ1wpWpmy08bGB2RWqWxCwB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHGDAdBgNVHQ4EFgQUJdbMCBXXtvCcwWdUUIzvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcwWdUUIzvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdT
ZWF0dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUX4Bh4MQ8
6Roh37VDRRX1MNOB3TcwEgYDVR0TAAQH/BAGwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQBnhocfH6ZIX6F5K9+Y9V4HFk8vSaaKL5ytw/P5td1h9ej94KF3xkZ5fyjN
URvGQv3kNmNJBONarcP9I7JIMjsNPmVzqWawyCEGCZImoARxSS3Fc5EAs2PyBfcD
9nCtzMTaK009Xyq0wqXVYn1xJsE5d5yBDsGrzaTHKjxo61+ezQ==
-----END CERTIFICATE-----

```

## RSA-2048

```

-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAMcyox4U0xxMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNTA4MTQw
ODU4MDJJaGA8yMTk1MDExNzA4NTgwM1owXDELMAKGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAClTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBZXWlGU2Vydm1jZXMgTEExDMIIIBjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIB
CgKCAQEAW45IhGZVbQcy1fHBqzR0h08Csrdzxj/WP4cRbJo/2DAnimVrCCDs5086
FA39Zo1xsDuJHD1wMKqeXYXkJXHYbcPwC6EYYAnR+P1LG+aNS0GUzsy202S03hT0
B20hWPCqpPp39itIrH64id6nbNRJ0zLm6evHuepMAHR4/0V7hyG0iGaV/v9zqiNA
pMCLhbh2xk0P035HCVBuWt3HUjsgeks2eEsu9Ws6H3JXTCfiqp0TjyRwapM290hA

```

```
cRfJ/d/+wBTz1fkW0Z7TF+EWRIN5ITEad1DTPnF1r8kBRuDcS/1IGFwr00HLo4C
cKoNgXkhTqDDBDu6oNBb2rS0K+sz3QIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAD
BgNVHQ4EFgQUqBy7D847Ya/w321Dfr+rBJGsGTwwgY4GA1UdIwSBhjCBg4AUqBy7
D847Ya/w321Dfr+rBJGsGTyhYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEsBX
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6
b24gV2ViIFNlcnZpY2VzIEExMQ4IJAMcyox4U0xxMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAC0oWSBf7b9A1cNr141r3QWwSc7k90/tUZa1
P1T0G30b12x9T/ZiBsQpbUvs01fotG0XqGVVHcIx3F8EbVwbw9KJGXBGSCJSEJKw
vGctc/jYMHXfhx675zmftm/MTYNvnzsyQQ3v8y3Rdah+xe1NPdpFrwmfL6xe3pFF
cY33KdHA/3PNLdn9CaEsHmcmj3ctaaXLFIZhQyyjtsrgGfTLvXeXRokktvsLDS/
YgKedQ+jFjzVJqgr4Njfy/Wt7/8kbbdhzaqlB5pCPjLLzv0zp/Xm06k+Jv0eP0Gh
JzGk5t1QrSju+MqNPfK3+107o910Vrhqw1QRB0gr1ExrviLbyfU=
-----END CERTIFICATE-----
```

## 中國 (北京) — cn-north-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIDNjCCAh4CCQD3yZ1w1AVkTzANBgkqhkiG9w0BAQsFADBcMQswCQYDVQQGEwJV
UzEZMBcGA1UECBMQV2FzaGluZ3RvbiBTdGF0ZTEQMA4GA1UEBxMHU2VhdHRsZTEg
MB4GA1UEChMXQW1hem9uIFdlYiBTZXJ2aWNlcyBMTEMwIBcNMTUwNTEzMDk10TE1
WhgPMjE5NDEwMTYwOTU5MTVaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEsBX
aW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6b24g
V2ViIFNlcnZpY2VzIEExMQzCCASiWdQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEB
AMWk9vyppSmDU3AxZ2Cy2bvKeK3F1UqNpMuyeriizi+NTsZ8tQqtNloaQcqhto/1
gsw9+QSnEJeYWnmivJW0Bdn9CyDpN7cpHVmeGgNjL2fvImWyWe2f2Kq/BL917N7C
P2ZT52/sH9orlck1n2z08xPi7MIItgPHQwu30xsGQsAdWucdxjHGtdchulpo1uJ31
jsTAPKZ3p1/sxPXBBAgBMatPHhRBqhwH0/Twm4J3GmTLWN7oVDds4W3bPKQfnw3r
vtBj/SM4/IgQ3xJs1Fc190TZbQbgxIi88R/gWTbs7GsyT2PzstU30yLdJhKfdZKz
/aIzraHvoDTWfa0dy0+00aECAwEAATANBgkqhkiG9w0BAQsFAA0CAQEAdSzn2+0E
V1BfR3DPWJHWRf1b7z1+1X/ZseW2hYE5r6YxrLv+1VPf/L5I6k7GEtqhZUqteY7
zAceoLrVu/70ynRyfQetJVGichaaxLNM31cr6kcx0owb+WQQ84cwrB3keykH4gRX
KHB2r1WSxta+2panSE01JX2q5jhcFP90rD0tZjlpYv57N/Z9iQ+dvQPJnChdq3BK
5pZ1nIDnVVxqRike7BFy8tKyPj7HzoPEF5mh9Kfnn1YoSVu+611MVv/qRjnyKfS9
c96nE98sYFj0ZVBzXw8Sq4Gh8FiVmFhbQp1peGC19id0UqxPxWsasWxQX00azYsP
9RyWLHKxH1dMuA==
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
```

```

MIIDCzCCAnSgAwIBAgIJALS0Mb0oU2svMA0GCSqGSIb3DQEBCwUAMFwx CzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXYYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0yMzA3MDQw
ODM1MzlaFw0yODA3MDIwODM1MzlaMFwx CzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBX
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQzCBnzANBqkqhkiG9w0BAQEFAA0BjQAwwYkCgYEA
uhhUN1qAZdcwWB/OSDVGk30A99EFz0n/mJlmcIQ/Xwu2dFJWmSCqEAE6gjuFCjQ
q3voxAhC2CF+e1KtJW/C0Sz/LYo60PUqd6iXF4h+upB9Hk00GuWHXsHBTsvgkgGA
1CGge14U0Cdq+23eANr8N8m28Uz1jjSnT1rYCHtzN4sCAwEAAaOB1DCB0TALBgNV
HQ8EBAMCB4AwHQYDVR00BBYEFBkZu3wT27NnYgrfH+xJz4HJaNJoMIG0BgNVHSMG
gYYwgY0AFBkZu3wT27NnYgrfH+xJz4HJaNJoWckXjBcMQswCQYDVQQGEwJVUzEZ
MBCGA1UECBMQV2FzaGluZ3RvbiBTdGF0ZTEQMA4GA1UEBxMHU2VhdHRsZTEgMB4G
A1UEChMXQW1hem9uIFd1YiBTZXJ2aWw1cyBMTE0CCQ0jjGzqFNrLzASBgNVHRMB
Af8ECDAGAQH/AgEAMA0GCSqGSIb3DQEBCwUAA4GBAECji43p+oPkYqzm117e8Hgb
oADS0ph+YUz5P/bUCm61wFj1xaTfwKcuTR3ytj7bFLow5Bm7Sa+TC1310Gb2taon
2h+9NirRK6JYk87LMNvbS40HGPFumJL2NzEsGUEK+MRiWu+0h5/lJGii3qw4YByx
SUD1RyNy1jJFstEZj0hs
-----END CERTIFICATE-----

```

## RSA-2048

```

-----BEGIN CERTIFICATE-----
MIID0zCCAi0gAwIBAgIJJA0trM5XLDsjCMA0GCSqGSIb3DQEBCwUAMFwx CzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXYYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNTA4MTQx
MDAxNDJaGA8yMTk1MDExNzEwMDE0M1owXDELMAkGA1UEBhMVCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACjTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWV2VydmljZXMgTExDMiIBIjANBgkqhkiG9w0BAQEFAA0CAQA8AMIIB
CgKCAQEAvBz+WQNdpM9S+aUUL0QEriTmNDUrrjLWlr7Sfa0JScBzis5D5ju0jh1
+qJdkbuGktFX50TWtm8pWhInX+hIo0S3exC4BaANoa1A3o6quoG+Rsv72qQf8LLH
sgEi6+LM1CN9TwnRK0ToEabmDKorss4zF17VSsbQJwcBSf0cIwbdRRaW9Ab6uJHu
79L+mBR3Ea+G7vSDrVIA8goAPkae6jY9WGw9Kxs0rcvNdQoEkqRVtHo4bs9fMRHU
Etphj2gh40bX1FN92VtvzD6QB3CcoFWgyWgvzg+dNG5VCbsiiuRdmii3kciZ3H
Nv1wCcZoEAqH72etVhsuvNRC/xAP8wIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQA8
ezx5LRjzUU9EYWyhyYIEShF1P1qDhs7F4L46/51c4pL8FPoQm5CZuAF31DJhYi/b
fcV7i3n++/ymQbCLC6kAg8DUB7NrcR0115ag8d/JXGzcTCn1DXLXx1905fPNa+jI
0q5quTmdmiSi0taeaKZmyUdhrB+a7ohWdSdlokEI0tbH1P+g5y113bI21eYE6Tm8
LKbyfK/532xJPq09abx4Ddn89ZEC6vvWVNDgTsxERg992Wi+/xoSw3XxkgAryIv1
zQ4dQ6irFmXwCWJqc6kHg/M5W+z60S/94+wGTXmp+19U6Rkq5jVMLh16XJXrXwHe
4KcgIS/aQGVgjM6wivVA
-----END CERTIFICATE-----

```

## 中國 (寧夏) — cn-northwest-1

## DSA

```
-----BEGIN CERTIFICATE-----
MIIDNjCCA4CCQD3yZ1w1AVkTzANBgkqhkiG9w0BAQsFAADBCMQswCQYDVQQGEwJV
UzEZMBcGA1UECBMQV2FzaGluZ3RvbiBTdGF0ZTEQMA4GA1UEBxMHU2VhdHRsZTEg
MB4GA1UEChMXQW1hem9uIFd1YiBTZXJ2aWNLcyBMTEMwIBcNMTUwNTEzMDk10TE1
WhgPMjE5NDEwMTYw0TU5MTVaMFwxZzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBXlYXNo
aW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6b24g
V2ViIFN1cnZpY2VzIEExMQzCCASiWdQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEB
AMWk9vyppSmDU3AxZ2Cy2bvKeK3F1UqNpMuyeriizi+NTsZ8tQqtNloaQcqhto/1
gsw9+QSnEJeYWnmivJW0Bdn9CyDpN7cpHVmeGgNJL2fvImWyWe2f2Kq/BL917N7C
P2ZT52/sH9orlck1n2z08xPi7MItgPHQwu30xsGQsAdWucdxjHGtdchulpo1uJ31
jsTAPKZ3p1/sxPXBBAgBMatPHhRBqhwH0/Twm4J3GmTLWN7oVDds4W3bPKQfnw3r
vtBj/SM4/IgQ3xJs1Fc190TZbQbgxIi88R/gWTbs7GsyT2PzstU30yLdJhKfdZKz
/aIzraHvoDTWfa0dy0+00aECAwEAATANBgkqhkiG9w0BAQsFAA0CAQEAdSzN2+0E
V1BfR3DPWJHWRf1b7z1+1X/ZseW2hYE5r6YxrLv+1VPf/L5I6kB7GEtqhZUqteY7
zAceoLrVu/70ynRyfQetJVGichaaxLNM31cr6kcx0owb+WQQ84cwrB3keykH4gRX
KHB2r1WSxta+2panSE01JX2q5jhcFP90rD0tZjlpYv57N/Z9iQ+dvQPJnChdq3BK
5pZlnIDnVVxqRike7BFy8tKyPj7HzoPEF5mh9Kfnn1YoSVu+611MVv/qRjnyKfS9
c96nE98sYFj0ZVBzXw8Sq4Gh8FiVmFhbQp1peGC19id0UqxPxWsasWxQX00azYsP
9RyWLHKxH1dMuA==
-----END CERTIFICATE-----
```

## RSA

```
-----BEGIN CERTIFICATE-----
MIIDCzCCANsGawIBAgIJALS0Mb0oU2svMA0GCSqGSIb3DQEBCwUAMFwxZzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXlYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0yMzA3MDQw
ODM1MzlaFw0yODA3MDIwODM1MzlaMFwxZzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBX
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQzCBnzANBgkqhkiG9w0BAQEFAA0BjQAwwYkCgYEA
uhhUN1qAZdcWwB/0SDVDGk30A99EFz0n/mJlmcIQ/Xwu2dFJWmSCqEAE6gjufCjQ
q3voxAhC2CF+e1KtJW/C0Sz/LYo60PUqd6iXF4h+upB9Hk00GuWHXsHBTsvgkgGA
1CGge14U0Cdq+23eANr8N8m28Uz1jjSnTlrYCHtzN4sCAwEA0B1DCB0TALBgNV
HQ8EBAMCB4AwHQYDVR00BBYEFBkZu3wT27NnYgrfH+xJz4HJaNJoMIG0BgNVHSM
eGYYwgY0AFBkZu3wT27NnYgrfH+xJz4HJaNJoWcKXjBcMQswCQYDVQQGEwJVUzEZ
MBcGA1UECBMQV2FzaGluZ3RvbiBTdGF0ZTEQMA4GA1UEBxMHU2VhdHRsZTEgMB4G
A1UEChMXQW1hem9uIFd1YiBTZXJ2aWNLcyBMTE0CCQC0jjGzqFNrLzASBgNVHRMB
Af8ECDAGAQH/AgEAMA0GCSqGSIb3DQEBCwUAA4GBAECji43p+oPkYqmqz117e8Hgb
oADS0ph+YUz5P/bUCm61wFj1xaTfwKcuTR3ytj7bFLow5Bm7Sa+TC1310Gb2taon
```

```
2h+9NirRK6JYk87LMNvbS40HGPFumJL2NzEsGUEk+MRiWu+0h5/1JGii3qw4YByx
SUDlRyNy1jJFstEZj0hs
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIID0zCCAi0gAwIBAgIJAPu4ssY3B1zcMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNTEyMDMy
MTI5MzJaGA8yMTk1MDUwODIxMjkzMlowXDELMAKGA1UEBhMVCVVMxGTAXBgNVBAGT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWVudmUyVydmljZXMgTEExMjI1IjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAs0iGi4A6+YTLzCdIyP8b8SCT2M/6PGKwzKJ5XbSBoL3gsnSWiFYqPg9c
uJPNbiy9wSA9vlyfWmd90qvTfiNrT6vewP813QdJ3EENZ0x4ERcf/Wd22tV72kxD
yw1Q3I10MH4b0ItGQAxU50tXCjBZEEUZoo0kU8RoUQ0U2Pq14NTiUpzWacNutAn5
HHS7Mdc4lUlsJqbN+5QW6fFrcNG/0Mrib3JbwdFUNhrQ5j+Yq5h78HarnUivnX/3
Ap+oPbentv1qd7wvPJU556LZuhfqI0TohiIT1Ah+yUdN5osoaMxTHKktf/CsSJ1F
w3qXqFJQA0VwsqjFyHXFI32I/G0upwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQCn
Um00QHvUsJSN6KATbghowLynHn3wZSQuS8E0C0pcFJFxp2SV0NYkERbXu0n/Vhi
yq5F8v4/bRA2/xpedLWmvFs7QWlomuXhSnYFkd33Z5gnXPb9vRkLwiMSw4uX1s35
qQrarczUJ9EXDhrv7VmngIk9H3YsxYr1DGEqh/oz4Ze4UL0gnfkauanHikk+BUeSg
/jTD+7e+niEzJPihHdsVKFDlud5pakEzyxovHwNJ1GS2I//yxrJFIL91mehjqEk
RLPdNse7N6UvSnuXc0okwu6l6kfzigGkJBxkcq4gre3szZFdCQCuioj7Z4xtuTL8
YMqfiDtN5cbD8R8ojw9Y
-----END CERTIFICATE-----
```

## AWS GovCloud (美國東部) — -1 us-gov-east

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgqhkiG9w0BAQ0DMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xMjI1MDUxMjUyMTJJaFw0z
ODAxMDUxMjUyMTJJaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbcwggEsBgqhkiG9w0BAQ0BMIIIBHwKBGQCjKvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MbcJl/U
```

```

hhy1KHVpCG19fueQ2s6IL0Ca0/buycU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmeve5f8LIE/Gf
MNMp9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKm11qx411HW
MXrs3IgIb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDlwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----

```

## RSA

```

-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIULVyrqjjwZ461qe1PCiShB1KCCj4wDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBAClTB1N1YXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBZXWVjZm1jZXMgTEExD
MB4XDTI0MDUwNzE1MjIzN1oXDTI0MDUwNzE1MjIzN1owXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAClTB1N1YXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBZXWVjZm1jZXMgTEExDMIGFMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCpohwYUVPH9I7Vbkb3WMe/JB0Y/bmfVj3VpcK445YBR09K80a1
esjgBc2tAX4KYg4Lht4EBKccLHTzaNi51YEGX1aLNrSmxhz1+WtzNLNUsyY3zD9z
vwX/3k1+JB2dRA+m+Cpwx4mjzZyAeQtHtegVaAytkmqtxQrSCexBxvqRqQIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHGDAdBgNVHQ4EFgQU1ZXneBYnPVYXkHV1Vjg7918V
gE8wgZkGA1UdIwSBkTCBjoAU1ZXneBYnPVYXkHV1Vjg7918VgE+hYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdT
ZWV0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IULVyrqjjw
Z461qe1PCiShB1KCCj4wEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQBfAL/YZv0y3zmVbXjyxQCsD1oeDCJjFKIu3ameEckeIWJbST9LMto0zViZ
puIAf05x6GQiEqfBmk+YmXJfcTmJB4Ebaj4egFls1JPSHyC2xuydHlr3B04INOH5
Z2oCM68u6GGbj0jZjg7GJonkReG9N72kDva/ukwZKgg8zErQVQ==
-----END CERTIFICATE-----

```

## RSA-2048

```

-----BEGIN CERTIFICATE-----
MIID0zCCAi0gAwIBAgIJALPB6hxFhay8MA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xODA0MTAx
MjMyND1aGA8yMTk3MDkxMzEyMzI0VowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAClTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBZXWVjZm1jZXMgTEExDMIIIBIjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIB
CgKCAQEAv9xsI9237KYb/SPWmeCVzi7giKNron8hoRDwlwwMC9+uHPd53UxzKLb
pTgtJWAPkZVxEdl2Gdhw3SULoKcKmkqE61tVFrvuPT33La1UufguT9k8ZDDu09C
hQNHUdSVEuVrK3bLjaSsM0S7Uxmnn71YT990IREowvnnBNBsBlcabfQTBV04xfUG0
/m0XUiuFj0xBqbNzkEib1W7vK7ydSjtFMS1jga54UAVXibQt9EAIIF7B8k912iLa

```

```

mu9yEjyQy+ZQICTuAvPUEWe6va2CHVY9gYQLA31/zU0VBKZPTNExjaqK4j8bKs1/
7d0V1so39sIGBz21cUBec1o+yCS5SwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQBt
h02W/Lm+Nk0qsXW6mqQFsAou0cASc/vtGNCyBfoFNX6aKXsVCHxq2aq2TUKWENS+
mKmYu1LZVhB0mLshy1lh3RRoL30hp3jCwXytkWQ7E1cGjDzNGc0FArZB8xFyQNdk
MNvXDi/ErzgrHGSpcvmGHi0hMf3UzChMwbIr6udoD1MbSI07+8F+jUJkh4X111Kb
YeN5fsLZp7T/6YvbFSPpmbn1YoE2vKtuGKx0bRrhU3h4JHdp1Zel1pZ6l1h5iM0ec
SD11SximGIYCjfZpRqI3q50mbxCd7ckULz+UUPwLrf0ds4VrVVSj+x0ZdY19P1v2
9shw5ez6Cn7E3IfzqNH0
-----END CERTIFICATE-----

```

## AWS GovCloud (美國西部) — -1 us-gov-west

### DSA

```

-----BEGIN CERTIFICATE-----
MIIC7TCCAqQCCQCWukjz5V4aZzAJBgqhkJ00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEyBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQKKExdBbWF6b24gV2ViIFN1cnZpY2VzIEEMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEyBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQKKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEEMQzCCAbcwggEsBgqhkJ00AQBMIIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nvwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJ1/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buyCU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3carVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKml1qx41lHW
MXrs3IgIb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAKGBYqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----

```

### RSA

```

-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUe5wGF3jfb71UHvzDxmM/ktGCLwwwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBAcTB1N1YXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWlU2Vydm1jZXMgTEEx
MB4XDTE0MDUwNzE3MzAzM1oXDTE1MDUwNjE3MzAzM1owXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1N1YXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWlU2Vydm1jZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUA

```

```
A4GNADCBiQKBgQCpohwYUVP9I7Vbkb3WMe/JB0Y/bmfVj3VpcK445YBR09K80a1
esjgBc2tAX4KYg4Lht4EBKccLHTzaNi51YEGX1aLnrSmxhz1+WtzNLNUsyY3zD9z
vwX/3k1+JB2dRA+m+Cpwx4mjzZyAeQtHtegVaAytkmqtxQrSCexBxvqRqQIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQU1ZXneBYnPvYXkHV1Vjg7918V
gE8wgZkGA1UdIwSBkTCBjoAU1ZXneBYnPvYXkHV1Vjg7918VgE+hYKReMFwxCzAJ
BgNVBAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdT
ZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUe5wGF3jf
b71UHvzDxmM/ktGCLwwwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQCbtDpx1Iob9SwUReY4exMn1wQ1mkTLyA8tYGWzchCJOJJEPfsw0ryy1A0H
YIuvyUty3rJdp9ib8h3GZR71BkZnNddHhy06kPs4p8ewF8+d80Wt0JQcI+ZnFfG4
KyM4rUsBr1jpG2a0Cm12iACEyrvgJJrS8VZwUDZS6mZEnn/1hA==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIID0zCCAi0gAwIBAgIJANCOF0Q6ohnuMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNTA5MTAx
OTQyNDdaGA8yMTk1MDIxMzE5NDI0N1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhZGUxEDA0BgNVBAcTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWVjU2Vydm1jZXMgTExDMIIIBIjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIB
CgKCAQEAzIcGTzNqie3f1o1rrqcfzGfbymSM2QfbTzDI0G6xXXeFrCDAm0q0wUhi
3fRCuoeh1K0WAPu76B9os71+zgF22dIDEVkpqHCjBrGzDQZXXUw0zhm+PmBUI8Z1
qvbVD4ZYhjCujWwzrsX6Z4yEK7PEFjtf4M4W8euw0RmiNwjy+knIFa+VxK6aQv94
1W98URFP2fD84xedHp6ozZ1r3+RZSIFZs0iyxYsgiwTbesRMI0Y7LnkKGCiHQ/XJ
0wSISwaCddbu59BZEADnyh14f+pWaSQpQQ1DpXvZAVBYvCH97J1oAxLfH8xcwgSQ
/se3wtn095VBt5b7qTVj0vy6vKZazwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQA/
S8+a9csfASKdtQU0LsBynAbsBCH9Gykq2m8JS7YE4TGvq1pnWehz78rFTzQwmz4D
fwq8byPk16DjdF9utqZ0JUo/Fxelxom0h6oievtB1SkmZJNbgc2WYm1zi6ptViup
Y+4S2+vWZyg/X1PXD7wyRWuETmykk73uEyeWFBYKCHWs09sI+6204Vf8Jkuj/cie
1NSJX8fkervfLrZSHBYhxLbL+actVEo00tiyZz8GnhgWx5faCY38D/k4Y/j5Vz99
71UX/+fWHT3+1TL8ZZK7f0QWh6NQpI0wTP9KtWqf0UwMIbgFQPoxkP00TWRmdmPz
W0wT0bEf9ouTnjG90Z20
-----END CERTIFICATE-----
```

## 執行個體身分角色

您啟動的每個執行個體都有一個代表其身分的執行個體身分角色。執行個體身分識別角色是 IAM 角色的一種。AWS 整合以使用執行個體身分識別角色的服務和功能可以使用它來識別服務的執行個體。



您可以從 `/identity-credentials/ec2/security-credentials/ec2-instance` 的執行個體中繼資料服務 (IMDS) 存取執行個體身分角色憑證。憑證由 AWS 臨時訪問 key pair 和會話令牌組成。它們是用來簽 AWS 署 Sigv4 要求，以使用執行個體身分識別角色的 AWS 服務。無論執行個體上是否已啟用使用執行個體身分角色的服務或功能，憑證都會顯示在執行個體中繼資料中。

執行個體身分角色在執行個體啟動時自動建立，沒有角色信任政策文件，且不受任何身分或資源政策的約束。

## 支援的服務

下列 AWS 服務使用執行個體識別角色：

- Amazon EC2 — EC2 [執行個體 Connect](#) 使用執行個體身分角色更新 Linux 執行個體的主機金鑰。
- Amazon GuardDuty — [執行階段監控](#) 使用執行個體身分角色，允許執行階段代理程式將安全性遙測傳送到 GuardDuty VPC 端點。
- AWS Security Token Service (AWS STS) — 執行個體身分識別角色認證可與 AWS STS [GetCallerIdentity](#) 動作搭配使用。
- AWS Systems Manager — 使用 [預設主機管理組態](#) 時，AWS Systems Manager 會使用執行個體身分角色提供的身分來註冊 EC2 執行個體。識別執行個體之後，Systems Manager 可將 `AWSManagedDefaultEC2InstanceManagementRole` IAM 角色傳遞給執行個體。

執行個體身分識別角色無法與其他 AWS 服務或功能搭配使用，因為這些角色未與執行個體身分識別角色整合。

## 執行個體身分角色 ARN

執行個體身分角色 ARN 採用下列格式：

```
arn:aws-partition:iam::account-number:assumed-role/aws:ec2-instance/instance-id
```

例如：

```
arn:aws:iam::0123456789012:assumed-role/aws:ec2-instance/i-0123456789example
```

如需 ARN 的詳細資訊，請參閱《IAM 使用者指南》中的 [Amazon Resource Name \(ARN\)](#)。

## 啟動時在您的 Amazon EC2 執行個體上執行命令

啟動 Amazon EC2 執行個體時，您可以將使用者資料傳遞到用於執行自動化組態任務的執行個體，或在執行個體啟動後執行指令碼。

如果您對更複雜的自動化案例感興趣，可以考慮 AWS CloudFormation 或 AWS OpsWorks。如需詳細資訊，請參閱下列內容：

- 《AWS CloudFormation 使用者指南》中的[使用 AWS CloudFormation 在 Amazon EC2 上部署應用程式](#)。
- [AWS OpsWorks 使用者指南](#)。

在 Linux 執行個體上，您可以將兩種類型的使用者資料傳遞給 Amazon EC2：殼層指令碼和雲端初始化指令。您也可以將這些資料以純文字、檔案形式傳遞至啟動執行個體精靈（這對於使用命令列工具啟動執行個體非常有用），或以 base64 編碼的文字（用於 API 呼叫）。

在 Windows 執行個體上，啟動代理程式會處理使用者資料指令碼。以下各節說明在每個作業系統上處理使用者資料的方式之間的差異。

### Amazon EC2 如何處理 Linux 執行個體的使用者資料

在下列範例中，在 [Amazon Linux 2 上安裝 LAMP 伺服器](#) 的命令會轉換為殼層指令碼和一組在執行個體啟動時執行的雲端初始化指令。在每個範例中，以下任務都會由使用者資料執行：

- 更新分佈軟體套件。
- 安裝必要 Web 伺服器、php 和 mariadb 套件。
- 透過 systemctl 啟動並開啟 httpd 服務。
- ec2-user 即會新增至 apache 群組。
- 接著便會為 Web 目錄和其中包含的檔案設定適當的所有權和檔案許可。
- 一個簡單的網頁便會建立，用於測試 Web 伺服器和 PHP 引擎。

#### 目錄

- [必要條件](#)
- [使用者資料與 Shell 指令碼](#)
- [使用者資料與主控台](#)
- [使用者資料與 cloud-init 指示詞](#)

- [使用者資料和 AWS CLI](#)
- [結合 shell 指令碼與 cloud-init 指令](#)

## 必要條件

此主題中的範例假設如下內容：

- 您的執行個體具有公有 DNS 名稱，且可從網際網路連線。
- 與執行個體相關聯的安全群組已設定為允許 SSH (連接埠 22) 流量，以便您可以連線到執行個體以檢視輸出日誌檔案。
- 您的執行個體是使用 Amazon Linux 2 AMI 啟動的。這些說明適用於與 Amazon Linux 2 搭配使用的情況，因此命令和指示詞可能無法在其他 Linux 發行版本上運作。如需其他分佈的詳細資訊，例如支援 cloud-init 的情形，請參閱其特定文件。

## 使用者資料與 Shell 指令碼

如果您熟悉 Shell 指令碼，這是在啟動時將指令傳送至執行個體的最簡單且最完整的方式。在開機階段新增這些任務也會增加開機執行個體所需要的時間長度。任務完成需要多花幾分鐘的時間，接著您便可以測試使用者指令碼是否已成功完成。

### Important

依預設，使用者資料指令碼和 cloud-init 指令只會在您第一次啟動執行個體時的開機週期過程中執行。您可以更新設定，以確保每次重新啟動執行個體時，您的使用者資料指令碼和 cloud-init 指令都會執行。如需詳細資訊，請參閱[如何利用使用者資料在每次重新啟動 Amazon EC2 Linux 執行個體時自動執行指令碼？](#) 在 AWS 知識中心。

使用者資料 shell 指令碼必須以 `#!` 字元，以及您希望讀取指令碼 (通常是 `/bin/bash`) 的解譯器路徑做為開頭。有關 shell 腳本的介紹，請參閱 [GNU 操作系統網站上的 Bash 參考手冊](#)。

做為使用者資料輸入的指令碼會以根使用者的身分執行，因此請不要在指令碼中使用 `sudo` 命令。請記得，您建立的任何檔案都會由根使用者擁有；若您需要讓非超級使用者擁有檔案存取權，建議您根據需求在指令碼中修改許可。此外，因為指令碼不會以互動方式執行，您無法包含需要使用者意見回饋的命令 (例如不帶有 `-y` 標記的 `yum update`)。

如果您在使用者資料指令碼中使用 AWS API (包括 AWS CLI)，則在啟動執行個體時必須使用執行個體設定檔。執行個體設定檔會提供使用者資料指令碼發出 API 呼叫所需的適當 AWS 憑證。如需詳細

資訊，請參閱《IAM 使用者指南》中的[使用執行個體描述檔](#)。您指派給 IAM 角色的許可取決於您使用 API 呼叫的服務。如需詳細資訊，請參閱[Amazon EC2 的 IAM 的角色](#)。

cloud-init 輸出日誌檔案會擷取主控台輸出，因此，若執行個體在啟動後未以您想要的方式運作，則可輕鬆地對您的指令碼進行除錯。若要檢視日誌檔案，[連線到執行個體](#)並開啟 `/var/log/cloud-init-output.log`。

處理使用者資料指令碼時，會將其複製到 `/var/lib/cloud/instances/instance-id/` 並從中執行。在執行指令碼之後，不會將其刪除。務必從 `/var/lib/cloud/instances/instance-id/` 中刪除使用者資料指令碼，然後再從執行個體中建立 AMI。否則，從 AMI 啟動任何執行個體時，指令碼將存在於這個目錄中。

## 使用者資料與主控台

您可以在啟動執行個體時指定執行個體使用者資料。若執行個體的根磁碟區為 EBS 磁碟區，您也可以停止執行個體並更新其使用者資料。

在啟動時指定執行個體使用者資料

請遵循[啟動執行個體](#)的程序。此 User data (使用者資料) 欄位位於啟動執行個體精靈的[進階詳細資訊](#)區段中。在 User data (使用者資料) 欄位輸入 shell 指令碼，然後完成執行個體啟動程序。

在以下範例指令碼中，指令碼會建立並設定我們的 Web 伺服器。

```
#!/bin/bash
yum update -y
amazon-linux-extras install -y lamp-mariadb10.2-php7.2 php7.2
yum install -y httpd mariadb-server
systemctl start httpd
systemctl enable httpd
usermod -a -G apache ec2-user
chown -R ec2-user:apache /var/www
chmod 2775 /var/www
find /var/www -type d -exec chmod 2775 {} \;
find /var/www -type f -exec chmod 0664 {} \;
echo "<?php phpinfo(); ?>" > /var/www/html/phpinfo.php
```

給予執行個體充分的時間啟動及執行您指令碼中的命令，然後檢查您的指令碼是否已完成您預計執行的任務。

針對我們的範例，在 Web 瀏覽器中，輸入指令碼建立之 PHP 測試檔案的 URL。此 URL 為您執行個體的公有 DNS 地址，其後跟隨斜線和檔案名稱。

```
http://my.public.dns.amazonaws.com/phpinfo.php
```

您現在應該會看見 PHP 資訊頁面。如果您未看見 PHP 資訊頁面，請檢查您使用的安全群組是否包含允許 HTTP (連接埠 80) 流量的規則。如需詳細資訊，請參閱 [將規則新增至安全群組](#)。

(選用) 若您的指令碼並未完成您預期執行的任務，或您只希望確認您的指令碼已順利完成，而沒有發生任何錯誤，請[連線到執行個體](#)，檢查 cloud-init 輸出日誌檔案 (/var/log/cloud-init-output.log)，並在輸出中查看是否有錯誤訊息。

如需額外的偵錯資訊，您可以使用以下指示詞建立包含 cloud-init 資料區段的 Mime 分段封存：

```
output : { all : '| tee -a /var/log/cloud-init-output.log' }
```

這個指示詞會將您指令碼的命令輸出傳送到 /var/log/cloud-init-output.log。如需 cloud-init 資料格式和建立 Mime 多部分封存的詳細資訊，請參閱 [cloud-init 格式](#)。

### 檢視及更新執行個體使用者資料

若要更新執行個體使用者資料，您必須先停止執行個體。如果執行個體正在執行，您可以檢視使用者資料，但無法進行修改。

#### Warning

停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區的資料，請將資料備份至持久性儲存。

### 修改執行個體使用者資料

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，並選取 Instance state (執行個體狀態)、Stop instance (停止執行個體)。如果此選項已停用，則執行個體已停止或其根設備是執行個體存放磁碟區。
4. 出現確認提示時，請選擇 Stop (停止)。停止執行個體可能需要幾分鐘。
5. 在仍然選取執行個體的情況下，選取動作 > 執行個體設定 > 編輯使用者資料。
6. 視需要修改使用者資料，然後選擇儲存。
7. 啟動實例。新的使用者資料便會在執行個體啟動之後出現在執行個體上；但不會執行使用者資料指令碼。

## 使用者資料與 cloud-init 指示詞

cloud-init 套件會在啟動時設定新 Amazon Linux 執行個體的特定部分；最明顯的是，它會設定 ec2-user 的 `.ssh/authorized_keys` 檔案，讓您可以使用自己的私有金鑰登入。如需雲端初始化套件針對 Amazon Linux 執行個體執行之組態任務的詳細資訊，請參閱 Amazon Linux 2 [使用者指南中的在 Amazon Linux 2 上使用雲端初始化](#)。

cloud-init 使用者指示詞可在啟動時，以傳遞指令碼相同的方式傳遞到執行個體，雖然語法不同。如需有關 cloud-init 的詳細資訊，請參閱 <http://cloudinit.readthedocs.org/en/latest/index.html>。

### Important

依預設，使用者資料指令碼和 cloud-init 指令只會在您第一次啟動執行個體時的開機週期過程中執行。您可以更新設定，以確保每次重新啟動執行個體時，您的使用者資料指令碼和 cloud-init 指令都會執行。如需詳細資訊，請參閱[如何利用使用者資料在每次重新啟動 Amazon EC2 Linux 執行個體時自動執行指令碼？](#) 在 AWS 知識中心。

在開機階段新增這些任務也會增加開機執行個體所需要的時間長度。任務完成需要多花幾分鐘的時間，接著您便可以測試您的使用者資料指示詞是否已完成。

使用使用者資料將 cloud-init 指示詞傳遞給執行個體

1. 請遵循[啟動執行個體](#)的程序。此 User data (使用者資料) 欄位位於啟動執行個體精靈的 [進階詳細資訊](#) 區段中。在 User data (使用者資料) 欄位中輸入您的 cloud-init 指示詞文字，然後完成執行個體啟動程序。

在以下範例中，指示詞會在 Amazon Linux 2 上建立及設定 Web 伺服器。頂端的 `#cloud-config` 為將命令識別為 cloud-init 指示詞的必要項目。

```
#cloud-config
repo_update: true
repo_upgrade: all

packages:
- httpd
- mariadb-server

runcmd:
- [sh, -c, "amazon-linux-extras install -y lamp-mariadb10.2-php7.2 php7.2"]
- systemctl start httpd
```

```
- sudo systemctl enable httpd
- [sh, -c, "usermod -a -G apache ec2-user"]
- [sh, -c, "chown -R ec2-user:apache /var/www"]
- chmod 2775 /var/www
- [find, /var/www, -type, d, -exec, chmod, 2775, {}, \;]
- [find, /var/www, -type, f, -exec, chmod, 0664, {}, \;]
- [sh, -c, 'echo "<?php phpinfo(); ?>" > /var/www/html/phpinfo.php']
```

2. 給予執行個體充分的時間啟動及執行您使用者資料中的指示詞，然後檢查您的指示詞是否已完成您預計執行的任務。

針對此範例，在 Web 瀏覽器中，輸入指示詞建立之 PHP 測試檔案的 URL。此 URL 為您執行個體的公有 DNS 地址，其後跟隨斜線和檔案名稱。

```
http://my.public.dns.amazonaws.com/phpinfo.php
```

您現在應該會看見 PHP 資訊頁面。如果您未看見 PHP 資訊頁面，請檢查您使用的安全群組是否包含允許 HTTP (連接埠 80) 流量的規則。如需詳細資訊，請參閱 [將規則新增至安全群組](#)。

3. (選用) 若您的指令並未完成您預期執行的任務，或您只希望確認您的指令已順利完成，而沒有發生任何錯誤，請[連線到執行個體](#)，檢查 cloud-init 輸出日誌檔案 (/var/log/cloud-init-output.log)，並在輸出中查看是否有錯誤訊息。如需其他偵錯資訊，您可以為您的指示詞新增下行：

```
output : { all : '| tee -a /var/log/cloud-init-output.log' }
```

這個指示詞會將 runcmd 的輸出傳送到 /var/log/cloud-init-output.log。

## 使用者資料和 AWS CLI

您可以使用指 AWS CLI 定、修改和檢視執行個體的使用者資料。如需使用執行個體中繼資料檢視您執行個體中使用者資料的資訊，請參閱[從執行個體擷取執行個體使用者資料](#)。

在視窗上，您可以使用，AWS Tools for Windows PowerShell 而不是使用 AWS CLI。如需詳細資訊，請參閱[使用者資料和視窗工具 PowerShell](#)。

**範例：**在啟動時指定使用者資料

若要在啟動您的執行個體時指定使用者資料，請搭配 --user-data 參數使用 [run-instances](#) 命令。使用時run-instances，會為您 AWS CLI 執行使用者資料的 base64 編碼。

以下範例顯示如何在命令列上將指令碼指定為字串：

```
aws ec2 run-instances --image-id ami-abcd1234 --count 1 --instance-type m3.medium \
--key-name my-key-pair --subnet-id subnet-abcd1234 --security-group-ids sg-abcd1234 \
--user-data echo user data
```

以下範例顯示如何使用文字檔案指定指令碼。請確認使用 `file://` 前綴來指定檔案。

```
aws ec2 run-instances --image-id ami-abcd1234 --count 1 --instance-type m3.medium \
--key-name my-key-pair --subnet-id subnet-abcd1234 --security-group-ids sg-abcd1234 \
--user-data file://my_script.txt
```

以下是具有 shell 指令碼的範例文字檔案。

```
#!/bin/bash
yum update -y
service httpd start
chkconfig httpd on
```

範例：修改已停止執行個體的使用者資料

您可以使用 [modify-instance-attribute](#) 命令修改已停止執行個體的使用者資料。使用時 `modify-instance-attribute`，AWS CLI 不會為您執行使用者資料的 base64 編碼。

- 在 Linux 電腦上，請使用 `base64` 命令來編碼使用者資料。

```
base64 my_script.txt >my_script_base64.txt
```

- 在 Windows 電腦上，請使用 `certutil` 命令編碼使用者資料。您必須先移除第一個 (開始憑證) 和最後一行 (結束憑證) 行 AWS CLI，才能搭配使用此檔案。

```
certutil -encode my_script.txt my_script_base64.txt
notepad my_script_base64.txt
```

請使用 `--attribute` 和 `--value` 參數，使用已編碼的文字檔案來指定使用者資料。請確認使用 `file://` 前綴來指定檔案。

```
aws ec2 modify-instance-attribute --instance-id i-1234567890abcdef0 --attribute
userData --value file://my_script_base64.txt
```



範例：清除已停止執行個體的使用者資料

若要刪除現有的使用者資料，請使用 [modify-instance-attribute](#) 命令，如下所示：

```
aws ec2 modify-instance-attribute --instance-id i-1234567890abcdef0 --user-data Value=
```

範例：檢視使用者資料

若要擷取執行個體的使用者資料，請使用 [describe-instance-attribute](#) 命令。使用時 describe-instance-attribute，AWS CLI 不會為您執行 base64 解碼使用者資料。

```
aws ec2 describe-instance-attribute --instance-id i-1234567890abcdef0 --attribute
userData
```

下列為已使用 base64 編碼之使用者資料的範例輸出。

```
{
 "UserData": {
 "Value":
 "IyEvYmluL2Jhc2gKeXVtIHVwZGF0ZSAteQpzZXJ2aWNlIGh0dHBkIHh0YXJ0CmNoa2NvbWZpZyBodHRwZCBvbG=="
 },
 "InstanceId": "i-1234567890abcdef0"
}
```

- 在 Linux 上，請使用 --query 選項取得已編碼的使用者資料，並使用 base64 命令將其解碼。

```
aws ec2 describe-instance-attribute --instance-id i-1234567890abcdef0 --attribute
userData --output text --query "UserData.Value" | base64 --decode
```

- 在 Windows 電腦上，使用 --query 選項以取得已編碼的使用者資料，並使用 certutil 命令將其解碼。請注意，編碼輸出會存放在檔案中，解碼輸出則會存放在另一個檔案中。

```
aws ec2 describe-instance-attribute --instance-id i-1234567890abcdef0 --attribute
userData --output text --query "UserData.Value" >my_output.txt
certutil -decode my_output.txt my_output_decoded.txt
type my_output_decoded.txt
```

下列為範例輸出。

```
#!/bin/bash
```

```
yum update -y
service httpd start
chkconfig httpd on
```

## 結合 shell 指令碼與 cloud-init 指令

根據預設，您一次只能在使用者資料中包含一種內容類型。不過，您可以使用 MIME 多部分檔案中的 `text/cloud-config` 和 `text/x-shellscript` 內容類型，以在使用者資料中包含 shell 指令碼和 cloud-init 指令。

以下展示 MIME 多部分格式。

```
Content-Type: multipart/mixed; boundary="//"
MIME-Version: 1.0

--//
Content-Type: text/cloud-config; charset="us-ascii"
MIME-Version: 1.0
Content-Transfer-Encoding: 7bit
Content-Disposition: attachment; filename="cloud-config.txt"

#cloud-config
cloud-init directives

--//
Content-Type: text/x-shellscript; charset="us-ascii"
MIME-Version: 1.0
Content-Transfer-Encoding: 7bit
Content-Disposition: attachment; filename="userdata.txt"

#!/bin/bash
shell script commands
--//--
```

例如，下列使用者資料包含 cloud-init 指令和 bash shell 指令碼。cloud-init 指令會建立檔案 (`/test-cloudinit/cloud-init.txt`)，並將 Created by cloud-init 寫入到該檔案。bash shell 指令碼會建立檔案 (`/test-userscript/userscript.txt`) 並將 Created by bash shell script 寫入到該檔案。

```
Content-Type: multipart/mixed; boundary="//"
MIME-Version: 1.0
```

```
--//
Content-Type: text/cloud-config; charset="us-ascii"
MIME-Version: 1.0
Content-Transfer-Encoding: 7bit
Content-Disposition: attachment; filename="cloud-config.txt"

#cloud-config
runcmd:
- [mkdir, /test-cloudinit]
write_files:
- path: /test-cloudinit/cloud-init.txt
 content: Created by cloud-init

--//
Content-Type: text/x-shellscript; charset="us-ascii"
MIME-Version: 1.0
Content-Transfer-Encoding: 7bit
Content-Disposition: attachment; filename="userdata.txt"

#!/bin/bash
mkdir test-userscript
touch /test-userscript/userscript.txt
echo "Created by bash shell script" >> /test-userscript/userscript.txt
--//--
```

## Amazon EC2 如何處理 Windows 執行個體的使用者資料

在 Windows 執行個體上，作業系統版本的預設啟動代理程式會如下處理使用者資料。

- [EC2Launch v2](#)在視窗服務器 2022 上運行用戶數據腳本
- [???](#)在視窗服務器 2016 和 2019 上運行用戶數據腳本
- [???](#)在視窗伺服器 2016 年之前的版本上執行使用者資料指令碼

如需範 AWS CloudFormation 本中UserData屬性組件的範例，請參閱 [Base64 編碼內 UserData 容](#)和 [Base64 編碼 UserData 屬性與 AccessKey](#)。SecretKey

如需在可搭配生命週期勾點使用之 Auto Scaling 群組中的執行個體上執行命令的範例，請參閱 Amazon EC2 Auto Scaling 使用者指南中的教學課程：[透過執行個體中繼資料設定使用者資料以擷取目標生命週期狀態](#)。

### 目錄

- [使用者資料指令碼](#)
- [使用者資料執行](#)
- [使用者資料與主控台](#)
- [使用者資料和視窗工具 PowerShell](#)

## 使用者資料指令碼

EC2Launch若要執行EC2Config或執行指令碼，當您將指令碼新增至使用者資料時，您必須將指令碼包含在特殊標記內。您使用的標籤取決於命令是在命令提示字元視窗 (批次命令) 中執行，還是使用 Windows PowerShell。

如果您同時指定批次指令碼和 Windows 指 PowerShell 令碼，則會先執行批次指令碼，然後再執行 Windows 指 PowerShell 令碼，而不論它們在執行個體使用者資料中出現的順序為何。

如果您在使用者資料指令碼中使用 AWS API (包括)，則在啟動執行個體時必須使用執行個體設定檔。AWS CLI執行個體設定檔會提供使用者資料指令碼進行 API 呼叫所需的適當 AWS 憑證。如需詳細資訊，請參閱 [執行個體描述檔](#)。您指派給 IAM 角色的許可取決於您使用 API 呼叫的服務。如需詳細資訊，請參閱 [Amazon EC2 的 IAM 的角色](#)。

### 指令碼類型

- [批次指令碼的語法](#)
- [語法視窗 PowerShell指令碼](#)
- [YAML 組態指令碼的語法](#)
- [Base64 編碼](#)

### 批次指令碼的語法

使用 `script` 標籤指定批次指令碼。使用換行符號分隔命令，如下列範例所示。

```
<script>
 echo Current date and time >> %SystemRoot%\Temp\test.log
 echo %DATE% %TIME% >> %SystemRoot%\Temp\test.log
</script>
```

在預設情況下，使用者資料指令碼僅會在您啟動執行個體時執行一次。若要在每次重新開機或啟動執行個體時執行使用者資料指令碼，請將 `<persist>>true</persist>` 新增至使用者資料。

```
<script>
echo Current date and time >> %SystemRoot%\Temp\test.log
echo %DATE% %TIME% >> %SystemRoot%\Temp\test.log
</script>
<persist>true</persist>
```

## EC2Launch v2 代理程式

若要透過 UserData 階段中的 EC2Launch v2 executeScript 任務，將 XML 使用者資料指令碼作為以分離的程序執行，請將下列標籤新增至您的使用者資料。

```
<detach>true</detach>
```

### Note

舊版啟動代理程式不支援分離標籤。

```
<script>
echo Current date and time >> %SystemRoot%\Temp\test.log
echo %DATE% %TIME% >> %SystemRoot%\Temp\test.log
</script>
<detach>true</detach>
```

## 語法視窗 PowerShell 指令碼

AWS Windows AMI 包含 [AWS Tools for Windows PowerShell](#)，因此您可以在使用者資料中指定這些指令程式。如果將 IAM 角色與執行個體建立關聯，則不需要為指令程式指定登入資料，因為在執行個體上執行的應用程式會使用角色的登入資料來存取 AWS 資源 (例如，Amazon S3 儲存貯體)。

使用此標記指定 Windows 指 PowerShell 令 <powershell> 碼。使用分行符號來區隔命令。<powershell> 標籤區分大小寫。

例如：

```
<powershell>
$file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
New-Item $file -ItemType file
</powershell>
```

根據預設，使用者資料指令碼會在您啟動執行個體時執行一次。若要在每次重新開機或啟動執行個體時執行使用者資料指令碼，請將 `<persist>>true</persist>` 新增至使用者資料。

```
<powershell>
$file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
New-Item $file -ItemType file
</powershell>
<persist>>true</persist>
```

您可以使用 `<powershellArguments>` 標籤指定一個或多個 PowerShell 引數。如果未傳遞任何引數，則根據預設，EC2Launch 和 EC2Launch v2 會新增下列引數：`-ExecutionPolicy Unrestricted`

範例：

```
<powershell>
$file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
New-Item $file -ItemType file
</powershell>
<powershellArguments>-ExecutionPolicy Unrestricted -NoProfile -NonInteractive</
powershellArguments>
```

## EC2Launch v2 代理程式

若要透過 UserData 階段中的 EC2Launch v2 `executeScript` 任務，將 XML 使用者資料指令碼作為以分離的程序執行，請將下列標籤新增至您的使用者資料。

```
<detach>>true</detach>
```

### Note

舊版啟動代理程式不支援分離標籤。

```
<powershell>
$file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
New-Item $file -ItemType file
</powershell>
<detach>>true</detach>
```

## YAML 組態指令碼的語法

如果您使用 EC2Launch v2 來執行指令碼，則可以使用 YAML 格式。若要檢視 EC2Launch v2 的組態任務、詳細資料和範例，請參閱 [EC2Launch v2 任務組態](#)。

指定具有 executeScript 任務的 YAML 指令碼。

### 執行指令碼的 YAML 語法範例 PowerShell

```
version: 1.0
tasks:
- task: executeScript
 inputs:
 - frequency: always
 type: powershell
 runAs: localSystem
 content: |-
 $file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
 New-Item $file -ItemType file
```

### 執行批次指令碼的 YAML 語法範例

```
version: 1.1
tasks:
- task: executeScript
 inputs:
 - frequency: always
 type: batch
 runAs: localSystem
 content: |-
 echo Current date and time >> %SystemRoot%\Temp\test.log
 echo %DATE% %TIME% >> %SystemRoot%\Temp\test.log
```

## Base64 編碼

如果您使用 Amazon EC2 API 或不會為您執行使用者資料的 base64 編碼的工具，就必須自行編碼使用者資料。若否，系統就會記錄一則錯誤，指出找不到要執行的 script 或 powershell 標籤。以下是使用 Windows PowerShell 進行編碼的範例。

```
$UserData =
[System.Convert]::ToBase64String([System.Text.Encoding]::ASCII.GetBytes($Script))
```

以下是使 PowerShell 用解碼的範例。

```
$Script =
[System.Text.Encoding]::UTF8.GetString([System.Convert]::FromBase64String($UserData))
```

如需有關 Base64 編碼的詳細資訊，請參閱 <https://www.ietf.org/rfc/rfc4648.txt>。

## 使用者資料執行

根據預設，所有 AWS Windows AMI 都會在初始啟動時啟用使用者資料執行。您可以指定在下一次重新開機或重新啟動執行個體時執行使用者資料指令碼。或者，您可以指定每次重新開機或重新啟動執行個體時都執行使用者資料指令碼。

### Note

在預設情況下，初始啟動後不會執行使用者資料。若要在重新開機或啟動執行個體後執行使用者資料，請參閱 [後續的重新開機或啟動](#)。

產生隨機密碼時，會透過本機管理員帳戶來執行使用者資料指令碼。否則，會透過系統帳戶執行使用者資料指令碼。

## 執行個體啟動

執行個體使用者資料中的指令碼僅會在執行個體初次啟動時執行。如果找到 `persist` 標籤，即會為後續的重新開機或啟動啟用使用者資料執行。EC2Launch v2、EC2Launch 和 EC2Config 的日誌檔案包含來自標準輸出和標準錯誤串流的輸出。

## EC2Launch v2

EC2Launch v2 的記錄檔是 `C:\ProgramData\Amazon\EC2Launch\log\agent.log`。

### Note

`C:\ProgramData` 資料夾可能隱藏不見。若要檢視此資料夾，您必須顯示隱藏的檔案和資料夾。

以下資訊會在執行使用者資料時記錄：



- Info: Converting user-data to yaml format - 如果使用者資料是以 XML 格式提供
- Info: Initialize user-data state - 開始執行使用者資料時
- Info: Frequency is: always - 如果使用者資料工作在每次開機時執行
- Info: Frequency is: once - 如果使用者資料任務只執行一次
- Stage: postReadyUserData execution completed - 使用者資料執行結束時

## EC2Launch

EC2Launch 的日誌文件是 C:\ProgramData\Amazon\EC2-Windows\Launch\Log\UserdataExecution.log。

C:\ProgramData 資料夾可能隱藏不見。若要檢視此資料夾，您必須顯示隱藏的檔案和資料夾。

以下資訊會在執行使用者資料時記錄：

- Userdata execution begins - 開始執行使用者資料時
- <persist> tag was provided: true - 如果找到 persist 標籤
- Running userdata on every boot - 如果找到 persist 標籤
- <powershell> tag was provided.. running powershell content - 如果找到 powershell 標籤
- <script> tag was provided.. running script content - 如果找到指令碼標籤
- Message: The output from user scripts - 如果執行使用者資料指令碼，則會記錄其輸出

## EC2Config

EC2Config 的日誌文件是 C:\Program Files\Amazon\Ec2ConfigService\Logs\Ec2Config.log。以下資訊會在執行使用者資料時記錄：

- Ec2HandleUserData: Message: Start running user scripts - 開始執行使用者資料時
- Ec2HandleUserData: Message: Re-enabled userdata execution - 如果找到 persist 標籤
- Ec2HandleUserData: Message: Could not find <persist> and </persist> - 如果沒有找到 persist 標籤
- Ec2HandleUserData: Message: The output from user scripts - 如果執行使用者資料指令碼，則會記錄其輸出

## 後續的重新開機或啟動

當您更新執行個體使用者資料時，使用者資料指令碼並未在您重新開機或啟動執行個體時自動執行。不過，您可以啟用使用者資料執行，以讓使用者資料指令碼在您重新開機或啟動執行個體時執行一次，或每次重新開機或啟動執行個體時都執行。

如果您選擇 Shutdown with Sysprep (使用 Sysprep 關機) 選項，使用者資料指令碼會在下次執行個體啟動或重新開機時執行，即使您並沒有為後續的重新開機或啟動啟用使用者資料執行亦同。使用者資料指令碼將不會在後續的重新開機或啟動時執行。

若要使用 EC2Launch v2 (預覽 AMI) 啟用使用者資料執行

- 若要在第一次開機時在使用者資料中執行工作，請將 frequency 設定為 once。
- 若要在每次開機時在使用者資料中執行工作，請將 frequency 設定為 always。

透過 EC2Launch 啟用使用者資料執行 (Windows Server 2016 或更新版本)

1. 連接至 Windows 執行個體。
2. 開啟命 PowerShell 令視窗並執行下列命令：

```
C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 -Schedule
```

3. 中斷連線您的 Windows 執行個體。若要在下一次啟動執行個體時執行更新的指令碼，請停止執行個體並更新使用者資料。

透過 EC2Config 啟用使用者資料執行 (Windows Server 2012 R2 或更早版本)

1. 連接至 Windows 執行個體。
2. Open C:\Program Files\Amazon\Ec2ConfigService\Ec2ConfigServiceSetting.exe.
3. 對於使用者資料，請選取啟用下次服務啟動時 UserData 執行。
4. 中斷連線您的 Windows 執行個體。若要在下一次啟動執行個體時執行更新的指令碼，請停止執行個體並更新使用者資料。

## 使用者資料與主控台

您可以在啟動執行個體時指定執行個體使用者資料。若執行個體的根磁碟區為 EBS 磁碟區，您也可以停止執行個體並更新其使用者資料。

## 在啟動時指定執行個體使用者資料

請遵循[啟動執行個體](#)的程序。此 User data (使用者資料) 欄位位於啟動執行個體精靈的 [進階詳細資訊](#) 區段中。在 [使用者資料] 欄位中輸入您的 PowerShell 指令碼，然後完成執行個體啟動程序。

在以下使用者資料欄位的螢幕擷取畫面中，範例指令碼會在 Windows 暫時資料夾中建立一個檔案，並在檔名中使用目前日期和時間。當您包括 `<persist>>true</persist>` 時，指令碼會在您每次重新開機或啟動執行個體時執行。如果您將使用者資料已使用 base64 編碼核取方塊留白，Amazon EC2 主控台會為您執行 base64 編碼。

### User data - optional **Info**

Enter user data in the field.

```
<powershell>
$file = $env:SystemRoot+"\Temp\"+(Get-Date).ToString("MM-dd-yy-hh-mm")
New-Item $file -ItemType file
</powershell>
<persist>true</persist>
```

User data has already been base64 encoded


## 檢視及更新執行個體使用者資料

您可以檢視任何執行個體的執行個體使用者資料，並更新已終止之執行個體的執行個體使用者資料。

### 使用主控台更新執行個體的使用者資料

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。

3. 選取執行個體，並選取 Actions (動作)、Instance state (執行個體狀態)、Stop instance (停止執行個體)。

 Warning

停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區的資料，請將資料備份至持久性儲存。

4. 出現確認提示時，請選擇 Stop (停止)。停止執行個體可能需要幾分鐘。
5. 在仍然選取執行個體的情況下，選取動作 > 執行個體設定 > 編輯使用者資料。您無法在執行個體仍在執行中時變更使用者資料，但您可以檢視它。
6. 在 Edit user data (編輯使用者資料) 對話方塊中，更新使用者資料，然後選擇 Save (儲存)。若要在每次重新開機或啟動執行個體時執行使用者資料指令碼，請新增 `<persist>true</persist>`，如下範例所示：

## Edit user data Info

Instance ID

 i-0655799f982552ec9

### Current user data

User data currently associated with this instance

```
<powershell>
$file = $env:SystemRoot+"\Temp\"+(Get-Date).ToString("MM-dd-yy-hh-mm")
New-Item $file -ItemType file
</powershell>
```

 Copy user data

### New user data

This user data will replace the current user data

**Modify user data as text**  
Add your user data below

**Modify user data by importing a file**  
Description of importing a file and what will happen to it

```
<powershell>
$file = $env:SystemRoot+"\Temp\"+(Get-Date).ToString("MM-dd-yy-hh-mm")
New-Item $file -ItemType file
</powershell>
<persist>>true</persist>
```

Input is already base64-encoded

Cancel

Save

7. 啟動實例。若您已為後續的重新開機或啟動啟用使用者資料執行，即會在執行個體啟動程序期間執行更新的使用者資料指令碼。

## 使用者資料和視窗工具 PowerShell

您可以使用 Windows 專用的工具 PowerShell 來指定、修改和檢視執行個體的使用者資料。如需使用執行個體中繼資料檢視您執行個體中使用者資料的資訊，請參閱[從執行個體擷取執行個體使用者資料](#)。若要取得有關使用者資料的資訊 AWS CLI，請參閱[使用者資料和 AWS CLI](#)。

範例：在啟動時指定執行個體使用者資料

使用執行個體使用者資料建立文字檔案。若要在每次重新開機或啟動執行個體時執行使用者資料指令碼，請新增 `<persist>>true</persist>`，如下範例所示。

```
<powershell>
$file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
New-Item $file -ItemType file
</powershell>
<persist>true</persist>
```

若要在啟動執行個體時指定執行個體使用者資料，請使用指[New-EC2Instance](#)令。此命令不會為您執行使用者資料的 base64 編碼。使用以下命令，將使用者資料編碼至名為 `script.txt` 的文字檔案：

```
PS C:\> $Script = Get-Content -Raw script.txt
PS C:\> $UserData =
[System.Convert]::ToBase64String([System.Text.Encoding]::ASCII.GetBytes($Script))
```

使用 `-UserData` 參數，將使用者資料傳遞到 `New-EC2Instance` 命令。

```
PS C:\> New-EC2Instance -ImageId ami-abcd1234 -MinCount 1 -MaxCount 1 -
InstanceType m3.medium \
-KeyName my-key-pair -SubnetId subnet-12345678 -SecurityGroupIds sg-1a2b3c4d \
-UserData $UserData
```

範例：更新已停止之執行個體的使用者資料

您可以使用[Edit-EC2InstanceAttribute](#)指令修改已停止執行個體的使用者資料。

使用新的指令碼建立文字檔案。使用以下命令，將使用者資料編碼至名為 `new-script.txt` 的文字檔案：

```
PS C:\> $NewScript = Get-Content -Raw new-script.txt
```

```
PS C:\> $NewUserData =
[System.Convert]::ToBase64String([System.Text.Encoding]::ASCII.GetBytes($NewScript))
```

使用 `-UserData` 和 `-Value` 參數指定使用者資料。

```
PS C:\> Edit-EC2InstanceAttribute -InstanceId i-1234567890abcdef0 -Attribute userData -
Value $NewUserData
```

範例：檢視執行個體使用者資料

若要擷取執行個體的使用者資料，請使用 [Get-EC2InstanceAttribute](#) 指令。

```
PS C:\> (Get-EC2InstanceAttribute -InstanceId i-1234567890abcdef0 -Attribute
userData).UserData
```

下列為範例輸出。請注意使用者資料已編碼。

```
PHBvd2Vyc2h1bGw
+DQpSZW5hbWUtQ29tcHV0ZXIgLlU51d05hbWUgdXNlci1kYXRhLXRlc3QNCjwvcG93ZXJzaGVsbD4=
```

使用以下命令，將已編碼的使用者資料存放至變數中，然後加以解碼。

```
PS C:\> $UserData_encoded = (Get-EC2InstanceAttribute -InstanceId i-1234567890abcdef0 -
Attribute userData).UserData
PS C:
> [System.Text.Encoding]::UTF8.GetString([System.Convert]::FromBase64String($UserData_encoded))
```

下列為範例輸出。

```
<powershell>
$file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
New-Item $file -ItemType file
</powershell>
<persist>true</persist>
```

範例：重新命名執行個體以符合標籤值

您可以使用指 [Get-EC2Tag](#) 令讀取標籤值、在第一次開機時重新命名執行個體以符合標籤值，然後重新啟動。若要成功執行此命令，您必須有連接至執行個體的角色具備 `ec2:DescribeTags` 許可，因為由 API 呼叫擷取標籤資訊。如需使用 IAM 角色設定許可的詳細資訊，請參閱 [將 IAM 角色連接至執行個體](#)。

**Note**

此指令碼在 2008 以前的 Windows Server 版本上失敗。

```
<powershell>
$instanceId = (invoke-webrequest http://169.254.169.254/latest/meta-data/instance-id -
UseBasicParsing).content
$nameValue = (get-ec2tag -filter @{Name="resource-id";Value=
$instanceid},@{Name="key";Value="Name"}).Value
$pattern = "^(?![0-9]{1,15}$)[a-zA-Z0-9-]{1,15}$"
#Verify Name Value satisfies best practices for Windows hostnames
If ($nameValue -match $pattern)
 {Try
 {Rename-Computer -NewName $nameValue -Restart -ErrorAction Stop}
 Catch
 {$ErrorMessage = $_.Exception.Message
 Write-Output "Rename failed: $ErrorMessage"}}
Else
 {Throw "Provided name not a valid hostname. Please ensure Name value is between 1
and 15 characters in length and contains only alphanumeric or hyphen characters"}
</powershell>
```

如果您的執行個體設定為從執行個體中繼資料存取標籤，則還可以使用執行個體中繼資料中的標籤重新命名執行個體，。如需詳細資訊，請參閱 [使用執行個體中繼資料中的執行個體標籤](#)。

**Note**

此指令碼在 2008 以前的 Windows Server 版本上失敗。

```
<powershell>
$nameValue = Get-EC2InstanceMetadata -Path /tags/instance/Name
$pattern = "^(?![0-9]{1,15}$)[a-zA-Z0-9-]{1,15}$"
#Verify Name Value satisfies best practices for Windows hostnames
If ($nameValue -match $pattern)
 {Try
 {Rename-Computer -NewName $nameValue -Restart -ErrorAction Stop}
 Catch
 {$ErrorMessage = $_.Exception.Message
 Write-Output "Rename failed: $ErrorMessage"}}
```



```
Else
```

```
{Throw "Provided name not a valid hostname. Please ensure Name value is between 1
and 15 characters in length and contains only alphanumeric or hyphen characters"}
</powershell>
```

## Connect 至您的 EC2 執行個體

Amazon EC2 使用者指南的這一部分提供相關資訊，協助您在啟動 Amazon EC2 執行個體後連接它。它也提供相關資訊，協助您將執行個體連線至其他 AWS 資源。

### 主題

- [連接至您的 Linux 執行個體](#)
- [連線至 Windows 執行個體](#)
- [使用工作階段管理員進行連接](#)
- [使用 EC2 執行個體 Connect 端點 Connect 至執行個體](#)
- [將 EC2 執行個體連線至 AWS 資源](#)

## 連接至您的 Linux 執行個體

有多種方法可連線至 Linux 執行個體。其中的某些方法會因連線的本機電腦作業系統而有所不同。像 EC2 Instance Connect 或 AWS Systems Manager 工作階段管理員這樣的其他方法則不會變化。您可在本節中了解如何連線至 Linux 執行個體，以及如何在本地電腦和執行個體之間傳輸檔案。

連線到您的 Linux 執行個體之前，請先完成下列先決條件。

- [取得執行個體的相關資訊](#)
- [找出私有金鑰並設定許可](#)
- [\(選用\) 取得執行個體指紋](#)

然後，請選擇以下其中一個選項來連線至 Linux 執行個體。

### 根據本機作業系統進行連線的選項

- [使用 SSH 從 Linux 或 macOS 本機電腦連線](#)
- [從 Windows 本機電腦連線](#)

## 從任何本機作業系統進行連線的選項

- [使用工作階段管理員進行連接](#)
- [使用 EC2 Instance Connect 連線至 Linux 執行個體。](#)

### Note

如需執行個體連線疑難排解秘訣，請參閱 [連線到 Linux 執行個體的疑難排解](#)。  
如果要對 [AWS Nitro System](#) 上建置的執行個體，進行開機、網路組態和其他問題的疑難排解，您可以使用 [亞馬遜 EC2 執行個體的 Amazon EC2 序列主控台](#)。

## 取得執行個體的相關資訊

若準備連線至執行個體，請從 Amazon EC2 主控台或使用 AWS CLI 取得下列資訊。

The screenshot displays the Amazon EC2 console interface. At the top, a green notification bar states 'Successfully started i-...'. Below this, the 'Instances (1/8)' table lists several instances. The first instance, 'Windows', is in a 'Running' state. The 'Instance ID' column for this instance is circled in red. The 'Public IPv4 DNS' column for the same instance is also circled in red. Below the table, the 'Instance: i-05e...' details page is open, with the 'Details' tab selected. The 'Instance summary' section contains several fields: 'Instance ID' (i-05e...), 'IPv6 address' (2600:1f1...), 'Public IPv4 address' (3.84...), 'Instance state' (Pending), 'Private IP DNS name (IPv4 only)' (ip-...), 'Instance type' (t2.micro), 'VPC ID' (vpc-ai...), and 'Subnet ID' (subnet-59...). The 'Public IPv4 DNS' field is circled in red. The 'Instance details' section is partially visible at the bottom.

- 取得執行個體的公有 DNS 名稱。

您可以從 Amazon EC2 主控台取得執行個體的公有 DNS。檢查執行個體窗格的公有 IPv4 DNS 資料欄。如果此資料欄已隱藏，請選擇畫面右上角的設定圖示



並選取公有 IPv4 DNS。您也可以執行個體窗格的執行個體資訊區段中找到公有 DNS。當您在 Amazon EC2 主控台的執行個體窗格中選取執行個體時，該執行個體的相關資訊將顯示在頁面的下半部。在詳細資訊索引標籤下，查找公有 IPv4 DNS。

如果您願意，您可以使用[描述實例 \(AWS CLI\) 或 Get-EC2Instance\(AWS Tools for Windows PowerShell\) 命令](#)。

如果未顯示公有 IPv4 DNS，則請確認執行個體狀態為執行中，且您尚未在私有子網路中啟動執行個體。如果您使用[啟動執行個體精靈](#)啟動執行個體，則可能已編輯網路設定下的自動指派公有 IP 欄位，並將值變更為停用。如果停用自動指派公有 IP 選項，則執行個體在啟動時不會被指派公有 IP 地址。

- (僅限 IPv6) 取得執行個體的 IPv6 地址。

如果您已將 IPv6 地址指派給執行個體，則您可使用其 IPv6 地址連線至執行個體，而不必使用公有 IPv4 地址或公有 IPv4 DNS 主機名稱。您的本機電腦必須擁有 IPv6 地址，且必須設定以使用 IPv6。您可以從 Amazon EC2 主控台取得執行個體的 IPv6 地址。檢查執行個體窗格的 IPv6 IP 資料欄。或者，您可以在執行個體資訊區段中找到 IPv6 地址。當您在 Amazon EC2 主控台的執行個體窗格中選取執行個體時，該執行個體的相關資訊將顯示在頁面的下半部。在詳細資訊索引標籤下，查找 IPv6 地址。

如果您願意，您可以使用[描述實例 \(AWS CLI\) 或 Get-EC2Instance\(AWS Tools for Windows PowerShell\) 命令](#)。如需有關 IPv6 的詳細資訊，請參閱[IPv6 地址](#)。

- 取得您的執行個體的使用者名稱。

使用您的使用者帳戶的使用者名稱或用於啟動執行個體的 AMI 預設使用者名稱，可連線至執行個體。

- 取得使用者帳戶的使用者名稱。

如需如何建立使用者帳戶的詳細資訊，請參閱[管理 Linux 執行個體上的系統使用者](#)。

- 取得您用於啟動執行個體的 AMI 的預設使用者名稱：

用於啟動執行個體的 AMI	預設使用者名稱
AL2023	ec2-user
Amazon Linux 2	

用於啟動執行個體的 AMI	預設使用者名稱
Amazon Linux	
CentOS	centos 或 ec2-user
Debian	admin
Fedora	fedora 或 ec2-user
RHEL	ec2-user 或 root
SUSE	ec2-user 或 root
Ubuntu	ubuntu
Oracle	ec2-user
Bitnami	bitnami
Rocky Linux	rocky
其他	檢查 AMI 提供者

## 找出私有金鑰並設定許可

您必須知道私有金鑰檔案的位置，才能連線至執行個體。若為 SSH 連線，則必須設定許可，以便只有您可以讀取檔案。

如需有關使用 Amazon EC2 時金鑰對如何運作的資訊，請參閱 [Amazon EC2 金鑰配對](#) 和 [Amazon EC2 執行個體](#)。

- 找出私有金鑰

取得當您啟動執行個體時所指定的金鑰對的 .pem 檔案在您電腦中的位置的完整路徑。如需詳細資訊，請參閱 [the section called “識別啟動時指定的公有金鑰”](#)。

如果找不到私密金鑰檔案，請參閱

[如果遺失了 EBS 後端執行個體的私有金鑰，您可以重新獲得對執行個體的存取權限。您必須停止執行個體、中斷其根磁碟區的連結，並將該磁碟區做為資料磁碟區連接至其他執行個體、使用新的](#)

公有金鑰修改 `authorized_keys` 檔案、將磁碟區移回到原始的執行個體，然後重新啟動執行個體。關於啟動、連線到和停用執行個體，詳細資訊請參閱 [執行個體生命週期](#)。

僅具有 EBS 根磁碟區的執行個體支援此程序。如果根裝置是執行個體存放區磁碟區，則無法使用此程序來重新獲取執行個體的存取權；您必須具備私有金鑰，才能連線到執行個體。若要確定執行個體的根裝置類型，請開啟 Amazon EC2 主控台，選擇執行個體，選取執行個體，選擇儲存索引標籤，然後在根裝置詳細資訊區段中，檢查根裝置類型的值。

此值為 EBS 或 INSTANCE-STORE。

除了下列步驟之外，遺失私有金鑰時還有其他方法可以連線到 Linux 執行個體。如需詳細資訊，請參閱 [如果我在初次啟動後遺失了 SSH 金鑰對，該如何連接到 Amazon EC2 執行個體？](#)

使用不同的金鑰對以連線到 EBS 後端執行個體的步驟

- [步驟 1：建立新的金鑰對](#)
- [步驟 2：取得關於原始執行個體及其根磁碟區的資訊](#)
- [步驟 3：停止原始執行個體](#)
- [步驟 4：啟動暫時執行個體](#)
- [步驟 5：將根磁碟區與原始執行個體分離，並將其連接到暫時執行個體](#)
- [步驟 6：在掛載到暫時執行個體的原始磁碟區上，將新的公有金鑰新增至的 `authorized\_keys`](#)
- [步驟 7：取消掛載原始磁碟區，並將該磁碟區從暫時執行個體分離，然後再重新連接到原始執行個體](#)
- [步驟 8：使用新的金鑰對連線到原始執行個體](#)
- [步驟 9：清理](#)

### 步驟 1：建立新的金鑰對

使用 Amazon EC2 主控台或第三方工具來產生新的金鑰對。如果希望新金鑰對的名稱，和已遺失私有金鑰的相同，您必須先刪除現有的金鑰對。如需建立新金鑰對的詳細資訊，請參閱 [使用 Amazon EC2 建立金鑰對](#) 或 [使用第三方工具來建立金鑰對](#)，然後將公有金鑰匯入 Amazon EC2。

### 步驟 2：取得關於原始執行個體及其根磁碟區的資訊

請儲存下列資訊，完成此程序將會需要用到這些資訊。

## 取得原始執行個體的相關資訊

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中選取 Instances (執行個體)，然後選取您要連線的執行個體。(我們會將此執行個體稱為「原始」執行個體。)
3. 在 Details (詳細資訊) 索引標籤上記下執行個體 ID 和 AMI ID。
4. 在網路索引標籤上記下「可用區域」。
5. 在 Storage (儲存) 索引標籤下的根裝置名稱，記下根磁碟區的裝置名稱 (例如：`/dev/xvda`)。接著，在區塊型儲存裝置下找出這個裝置名稱，並記下磁碟區 ID (例如：`vol-0a1234b5678c910de`)。

### 步驟 3：停止原始執行個體

選擇 Instance state (執行個體狀態)、Stop instance (停止執行個體)。如果此選項已停用，則執行個體已停止或其根設備是執行個體存放磁碟區。

#### Warning

停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區的資料，請將資料備份至持久性儲存。

### 步驟 4：啟動暫時執行個體

#### New console

#### 啟動暫時執行個體

1. 在導覽窗格中，選擇 Instances (執行個體)，然後選擇 Launch instances (啟動執行個體)。
2. 在 Name and tags (名稱和標籤) 區段，針對 Name (名稱)，輸入 Temporary (暫時)。
3. 在 Application and OS Images (應用程式和作業系統映像) 區段，選取您用來啟動原始執行個體的同一個 AMI。如果此 AMI 無法使用，您可以從已停止的執行個體，建立能夠使用的 AMI。如需詳細資訊，請參閱 [創建一個 Amazon EBS-backed AMI](#)。
4. 在 Instance type (執行個體類型) 區段，保留預設執行個體類型。
5. 在 Key pair (金鑰對) 區段，針對 Key pair name (金鑰對名稱)，選取要使用的現有金鑰對或建立新的金鑰對。

- 在 Network settings (網路設定) 區段，選取 Edit (編輯)，然後針對 Subnet (子網)，選取與原始執行個體相同可用區域中的子網。

- 在 Summary (摘要) 面板中，選擇 Launch (啟動)。

#### Old console

選擇 Launch Instance (啟動執行個體)，接著再使用 Launch Wizard，透過下列選項來啟動 temporary (暫時) 執行個體：

- 在 Choose an AMI (選取 AMI) 頁面上，選取您之前用來啟動原始執行個體的那個 AMI。如果此 AMI 無法使用，您可以從已停止的執行個體，建立能夠使用的 AMI。如需詳細資訊，請參閱 [創建一個 Amazon EBS-backed AMI](#)。
- 在 Choose an Instance Type (選取執行個體類型) 頁面上，保留精靈為您選取的預設執行個體類型。
- 在 Configure Instance Details (設定執行個體詳細資訊) 頁面中，指定相同的可用區域做為原始執行個體。如果在 VPC 中啟動執行個體，請選取這個可用區域中的子網。
- 在 Add Tags (新增標籤) 頁面上，將 Name=Temporary 標籤加入執行個體，來表示這是暫時執行個體。
- 在 Review (檢閱) 頁面上，選擇 Launch (啟動)。選擇您在步驟 1 建立的金鑰對，然後選擇 Launch Instances (啟動執行個體)。

#### 步驟 5：將根磁碟區與原始執行個體分離，並將其連接到暫時執行個體

- 在導覽窗格中選取 Volumes (磁碟區)，接著再選取原始執行個體的根裝置磁碟區 (您已在先前的步驟中記下其磁碟區 ID)。選取 Actions (動作)、Detach Volume (分離磁碟區)，然後選取 Detach (分離)。等待磁碟區的狀態變成 available。(您可能需要選取 Refresh (重新整理) 圖示。)
- 在已選取磁碟區的狀態下，選取 Actions (動作)，然後再選取 Attach volume (連接磁碟區)。選取暫時執行個體的執行個體 ID、記下 Device name (裝置名稱) 中的指定裝置名稱 (例如：/dev/sdf)，接著再選取 Attach volume (連接磁碟區)。

#### Note

如果您從 AWS Marketplace AMI 啟動原始執行個體，且磁碟區包含 AWS Marketplace 代碼，則必須先停止暫存執行個體，然後才能連接磁碟區。

## 步驟 6：在掛載到暫時執行個體的原始磁碟區上，將新的公有金鑰新增至的 `authorized_keys`

1. 連線到暫時執行個體。
2. 從暫時執行個體掛載已連結到執行個體的磁碟區，如此您就能存取其檔案系統。例如，如果裝置名稱為 `/dev/sdf`，請使用下列的指令來將磁碟區掛載為 `/mnt/tempvol`。

### Note

在執行個體上所顯示的裝置名稱可能會有不同。例如，掛載為 `/dev/sdf` 的裝置，在執行個體上可能會顯示為 `/dev/xvdf`。Red Hat 的某些版本 (或是其變體版本，例如 CentOS) 甚至可能會將結尾的字母增加 4 個字元，其中 `/dev/sdf` 會變成 `/dev/xvdk`。

- a. 使用 `lsblk` 命令來查看磁碟區是否已分割。

```
[ec2-user ~]$ lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
xvda 202:0 0 8G 0 disk
##xvda1 202:1 0 8G 0 part /
xvdf 202:80 0 101G 0 disk
##xvdf1 202:81 0 101G 0 part
xvdg 202:96 0 30G 0 disk
```

在上述的範例中，`/dev/xvda` 和 `/dev/xvdf` 為分割的磁碟區，而 `/dev/xvdg` 則不是。如果磁碟區已分割，則在接下來的步驟中，您會掛載 (`/dev/xvdf1`) 而非原始裝置 (`/dev/xvdf`)。

- b. 建立暫時的目錄來掛載磁碟區。

```
[ec2-user ~]$ sudo mkdir /mnt/tempvol
```

- c. 使用您先前找出的磁碟區名稱或裝置名稱，將磁碟區 (或分割磁碟區) 掛載至暫時掛載點。必要命令取決於作業系統的檔案系統。請注意，在執行個體上所顯示的設備名稱可能會有不同。如需詳細資訊，請參閱步驟 6 中的 [note](#)。
  - Amazon Linux、Ubuntu 和 Debian



```
[ec2-user ~]$ sudo mount /dev/xvdf1 /mnt/tempvol
```

- Amazon Linux 2、CentOS、SUSE Linux 12 和 RHEL 7.x

```
[ec2-user ~]$ sudo mount -o nouuid /dev/xvdf1 /mnt/tempvol
```

### Note

如果您收到檔案系統毀損的錯誤，請執行下列命令，使用 `fsck` 公用程式來檢查檔案系統並修復任何問題：

```
[ec2-user ~]$ sudo fsck /dev/xvdf1
```

3. 從暫時執行個體使用下列的指令，用暫時執行個體的 `authorized_keys` 所產生的新公有金鑰，來更新已掛載磁碟區上的 `authorized_keys`。

### Important

下列的範例使用 Amazon Linux 使用者名稱 `ec2-user`。您可以需要換成不同的使用者名稱，例如適用於 Ubuntu 執行個體的 `ubuntu`。

```
[ec2-user ~]$ cp .ssh/authorized_keys /mnt/tempvol/home/ec2-user/.ssh/authorized_keys
```

如果此項複製動作成功，您就可以進行下一個步驟。

(選用) 否則，如果您沒有編輯 `/mnt/tempvol` 中的檔案的許可，必須使用 `sudo` 來更新檔案，然後檢查對於該檔案的許可，以確認您是否能夠登入原始執行個體。使用下列命令以檢查對檔案的許可。

```
[ec2-user ~]$ sudo ls -l /mnt/tempvol/home/ec2-user/.ssh
total 4
-rw----- 1 222 500 398 Sep 13 22:54 authorized_keys
```

在本範例輸出中，**222** 為使用者 ID，而 **500** 為群組 ID。接下來，請使用 `sudo` 來重新執行先前失敗的複製命令。

```
[ec2-user ~]$ sudo cp .ssh/authorized_keys /mnt/tempvol/home/ec2-user/.ssh/authorized_keys
```

再次執行下列命令，以確定許可是否已變更。

```
[ec2-user ~]$ sudo ls -l /mnt/tempvol/home/ec2-user/.ssh
```

如果使用者 ID 和群組 ID 已變更，請使用下列命令來還原這些 ID。

```
[ec2-user ~]$ sudo chown 222:500 /mnt/tempvol/home/ec2-user/.ssh/authorized_keys
```

## 步驟 7：取消掛載原始磁碟區，並將該磁碟區從暫時執行個體分離，然後再重新連接到原始執行個體

1. 從暫時執行個體取消掛載已連結的磁碟區，如此您就能將該磁碟區重新連結到原始執行個體。例如，使用以下命令來取消掛載位於 `/mnt/tempvol` 的磁碟區。

```
[ec2-user ~]$ sudo umount /mnt/tempvol
```

2. 將磁碟區從暫時執行個體分離 (您在上一步已卸載該磁碟區)：從 Amazon EC2 主控台，選取導覽窗格中的 Volumes (磁碟區)，選取原始執行個體的根裝置磁碟區 (您在上一步已記下磁碟區 ID)，選取 Actions (動作)、Detach volume (分離磁碟區)，接著再選取 Detach (分離)。等待磁碟區的狀態變成 available。(您可能需要選取 Refresh (重新整理) 圖示。)
3. 將磁碟區重新連接到原始執行個體：在已選取磁碟區的狀態下，選取 Actions (動作)、Attach volume (連接磁碟區)。選取原始執行個體的執行個體 ID、指定您先前在 [步驟 2](#) 中為原始根裝置連接 (`/dev/sda1` 或 `/dev/xvda`) 記錄的裝置名稱，然後選取 Attach volume (連接磁碟區)。

**⚠ Important**

如果您並未指定與原始連結相同的裝置名稱，則無法啟動原始執行個體。Amazon EC2 預期根裝置磁碟區位於 `sda1` 或 `/dev/xvda`。

## 步驟 8：使用新的金鑰對連線到原始執行個體

選取原始執行個體，然後依序選取 Instance state (執行個體狀態)、Start instance (啟動執行個體)。在執行個體進入 `running` 狀態之後，您可以使用新金鑰對的私有金鑰檔案，來連線到該執行個體。

**i Note**

如果新金鑰對的名稱和對應的私有金鑰檔案，與原始金鑰對的名稱不同，請務必在連線到執行個體時，指定新私有金鑰檔案的名稱。

## 步驟 9：清理

(選用) 如果不需再使用時，您可以終止暫時執行個體。選取暫時執行個體，再依序選取 Instance state (執行個體狀態)、Terminate instance (終止執行個體)。

如果使用 Putty 連線至執行個體，且需要將 `.pem` 檔案轉換為 `.ppk`，則請參閱本節 [使用 PuTTY 從 Windows 連線至 Linux 執行個體](#) 主題中的 [使用 PuTTYgen 轉換私有金鑰](#)。

- 設定私有金鑰的許可，確保只有您可以讀取
- 從 macOS 或 Linux 進行連線

(Linux 執行個體) 如果您打算在 macOS 或 Linux 電腦上使用安全殼層用戶端連線到 Linux 執行個體，請使用下列指令來設定私密金鑰檔案的權限，以便只有您才能讀取它。

```
chmod 400 key-pair-name.pem
```

如果您未設定這些許可，則無法使用此金鑰對來連接至您的執行個體。如需詳細資訊，請參閱 [錯誤：未受保護的私有金鑰檔案](#)。

- 從 Windows 進行連線

開啟檔案總管並在 .pem 檔案上按一下滑鼠右鍵。選取屬性 > 安全性索引標籤，然後選擇進階。選擇停用繼承。移除目前使用者以外的所有使用者存取權。

## (選用) 取得執行個體指紋

為了保護自己免受 man-in-the-middle 攻擊，您可以驗證顯示的指紋來驗證即將連線的執行個體的真實性。如果您從第三方提供的公用 AMI 啟動執行個體，則驗證指紋非常有用。

### 任務概觀

首先，從執行個體取得執行個體指紋。然後，當您連線至執行個體並提示您驗證指紋時，請將您在此程序中取得的指紋與顯示的指紋進行比較。如果指紋不匹配，可能有人試圖 man-in-the-middle 攻擊。如果相符，您就能很有信心地連線至您的執行個體。

### 取得執行個體指紋的先決條件

- 執行個體必不得處於 pending 狀態。只有在執行個體的第一次開機完成後，才能使用指紋。
- 您必須是執行個體擁有者才能取得主控台輸出。
- 有多種方法可以獲取實例指紋。如果您要使用 AWS CLI，則必須將其安裝在本機電腦上。若要取得有關安裝的資訊 AWS CLI，請參閱《AWS Command Line Interface 使用指南》AWS Command Line Interface 中的〈[安裝](#)〉。

### 取得執行個體指紋

在步驟 1 中，您將獲得控制台輸出，其中包括實例指紋。在步驟 2 中，您會在主控台輸出中找到執行個體指紋。

1. 使用下列其中一種方法取得主控台輸出。

#### Console

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 從左側瀏覽器中，選擇「執行環境」。
3. 選取執行個體，然後選擇 [動作]、[監控和疑難排解]、[取得系統記錄]。

## AWS CLI

在您的本機電腦上 (不在您要連線的執行個體上)，使用 `get-Console-output ()` 指令。AWS CLI 如果輸出很大，[可以將輸出傳送至文字檔案](#)，這可能更易於讀取。請注意，當您使用 AWS 區域時，您必須明確指定 AWS CLI，或透過設定預設「區域」(Region) 來指定。如需有關如何設定或指定區域的資訊，請參閱《AWS Command Line Interface 使用者指南》中的[組態基礎概念](#)。

```
aws ec2 get-console-output --instance-id instance_id --query Output --output text > temp.txt
```

2. 在控制台輸出中，找到位於下的實例 (主機) 指紋BEGIN SSH HOST KEY FINGERPRINTS。可能有幾個實例指紋。當您連線至執行個體時，它只會顯示其中一個指紋。

具體的輸出可能因作業系統、AMI 版本以及是否使用 AWS 建立的金鑰對而有所不同。下列為範例輸出。

```
ec2:#####
ec2: -----BEGIN SSH HOST KEY FINGERPRINTS-----
ec2: 256 SHA256:l4UB/neBad9tvkgJf1QZWxheQmR59WgrgzEimCG6kZY no comment (ECDSA)
ec2: 256 SHA256:kpEa+rw/Uq3zxaYZN8KT501iBtJ0IdHG52dFi66EEfQ no comment (ED25519)
ec2: 2048 SHA256:L8l6pepcA7iqW/jBecQjVZClUrKY+o2cHLI0iHerbVc no comment (RSA)
ec2: -----END SSH HOST KEY FINGERPRINTS-----
ec2: #####
```

### Note

當您連線至執行個體時，您會參考此指紋。

## 使用 SSH 從 Linux 或 macOS 連線至 Linux 執行個體

您可以使用安全殼層 (SSH) 從執行 Linux 或 macOS 作業系統的本機電腦 Connect 線到 Linux 執行個體，也可以使用獨立於平台的連線工具，例如 EC2 執行個體連線或工作 AWS Systems Manager 階段管理員。如需有關平台獨立工具的詳細資訊，請參閱[連接至您的 Linux 執行個體](#)。

本頁旨在說明如何透過 SSH 用戶端連線至執行個體。若要從 Windows 連線至 Linux 執行個體，請參閱[從 Windows 進行連線](#)。

**Note**

如果您在嘗試連線至執行個體時收到錯誤訊息，請確認執行個體符合所有 [SSH 連線先決條件](#)。如果其符合所有先決條件，但您仍然無法連線至 Linux 執行個體，請參閱 [連線到 Linux 執行個體的疑難排解](#)。

## 目錄

- [SSH 連線先決條件](#)
- [使用 SSH 用戶端連接至 Linux 執行個體](#)
- [使用 SCP 用戶端將檔案傳輸至 Linux 執行個體](#)

## SSH 連線先決條件

連線到您的 Linux 執行個體之前，請先完成下列先決條件。

## 檢查您的執行個體狀態

啟動執行個體之後，執行個體可能需要幾分鐘的時間準備就緒讓您連接。請確認您的執行個體已通過狀態檢查。您可以在 Instances (執行個體) 頁面的 Status check (狀態檢查) 欄檢視此資訊。

取得公有 DNS 名稱和使用者名稱，以連線到執行個體

若要尋找執行個體的公有 DNS 名稱或 IP 地址，以及連接執行個體時應使用的使用者名稱，請參閱 [取得執行個體的相關資訊](#)。

## 找出私有金鑰並設定許可

若要找出連線到執行個體所需的私有金鑰，並設定金鑰許可，請參閱 [找出私有金鑰並設定許可](#)。

## 視需要在本機電腦安裝 SSH 用戶端

您的本機電腦可能已預設安裝 SSH 用戶端。您可以透過在命令列輸入 ssh 來進行驗證。若您的電腦無法識別該命令，您可以安裝 SSH 用戶端。

- Windows Server 2019 和 Windows 10 的最新版本 - 已隨附 OpenSSH 並做為可安裝的元件提供。如需詳細資訊，請參閱 [Windows 中的 OpenSSH](#)。
- 舊版 Windows - 下載並安裝 OpenSSH。如需詳細資訊，請參閱 [Win32-OpenSSH](#)。
- Linux 和 macOS X - 下載並安裝 OpenSSH。如需詳細資訊，請參閱 <https://www.openssh.com>。

## 使用 SSH 用戶端連接至 Linux 執行個體

請使用下列程序使用 SSH 用戶端連線至您的 Linux 執行個體。如果您在嘗試連線至執行個體時遇到錯誤，請參閱 [連線到 Linux 執行個體的疑難排解](#)。

### 使用 SSH 連線至您的執行個體

1. 在終端機視窗中，使用 `ssh` 命令來連接到執行個體。您指定私密金鑰 (`.pem`) 的路徑和檔案名稱、執行個體的使用者名稱，以及執行個體的公有 DNS 名稱或 IPv6 位址。如需如何尋找私密金鑰、您的執行個體的使用者名稱，以及執行個體 DNS 名稱或 IPv6 位址的詳細資訊，請參閱 [找出私有金鑰並設定許可](#) 和 [取得執行個體的相關資訊](#)。若要連線到您的執行個體，請執行下列其中一個命令。

- (公有 DNS) 若要使用執行個體的公有 DNS 名稱進行連線，請輸入下列命令。

```
ssh -i /path/key-pair-name.pem instance-user-name@instance-public-dns-name
```

- (IPv6) 或者，如果您的執行個體具有 IPv6 位址，若要使用執行個體的 IPv6 位址進行連線，請輸入下列命令。

```
ssh -i /path/key-pair-name.pem instance-user-name@instance-IPv6-address
```

您會看到如下的回應：

```
The authenticity of host 'ec2-198-51-100-1.compute-1.amazonaws.com (198-51-100-1)'
can't be established.
ECDSA key fingerprint is l4UB/neBad9tvkgJf1QZWxheQmR59WgrgzEimCG6kZY.
Are you sure you want to continue connecting (yes/no)?
```

2. (選用) 確認安全提醒中的指紋是否符合您在 [\(選用\) 取得執行個體指紋](#) 中取得的指紋。如果這些指紋不匹配，可能是有人試圖 man-in-the-middle 攻擊。如果它們符合，請繼續下一個步驟。
3. 輸入 **yes**。

您會看到如下的回應：

```
Warning: Permanently added 'ec2-198-51-100-1.compute-1.amazonaws.com' (ECDSA) to
the list of known hosts.
```

## 使用 SCP 用戶端將檔案傳輸至 Linux 執行個體

在本機電腦和 Linux 執行個體之間傳輸檔案的一種方式是使用安全複製通訊協定 (SCP)。本節說明如何使用 SCP 傳輸檔案。此程序類似使用 SSH 連線到執行個體的程序。

### 必要條件

- 確認將檔案傳輸到執行個體的一般先決條件。

在本機電腦和執行個體之間傳輸檔案前，請先執行以下動作，確保自己擁有所需的所有資訊。

- [取得執行個體的相關資訊](#)
- [找出私有金鑰並設定許可](#)
- [\(選用\) 取得執行個體指紋](#)
- 安裝 SCP 用戶端

大部分的 Linux、Unix 和 Apple 電腦預設都包含 SCP 用戶端。如果您的電腦沒有 SCP 用戶端，OpenSSH 專案會提供 SSH 工具完整套件的免費實作，包括 SCP 用戶端。如需詳細資訊，請參閱 <https://www.openssh.com>。

下列程序會逐步引導您使用 SCP 使用執行個體的公有 DNS 名稱傳輸檔案，或是使用 IPv6 地址 (如果執行個體具備的話) 傳輸檔案。

### 使用 SCP 在電腦和執行個體之間傳輸檔案

1. 決定來源檔案在電腦上的位置，以及執行個體上的目的地路徑。在下列範例中，私有金鑰檔案的名稱為 `key-pair-name.pem`，要傳輸的檔案為 `my-file.txt`，執行個體的使用者名稱為 `ec2-user`，執行個體的公有 DNS 名稱為 `instance-public-dns-name`，而執行個體的 IPv6 地址為 `instance-IPv6-address`。
  - (公用 DNS) 若要將檔案傳輸到執行個體上的目的地，請從您的電腦輸入下列命令。

```
scp -i /path/key-pair-name.pem /path/my-file.txt ec2-user@instance-public-dns-name:path/
```

- (IPv6) 若要將檔案傳輸到執行個體上的目的地 (如果執行個體具有 IPv6 地址)，請從您的電腦輸入下列命令。IPv6 位址必須用方括號 ([ ]) 括住，必須逸出 (\)。

```
scp -i /path/key-pair-name.pem /path/my-file.txt ec2-user@[instance-IPv6-address]:path/
```



2. 如果您尚未使用 SSH 連接到執行個體，則會看到如下所示的回應：

```
The authenticity of host 'ec2-198-51-100-1.compute-1.amazonaws.com (10.254.142.33)'
can't be established.
RSA key fingerprint is 1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f.
Are you sure you want to continue connecting (yes/no)?
```

(選用) 您可以選用驗證安全性警示中的指紋是否符合執行個體指紋。如需詳細資訊，請參閱 [\(選用\) 取得執行個體指紋](#)。

輸入 **yes**。

3. 如果傳輸成功，則回應會和下列內容相似：

```
Warning: Permanently added 'ec2-198-51-100-1.compute-1.amazonaws.com' (RSA)
to the list of known hosts.
my-file.txt 100% 480 24.4KB/s 00:00
```

4. 若要反向 (從 Amazon EC2 執行個體到您的電腦) 傳輸檔案，請反轉主機參數的順序。例如：您可以從 EC2 執行個體傳輸 `my-file.txt` 到本機電腦上的目的地 `my-file2.txt`，如下列範例所示。

- (公用 DNS) 若要將檔案傳輸到電腦上的目的地，請從您的電腦輸入下列命令。

```
scp -i /path/key-pair-name.pem ec2-user@instance-public-dns-name:path/my-
file.txt path/my-file2.txt
```

- (IPv6) 如果執行個體具有 IPv6 地址，欲將檔案傳輸到電腦上的目的地，請從您的電腦輸入下列命令。IPv6 位址必須用方括號 ([ ]) 括住，必須逸出 (\)。

```
scp -i /path/key-pair-name.pem ec2-user@[instance-IPv6-address]:path/my-
file.txt path/my-file2.txt
```

## 從 Windows 連線至 Linux 執行個體

您可以使用以下方法，從執行 Windows 作業系統的本機電腦連線至 Linux 執行個體。

## 使用 OpenSSH 從 Windows 連線至 Linux 執行個體

下列程序說明如何使用 OpenSSH (它是一個開放原始碼連線工具，使用 SSH 通訊協定進行遠端登入) 從 Windows 連線至您的 Linux 執行個體。OpenSSH 在 Windows Server 2019 和更高版本的作業系統中受支援。

### 內容

- [必要條件](#)
- [使用下列方式安裝 OpenSSH PowerShell](#)
- [使用 OpenSSH 從 Windows 連線至您的 Linux 執行個體](#)
- [使用從視窗解除安裝 OpenSSH PowerShell](#)

### 必要條件

在使用 OpenSSH 從 Windows 連線到您的 Linux 執行個體之前，請先完成下列先決條件。

#### 確認執行個體已準備就緒

啟動執行個體之後，執行個體可能需要幾分鐘的時間準備就緒讓您連接。請確認您的執行個體已通過狀態檢查。您可以在 Instances (執行個體) 頁面的 Status check (狀態檢查) 欄檢視此資訊。

#### 驗證連接到執行個體的一般先決條件

若要尋找執行個體的公有 DNS 名稱或 IP 地址，以及連線執行個體時應使用的使用者名稱，請參閱 [取得執行個體的相關資訊](#)。

#### 驗證您的 Windows 版本

若要使用 OpenSSH 從 Windows 連線到您的 Linux 執行個體，Windows 版本必須是 Windows Server 2019 及更高版本。

#### 確認 PowerShell 先決條件

若要使用在 Windows 作業系統上安裝 OpenSSH PowerShell，您必須執行 5.1 或更新 PowerShell 版本，而且您的帳戶必須是內建管理員群組的成員。`$PSVersionTable.PSVersion` 從運行 PowerShell 以檢查您的 PowerShell 版本。

若要檢查您是否是內建管理員群組的成員，請執行下列 PowerShell 命令：

```
(New-Object Security.Principal.WindowsPrincipal([Security.Principal.WindowsIdentity]::GetCurrent())).Is
```

如果您是內建管理員群組的成員，則輸出為 True。

使用下列方式安裝 OpenSSH PowerShell

若要使用安裝 OpenSSH PowerShell，請執行下列 PowerShell 命令：

```
Add-WindowsCapability -Online -Name OpenSSH.Client~~~~0.0.1.0
```

預期的輸出結果：

```
Path :
Online : True
RestartNeeded : False
```

使用 OpenSSH 從 Windows 連線至您的 Linux 執行個體

在安裝 OpenSSH 之後，請使用下列程序透過 OpenSSH 從 Windows 連線至您的 Linux 執行個體。如果您在嘗試連線至執行個體時遇到錯誤，請參閱 [連線到 Linux 執行個體的疑難排解](#)。

若要使用 OpenSSH 連線到您的執行個體

1. 在 PowerShell 或命令提示字元中，使用 ssh 指令連線至執行個體。您指定私有金鑰 (.pem) 的路徑和檔案名稱、執行個體的使用者名稱，以及執行個體的公有 DNS 名稱或 IPv6 地址。如需有關如何尋找私有金鑰、執行個體的使用者名稱，以及執行個體 DNS 名稱或 IPv6 地址的詳細資訊，請參閱 [找出私有金鑰並設定許可](#) 和 [取得執行個體的相關資訊](#)。若要連線到您的執行個體，請執行下列其中一個命令。
  - (公有 DNS) 若要使用執行個體的公有 DNS 名稱進行連線，請輸入下列命令。

```
ssh -i /path/key-pair-name.pem instance-user-name@instance-public-dns-name
```

- (IPv6) 或者，如果您的執行個體具有 IPv6 位址，若要使用執行個體的 IPv6 位址進行連線，請輸入下列命令。

```
ssh -i /path/key-pair-name.pem instance-user-name@instance-IPv6-address
```

您會看到如下的回應：

```
The authenticity of host 'ec2-198-51-100-1.compute-1.amazonaws.com (198-51-100-1)'
can't be established.
```

```
ECDSA key fingerprint is 14UB/neBad9tvkgJf1QZWxheQmR59WgrgzEimCG6kZY.
Are you sure you want to continue connecting (yes/no/[fingerprint])?
```

- (選用) 確認安全提醒中的指紋是否符合您在[\(選用\) 取得執行個體指紋](#)中取得的指紋。如果這些指紋不匹配，可能是有人試圖 man-in-the-middle 攻擊。如果它們符合，請繼續下一個步驟。
- 輸入 **yes**。

您會看到如下的回應：

```
Warning: Permanently added 'ec2-198-51-100-1.compute-1.amazonaws.com' (ECDSA) to
the list of known hosts.
```

## 使用從視窗解除安裝 OpenSSH PowerShell

若要使用解除安裝 OpenSSH PowerShell，請執行下列 PowerShell 命令：

```
Remove-WindowsCapability -Online -Name OpenSSH.Client~~~~0.0.1.0
```

預期的輸出結果：

```
Path :
Online : True
RestartNeeded : True
```

## 使用 PuTTY 從 Windows 連線至 Linux 執行個體

如果執行的是 Windows Server 2019 或更高版本，建議您使用 OpenSSH。OpenSSH 是一種開放原始碼連線工具，可透過 SSH 通訊協定進行遠端登入。如需使用 OpenSSH 從 Windows 連線至 Linux 執行個體的步驟，請參閱 [使用 OpenSSH 從 Windows 連線至 Linux 執行個體](#)。

下列指示會說明如何使用 PuTTY (Windows 免費的 SSH 用戶端) 連線至您的執行個體。如果您在嘗試連線至執行個體時遇到錯誤，請參閱 [連線到 Linux 執行個體的疑難排解](#)。

### 內容

- [必要條件](#)
  - [使用 PuTTYgen 轉換私有金鑰](#)
- [連接至您的 Linux 執行個體](#)
- [使用 PuTTY 安全複製用戶端將檔案傳輸到您的 Linux 執行個體](#)

- [使用 WinSCP 將檔案傳輸到您的 Linux 執行個體](#)

## 必要條件

使用 PuTTY 連線到您的 Linux 執行個體之前，請先完成下列先決條件。

### 確認執行個體已準備就緒

啟動執行個體之後，執行個體可能需要幾分鐘的時間準備就緒讓您連接。請確認您的執行個體已通過狀態檢查。您可以在 Instances (執行個體) 頁面的 Status check (狀態檢查) 欄檢視此資訊。

### 驗證連接到執行個體的一般先決條件

若要尋找執行個體的公有 DNS 名稱或 IP 地址，以及連線執行個體時應使用的使用者名稱，請參閱 [取得執行個體的相關資訊](#)。

### 在本機電腦安裝 PuTTY

從 [PuTTY 下載頁面](#) 下載並安裝 PuTTY。如已安裝舊版的 PuTTY，建議您下載最新版本。務必安裝整個套件。

### 使用 PuTTYgen 轉換私有 .pem 金鑰

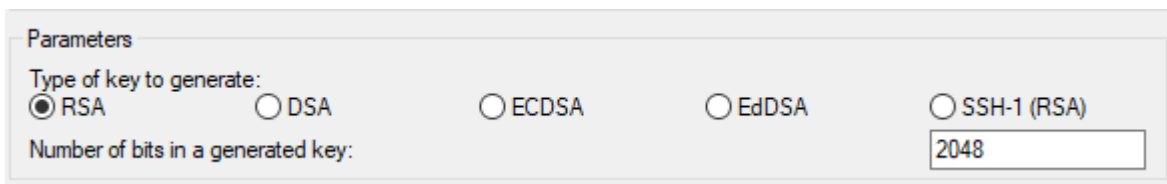
對於您在啟動執行個體時所指定的金鑰對，如果您選擇以 .pem 格式建立私有金鑰，則必須將其轉換為 .ppk 檔案，才能與 PuTTY 搭配使用。找到私有 .pem 檔案，然後按照下一節中的步驟進行。

### 使用 PuTTYgen 轉換私有金鑰

PuTTY 原生並不支援 PEM 格式的 SSH 金鑰。PuTTY 提供一套名為 PuTTYgen 的工具，可將 PEM 金鑰轉換為 PuTTY 所需的 PPK 格式。您必須先如下所示將您的私有金鑰 (.pem 檔案) 轉換為此格式 (.ppk 檔案)，才能使用 PuTTY 連線至您的執行個體。

#### 將私有 .pem 金鑰轉換成 .ppk

1. 從 Start (開始) 功能表中選擇 All Programs (所有程式)、PuTTY、PuTTYgen。
2. 在 Type of key to generate (要產生的金鑰類型) 下，選擇 RSA (SSH-2 RSA)。如果您的 PuTTYgen 版本不包含此選項，請選擇 SSH-2 RSA。



The image shows a screenshot of the PuTTYgen 'Parameters' dialog box. It features a section titled 'Type of key to generate:' with five radio button options: RSA (selected), DSA, ECDSA, EdDSA, and SSH-1 (RSA). Below this, there is a text input field for 'Number of bits in a generated key:' with the value '2048' entered.

- 選擇 Load (載入)。根據預設，PuTTYgen 只會顯示副檔名為 .ppk 的檔案。若要尋找您的 .pem 檔案，請選擇顯示所有類型之檔案的選項。



- 為您在啟動執行個體時所指定的金鑰對選取 .pem 檔案，然後選取 Open (開啟)。PuTTYgen 會顯示一個通知，指出 .pem 檔案已成功匯入。選擇 OK (確定)。
- 若要以 PuTTY 可使用的格式儲存金鑰，請選擇 Save private key (儲存私有金鑰)。PuTTYgen 會顯示有關儲存沒有密碼短語之金鑰的警告。選擇 Yes (是)。

#### Note

私有金鑰上的複雜密碼是額外一層保護。即使有人探索到您的私有金鑰，在沒有複雜密碼的情況下就無法使用您的私有金鑰。使用複雜密碼的缺點是提高自動化的難度，因為需要人力介入登入執行個體，或將檔案複製到執行個體。

- 為您用於金鑰對的金鑰指定相同名稱 (例如，key-pair-name)，然後選擇 Save (儲存)。PuTTY 會自動新增 .ppk 副檔名。

您的私有金鑰對現已有正確的格式可供 PuTTY 使用。您現在可以使用 PuTTY 的 SSH 用戶端連線至您的執行個體。

### 連接至您的 Linux 執行個體

請使用下列程序使用 PuTTY 連線至您的 Linux 執行個體。需要您為私有金鑰建立的 .ppk 檔案。如需詳細資訊，請參閱上一節中的[使用 PuTTYgen 轉換私有金鑰](#)。如果您在嘗試連線至執行個體時遇到錯誤，請參閱[連線到 Linux 執行個體的疑難排解](#)。

最後測試 PuTTY 的版本：.78

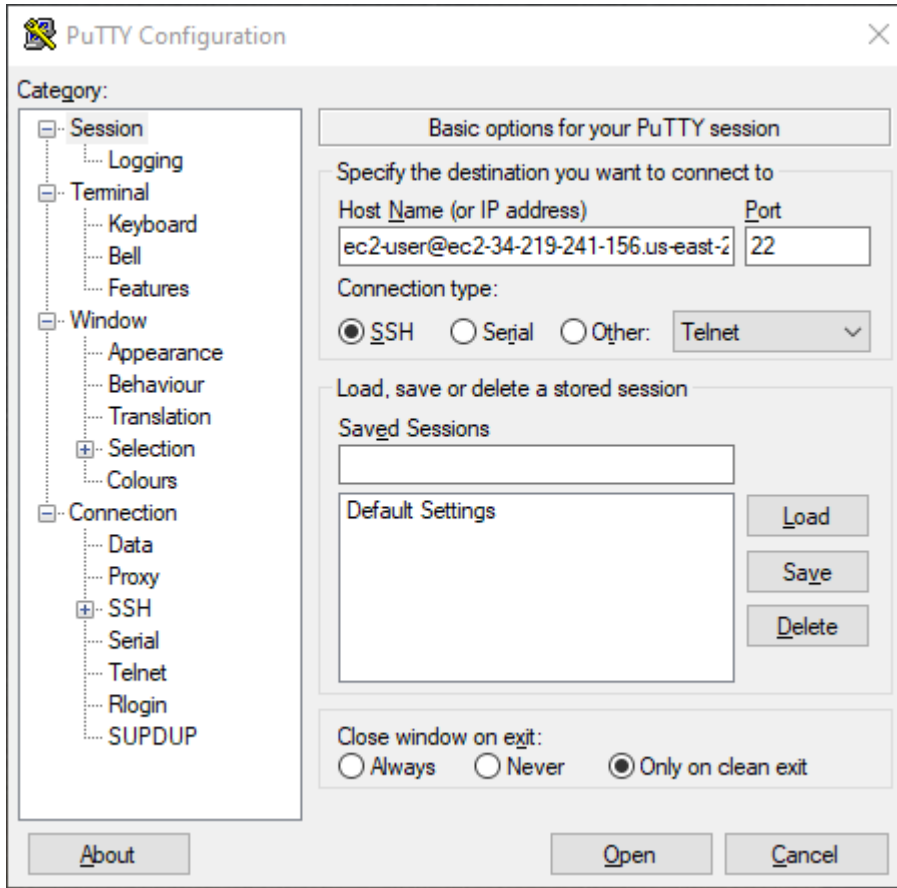
### 使用 PuTTY 連線到您的執行個體

- 啟動 PuTTY (在開始選單搜尋 PuTTY，然後選擇開啟)。
- 在 Category (類別) 窗格中選擇 Session (工作階段)，然後填妥下列欄位：
  - 在 Host Name (主機名稱) 方塊中，執行下列其中一項作業：
    - (公有 DNS) 若要使用執行個體的公有 DNS 名稱連線，請輸入 *instance-user-name@instance-public-dns-name*。

- (IPv6) 或者，如果您的執行個體具有 IPv6 位址，若要使用執行個體的 IPv6 位址進行連線，請輸入 `instance-user-name@instance-IPv6-address`。


如需有關如何尋找您執行個體的使用者名稱，以及執行個體的公有 DNS 名稱或 IPv6 地址的詳細資訊，請參閱 [取得執行個體的相關資訊](#)。

- b. 確保 Port (連接埠) 值是 22。
- c. 在 Connection type (連線類型) 下，選取 SSH。



3. (選用) 您可設定 PuTTY 按定期間隔自動傳送保持連線資料，以保持工作階段持續作用。這對避免您執行個體因工作階段閒置而中斷連線很有幫助。在類別窗格中選擇連線，然後在保持連線之間的秒數中輸入所需間隔。例如，如果您的工作階段在閒置 10 分鐘後中斷連線，請輸入 180 以設定 PuTTY 每 3 分鐘傳送一次保持連線資料。
4. 在類別窗格中，展開連線、展開 SSH，然後選擇身分驗證。選擇憑證。
5. 在要驗證的私有金鑰檔案旁，選擇瀏覽。在選取私有金鑰檔案對話方塊中，選取您為金鑰對產生的 .ppk 檔案。您可以按兩下檔案，或在選取私有金鑰檔案對話方塊中選擇開啟。

6. (選用) 如果您打算稍後再次連線到此執行個體，可以儲存工作階段資訊以供日後使用。在類別窗格中，選擇工作階段。在儲存的工作階段中輸入工作階段的名稱，然後選擇儲存。
7. 若要連線至執行個體，請選擇開啟。
8. 如果這是您第一次連線到此執行個體，PuTTY 會顯示安全提醒對話方塊，詢問您是否信任要連線的主機。
  - a. (選用) 確認安全提醒對話方塊中的指紋是否符合您之前在[\(選用\) 取得執行個體指紋](#)中取得的指紋。如果這些指紋不符合，可能有人嘗試發動中間人攻擊。如果它們符合，請繼續下一個步驟。
  - b. 選擇 Accept (接受)。隨即開啟視窗，您會連線到您的執行個體。

 Note

當您將私有金鑰轉換成 PuTTY 格式時，如已指定複雜密碼，您必須在登入執行個體時提供該複雜密碼。

如果您在嘗試連線至執行個體時遇到錯誤，請參閱 [連線到 Linux 執行個體的疑難排解](#)。

使用 PuTTY 安全複製用戶端將檔案傳輸到您的 Linux 執行個體

PuTTY 安全複製用戶端 (PSCP) 是命令列工具，您可用來在 Windows 電腦和 Linux 執行個體間傳輸檔案。如果您偏好圖形使用者介面 (GUI)，您可使用名為 WinSCP 的開放原始碼 GUI 工具。如需詳細資訊，請參閱 [使用 WinSCP 將檔案傳輸到您的 Linux 執行個體](#)。

若要使用 PSCP，需要您在[使用 PuTTYgen 轉換私有金鑰](#)中產生的私有金鑰。您還需要 Linux 執行個體的公有 DNS 名稱，或如果執行個體有 IPv6 位址，則還需要它。

下列範例會將檔案 Sample\_file.txt 從 Windows 電腦的 C:\ 磁碟機傳輸到 Amazon Linux 執行個體的 instance-user-name 主目錄。若要傳輸檔案，請使用下列其中一個命令。

- (公有 DNS) 若要使用執行個體的公有 DNS 名稱傳輸檔案，請輸入下列命令。

```
pscp -i C:\path\my-key-pair.ppk C:\path\Sample_file.txt instance-user-name@instance-public-dns-name:/home/instance-user-name/Sample_file.txt
```

- (IPv6) 或者，如果您的執行個體具有 IPv6 位址，若要使用您的執行個體的 IPv6 位址進行連線，請輸入下列命令。IPv6 位址必須用方括號 ([ ]) 括住。



```
pscp -i C:\path\my-key-pair.ppk C:\path\Sample_file.txt instance-user-name@[instance-IPv6-address]:/home/instance-user-name/Sample_file.txt
```

## 使用 WinSCP 將檔案傳輸到您的 Linux 執行個體

WinSCP 是 Windows 的 GUI 型檔案管理員，能讓您透過其使用 SFTP、SCP、FTP 和 FTPS 通訊協定將檔案上傳及傳輸到遠端電腦。若使用 WinSCP，您可將 Windows 電腦的檔案拖放到 Linux 執行個體，或在兩個系統間同步整個目錄結構。


### 要求

- 您必須具備在 [使用 PuTTYgen 轉換私有金鑰](#) 中產生的私有金鑰。
- 您也需要 Linux 執行個體的公有 DNS 名稱。
- 您的 Linux 執行個體必須已安裝 scp。對於某些作業系統，您可以安裝 openssh-clients 套件。對於其他作業系統，例如，Amazon ECS 最佳化 AMI，您可以安裝 scp 套件。請查看 Linux 發行版的說明文件。

## 使用 WinSCP 連線到您的執行個體

1. 請從 <http://winscp.net/eng/download.php> 下載並安裝 WinSCP。大部分的使用者使用預設安裝選項即可。
2. 啟動 WinSCP。
3. 在 WinSCP login (WinSCP 登入) 畫面中，針對 Host name (主機名稱)，輸入下列其中一項：
  - (公有 DNS 或 IPv4 位址) 若要使用執行個體的公有 DNS 名稱或公有 IPv4 位址登入，請輸入執行個體的公有 DNS 名稱或公有 IPv4 位址。
  - (IPv6) 或者，如果您的執行個體具有 IPv6 位址，若要使用執行個體的 IPv6 位址登入，請輸入您的執行個體的 IPv6 位址。
4. 在使用者名稱欄位中輸入您 AMI 的預設使用者名稱。
  - 對於 AL2023、Amazon Linux 2 或 Amazon Linux AMI，使用者名稱為 ec2-user。
  - 若使用 CentOS AMI，使用者名稱為 centos 或 ec2-user。
  - 若為 Debian AMI，使用者名稱為 admin。
  - 對於 Fedora AMI，使用者名稱為 fedora 或 ec2-user。
  - 對於 RHEL AMI，使用者名稱為 ec2-user 或 root。

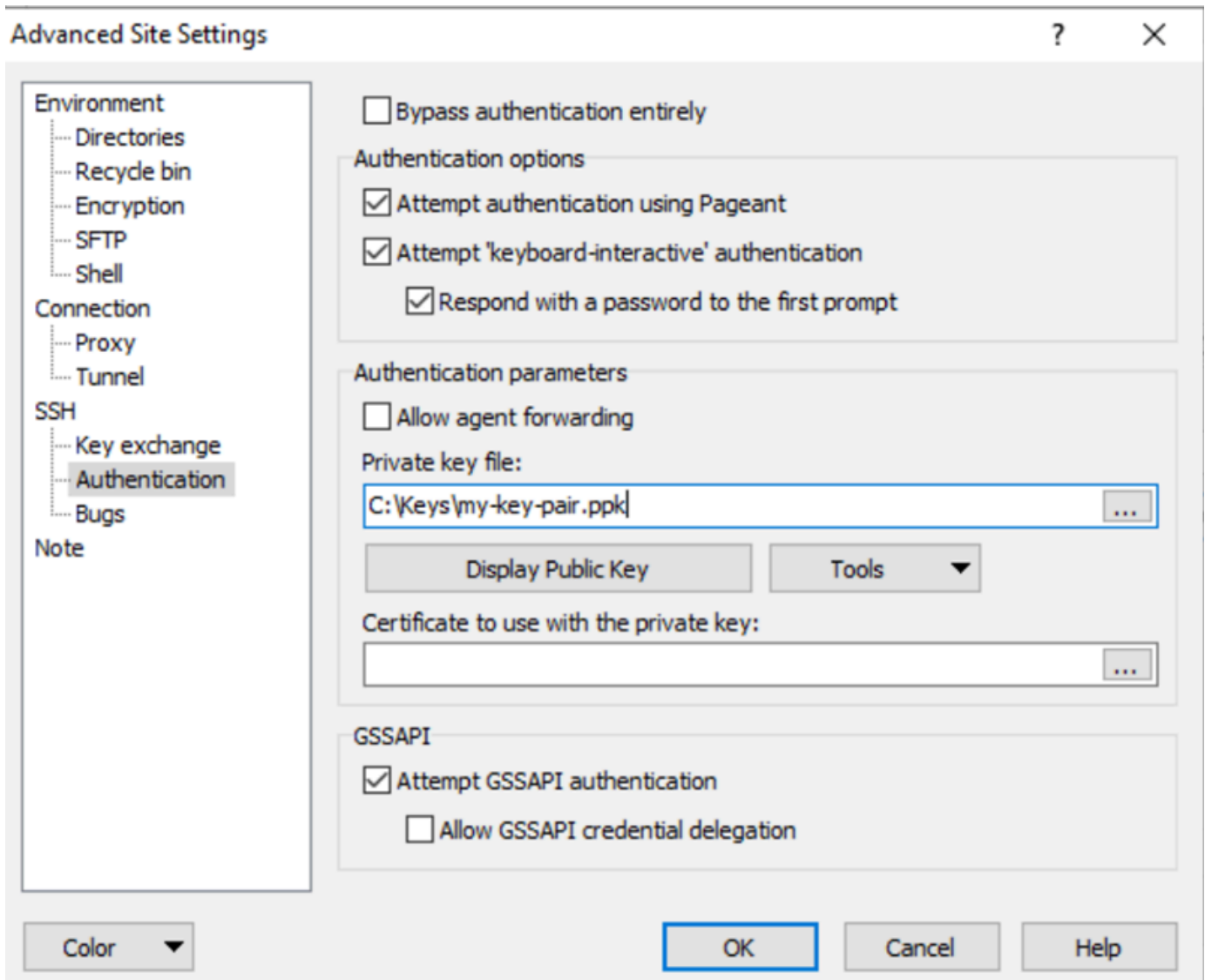
- 針對 SUSE AMI，使用者名稱為 `ec2-user` 或 `root`。
- 若為 Ubuntu AMI，使用者名稱為 `ubuntu`。
- 若使用 Oracle AMI，使用者名稱為 `ec2-user`。
- 若使用 Bitnami AMI，使用者名稱為 `bitnami`。

 Note

若要尋找其他 Linux 發行版的預設使用者名稱，請洽詢 AMI 提供者。

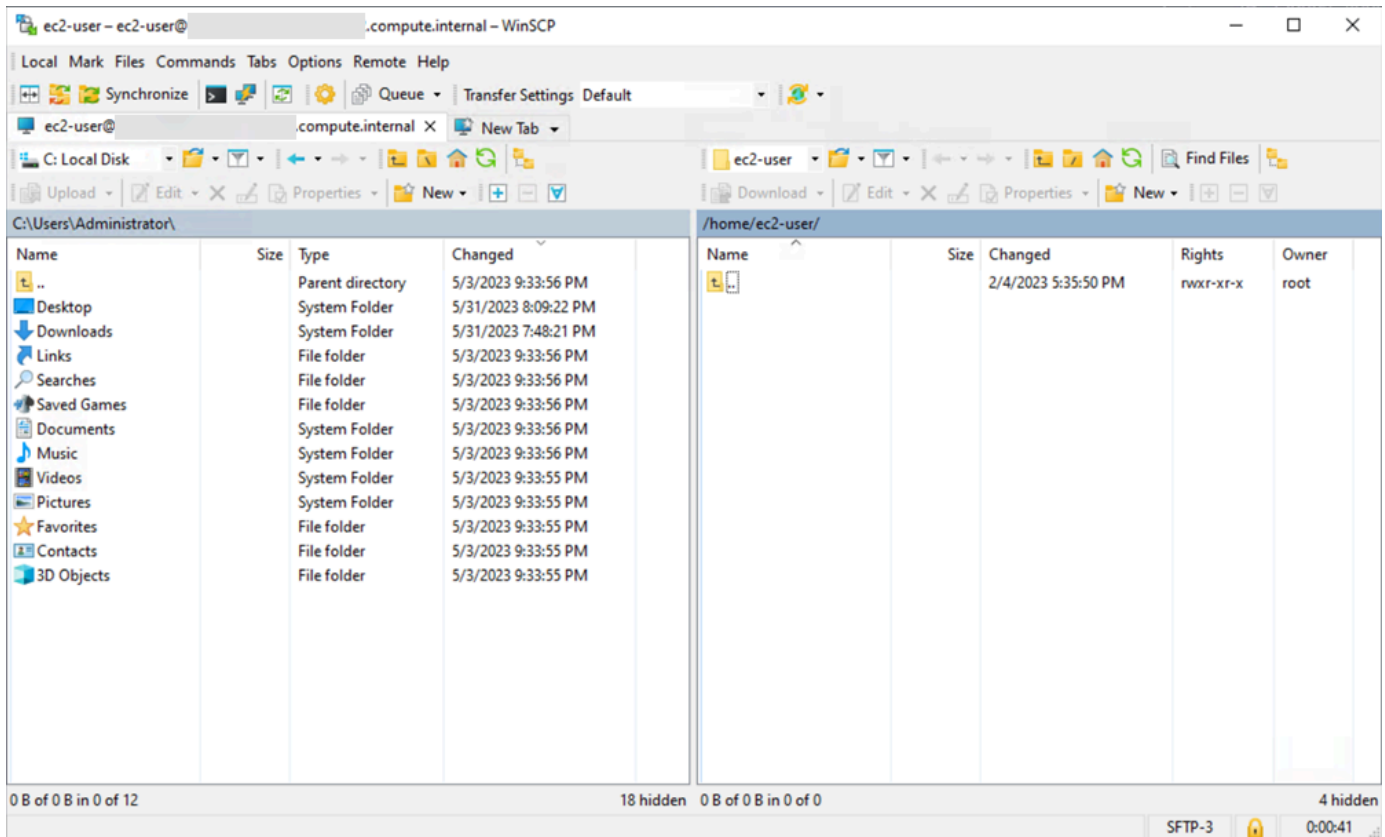
5. 指定您執行個體的私有金鑰檔案。
  - a. 選擇進階... 按鈕。
  - b. 在 SSH 下，選擇驗證。
  - c. 指定您私有金鑰檔案的路徑，或選擇 ... 按鈕，瀏覽至金鑰對檔案。
  - d. 選擇確定。

以下為 WinSCP 6.1 版的螢幕截取畫面：



WinSCP 需要 PuTTY 私有金鑰檔案 (.ppk)。您可使用 PuTTYgen 將 .pem 安全金鑰檔案轉換為 .ppk 格式。如需詳細資訊，請參閱 [使用 PuTTYgen 轉換私有金鑰](#)。

- (選用) 在左面板中，選擇 Directories (目錄)。若是 Remote directory (遠端目錄)，請輸入要新增檔案的目錄路徑。若要在較新版的 WinSCP 開啟進階網站設定，請選擇 Advanced (進階)。若要尋找 Remote directory (遠端目錄) 設定，在 Environment (環境) 下，選擇 Directories (目錄)。
- 選擇 Login (登入)。選擇 Yes (是) 將主機指紋新增至主機快取。



8. 建立連線後，在連線視窗中，您的 Linux 執行個體在右，而您的本機電腦在左。您可以在遠端檔案系統和本機電腦之間拖放檔案。如需 WinSCP 的詳細資訊，請參閱 <http://winscp.net/eng/docs/start> 中的專案文件。

如果您收到無法執行 SCP 以開始傳輸的錯誤，請確認您已在 Linux 執行個體上安裝 scp。

使用適用於 Linux 的 Windows 子系統 (WSL)，從 Windows 連線至 Linux 執行個體

啟動執行個體之後，您可以連接到此執行個體，並以一般使用電腦的方式使用。

下列指示說明如何在適用於 Linux 的 Windows 子系統 (WSL) 上使用 Linux 發行版本連線至執行個體。WSL 是免費下載的軟體，可讓您直接在 Windows 上執行原生 Linux 命令列工具與傳統 Windows 桌面，而不需要額外的虛擬機器。

安裝 WSL 後，您可以使用原生 Linux 環境來連接至 Linux EC2 執行個體，而不需要使用 PuTTY 或 PuTTYgen。Linux 環境可讓您輕鬆連接至 Linux 執行個體，因為此環境隨附一個原生 SSH 用戶端，可用來連接至 Linux 執行個體和變更 .pem 金鑰檔案的許可。Amazon EC2 主控台可提供 SSH 命令以連接至 Linux 執行個體，且您可以透過 SSH 命令取得詳細輸出來進行故障排解。如需詳細資訊，請參閱 [適用於 Linux 的 Windows 子系統文件](#)。

**Note**

安裝 WSL 後，所有先決條件和步驟都與[使用 SSH 從 Linux 或 macOS 連線至 Linux 執行個體](#)中所述的先決條件和步驟相同，而且其體驗就與使用原生 Linux 一樣。

如果您在嘗試連線至執行個體時遇到錯誤，請參閱[連線到 Linux 執行個體的疑難排解](#)。

## 內容

- [必要條件](#)
- [使用 WSL 連線至您的 Linux 執行個體](#)
- [使用 SCP 將檔案從 Linux 傳輸到 Linux 執行個體](#)
- [解除安裝 WSL](#)

## 必要條件

連線到您的 Linux 執行個體之前，請先完成下列先決條件。

### 確認執行個體已準備就緒

啟動執行個體之後，執行個體可能需要幾分鐘的時間準備就緒讓您連接。請確認您的執行個體已通過狀態檢查。您可以在 Instances (執行個體) 頁面的 Status check (狀態檢查) 欄檢視此資訊。

### 驗證連接到執行個體的一般先決條件

若要尋找執行個體的公有 DNS 名稱或 IP 地址，以及連接執行個體時應使用的使用者名稱，請參閱[取得執行個體的相關資訊](#)。

### 在本機電腦安裝適用於 Linux 的 Windows 子系統 (WSL) 和 Linux 發行版本

使用 [Windows 10 安裝指南](#) 中的指示來安裝 WSL 和 Linux 發行版本。指示中的範例是安裝 Linux 的 Ubuntu 發行版本，但您可以安裝任何發行版本。您將會收到提示，要求您重新啟動電腦以讓變更生效。

### 將私有金鑰從 Windows 複製到 WSL

在 WSL 終端機視窗中，將 (啟動執行個體時所指定金鑰對的) .pem 檔案從 Windows 複製到 WSL。請記下 .pem 檔案在 WSL 上的完整路徑，以便於在連接至執行個體時使用。如需有關如何指定 Windows 硬碟路徑的資訊，請參閱[如何存取 C 磁碟機?](#)。如需有關金鑰對和 Windows 執行個體的詳細資訊，請參閱[Amazon EC2 金鑰對和 Windows 執行個體](#)。

```
cp /mnt/<Windows drive letter>/path/my-key-pair.pem ~/WSL-path/my-key-pair.pem
```

## 使用 WSL 連線至您的 Linux 執行個體

根據下列程序使用適用於 Linux 的 Windows 子系統 (WSL) 來連接至 Linux 執行個體。如果您在嘗試連線至執行個體時遇到錯誤，請參閱 [連線到 Linux 執行個體的疑難排解](#)。

## 使用 SSH 連線到您的執行個體

1. 在終端機視窗中，使用 ssh 命令來連接到執行個體。您指定私密金鑰 (.pem) 的路徑和檔案名稱、執行個體的使用者名稱，以及執行個體的公有 DNS 名稱或 IPv6 位址。如需如何尋找私密金鑰、您的執行個體的使用者名稱，以及執行個體 DNS 名稱或 IPv6 位址的詳細資訊，請參閱 [找出私有金鑰並設定許可](#) 和 [取得執行個體的相關資訊](#)。若要連線到您的執行個體，請執行下列其中一個命令。

- (公有 DNS) 若要使用執行個體的公有 DNS 名稱進行連線，請輸入下列命令。

```
ssh -i /path/key-pair-name.pem instance-user-name@my-instance-public-dns-name
```

- (IPv6) 或者，如果您的執行個體具有 IPv6 位址，則可以使用其 IPv6 位址連線到該執行個體。使用私有金鑰 (.pem) 檔案的路徑、適當的使用者名稱和 IPv6 地址來指定 ssh 命令。

```
ssh -i /path/key-pair-name.pem instance-user-name@my-instance-IPv6-address
```

您會看到如下的回應：

```
The authenticity of host 'ec2-198-51-100-1.compute-1.amazonaws.com (10.254.142.33)'
can't be established.
RSA key fingerprint is 1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f.
Are you sure you want to continue connecting (yes/no)?
```

2. (選用) 確認安全提醒中的指紋是否符合您在 [\(選用\) 取得執行個體指紋](#) 中取得的指紋。如果這些指紋不符合，可能有人嘗試發動中間人攻擊。如果它們符合，請繼續下一個步驟。
3. 輸入 yes。

您會看到如下的回應：

```
Warning: Permanently added 'ec2-198-51-100-1.compute-1.amazonaws.com' (RSA)
to the list of known hosts.
```

## 使用 SCP 將檔案從 Linux 傳輸到 Linux 執行個體

在本機電腦和 Linux 執行個體之間傳輸檔案的一種方式是使用安全複製通訊協定 (SCP)。本節說明如何使用 SCP 傳輸檔案。此程序類似使用 SSH 連線到執行個體的程序。

### 必要條件

- 確認將檔案傳輸到執行個體的一般先決條件。

在本機電腦和執行個體之間傳輸檔案前，請先執行以下動作，確保自己擁有所需的所有資訊。

- [取得執行個體的相關資訊](#)
- [找出私有金鑰並設定許可](#)
- [\(選用\) 取得執行個體指紋](#)
- 安裝 SCP 用戶端

大部分的 Linux、Unix 和 Apple 電腦預設都包含 SCP 用戶端。如果您的電腦沒有 SCP 用戶端，OpenSSH 專案會提供 SSH 工具完整套件的免費實作，包括 SCP 用戶端。如需詳細資訊，請參閱 <https://www.openssh.com>。

下列程序會逐步引導您使用 SCP 傳輸檔案。如已使用 SSH 連線到執行個體，並已確認其指紋，您可開始使用包含 SCP 命令的步驟 (步驟 4)。

### 使用 SCP 傳輸檔案

1. 使用執行個體的公有 DNS 名稱將檔案傳輸到您的執行個體。例如，如果私有金鑰檔案的名稱為 `key-pair-name`，要傳輸的檔案為 `SampleFile.txt`，使用者名稱為 `instance-user-name`，且執行個體的公有 DNS 名稱為 `my-instance-public-dns-name`，或 IPv6 位址為 `my-instance-IPv6-address`，請使用下列命令將檔案複製到 `instance-user-name` 主目錄。
  - (公有 DNS) 若要使用執行個體的公有 DNS 名稱傳輸檔案，請輸入下列命令。

```
scp -i /path/key-pair-name.pem /path/SampleFile.txt instance-user-name@my-instance-public-dns-name:~
```

- (IPv6) 或者，如果您的執行個體具有 IPv6 位址，則可以使用該執行個體的 IPv6 位址傳輸檔案。IPv6 位址必須用方括號 ([ ]) 括住，必須逸出 (\)。

```
scp -i /path/key-pair-name.pem /path/SampleFile.txt instance-user-name@[my-
instance-IPv6-address]:~
```

您會看到如下的回應：

```
The authenticity of host 'ec2-198-51-100-1.compute-1.amazonaws.com (10.254.142.33)'
can't be established.
RSA key fingerprint is 1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f.
Are you sure you want to continue connecting (yes/no)?
```

2. (選用) 確認安全提醒中的指紋是否符合您在(選用) [取得執行個體指紋](#)中取得的指紋。如果這些指紋不符合，可能有人嘗試發動中間人攻擊。如果它們符合，請繼續下一個步驟。
3. 輸入 **yes**。

您會看到如下的回應：

```
Warning: Permanently added 'ec2-198-51-100-1.compute-1.amazonaws.com' (RSA)
to the list of known hosts.
Sending file modes: C0644 20 SampleFile.txt
Sink: C0644 20 SampleFile.txt
SampleFile.txt 100% 20 0.0KB/s 00:00
```

如果您收到 "bash: scp: command not found" (bash: scp: 找不到命令) 錯誤，必須先在您的 Linux 執行個體上安裝 scp。此項目在某些作業系統中位於 openssh-clients 套件中。針對 Amazon ECS 最佳化的 AMI 等 Amazon Linux 變體，請使用下列命令來安裝 scp：

```
[ec2-user ~]$ sudo yum install -y openssh-clients
```

4. 若要反向 (從 Amazon EC2 執行個體到本機電腦) 傳輸檔案，請反轉主機參數的順序。例如，若要將 SampleFile.txt 檔案以 SampleFile2.txt 的形式，從 EC2 執行個體傳回到本機電腦的主目錄，請在本機電腦上使用下列其中一個命令。
  - (公有 DNS) 若要使用執行個體的公有 DNS 名稱傳輸檔案，請輸入下列命令。

```
scp -i /path/key-pair-name.pem instance-user-
name@ec2-198-51-100-1.compute-1.amazonaws.com:~/SampleFile.txt ~/
SampleFile2.txt
```

- (IPv6) 或者，如果您的執行個體具有 IPv6 位址，若要使用該執行個體的 IPv6 位址，向另一個方向傳輸檔案，請輸入下列命令。



```
scp -i /path/key-pair-name.pem instance-user-name@
\[2001:db8:1234:1a00:9691:9503:25ad:1761\]:~/SampleFile.txt ~/SampleFile2.txt
```

## 解除安裝 WSL

如需有關解除安裝適用於 Linux 的 Windows 子系統的資訊，請參閱[如何解除安裝 WSL 發行版本？](#)。

## 使用 EC2 Instance Connect 連線至 Linux 執行個體

Amazon EC2 Instance Connect 提供使用 Secure Shell (SSH) 連線至 Linux 執行個體的簡單安全方式。使用 EC2 執行個體 Connect 時，您可以使用 AWS Identity and Access Management (IAM) [政策](#)和[主體](#)控制對執行個體的 SSH 存取，無需共用和管理安全殼層金鑰。使用 EC2 執行個體 Connect 的所有連接請求都會[記錄到](#)，以 [AWS CloudTrail](#) 使您可以稽核連接請求。

可以使用 Amazon EC2 主控台或您選擇的 SSH 用戶端，透過 EC2 Instance Connect 連線到執行個體。

當您使用 EC2 Instance Connect 連接至執行個體時，Instance Connect API 會將 SSH 公有金鑰推送至[執行個體中繼資料](#)，此金鑰可保留 60 秒。連接至使用者的 IAM 政策會授權使用者將公有金鑰推送至執行個體中繼資料。SSH 協助程式使用安裝 Instance Connect 時設定的 `AuthorizedKeysCommand` 和 `AuthorizedKeysCommandUser`，從執行個體中繼資料查詢身分驗證所用的公有金鑰並將您連接至執行個體。

您可以使用 EC2 Instance Connect 連線至具有公用或私有 IP 地址的執行個體。如需詳細資訊，請參閱 [使用 EC2 Instance Connect 連線](#)。

如需討論如何使用 EC2 Instance Connect 來改善堡壘主機安全性的部落格文章，請參閱[使用 Amazon EC2 Instance Connect 保護您的堡壘主機](#)。

### Tip

EC2 Instance Connect 是連線到 Linux 執行個體的其中一個選項。如需有關其他選項的詳細資訊，請參閱 [連接至您的 Linux 執行個體](#)。若要連線至 Windows 執行個體，請參閱[連線至 Windows 執行個體](#)。

## 目錄

- [教學課程：完成使用 EC2 執行個體 Connect 連線至執行個體所需的組態](#)

- [必要條件](#)
- [為 EC2 Instance Connect 授予 IAM 許可](#)
- [在 EC2 執行個體上安裝 EC2 Instance Connect](#)
- [使用 EC2 Instance Connect 連線](#)
- [解除安裝 EC2 Instance Connect](#)

教學課程：完成使用 EC2 執行個體 Connect 連線至執行個體所需的組態

若要使用 Amazon EC2 主控台內的 EC2 執行個體 Connect 連接到執行個體，您首先需要完成先決條件組態，以便成功連線到執行個體。本教學課程的目的是引導您完成工作，以完成必要條件組態。

## 教程概述

在本教學課程中，您將完成以下四項工作：

- [工作 1：建立並附加 IAM 政策以允許您使用 EC2 執行個體 Connect](#)

首先，您將建立包含 IAM 許可的 IAM 政策，以便將公開金鑰推送至執行個體中繼資料。您會將此政策附加到 IAM 身分 (使用者、使用者群組或角色)，以便 IAM 身分取得這些許可。

- [工作 2：建立安全群組，以允許從 EC2 執行個體 Connect 服務傳入流量到您的執行個體](#)

然後，您將建立一個安全群組，允許來自 EC2 執行個體 Connect 服務的流量到您的執行個體。當您在 Amazon EC2 主控台中使用 EC2 執行個體 Connect 來連接到執行個體時，這是必要的。

- [工作 3：啟動執行個體](#)

然後，您將使用預先安裝在 EC2 執行個體 Connect 的 AMI 啟動 EC2 執行個體，然後新增在上一步中建立的安全群組。

- [工作 4：Connect 至執行個體](#)

最後，您將使用 Amazon EC2 主控台內的 EC2 執行個體 Connect 來連線到您的執行個體。如果您可以連線，則可以確定您在工作 1、2 和 3 中完成的先決條件組態是否成功。

## 工作 1：建立並附加 IAM 政策以允許您使用 EC2 執行個體 Connect

當您使用 EC2 Instance Connect 連接至執行個體時，EC2 Instance Connect API 會將 SSH 公有金鑰推送至[執行個體中繼資料](#)，此金鑰可保留 60 秒。您需要將 IAM 政策附加到 IAM 身分 (使用者、使用者群組或角色)，才能授與將公開金鑰推送至執行個體中繼資料的所需權限。

## 任務目標

在此工作中，您將建立 IAM 政策，以授與將公開金鑰推送至執行個體的權限。要允許的特定動作為 `ec2-instance-connect:SendSSHPublicKey`。您還必須允許該 `ec2:DescribeInstances` 動作，以便在 Amazon EC2 主控台中檢視和選取執行個體。

建立政策後，您會將政策附加到 IAM 身分 (使用者、使用者群組或角色)，以便您的 IAM 身分取得許可。

您將建立如下設定的原則：

```
{
 "Version": "2012-10-17",
 "Statement": [{
 "Effect": "Allow",
 "Action": "ec2-instance-connect:SendSSHPublicKey",
 "Resource": "*"
 },
 {
 "Effect": "Allow",
 "Action": "ec2:DescribeInstances",
 "Resource": "*"
 }
]
```

### Important

本教學課程中建立的 IAM 政策是高度寬鬆的政策；它可讓您使用任何 AMI 使用者名稱連線至任何執行個體。我們正在使用這個高度寬鬆的政策來保持教程簡單，並專注於本教程所教學的特定配置。不過，在生產環境中，建議您將 IAM 政策設定為提供[最低權限](#)許可。如需 IAM 政策的範例，請參閱[為 EC2 Instance Connect 授予 IAM 許可](#)。

## 建立和附加 IAM 政策的步驟

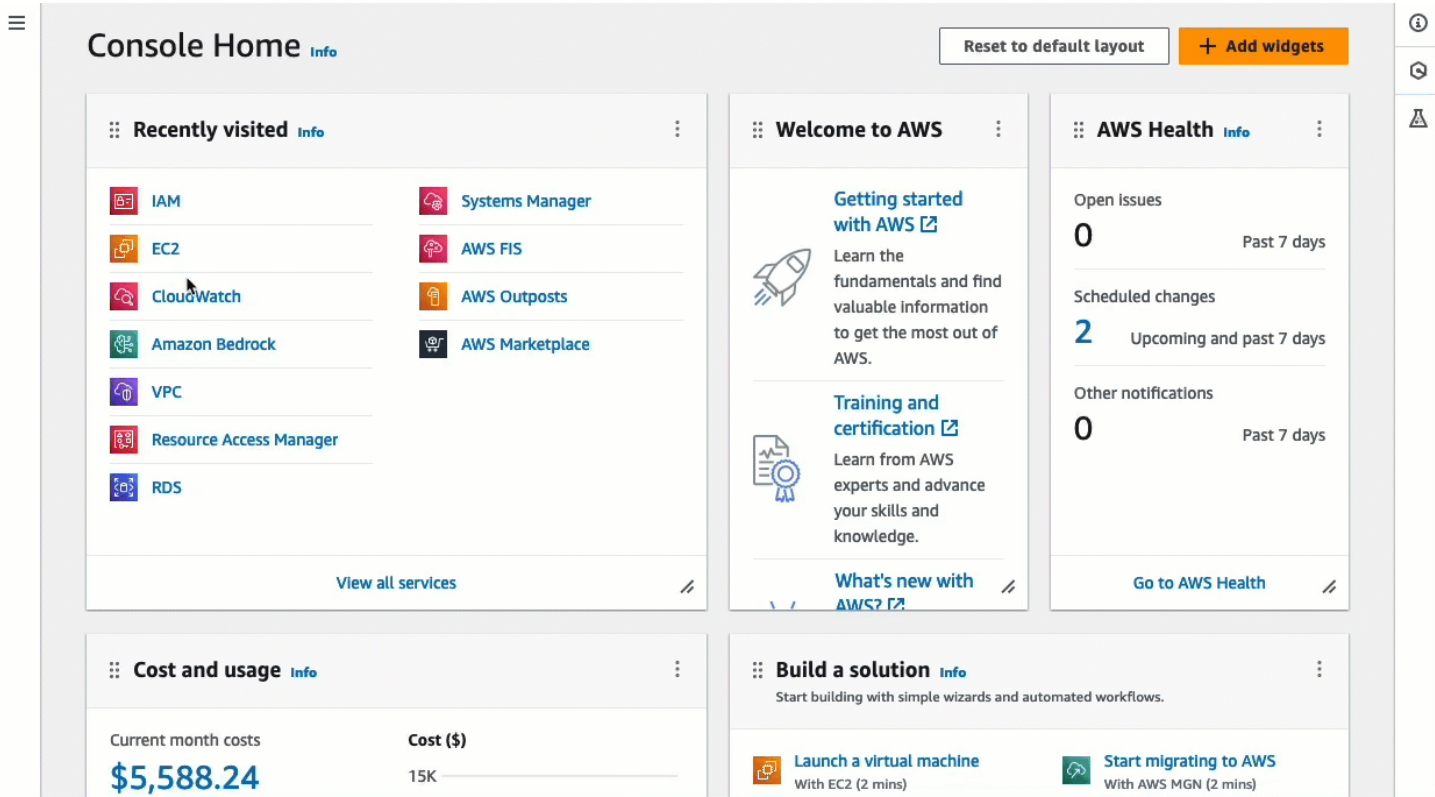
使用下列步驟建立並附加 IAM 政策。若要檢視步驟的動畫，請參閱[檢視動畫：建立 IAM 政策](#)和[檢視動畫：附加 IAM 政策](#)。

建立並附加可讓您使用 EC2 執行個體 Connect 連線到執行個體的 IAM 政策

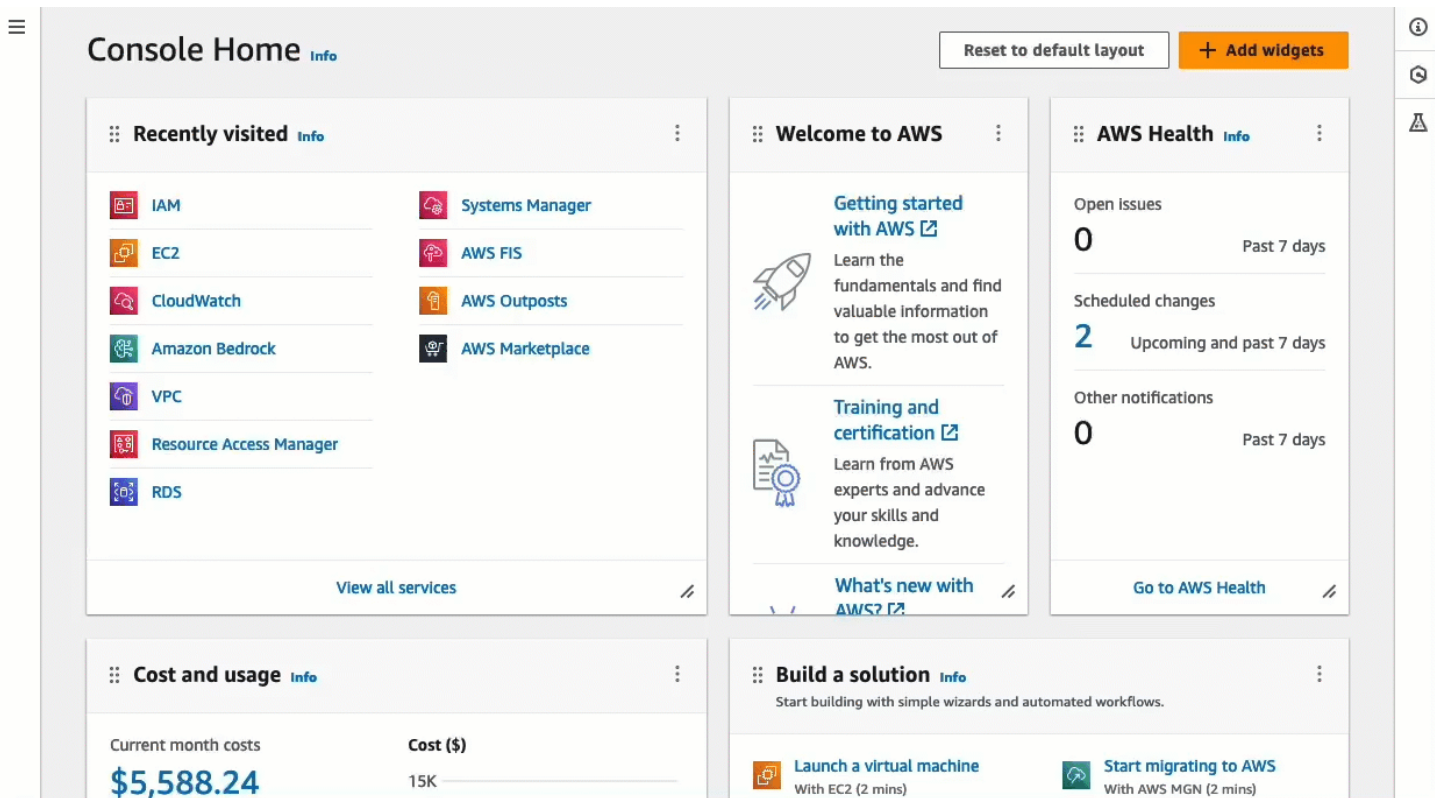
### 1. 首先建立 IAM 政策

- a. 前往 <https://console.aws.amazon.com/iam/> 開啟 IAM 主控台。
  - b. 在導覽窗格中，選擇政策。
  - c. 選擇 Create policy (建立政策)。
  - d. 在 [指定權限] 頁面上，執行下列動作：
    - i. 對於服務，請選擇 EC2 執行個體 Connect。
    - ii. 在 [允許的動作] 下，在搜尋欄位中開始輸入 **send** 以顯示相關動作，然後選取 [Send PublicKey SSH]。
    - iii. 在資源下，選擇全部。對於生產環境，我們建議您使用 ARN 指定執行個體，但在本教學課程中，您允許所有執行個體。
    - iv. 選擇 [新增更多權限]。
    - v. 在 Service (服務) 欄位中，選擇 EC2。
    - vi. 在「允許的動作」下，在搜尋欄位中開始輸入 **describein** 以顯示相關動作，然後選取 DescribeInstances。
    - vii. 選擇下一步。
  - e. 在「檢閱並建立」頁面上，執行下列動作：
    - i. 針對 Policy Name (政策名稱)，請輸入政策的名稱。
    - ii. 選擇建立政策。
2. 然後將政策附加到您的身份
- a. 在 IAM 主控台的導覽窗格中，選擇 Policies (政策)。
  - b. 在策略清單中，選取您建立之策略名稱旁邊的選項按鈕。您可以使用搜尋方塊來篩選政策清單。
  - c. 選擇 Actions (動作)、Attach (連接)。
  - d. 在 IAM 實體下，選取身分 (使用者、使用者群組或角色) 旁邊的核取方塊。您可以使用搜尋方塊來篩選實體清單。
  - e. 選擇連接政策。

### 檢視動畫：建立 IAM 政策



### 檢視動畫：附加 IAM 政策



## 工作 2：建立安全群組，以允許從 EC2 執行個體 Connect 服務傳入流量到您的執行個體

當您在 Amazon EC2 主控台中使用 EC2 執行個體 Connect 連線到執行個體時，必須允許到達執行個體的流量是來自 EC2 執行個體 Connect 服務的流量。這與從本機電腦連線到執行個體不同；在這種情況下，您必須允許從本機電腦傳輸到執行個體的流量。若要允許來自 EC2 執行個體 Connect 服務的流量，您必須建立一個安全群組，以允許來自 EC2 執行個體 Connect 服務 IP 地址範圍的輸入 SSH 流量。

這些 AWS 服務的 IP 位址範圍可在 <https://ip-ranges.amazonaws.com/ip-ranges.json> 取得。EC2 執行個體 Connect IP 位址範圍由識別 "service": "EC2\_INSTANCE\_CONNECT"。

### 任務目標

在這項工作中，首先您會找到執行個體所 EC2\_INSTANCE\_CONNECT AWS 區域 在的 IP 位址範圍。然後，您將建立一個安全群組，允許該 IP 位址範圍的連接埠 22 上的輸入 SSH 流量。

### 建立安全性群組的步驟

使用下列步驟來建立安全性群組。若要檢視步驟的動畫，請參閱 [檢視動畫：取得特定區域的 EC2 執行個體 Connect IP 位址範圍](#) 和 [檢視動畫：設定安全性群組](#)。

建立允許從 EC2 執行個體 Connect 服務傳入流量到執行個體的安全群組

#### 1. 首先獲取 EC2 實例 Connect 服務的 IP 地址範圍

- a. [在以下 AWS 位置開啟 IP 位址範圍 JSON 檔案。](https://ip-ranges.amazonaws.com/ip-ranges.json) <https://ip-ranges.amazonaws.com/ip-ranges.json>
- b. 選擇「原始資料」。
- c. 尋找執行個體 EC2\_INSTANCE\_CONNECT 所 AWS 區域 在的 IP 位址範圍。您可以使用瀏覽器搜尋欄位來搜尋服務 EC2\_INSTANCE\_CONNECT，並繼續搜尋，直到找到執行個體所在的區域為止。

例如，如果您的執行個體位於美國東部 (維吉尼亞北部 us-east-1) () 區域，則該區域的 IP EC2\_INSTANCE\_CONNECT 位址範圍為 18.206.107.24/29。

#### Note

每個 IP 位址範圍都不同 AWS 區域。

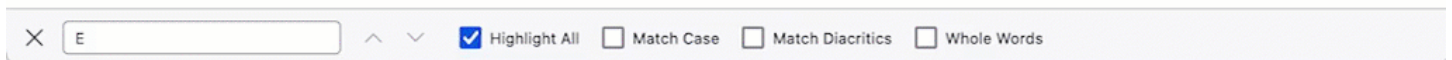
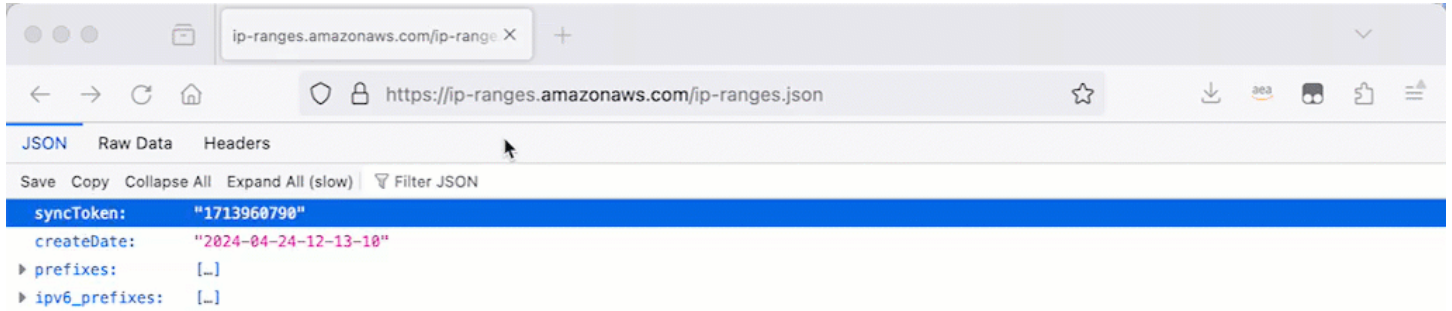
- d. 複製出現在旁邊的 IP 位址範圍 ip\_prefix。您將在稍後的程序中使用此 IP 位址範圍。

如需有關下載 AWS IP 位址範圍 JSON 檔案和依服務篩選的詳細資訊，請參閱 Amazon VPC 使用者指南中的 [AWS IP 位址範圍](#)。

2. 然後使用輸入規則建立安全群組，以允許來自複製 IP 位址範圍的流量
  - a. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
  - b. 在導覽窗格中，選擇 Security Groups (安全群組)。
  - c. 選擇 Create Security Group (建立安全群組)。
  - d. 在 Basic details (基本詳細資料) 下，執行下列動作：
    - i. 在「安全性群組名稱」中，輸入有意義的安全性群組名稱。
    - ii. 在說明中，為您的安全性群組輸入有意義的說明。
  - e. 在「輸入規則」下，執行下列動作：
    - i. 選擇新增規則。
    - ii. 針對 Type (類型)，選擇 SSH。
    - iii. 對於來源，請保留自訂。
    - iv. 在「來源」旁邊的欄位中，貼上您先前在此程序中複製的 EC2 執行個體 Connect 服務的 IP 位址範圍。

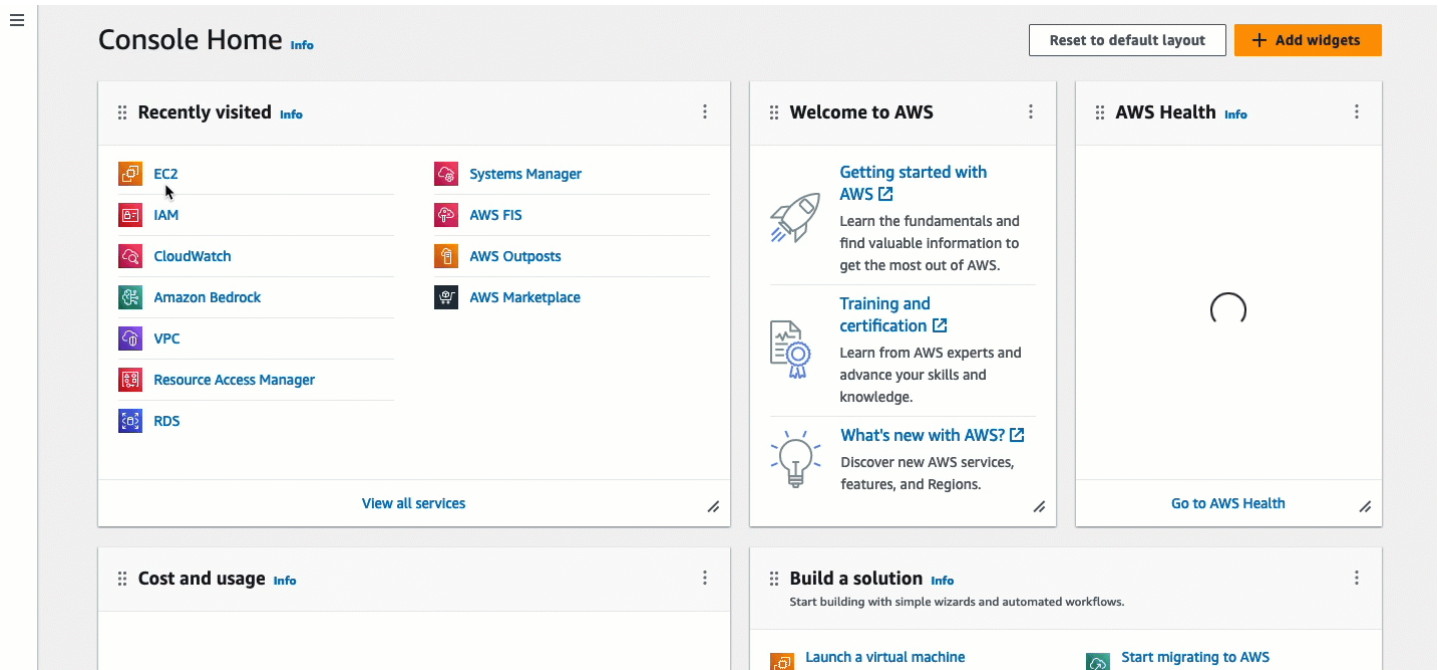
例如，如果您的執行個體位於美國東部 (維吉尼亞北部) (us-east-1) 區域，請將下列 IP 位址範圍貼到欄位中：18.206.107.24/29
  - f. 選擇建立安全群組。

## 檢視動畫：取得特定區域的 EC2 執行個體 Connect IP 位址範圍





## 檢視動畫：設定安全性群組



### 工作 3：啟動執行個體

啟動執行個體時，您必須指定包含啟動執行個體所需資訊的 AMI。您可以選擇啟動具有或不預先安裝 EC2 執行個體 Connect 的執行個體。在此工作中，我們會指定 EC2 執行個體 Connect 預先安裝的 AMI。

如果您在未預先安裝 EC2 執行個體 Connect 的情況下啟動執行個體，並且想要使用 EC2 執行個體 Connect 連接到執行個體，則需要執行其他設定步驟。這些步驟不在本教學課程的範圍內。

### 任務目標


在此任務中，您將使用 Amazon Linux 2023 AMI 啟動一個實例，該實例已預先安裝在 EC2 執行個體 Connect 中。您還將指定先前建立的安全群組，以便在 Amazon EC2 主控台中使用 EC2 執行個體 Connect 來連接到執行個體。由於您將使用 EC2 執行個體 Connect 連線至執行個體 (該執行個體會將公開金鑰推送至執行個體的中繼資料)，因此在啟動執行個體時不需要指定安全殼層金鑰。但是您必須確保執行個體具有公有 IPv4 地址，因為在 Amazon EC2 主控台中使用 EC2 執行個體 Connect 只支援連線到具有公有 IPv4 地址的執行個體。

### 啟動執行個體的步驟

請使用下列步驟啟動執行個體。若要檢視步驟的動畫，請參閱[檢視動畫：啟動您的執行個體](#)。

在 Amazon EC2 主控台中啟動可以使用 EC2 執行個體 Connect 進行連線的執行個體

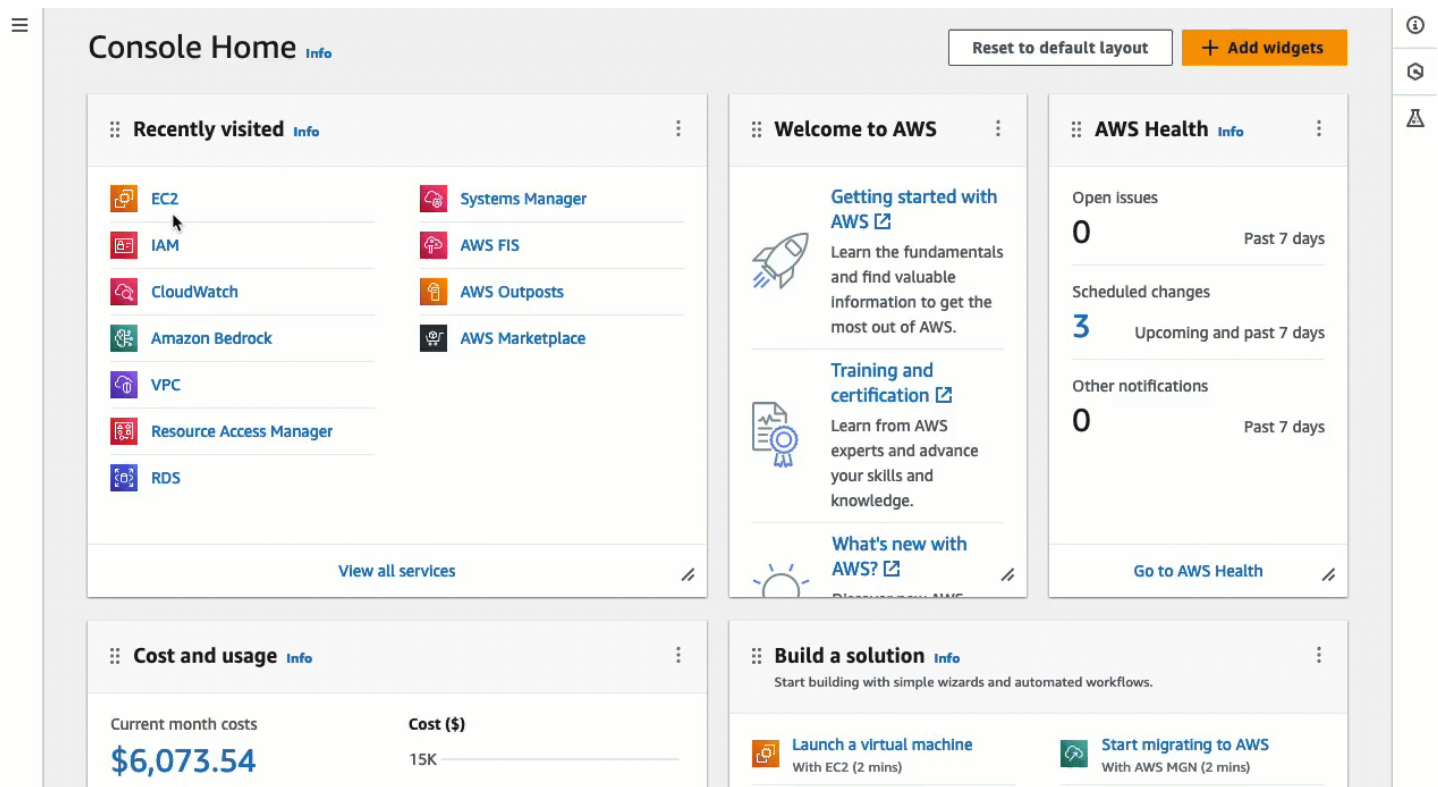
1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在畫面頂端的導覽列中，會顯示目前的「區域」(AWS Region) (例如，愛爾蘭)。選取要在其中啟動執行個體的區域。這個選擇很重要，因為您建立了允許特定區域流量的安全性群組，因此您必須選取要在其中啟動執行個體的相同區域。
3. 從 Amazon EC2 主控台儀表板選擇 Launch Instance (啟動執行個體)。
4. (選用) 在 Name and tags (名稱和標籤) 下，針對 Name (名稱)，輸入執行個體的描述性名稱。
5. 在應用程式和作業系統映像 (Amazon 機器映像) 下，選擇快速入門。依預設會選取 Amazon Linux。在 Amazon 機器映像 (AMI) 下，默認情況下選擇了 Amazon Linux 2023 AMI。保留此工作的預設選項。
6. 在「例證類型」下，對於「例證類型」，保留預設選取項，或選擇不同的例證類型。
7. 在 key pair (登入) 下，對於金鑰配對名稱，選擇不使用金鑰配對繼續 (不建議)。當您使用 EC2 執行個體 Connect 連線至執行個體時，EC2 執行個體 Connect 會將 key pair 推送至執行個體的中繼資料，而這個 key pair 用於連線。
8. 在 Network settings (網路設定) 中，執行下列操作：
  - a. 對於自動指派公用 IP，請離開 [啟用]。

 Note

若要在 Amazon EC2 主控台中使用 EC2 執行個體 Connect 來連接執行個體，該執行個體必須具有公有 IPv4 地址。

- b. 對於「防火牆 (安全群組)」，選擇「選取現有安全群組」。
  - c. 在 [一般安全性群組] 底下，選擇您先前建立的安全性群組。
9. 在 Summary (摘要) 面板中，選擇 Launch instance (啟動執行個體)。

## 檢視動畫：啟動您的執行個體



### 工作 4：Connect 至執行個體

當您使用 EC2 Instance Connect 連接至執行個體時，EC2 Instance Connect API 會將 SSH 公有金鑰推送至 [執行個體中繼資料](#)，此金鑰可保留 60 秒。SSH 精靈會使用 `AuthorizedKeysCommand` 和 `AuthorizedKeysCommandUser` 查詢執行個體中繼資料中的公開金鑰以進行驗證，並將您連線至執行個體。

#### 任務目標

在此任務中，您將使用 Amazon EC2 主控台中的 EC2 執行個體 Connect 連接到執行個體。如果您已完成先決條件工作 1、2 和 3，連線應該會成功。

#### 連線至執行個體的步驟

請使用下列步驟連線至您的執行個體。若要檢視步驟的動畫，請參閱 [檢視動畫：Connect 至您的執行個體](#)。

使用 Amazon EC2 主控台中的 EC2 執行個體 Connect 執行個體

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。

2. 在畫面頂端的導覽列中，會顯示目前的「區域」(AWS Region) (例如，愛爾蘭)。選取執行個體所在的區域。
3. 在導覽窗格中，選擇執行個體。
4. 選取執行個體並選擇 [Connect]。
5. 選擇 EC2 Instance Connect 索引標籤。
6. 針對 [連線類型]，選擇 [使用 EC2 執行個體 Connect 線連線]
7. 選擇連線。

終端機視窗會在瀏覽器中開啟，且您已連線至執行個體。

### 檢視動畫：Connect 至您的執行個體

The screenshot displays the AWS Management Console Home page. At the top, there's a navigation bar with 'Console Home' and buttons for 'Reset to default layout' and '+ Add widgets'. The main content area is divided into several sections:

- Recently visited:** A grid of service tiles including EC2, IAM, CloudWatch, Amazon Bedrock, VPC, Resource Access Manager, RDS, Systems Manager, AWS FIS, AWS Outposts, and AWS Marketplace. A mouse cursor is hovering over the EC2 tile.
- Welcome to AWS:** A section with three sub-sections: 'Getting started with AWS' (Learn the fundamentals and find valuable information to get the most out of AWS.), 'Training and certification' (Learn from AWS experts and advance your skills and knowledge.), and 'What's new with AWS?'.
- AWS Health:** A section showing 'Open issues' (0), 'Scheduled changes' (3), and 'Other notifications' (0), all for the 'Past 7 days' period.
- Cost and usage:** A section showing 'Current month costs' as \$6,073.54, which is a 3% decrease compared to the last month for the same period. It includes a bar chart for 'Cost (\$)' with markers at 10K and 15K.
- Build a solution:** A section titled 'Start building with simple wizards and automated workflows.' with four tiles: 'Launch a virtual machine' (With EC2 (2 mins)), 'Start migrating to AWS' (With AWS MGN (2 mins)), 'Register a domain', and 'Host a static web app'.

### 必要條件

以下是安裝 EC2 Instance Connect 以及使用 EC2 Instance Connect 連線到執行個體的先決條件：

- [AWS 區域](#)
- [本機區域](#)
- [AMI](#)

- [安裝 EC2 Instance Connect](#)
- [IPv4 地址](#)
- [網路存取](#)
- [安全群組規則](#)
- [授予許可](#)
- [本機電腦設定](#)
- [使用者名稱](#)

## AWS 區域

AWS 區域 除加拿大西部 ( 卡爾加里 ) 以外的所有支持。

## 本機區域

不支援。

## AMI

EC2 執行個體 Connect 已預先安裝在下列 AMI 上：

- AL2023
- Amazon Linux 2 2.0.20190618 或更新版本
- macOS 索諾瑪 14.2.1 或更高版本
- macOS 文圖拉 13.6.3 或更高版本
- macOS 蒙特雷 12.7.2 或更高版本
- Ubuntu 20.04 或更新版本

EC2 執行個體 Connect 並未預先安裝在下列 AMI 上，但您可以將其安裝在使用下列 AMI 啟動的執行個體上：

- Amazon Linux 2 在 2.0.20190618 之前的版本
- CentOS Stream 8 和 9
- macOS 索諾瑪之前 14.2.1, 文圖拉之前 13.6.3, 和蒙特雷之前 12.7.2
- Red Hat Enterprise Linux (RHEL) 8 和 9

- Ubuntu 16.04 或 18.04

## 安裝 EC2 Instance Connect

若要使用 EC2 Instance Connect 連線至執行個體，請務必安裝 EC2 Instance Connect。您可以使用預先安裝 EC2 執行個體 Connect 的 AMI 啟動執行個體，也可以在使用支援 AMI 啟動的執行個體上安裝 EC2 執行個體 Connect。如需支援的 AMI，請參閱先前部分。如需安裝指示，請參閱 [在 EC2 執行個體上安裝 EC2 Instance Connect](#)。

## IPv4 地址

您的執行個體必須具有 IPv4 地址 (私有或公有)。EC2 Instance Connect 不支援使用 IPv6 地址進行連線。

## 網路存取

執行個體可設定為允許使用者透過網際網路或執行個體的私有 IP 地址連線至您的執行個體。視您的使用者透過 EC2 Instance Connect 連線到執行個體的方式而定，您必須設定以下網路存取：

- 如果使用者將透過網際網路連線到執行個體，則該執行個體必須具有公有 IP 地址並位於公有子網路中。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的 [啟用網際網路存取權](#)。
- 如果您的使用者將透過執行個體的私有 IP 地址連線至您的執行個體，則您必須建立與 VPC 的私人網路連線，例如使用 AWS Direct Connect AWS Site-to-Site VPN、或 VPC 對等，以便使用者可以連線到執行個體的私有 IP 地址。

如果您的執行個體沒有公有 IPv4 地址，且您不想按上述說明設定網路存取權，則可將 EC2 Instance Connect Endpoint 視為 EC2 Instance Connect 的替代方案。EC2 Instance Connect Endpoint 允許您透過 SSH 或 RDP 連線至執行個體，而不需要執行個體擁有公有 IPv4 地址。如需詳細資訊，請參閱 [使用 Amazon EC2 主控台連線至 Linux 執行個體](#)。

## 安全群組規則

請確認與執行個體關聯的安全群組在連接埠 22 上 [允許來自您 IP 地址或網路的傳入 SSH 流量](#)。VPC 的預設安全群組根據預設並不允許傳入 SSH 流量。啟動執行個體精靈建立的安全群組預設會允許傳入 SSH 流量。如需詳細資訊，請參閱 [從您電腦連線到執行個體的規則](#)。

EC2 Instance Connect 使用特定的 IP 地址範圍進行執行個體的基於瀏覽器的 SSH 連線 (當使用者使用 Amazon EC2 主控台連線至執行個體時)。如果使用者將使用 Amazon EC2 主控台連線到執行個體，請確認與執行個體關聯的安全群組允許從 EC2\_INSTANCE\_CONNECT 的 IP 地址範圍傳入 SSH 流量。

若要識別位址範圍，請下載提供的 JSON 檔案，AWS 並使用 EC2\_INSTANCE\_CONNECT 服務值篩選 EC2 執行個體 Connect 的子集。這些 IP 位址範圍之間有所不同 AWS 區域。如需有關下載 JSON 檔案和依服務進行篩選的詳細資訊，請參閱《Amazon VPC 使用者指南》中的 [AWS IP 地址範圍](#)。

## 授予許可

您必須將所需許可授予給每個將使用 EC2 Instance Connect 連線到執行個體的 IAM 使用者。如需詳細資訊，請參閱 [為 EC2 Instance Connect 授予 IAM 許可](#)。

## 本機電腦設定

如果您的使用者將使用 SSH 進行連線，則必須確保其本機電腦具有 SSH 用戶端。

使用者的本機電腦很可能預設已安裝 SSH 用戶端。他們可以在命令列鍵入 ssh 來檢查 SSH 用戶端。若本機電腦無法識別該命令，可以安裝 SSH 用戶端。如需在 Linux 或 macOS X 上安裝 SSH 用戶端的相關資訊，請參閱 <http://www.openssh.com>。如需在 Windows 10 上安裝 SSH 用戶端的相關資訊，請參閱 [Windows 中的 OpenSSH](#)。

如果使用者僅使用 Amazon EC2 主控台連線至執行個體，就無需在本機電腦安裝 SSH 用戶端。

## 使用者名稱

使用 EC2 Instance Connect 連線至執行個體時，使用者名稱必須符合下列先決條件：

- 第一個字元：必須是字母 (A-Z, a-z)、數字 (0-9) 或底線 (\_)
- 後續字元：可以是字母 (A-Z、a-z)、數字 (0-9) 或 @ . \_ - 這些字元
- 長度下限：1 個字元
- 長度上限：31 個字元

## 為 EC2 Instance Connect 授予 IAM 許可

若要使用 EC2 Instance Connect 連線至執行個體，請務必建立 IAM 政策來授予使用者執行下列動作和條件的許可：

- `ec2-instance-connect:SendSSHPublicKey` 動作 – 准許將公有金鑰推送至執行個體。
- `ec2:osuser` 條件 – 指定可將公有金鑰推送至執行個體的 OS 使用者名稱。使用您用來啟動執行個體之 AMI 的預設使用者名稱。AL2023 和 Amazon Linux 2 的默認用戶名是 `ec2-user`，對於 Ubuntu 而言，它是 `ubuntu`。
- `ec2:DescribeInstances` 動作 – 使用 EC2 主控台時必需，因為包裝函數會呼叫此動作。使用者可能已有從另一個政策呼叫此動作的許可。

考慮限制對特定 EC2 執行個體的存取。否則，具有 `ec2-instance-connect:SendSSHPublicKey` 動作許可的所有 IAM 主體都可以連線到所有 EC2 執行個體。您可以指定資源 ARN 或將資源標籤做為 [條件索引鍵](#) 使用，來限制存取。

如需詳細資訊，請參閱 [適用於 Amazon EC2 Instance Connect 的動作、資源和條件金鑰](#)。

如需有關建立和編輯 IAM 政策的資訊，請參閱《IAM 使用者指南》中的 [建立 IAM 政策](#)。

允許使用者連線至特定執行個體

下列 IAM 政策會授予連線至特定執行個體的許可，由其資源 ARN 識別。

在下列 IAM 政策範例中，指定了下列動作和條件：

- 此 `ec2-instance-connect:SendSSHPublicKey` 動作准許使用者連線至兩個執行個體，由資源 ARN 指定。若要准許使用者連線至所有 EC2 執行個體，請以 \* 萬用字元取代資源 ARN。
- 只有在連線時指定 `ami-username`，`ec2:osuser` 條件才會准許連線至執行個體。
- 指定 `ec2:DescribeInstances` 動作，准許使用主控台的使用者連線至執行個體。如果使用者僅使用 SSH 用戶端連線至執行個體，則可以省略 `ec2:DescribeInstances`。注意，`ec2:Describe*` API 動作不支援資源層級許可。因此，在 `Resource` 元素中必須包含 \* 萬用字元。

```
{
 "Version": "2012-10-17",
 "Statement": [{
 "Effect": "Allow",
 "Action": "ec2-instance-connect:SendSSHPublicKey",
 "Resource": [
 "arn:aws:ec2:region:account-id:instance/i-1234567890abcdef0",
 "arn:aws:ec2:region:account-id:instance/i-0598c7d356eba48d7"
],
 "Condition": {
 "StringEquals": {
 "ec2:osuser": "ami-username"
 }
 }
 }],
 {
 "Effect": "Allow",
 "Action": "ec2:DescribeInstances",
 "Resource": "*"
 }
}
```



```

}
]
}

```

## 允許使用者連線至具有特定標籤的執行個體

以屬性為基礎的存取控制 (ABAC) 是一種授權策略，根據可附加至使用者和資源的標籤來定義權限。AWS 您可使用資源標籤來控制對執行個體的存取。如需有關使用標籤來控制 AWS 資源存取權的詳細資訊，請參閱 IAM 使用者指南中的[控制 AWS 資源存取](#)。

在下列 IAM 政策範例中，`ec2-instance-connect:SendSSHPublicKey` 動作准許使用者連線至任何執行個體 (由資源 ARN 中的 \* 萬用字元表示)，條件是執行個體的資源標記中 `key=tag-key` 和 `value=tag-value`。

指定 `ec2:DescribeInstances` 動作，准許使用主控台的使用者連線至執行個體。如果使用者僅使用 SSH 用戶端連線至執行個體，則可以省略 `ec2:DescribeInstances`。注意，`ec2:Describe*` API 動作不支援資源層級許可。因此，在 `Resource` 元素中必須包含 \* 萬用字元。

```

{
 "Version": "2012-10-17",
 "Statement": [{
 "Effect": "Allow",
 "Action": "ec2-instance-connect:SendSSHPublicKey",
 "Resource": "arn:aws:ec2:region:account-id:instance/*",
 "Condition": {
 "StringEquals": {
 "aws:ResourceTag/tag-key": "tag-value"
 }
 }
 }],
 {
 "Effect": "Allow",
 "Action": "ec2:DescribeInstances",
 "Resource": "*"
 }
]
}

```

## 在 EC2 執行個體上安裝 EC2 Instance Connect

若要使用 EC2 Instance Connect 連線至執行個體，則該執行個體必須安裝 EC2 Instance Connect。

下列 AMI 已預先安裝 EC2 執行個體 Connect：

- AL2023 標準型 AMI
- Amazon Linux 2 2.0.20190618 或更新版本
- macOS 索諾瑪 14.2.1 或更高版本
- macOS 文圖拉 13.6.3 或更高版本
- macOS 蒙特雷 12.7.2 或更高版本
- Ubuntu 20.04 或更新版本

如果使用前面清單中的 AMI 之一啟動執行個體，則可以略過此程序。

#### Note

如果您對 SSH 身分驗證設定 `AuthorizedKeysCommand` 和 `AuthorizedKeysCommandUser`，則 EC2 Instance Connect 安裝不會進行更新。因此，您無法使用 EC2 Instance Connect。

安裝 EC2 Instance Connect 的先決條件

- 使用下列其中一個支援的 AMI 啟動執行個體：

Amazon Linux 2 在 2.0.20190618 之前的版本

AL2023 最小 AMI 或 Amazon ECS 優化 AMI

CentOS Stream 8 和 9

macOS 索諾瑪之前 14.2.1, 文圖拉之前 13.6.3, 和蒙特雷之前 12.7.2

Red Hat Enterprise Linux (RHEL) 8 和 9

Ubuntu 16.04 和 18.04

如果您的執行個體是使用較新版本的 Amazon Linux 2、macOS 索諾瑪、文圖拉或蒙特雷或 Ubuntu 啟動的，則預先安裝了 EC2 執行個體 Connect，您可以跳過此程序。

- 驗證 EC2 Instance Connect 的一般先決條件。

如需詳細資訊，請參閱 [必要條件](#)。

- 確認在本機電腦上使用 SSH 用戶端連線至執行個體的先決條件。

如果本機電腦是 Linux 或 macOS，請參閱 [使用 SSH 從 Linux 或 macOS 連線至 Linux 執行個體](#)。  
如果本機電腦是 Windows，請參閱 [必要條件](#)。

如需詳細資訊，請參閱 [SSH 連線先決條件](#)。

- 取得執行個體的 ID。

您可以使用 Amazon EC2 主控台取得執行個體的 ID (從執行個體 ID 資料欄中)。如果您願意，您可以使用 [描述實例 \(AWS CLI\)](#) 或 [Get-EC2Instance\(AWS Tools for Windows PowerShell\)](#) 命令。

- 在本機電腦安裝 SSH 用戶端。

根據預設，您的本機電腦很可能已安裝 SSH 用戶端。您可以藉由在命令列鍵入 ssh 來檢查 SSH 用戶端。若您的本機電腦無法識別該命令，您可以安裝 SSH 用戶端。如需在 Linux 或 macOS X 上安裝 SSH 用戶端的相關資訊，請參閱 <http://www.openssh.com>。如需在 Windows 10 上安裝 SSH 用戶端的相關資訊，請參閱 [Windows 中的 OpenSSH](#)。

- (Ubuntu) AWS CLI 在您的執行個體上安裝。

若要在 Ubuntu 執行個體上安裝 EC2 執行個體 Connect，您必須在執行個體 AWS CLI 上使用。若要取得有關安裝的更多資訊 AWS CLI，請參閱《AWS Command Line Interface 使用指南》AWS CLI 中的 [〈安裝〉](#)。

## 安裝 EC2 Instance Connect

安裝 EC2 Instance Connect 會在執行個體上設定 SSH 常駐程式。

根據執行個體的作業系統，使用以下其中一種程序來安裝 EC2 Instance Connect。

### Amazon Linux 2

若要在使用 Amazon Linux 2 啟動的執行個體上安裝 EC2 Instance Connect

1. 使用 SSH 連接至您的執行個體。

將以下命令中的範例值取代為您自己的值。使用啟動執行個體時指派給執行個體的 SSH key pair，以及用來啟動執行個體之 AMI 的預設使用者名稱。對於 Amazon Linux 2，默認的用戶名是 ec2-user。

```
$ ssh -i my_ec2_private_key.pem ec2-user@ec2-a-b-c-d.us-west-2.compute.amazonaws.com
```

如需連線至執行個體的詳細資訊，請參閱[使用 SSH 從 Linux 或 macOS 連線至 Linux 執行個體](#)。

2. 在執行個體上安裝 EC2 Instance Connect 套件。

```
[ec2-user ~]$ sudo yum install ec2-instance-connect
```

您應會在 `/opt/aws/bin/` 資料夾中看到三個新指令碼：

```
eic_curl_authorized_keys
eic_parse_authorized_keys
eic_run_authorized_keys
```

3. (選用) 確認 EC2 Instance Connect 是否已順利安裝在您的執行個體上。

```
[ec2-user ~]$ sudo less /etc/ssh/sshd_config
```

如果 `AuthorizedKeysCommand` 和 `AuthorizedKeysCommandUser` 行包含以下值，則 EC2 Instance Connect 的安裝已順利完成：

```
AuthorizedKeysCommand /opt/aws/bin/eic_run_authorized_keys %u %f
AuthorizedKeysCommandUser ec2-instance-connect
```

- `AuthorizedKeysCommand` 會設定 `eic_run_authorized_keys` 指令碼來從執行個體中繼資料查閱金鑰
- `AuthorizedKeysCommandUser` 會將系統使用者設為 `ec2-instance-connect`

#### Note

如果您先前已設定 `AuthorizedKeysCommand` 和 `AuthorizedKeysCommandUser`，EC2 Instance Connect 安裝不會變更這些值且您將無法使用 EC2 Instance Connect。

## CentOS

若要在使用 CentOS 啟動的執行個體上安裝 EC2 Instance Connect

1. 使用 SSH 連接至您的執行個體。

將以下命令中的範例值取代為您自己的值。使用啟動執行個體時指派給執行個體的 SSH key pair，以及用來啟動執行個體之 AMI 的預設使用者名稱。對於 CentOS 而言，預設的使用者名稱為 `centos` 或 `ec2-user`。

```
$ ssh -i my_ec2_private_key.pem centos@ec2-a-b-c-d.us-west-2.compute.amazonaws.com
```

如需連線至執行個體的詳細資訊，請參閱[使用 SSH 從 Linux 或 macOS 連線至 Linux 執行個體](#)。

2. 如果您使用 HTTP 或 HTTPS 代理伺服器，則必須在目前的 Shell 工作階段中設定 `http_proxy` 或 `https_proxy` 環境變數。

如果您不使用代理伺服器，則可以略過此步驟。

- 若使用 HTTP 代理伺服器，請執行下列命令：

```
$ export http_proxy=http://hostname:port
$ export https_proxy=http://hostname:port
```

- 若使用 HTTPS 代理伺服器，請執行下列命令：

```
$ export http_proxy=https://hostname:port
$ export https_proxy=https://hostname:port
```

3. 執行下列命令，在執行個體上安裝 EC2 Instance Connect 套件。

適用於 CentOS 的 EC2 執行個體 Connect 組態檔案是以 Red Hat Package Manager (RPM) 套件提供，搭配以下項目適用的不同 RPM 套件：CentOS 8 和 CentOS 9，以及在 Intel /AMD (x86\_64) 或 ARM (AArch64) 上執行的執行個體類型。

為您的作業系統和 CPU 架構使用命令區塊。

- CentOS 8

### Intel/AMD (x86\_64)

```
[ec2-user ~]$ mkdir /tmp/ec2-instance-connect
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect.rhel8.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect.rpm
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect-selinux.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

## ARM (AArch64)

```
[ec2-user ~]$ mkdir /tmp/ec2-instance-connect
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-west-2.amazonaws.com/latest/linux_arm64/ec2-instance-connect.rhel8.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect.rpm
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect-selinux.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

- CentOS 9

## Intel/AMD (x86\_64)

```
[ec2-user ~]$ mkdir /tmp/ec2-instance-connect
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect.rhel8.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect.rpm
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect-selinux.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

## ARM (AArch64)

```
[ec2-user ~]$ mkdir /tmp/ec2-instance-connect
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_arm64/ec2-instance-connect.rpm -o /tmp/ec2-
instance-connect/ec2-instance-connect.rpm
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect-
selinux.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-
selinux.rpm
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-
connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

您應會在 `/opt/aws/bin/` 資料夾中看到以下新指令碼：

```
eic_run_authorized_keys
```

4. (選用) 確認 EC2 Instance Connect 是否已順利安裝在您的執行個體上。

- CentOS 8 :

```
[ec2-user ~]$ sudo less /lib/systemd/system/ssh.service.d/ec2-instance-
connect.conf
```

- CentOS 9 :

```
[ec2-user ~]$ sudo less /etc/ssh/sshd_config.d/60-ec2-instance-connect.conf
```

如果 `AuthorizedKeysCommand` 和 `AuthorizedKeysCommandUser` 行包含以下值，則 EC2 Instance Connect 的安裝已順利完成：

```
AuthorizedKeysCommand /opt/aws/bin/eic_run_authorized_keys %u %f
AuthorizedKeysCommandUser ec2-instance-connect
```

- `AuthorizedKeysCommand` 會設定 `eic_run_authorized_keys` 指令碼來從執行個體中繼資料查閱金鑰
- `AuthorizedKeysCommandUser` 會將系統使用者設為 `ec2-instance-connect`

**Note**

如果您先前已設定 `AuthorizedKeysCommand` 和 `AuthorizedKeysCommandUser`，EC2 Instance Connect 安裝不會變更這些值且您將無法使用 EC2 Instance Connect。

## macOS

若要在使用 macOS 啟動的執行個體上安裝 EC2 Instance Connect

1. 使用 SSH 連接至您的執行個體。

將以下命令中的範例值取代為您自己的值。使用啟動執行個體時指派給執行個體的 SSH key pair，以及用來啟動執行個體之 AMI 的預設使用者名稱。對於 macOS 執行個體，預設使用者名稱為 `ec2-user`。

```
$ ssh -i my_ec2_private_key.pem ec2-user@ec2-a-b-c-d.us-west-2.compute.amazonaws.com
```

如需連線至執行個體的詳細資訊，請參閱[使用 SSH 從 Linux 或 macOS 連線至 Linux 執行個體](#)。

2. 使用以下命令更新 Homebrew。更新將列出 Homebrew 已知的軟體。EC2 執行個體 Connect 套件是透過 macOS 執行個體上的自製軟體提供。如需詳細資訊，請參閱[更新 Mac 實例上的操作系統和軟體](#)。

```
[ec2-user ~]$ brew update
```

3. 在執行個體上安裝 EC2 Instance Connect 套件。這會安裝軟體，並設定 `sshd` 以使用該軟體。

```
[ec2-user ~]$ brew install ec2-instance-connect
```

您應會在 `/opt/aws/bin/` 資料夾中看到以下新指令碼：

```
eic_run_authorized_keys
```

4. (選用) 確認 EC2 Instance Connect 是否已順利安裝在您的執行個體上。



```
[ec2-user ~]$ sudo less /etc/ssh/sshd_config.d/60-ec2-instance-connect.conf
```

如果 `AuthorizedKeysCommand` 和 `AuthorizedKeysCommandUser` 行包含以下值，則 EC2 Instance Connect 的安裝已順利完成：

```
AuthorizedKeysCommand /opt/aws/bin/eic_run_authorized_keys %u %f
AuthorizedKeysCommandUser ec2-instance-connect
```

- `AuthorizedKeysCommand` 會設定 `eic_run_authorized_keys` 指令碼來從執行個體中繼資料查閱金鑰
- `AuthorizedKeysCommandUser` 會將系統使用者設為 `ec2-instance-connect`

#### Note

如果您先前已設定 `AuthorizedKeysCommand` 和 `AuthorizedKeysCommandUser`，EC2 Instance Connect 安裝不會變更這些值且您將無法使用 EC2 Instance Connect。

## RHEL

若要在使用 Red Hat Enterprise Linux (RHEL) 啟動的執行個體上安裝 EC2 Instance Connect

1. 使用 SSH 連接至您的執行個體。

將以下命令中的範例值取代為您自己的值。使用啟動執行個體時指派給執行個體的 SSH key pair，以及用來啟動執行個體之 AMI 的預設使用者名稱。對於 RHEL，預設使用者名稱為 `ec2-user` 或 `root`。

```
$ ssh -i my_ec2_private_key.pem ec2-user@ec2-a-b-c-d.us-west-2.compute.amazonaws.com
```

如需連線至執行個體的詳細資訊，請參閱 [使用 SSH 從 Linux 或 macOS 連線至 Linux 執行個體](#)。

2. 如果您使用 HTTP 或 HTTPS 代理伺服器，則必須在目前的 Shell 工作階段中設定 `http_proxy` 或 `https_proxy` 環境變數。

如果您不使用代理伺服器，則可以略過此步驟。

- 若使用 HTTP 代理伺服器，請執行下列命令：

```
$ export http_proxy=http://hostname:port
$ export https_proxy=http://hostname:port
```

- 若使用 HTTPS 代理伺服器，請執行下列命令：

```
$ export http_proxy=https://hostname:port
$ export https_proxy=https://hostname:port
```

3. 執行下列命令，在執行個體上安裝 EC2 Instance Connect 套件。

適用於 RHEL 的 EC2 執行個體 Connect 組態檔案是以 Red Hat Package Manager (RPM) 套件提供，搭配以下項目適用的不同 RPM 套件：RHEL 8 和 RHEL 9，以及在 Intel /AMD (x86\_64) 或 ARM (AArch64) 上執行的執行個體類型。

為您的作業系統和 CPU 架構使用命令區塊。

- RHEL 8

Intel/AMD (x86\_64)

```
[ec2-user ~]$ mkdir /tmp/ec2-instance-connect
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect.rhel8.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect.rpm
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect-selinux.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

ARM (AArch64)

```
[ec2-user ~]$ mkdir /tmp/ec2-instance-connect
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-west-2.amazonaws.com/latest/linux_arm64/ec2-instance-connect.rhel8.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect.rpm
```

```
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect-selinux.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

- RHEL 9

#### Intel/AMD (x86\_64)

```
[ec2-user ~]$ mkdir /tmp/ec2-instance-connect
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect.rpm
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect-selinux.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

#### ARM (AArch64)

```
[ec2-user ~]$ mkdir /tmp/ec2-instance-connect
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-west-2.amazonaws.com/latest/linux_arm64/ec2-instance-connect.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect.rpm
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect-selinux.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

您應會在 `/opt/aws/bin/` 資料夾中看到以下新指令碼：

```
eic_run_authorized_keys
```

4. (選用) 確認 EC2 Instance Connect 是否已順利安裝在您的執行個體上。

- 針對 RHEL 8 :

```
[ec2-user ~]$ sudo less /lib/systemd/system/ssh.service.d/ec2-instance-connect.conf
```

- 針對 RHEL 9 :

```
[ec2-user ~]$ sudo less /etc/ssh/sshd_config.d/60-ec2-instance-connect.conf
```

如果 `AuthorizedKeysCommand` 和 `AuthorizedKeysCommandUser` 行包含以下值，則 EC2 Instance Connect 的安裝已順利完成：

```
AuthorizedKeysCommand /opt/aws/bin/eic_run_authorized_keys %u %f
AuthorizedKeysCommandUser ec2-instance-connect
```

- `AuthorizedKeysCommand` 會設定 `eic_run_authorized_keys` 指令碼來從執行個體中繼資料查閱金鑰
- `AuthorizedKeysCommandUser` 會將系統使用者設為 `ec2-instance-connect`

#### Note

如果您先前已設定 `AuthorizedKeysCommand` 和 `AuthorizedKeysCommandUser`，EC2 Instance Connect 安裝不會變更這些值且您將無法使用 EC2 Instance Connect。

## Ubuntu

若要在使用 Ubuntu 16.04 或更新版本啟動的執行個體上安裝 EC2 Instance Connect

1. 使用 SSH 連接至您的執行個體。

將以下命令中的範例值取代為您自己的值。使用啟動執行個體時指派給執行個體的 SSH key pair，並使用您用來啟動執行個體之 AMI 的預設使用者名稱。對於 Ubuntu AMI，使用者名稱為 `ubuntu`。

```
$ ssh -i my_ec2_private_key.pem ubuntu@ec2-a-b-c-d.us-west-2.compute.amazonaws.com
```

如需連線至執行個體的詳細資訊，請參閱[使用 SSH 從 Linux 或 macOS 連線至 Linux 執行個體](#)。

- (選用) 請確認您的執行個體具有最新的 Ubuntu AMI。

執行下列命令以更新您的執行個體上的所有套件。

```
ubuntu:~$ sudo apt-get update
```

```
ubuntu:~$ sudo apt-get upgrade
```

- 在執行個體上安裝 EC2 Instance Connect 套件。

```
ubuntu:~$ sudo apt-get install ec2-instance-connect
```

您應會在 `/usr/share/ec2-instance-connect/` 資料夾中看到三個新指令碼：

```
eic_curl_authorized_keys
eic_parse_authorized_keys
eic_run_authorized_keys
```

- (選用) 確認 Instance Connect 是否已順利安裝在您的執行個體上。

```
ubuntu:~$ sudo less /lib/systemd/system/ssh.service.d/ec2-instance-connect.conf
```

如果 `AuthorizedKeysCommand` 和 `AuthorizedKeysCommandUser` 行包含以下值，則 EC2 Instance Connect 的安裝已順利完成：

```
AuthorizedKeysCommand /usr/share/ec2-instance-connect/eic_run_authorized_keys %
%u %f
AuthorizedKeysCommandUser ec2-instance-connect
```

- `AuthorizedKeysCommand` 會設定 `eic_run_authorized_keys` 指令碼來從執行個體中繼資料查閱金鑰
- `AuthorizedKeysCommandUser` 會將系統使用者設為 `ec2-instance-connect`

**Note**

如果您先前已設定 `AuthorizedKeysCommand` 和 `AuthorizedKeysCommandUser`，EC2 Instance Connect 安裝不會變更這些值且您將無法使用 EC2 Instance Connect。

如需 EC2 執行個體 Connect 套件的詳細資訊，請參閱網站上的 [aws/aws-ec2-instance-connect-config](https://aws.amazon.com/ec2/instance-connect-config/)。GitHub

## 使用 EC2 Instance Connect 連線

下列指示會說明如何使用 EC2 Instance Connect 連線至 Linux 執行個體。

決定要使用的連線選項。使用的連線選項取決於執行個體是否具有公有 IPv4 地址：

- Amazon EC2 主控台 – 若要使用 Amazon EC2 主控台進行連線，執行個體必須設有公有 IPv4 地址。
- SSH 用戶端 – 如果執行個體沒有公有 IP 地址，您可以使用 SSH 用戶端透過私有網路連線到執行個體。例如，您可以從相同 VPC 內，或透過 VPN 連接、Transit Gateway 或 AWS Direct Connect 來連線。

EC2 Instance Connect 不支援使用 IPv6 地址進行連線。

**Tip**

EC2 Instance Connect 是連線到 Linux 執行個體的其中一個選項。如需有關其他選項的詳細資訊，請參閱 [連接至您的 Linux 執行個體](#)。若要連線至 Windows 執行個體，請參閱 [連線至 Windows 執行個體](#)。

## EC2 執行個體連線的連線選項

- [使用 Connect Amazon EC2 主控台進行連線](#)
- [使用您自己的金鑰和 SSH 用戶端來進行連接](#)
- [使用 Connect AWS CLI](#)
- [疑難排解](#)

## 使用 Connect Amazon EC2 主控台進行連線

您可以從主控台選取執行個體並選取使用 EC2 Instance Connect 來進行連線，以便使用 Amazon EC2 主控台連線至執行個體。Instance Connect 會處理許可並提供成功的連線。

若要使用 Amazon EC2 主控台進行連線，執行個體必須設有公用 IPv4 地址。連接之前，請務必檢閱所有[先決條件](#)。

從 Amazon EC2 主控台使用以瀏覽器為基礎的用戶端連線至執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選取 Connect (連線)。
4. 選擇 EC2 Instance Connect 索引標籤。
5. 針對 [連線類型]，選擇 [使用 EC2 執行個體 Connect 線連線]
6. 對於使用者名稱，驗證使用者名稱
7. 選擇連線以開啟終端視窗。

使用您自己的金鑰和 SSH 用戶端來進行連接

當您使用 EC2 Instance Connect API 時，您可以使用自己的 SSH 金鑰和從您選擇的 SSH 用戶端連接執行個體。這可讓您從 Instance Connect 發佈公有的金鑰至執行個體。此連線方法適用於具有公有和私有 IP 地址的執行個體。

要求

- 金鑰對的需求
  - 支援的類型：RSA (OpenSSH 和 SSH2) 和 ED25519
  - 支援的長度：2048 和 4096
  - 如需詳細資訊，請參閱 [使用第三方工具來建立金鑰對，然後將公有金鑰匯入 Amazon EC2](#)。
- 連線到僅具有私有 IP 地址的執行個體時，從中起始 SSH 工作階段的本機電腦必須具有連線到 EC2 Instance Connect 服務端點的能力 (以將 SSH 公有金鑰推送至執行個體)，以及透過網路連線到執行個體私有 IP 地址的能力，以建立 SSH 工作階段。EC2 Instance Connect 服務端點可以透過網際網路或 AWS Direct Connect 公有虛擬介面連線。若要連線到執行個體的私有 IP 地址，您可以使用服務，例如 [AWS Direct Connect](#)、[AWS Site-to-Site VPN](#) 或 [VPC 互連](#)。

連接之前，請務必檢閱所有[先決條件](#)。

使用您自己的金鑰和任何 SSH 用戶端來連接執行個體

### 1. (選用) 產生新 SSH 私有和公有金鑰

您可以使用以下命令來產生新 SSH 私有和公有金鑰 (my\_key 和 my\_key.pub) :

```
ssh-keygen -t rsa -f my_key
```

### 2. 將 SSH 公有金鑰推送至執行個體

使用 [send-ssh-public-key](#) 命令將 SSH 公開金鑰推送至執行個體。如果您使用 AL2023 或 Amazon Linux 2 啟動執行個體，則 AMI 的預設使用者名稱為 ec2-user。如果您使用 Ubuntu 啟動執行個體，AMI 的預設使用者名稱為 ubuntu。

下列範例將公開金鑰推送至特定可用區域的特定執行個體，若要認證 ec2-user。

```
aws ec2-instance-connect send-ssh-public-key \
 --region us-west-2 \
 --availability-zone us-west-2b \
 --instance-id i-001234a4bf70dec41EXAMPLE \
 --instance-os-user ec2-user \
 --ssh-public-key file://my_key.pub
```

### 3. 使用私有金鑰連線至執行個體

使用 ssh 命令，來使用私有金鑰連接至執行個體，之後系統會將公有金鑰從執行個體中繼資料中移除 (在系統移除金鑰前您有 60 秒的時間)。指定與公開金鑰對應的私密金鑰、用來啟動執行個體之 AMI 的預設使用者名稱，以及執行個體的公用 DNS 名稱 (如果透過私人網路連線，請指定私人 DNS 名稱或 IP 位址)。新增該 IdentitiesOnly=yes 選項以確保連線只使用 ssh 組態中的檔案和指定的金鑰。

```
ssh -o "IdentitiesOnly=yes" -i my_key ec2-
user@ec2-198-51-100-1.compute-1.amazonaws.com
```

## 使用 Connect AWS CLI

如果您知道執行個體 ID，就可以使用 [ec2 執行個體 connect AWS CLI 指令](#)，透過安全殼層用戶端連線至執行個體。如果您未指定連線類型，則 EC2 Instance Connect 會自動嘗試連線至執行個體的公有 IPv4 地址。如果您的執行個體沒有公有 IPv4 地址，則 EC2 Instance Connect 會嘗試透過 [EC2 Instance Connect Endpoint](#) 連接到執行個體的私有 IPv4 地址。如果執行個體沒有公有 IPv4 地址，或



者 VPC 沒有 EC2 Instance Connect Endpoint，則 EC2 Instance Connect 會嘗試連線到執行個體的 IPv6 地址。

### Important

使用此方法連線之前，請確定您已設定 AWS CLI，包括其使用的認證，並且您使用的是最新版本的 AWS CLI。如需詳細資訊，請參閱《AWS Command Line Interface 使用者指南》中的[安裝或更新 AWS CLI 的最新版本](#)和[設定 AWS CLI](#)。

## 連線類型

### auto (default)

CLI 會嘗試依下列順序和對應的連線類型，使用執行個體的 IP 地址進行連線：

- 公有 IPv4 : direct
- 私有 IPv4 : eice
- IPv6 : direct

### direct

CLI 會嘗試按照以下順序使用執行個體的 IP 地址進行連線 (不會透過 EC2 Instance Connect Endpoint 進行連線)：

- 公有 IPv4
- IPv6
- 私有 IPv4

### eice

CLI 一律使用執行個體的私有 IPv4 地址。

### Note

將來，我們可能會變更 auto 連線類型的行為。若要確保使用您想要的連線類型，建議您明確地將 `--connection-type` 設定為 `direct` 或 `eice`。

當您使用 EC2 Instance Connect 連接至執行個體時，EC2 Instance Connect API 會將 SSH 公有金鑰推送至[執行個體中繼資料](#)，此金鑰可保留 60 秒。連接至使用者的 IAM 政策會授權使用者將公有金鑰推送至執行個體中繼資料。

### 使用執行個體 ID 連線到執行個體

如果您只知道執行個體 ID，並希望讓 EC2 執行個體 Connect 線確定連線到執行個體時要使用的連線類型，請使用 [ec2 執行個體連線](#) CLI 指令並指定 `ssh` 參數和執行個體 ID。

```
aws ec2-instance-connect ssh --instance-id i-1234567890example
```

#### Tip

如果使用此命令時出現錯誤，請確定您使用的是 AWS CLI 版本 2。該 `ssh` 參數僅在 AWS CLI 版本 2 中可用。如需詳細資訊，請參閱《使用指南》中的 AWS Command Line Interface [〈關於 AWS CLI 版本 2〉](#)。

### 使用執行個體 ID 和 EC2 Instance Connect Endpoint 連線至執行個體

如果要透過 [EC2 Instance Connect Endpoint](#) 連線到執行個體，則請使用上述命令並指定具有 `eice` 值的 `--connection-type` 參數。

```
aws ec2-instance-connect ssh --instance-id i-1234567890example --connection-type eice
```

### 使用執行個體 ID 和您自己的私有金鑰檔案連線至執行個體

如果想透過 EC2 Instance Connect Endpoint 使用自己的私有金鑰連線到執行個體，則請指定執行個體 ID 和私有金鑰檔案的路徑。請勿在路徑中包含 `file://`；下列範例會無效：`file:///path/to/key`。

```
aws ec2-instance-connect ssh --instance-id i-1234567890example --private-key-file /
path/to/key.pem
```

### 疑難排解

如果您在嘗試連線至執行個體時遇到錯誤，請參閱下文：

- [連線到 Linux 執行個體的疑難排解](#)
- [如何對使用 EC2 Instance Connect 連線至 EC2 執行個體的問題進行疑難排解？](#)

## 解除安裝 EC2 Instance Connect

若要停用 EC2 Instance Connect，連接至您的執行個體並解除安裝在 OS 上的 `ec2-instance-connect` 套件。如果 `sshd` 組態與您安裝在 EC2 Instance Connect 設定的內容相符，解除安裝 `ec2-instance-connect` 也會移除 `sshd` 組態。如果您在安裝 EC2 Instance Connect 之後修改 `sshd` 組態，您必須手動更新。

### Amazon Linux

您可以在 AL2023 和 Amazon Linux 2 2.0.20190618 或更新版本中 (EC2 Instance Connect 已預先設定) 解除安裝 EC2 Instance Connect。

若要在使用 Amazon Linux 2 啟動的執行個體上解除安裝 EC2 Instance Connect

1. 使用 SSH 連接至您的執行個體。指定啟動執行個體時用於執行個體的安全殼層 key pair，以及 AL2023 或 Amazon Linux 2 AMI 的預設使用者名稱，也就是說 `ec2-user`。

例如，以下 `ssh` 命令會使用 `my_ec2_private_key.pem` 金鑰對連接到公有 DNS 名稱為 `ec2-a-b-c-d.us-west-2.compute.amazonaws.com` 的執行個體。

```
$ ssh -i my_ec2_private_key.pem ec2-user@ec2-a-b-c-d.us-west-2.compute.amazonaws.com
```

2. 使用 `yum` 指令，解除安裝 `ec2-instance-connect` 套件。

```
[ec2-user ~]$ sudo yum remove ec2-instance-connect
```

### Ubuntu

若要在使用 Ubuntu AMI 啟動的執行個體上解除安裝 EC2 Instance Connect

1. 使用 SSH 連接至您的執行個體。指定啟動執行個體時用於執行個體的安全殼層 key pair，以及 Ubuntu AMI 的預設使用者名稱 `ubuntu`。

例如，以下 `ssh` 命令會使用 `my_ec2_private_key.pem` 金鑰對連接到公有 DNS 名稱為 `ec2-a-b-c-d.us-west-2.compute.amazonaws.com` 的執行個體。

```
$ ssh -i my_ec2_private_key.pem ubuntu@ec2-a-b-c-d.us-west-2.compute.amazonaws.com
```

2. 使用 `apt-get` 指令，解除安裝 `ec2-instance-connect` 套件。

```
ubuntu:~$ sudo apt-get remove ec2-instance-connect
```

## 連線至 Windows 執行個體

您可以使用遠端桌面，連接至從多數 Windows Amazon Machine Image (AMI) 建立的 Amazon EC2 執行個體。遠端桌面使用[遠端桌面協定 \(RDP\)](#)，可讓您以一般使用電腦的方式連接和使用執行個體，就像您使用坐在您面前的電腦 (本機電腦) 一樣。其適用於多數的 Windows 版本，也適用於 Mac OS。

Windows Server 作業系統的授權允許為了管理用途而同時有兩個遠端連線。Windows Server 的授權包含在您的 Windows 執行個體價格中。如果您同時需要超過兩個的遠端連線，就必須購買 Remote Desktop Services (RDS) 授權。如果您嘗試第三個連線，就會出現錯誤。

### Tip

如果您需要連線到執行個體，以便對 [AWS Nitro 系統](#) 上建置的執行個體，進行開機、網路設定和其他問題的疑難排解，您可以使用 [亞馬遜 EC2 執行個體的 Amazon EC2 序列主控台](#)。

## 目錄

- [使用 RDP 用戶端 Connect 至您的 Windows 執行個體](#)
- [使用 Fleet Manager 連線到 Windows 執行個體](#)
- [設定帳戶](#)
- [傳輸檔案至 Windows 執行個體](#)

## 使用 RDP 用戶端 Connect 至您的 Windows 執行個體

下節詳細說明使用執行個體的 IPv4 或 IPv6 位址搭配 RDP 用戶端連線至執行個體的必要條件和程序。

### 必要條件

您必須符合下列先決條件，才能使用 RDP 用戶端連線到 Windows 執行個體。

- 安裝 RDP 用戶端
  - (視窗) 視窗預設包含 RDP 用戶端。若要確認，請在命令提示視窗下輸入 `mstsc`。如果電腦無法辨識此命令，請參閱 [Windows 首頁](#) 並搜尋下載 Microsoft 遠端桌面應用程式。

- ( macOS X ) 從 Mac 應用程序商店下載 [Microsoft 遠程桌面](#) 應用程序。
- (Linux) 使用 [雷米娜](#)。
- 找出私有金鑰

取得當您啟動執行個體時所指定的金鑰對的 .pem 檔案在您電腦中的位置的完整路徑。如需詳細資訊，請參閱 [the section called “識別啟動時指定的公有金鑰”](#)。

如果找不到私密金鑰檔案，請參閱

---

[當您連線到新啟動的 Windows 執行個體時，使用您在啟動執行個體時所指定金鑰對中的私有金鑰，將管理員帳戶的密碼解密。](#)

---

[如果您遺失了管理員密碼，而且也不再擁有私有金鑰，則必須重設密碼或建立新的執行個體。如需詳細資訊，請參閱 \[重設遺失或過期的 Windows 管理員密碼\]\(#\)。如需使用 Systems Manager 重設密碼之步驟的文件，請參閱《AWS Systems Manager 使用者指南》中的 \[在 EC2 執行個體上重設密碼和 SSH 金鑰\]\(#\)。](#)

---

- 啟用從 IP 地址到執行個體的傳入 RDP 流量

請確認與執行個體關聯的安全群組允許來自您 IP 地址的傳入 RDP 流量 (port 3389)。預設的安全群組根據預設並不允許傳入 RDP 流量。如需詳細資訊，請參閱 [從您電腦連線到執行個體的規則](#)。

#### Tip

您可以建立 [EC2 執行個體 Connect 端點](#)，以便在沒有公有 IPv4 地址的情況下使用 RDP 連接到您的 Windows 執行個體。

## 使用 RDP 及其 IPv4 位址 Connect 線至 Windows 執行個體

若要連線至 Windows 執行個體，您必須擷取初始管理員密碼，並在使用遠端桌面連線至執行個體時使用此密碼。在執行個體啟動後，需要幾分鐘的時間處理才能提供密碼。

系統管理員帳戶的預設使用者名稱取決於 AMI 中包含的作業系統 (OS) 的語言。要確定正確的使用者名稱，請確定 AMI 操作系統的語言，然後選擇相應的使用者名稱。例如，對於英語操作系統，使用者名稱是 Administrator，對於法國操作系統來說 Administrateur，對於葡萄牙語操作系統而言，它是 Administrador。如果作業系統的語言版本沒有相同語言的使用者名稱，請選擇使用者名稱 Administrator (Other)。如需詳細資訊，請參閱 Microsoft TechNet Wiki [中的 Windows 系統管理員帳戶的本地化名稱](#)。

如果您已將執行個體加入網域，則可以使用已在 AWS Directory Service 中定義的網域憑證連線至您的執行個體。在遠端桌面登入畫面上，不要使用本機電腦名稱和產生的密碼，而是使用系統管理員的合格使用者名稱 (例如 `corp.example.com\Admin`) 和此帳戶的密碼。

如果您在嘗試連線至執行個體時遇到錯誤，請參閱 [the section called “遠端桌面無法連線到遠端電腦”](#)。

使用 RDP 用戶端連線至您的 Windows 執行個體。

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體，然後選取 Connect (連線)。
4. 在 [Connect 至執行個體] 頁面上，選擇 [RDP 用戶端] 索引標籤。
5. 在使用者名稱中，選擇管理員帳戶的預設使用者名稱。您選擇的使用者名稱必須與您用來啟動執行個體的 AMI 中所包含的作業系統 (OS) 語言相符。如果沒有與您的作業系統相同語言的使用者名稱，請選擇「管理員 (其他)」。
6. 選擇 [取得密碼]。
7. 在 [取得 Windows 密碼] 頁面上，執行下列動作：
  - a. 選擇「上傳私密金鑰檔案」，然後瀏覽至您在啟動執行個體時指定的私密金鑰 (.pem) 檔案。選取檔案並選取 Open (開啟)，將檔案的完整內容複製至此視窗。
  - b. 選擇「解密密碼」。[取得 Windows 密碼] 頁面會關閉，且執行個體的預設管理員密碼會顯示在 [密碼] 底下，取代先前顯示的 [取得密碼] 連結。
  - c. 複製密碼並將其保存在安全的地方。必須使用此密碼來連接至執行個體。
8. 選擇 Download Remote Desktop File (下載遠端桌面檔案)。當您完成下載檔案時，請選擇 Cancel (取消) 以返回 Instances (執行個體) 頁面。導覽至您的下載目錄，然後開啟 RDP 檔案。
9. 您可能會收到警告提示遠端連線的發佈者為未知。選擇 Connect (連接) 以繼續連接到您的執行個體。
10. 預設選擇系統管理員帳戶。貼上先前複製的密碼，然後選擇 [確定]。
11. 由於自我簽署憑證的性質，您可能會收到安全憑證無法驗證的警告。執行以下任意一項：
  - 如果您信任憑證，請選擇 [是] 連線至您的執行個體。
  - [Windows] 在繼續之前，請將證書的指紋與系統日誌中的值進行比較，以確認遠端電腦的身份。選擇檢視憑證，然後從詳細資料索引標籤中選擇指紋。將此值與 [動作]、[監視及疑難排解]、[取得系統記錄] RDPCERTIFICATE-THUMBPRINT 中的值進行比較。

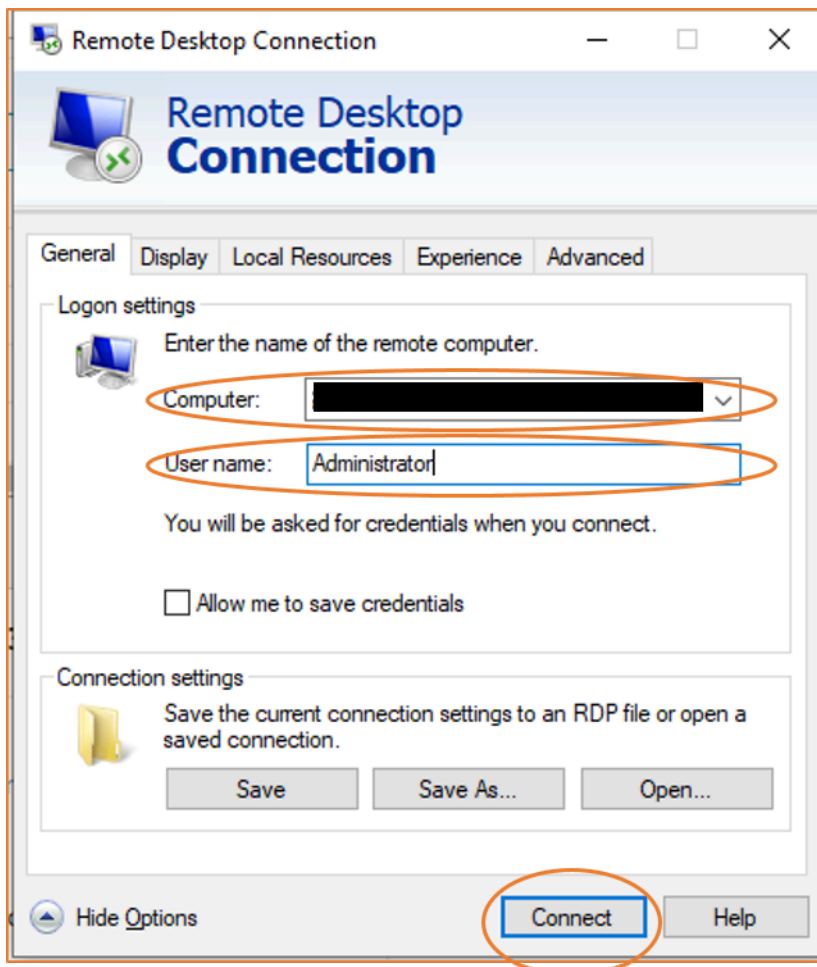
- [Mac OS X] 繼續之前，請先將憑證的指紋與系統記錄檔中的值進行比較，以確認遠端電腦的身分識別。選擇顯示憑證，展開詳細資料，然後選擇 SHA1 指紋。將此值與 [動作]、[監視及疑難排解]、[取得系統記錄] RDPCERTIFICATE-THUMBPRINT 中的值進行比較。

## 使用 RDP 及其 IPv6 位址 Connect 至視窗執行個體

如果您已啟用支援 IPv6 的 VPC 並指派 IPv6 地址給 Windows 執行個體，則您可使用 RDP 用戶端透過 IPv6 地址 (例如 2001:db8:1234:1a00:9691:9503:25ad:1761) 連接至執行個體，不必使用公有 IPv4 地址或公有 DNS 主機名稱。

## 使用 IPv6 地址連接至 Windows 執行個體

1. 取得執行個體的初始管理員密碼，如 [使用 RDP 用戶端 Connect 至您的 Windows 執行個體](#) 中所述。必須使用此密碼來連接至執行個體。
2. (Windows) 在 Windows 電腦上開啟 RDP 用戶端，選擇「顯示選項」，然後執行下列動作：



- 在「Computer (電腦)」中，輸入 Windows 執行個體的 IPv6 位址。

- 在 User name (使用者名稱) 中，輸入 Administrator (管理員)。
- 選擇 Connect (連線)。
- 出現提示時，請輸入您先前儲存的密碼。

(macOS X) 在您的電腦上開啟 RDP 用戶端，然後執行下列動作：

- 選擇 New (新增)。
  - 在「PC Name (電腦名稱)」中輸入 Windows 執行個體的 IPv6 地址。
  - 在 User name (使用者名稱) 中，輸入 Administrator (管理員)。
  - 關閉對話方塊。在「My Desktops (我的桌面)」下，選取連接，然後選取「Start (啟動)」。
  - 出現提示時，請輸入您先前儲存的密碼。
3. 由於自我簽署憑證的性質，您可能會收到安全憑證無法驗證的警告。如果您信任憑證，您可以選擇「Yes (是)」或「Continue (繼續)」。否則，您可以確認遠端電腦的身分識別，如 [使用 RDP 用戶端 Connect 至您的 Windows 執行個體](#) 中所述。

## 使用 Fleet Manager 連線到 Windows 執行個體

您可以使用叢集管理員 (一種功能) AWS Systems Manager，使用遠端桌面通訊協定 (RDP) 連線到 Windows 執行個體，並在中的相同頁面上顯示最多四個 Windows 執行個體 AWS Management Console。您可以從 Amazon EC2 主控台的執行個體頁面中直接連線至 Fleet Manager 遠端桌面的第一個執行個體。如需有關 Fleet Manager 的詳細資訊，請參閱《AWS Systems Manager 使用者指南》中的 [使用遠端桌面連線到受管節點](#)。

嘗試使用 Fleet Manager 連接到執行個體之前，請確保必要的設定步驟已經完成。如需詳細資訊，請參閱 [設定 Fleet Manager](#)。

### Note

如果您使用 Fleet Manager 進行連線，則不需要特別允許來自您 IP 地址的傳入 RDP 流量。Fleet Manager 會為您處理。

若要使用 Fleet Manager (主控台) 透過 RDP 連線至執行個體

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 從導覽窗格中選擇 Instances (執行個體)。



3. 選取執行個體，然後選取 Connect (連線)。
4. 在 Connect to instance (連線至執行個體) 頁面上，選擇選項 Connect using Fleet Manager (使用 Fleet Manager 進行連線)，然後選擇 Fleet Manager Remote Desktop (Fleet Manager 遠端桌面)。這將開啟 AWS Systems Manager 主控台中的 Fleet Manager Remote Desktop (Fleet Manager 遠端桌面) 頁面。

**Connect to instance** [Info](#)

Connect to your instance i-  (periscope\_test\_instance) using any of these options

**Session Manager** | **RDP client** | EC2 serial console

---

Instance ID  
i-  (periscope\_test\_instance)

Connection Type

Connect using RDP client  
Download a file to use with your RDP client and retrieve your password.

**Connect using Fleet Manager**  
Connect to your instance using Fleet Manager Remote Desktop.

When prompted, connect to your instance using the following details:

User name  
Administrator

Password [Get password](#)

**Fleet Manager Remote Desktop** [↗](#)

**i** If you've joined your instance to a directory, you can use your directory credentials to connect to your instance.

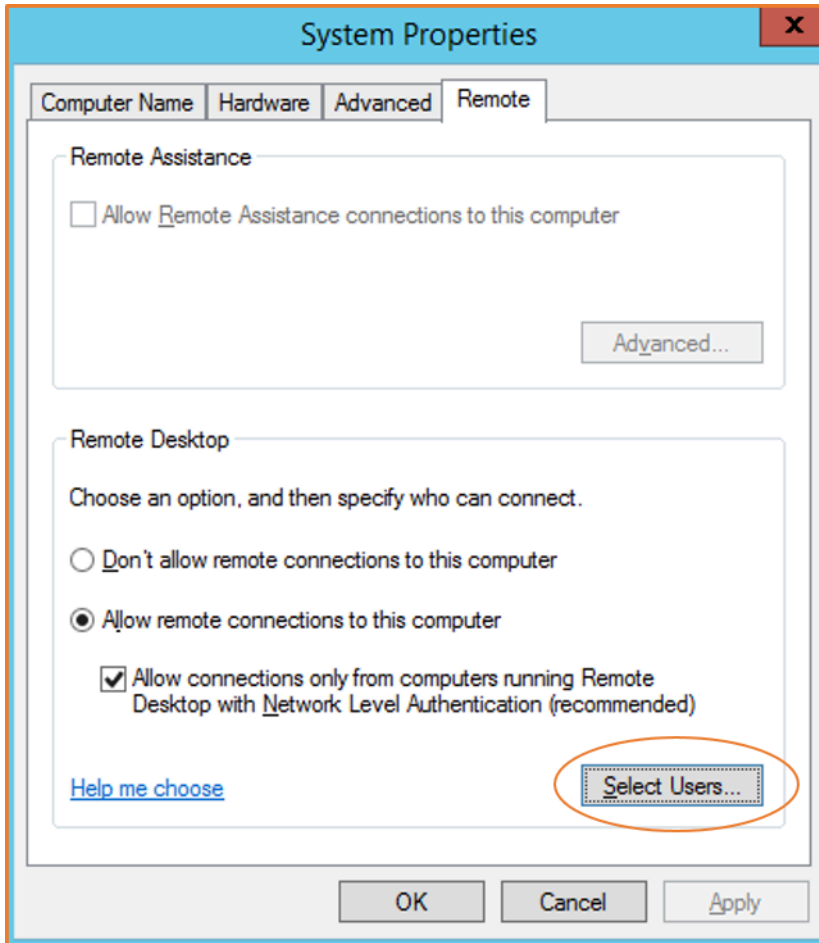
Cancel

如需有關從 Fleet Manager 遠端桌面頁面連線至 Windows 執行個體的詳細資訊，請參閱《AWS Systems Manager 使用者指南》中的[使用遠端桌面進行連線](#)。

## 設定帳戶

透過 RDP 連線之後，我們建議您執行下列動作：

- 變更預設的管理員密碼。您可以在登入執行個體後變更密碼，就像使用任何其他執行 Windows 伺服器的電腦一樣。
- 在執行個體上建立其他擁有管理員權限的使用者。如果您忘記管理員密碼或管理員帳戶發生問題，這可作為保護措施。新使用者必須具備可遠端存取執行個體的許可。在 Windows 桌面或檔案總管上，以滑鼠右鍵按一下 This PC (本機)，選取內容，以開啟系統內容。選取遠端設定，然後選取選取使用者，將使用者新增至遠端桌面使用者群組。



## 傳輸檔案至 Windows 執行個體

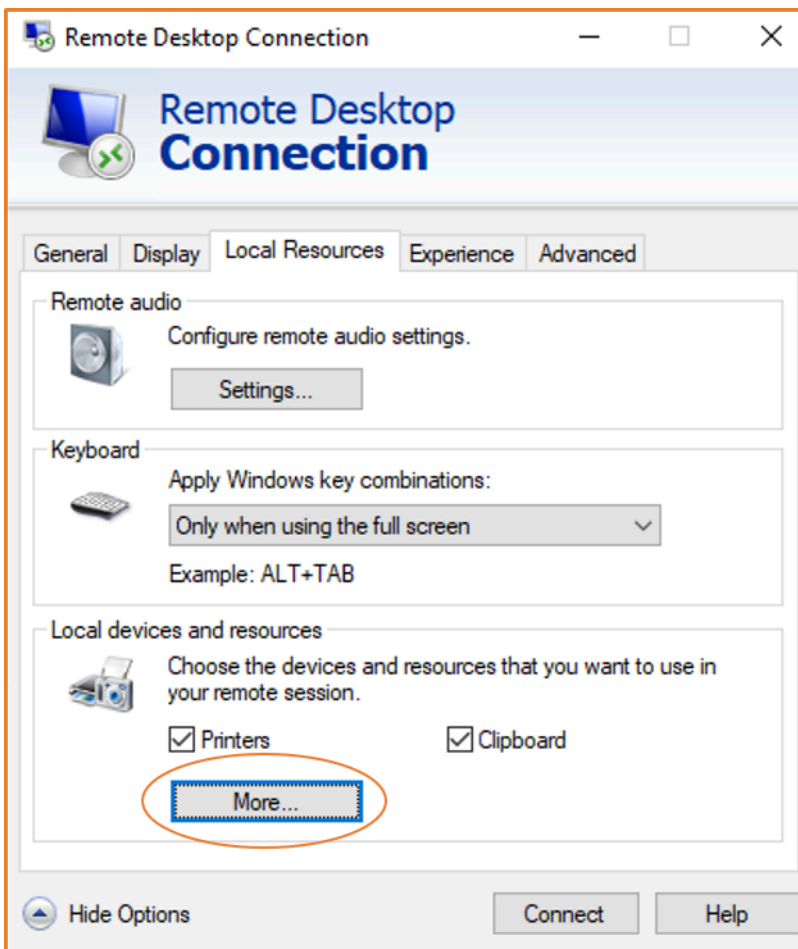
您可用與任何 Windows 伺服器相同的使用方式來使用 Windows 執行個體。例如，您可以使用 Microsoft 遠端桌面連線 (RDP) 軟體的本機檔案共用功能，在 Windows 執行個體和本機電腦之間傳輸檔案。本機檔案可以位在硬碟、DVD 光碟、可攜式媒體磁碟機和映射的網路磁碟機。

如要從 Windows 執行個體存取本機檔案，您必須透過將遠端工作階段磁碟機映射至您本機磁碟機來啟用本機檔案共享功能。這些步驟略有不同，取決於您的本機電腦作業系統是 Windows 還是 macOS X。

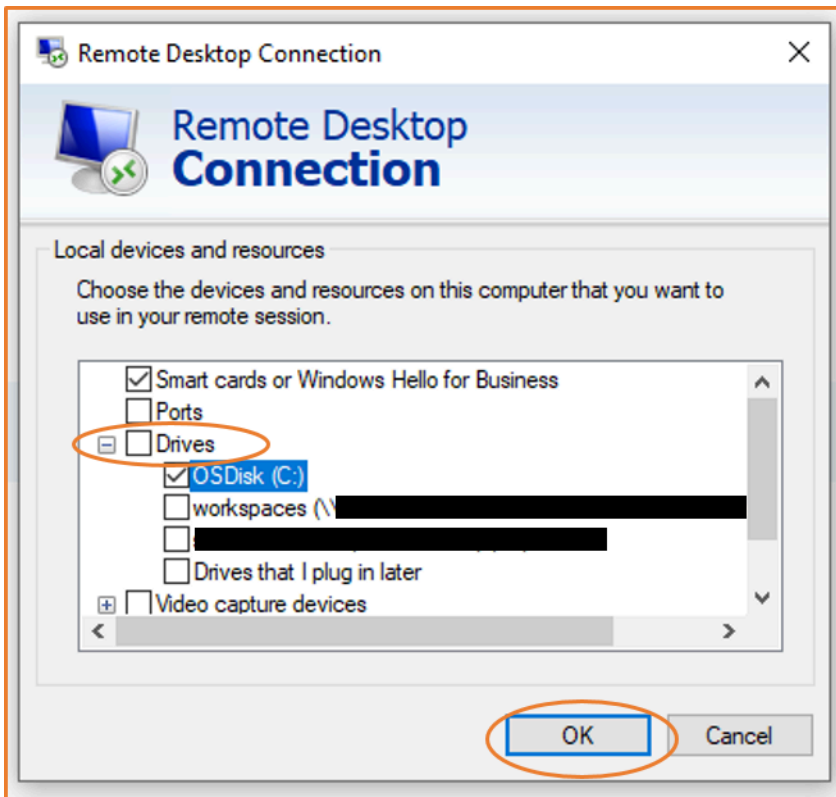
## Windows

如要將遠端工作階段磁碟機映射至您本機 Windows 電腦的本機磁碟機

1. 開啟遠端桌面連線用戶端。
2. 選擇 Show Options (顯示選項)。
3. 將執行個體主機名稱新增至 Computer (電腦) 欄位，及使用者名稱至 User name (使用者名稱) 欄位，如下所示：
  - a. 在 Connection settings (連線設定) 下，選擇 Open... (開啟...)，然後瀏覽您從 Amazon EC2 主控台下載的 RDP 捷徑檔案。該檔案包含標識執行個體的公有 IPv4 DNS 主機名稱和管理員使用者名稱。
  - b. 選取檔案，然後選取 Open (開啟)。Computer (電腦) 和 User name (使用者名稱) 欄位使用 RDP 捷徑檔案的值進行填入。
  - c. 選擇儲存。
4. 選擇 Local Resources (本機資源) 標籤。
5. 在 Local Devices and resources (本機裝置和資源) 下，選擇 More... (更多...)



6. 開啟 Drives (磁碟機) 並選取本機磁碟機，以映射到 Windows 執行個體。
7. 選擇 OK (確定)。

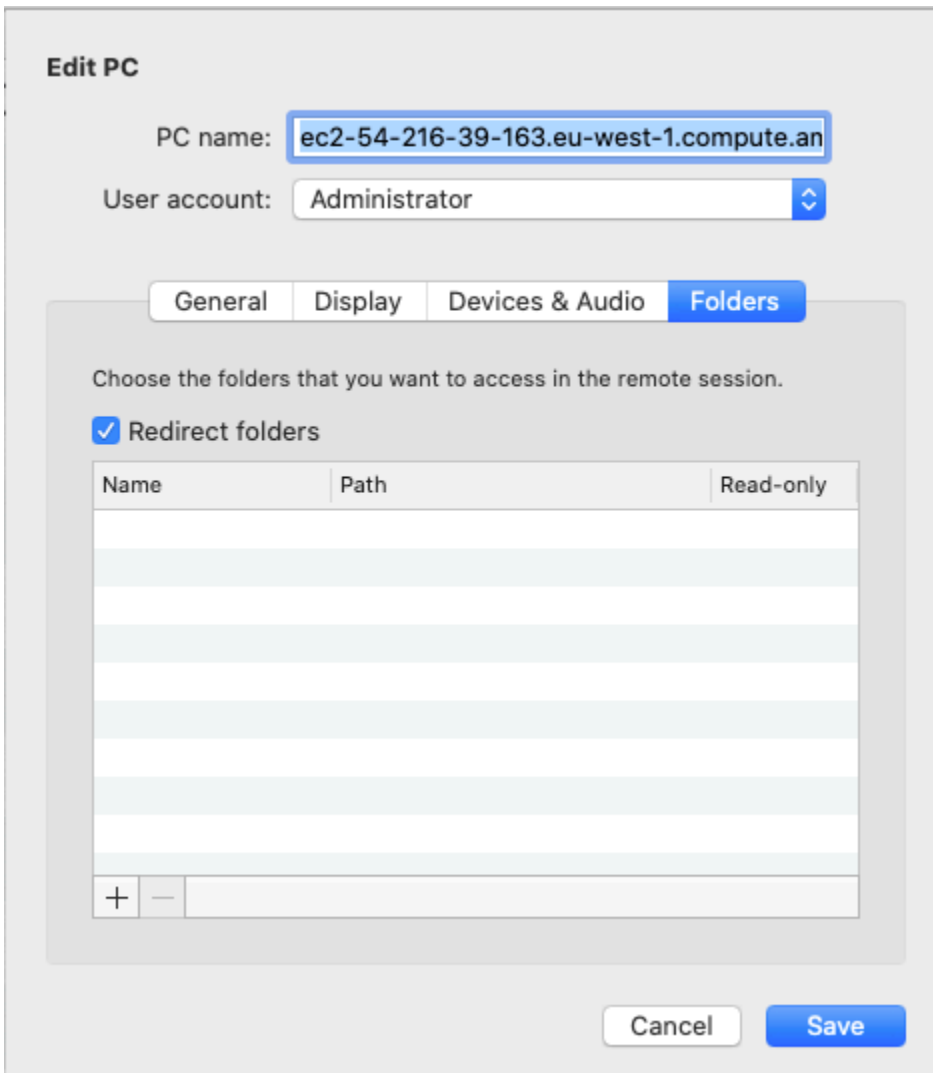


8. 選擇 Connect (連接) 以連接到您的 Windows 執行個體。

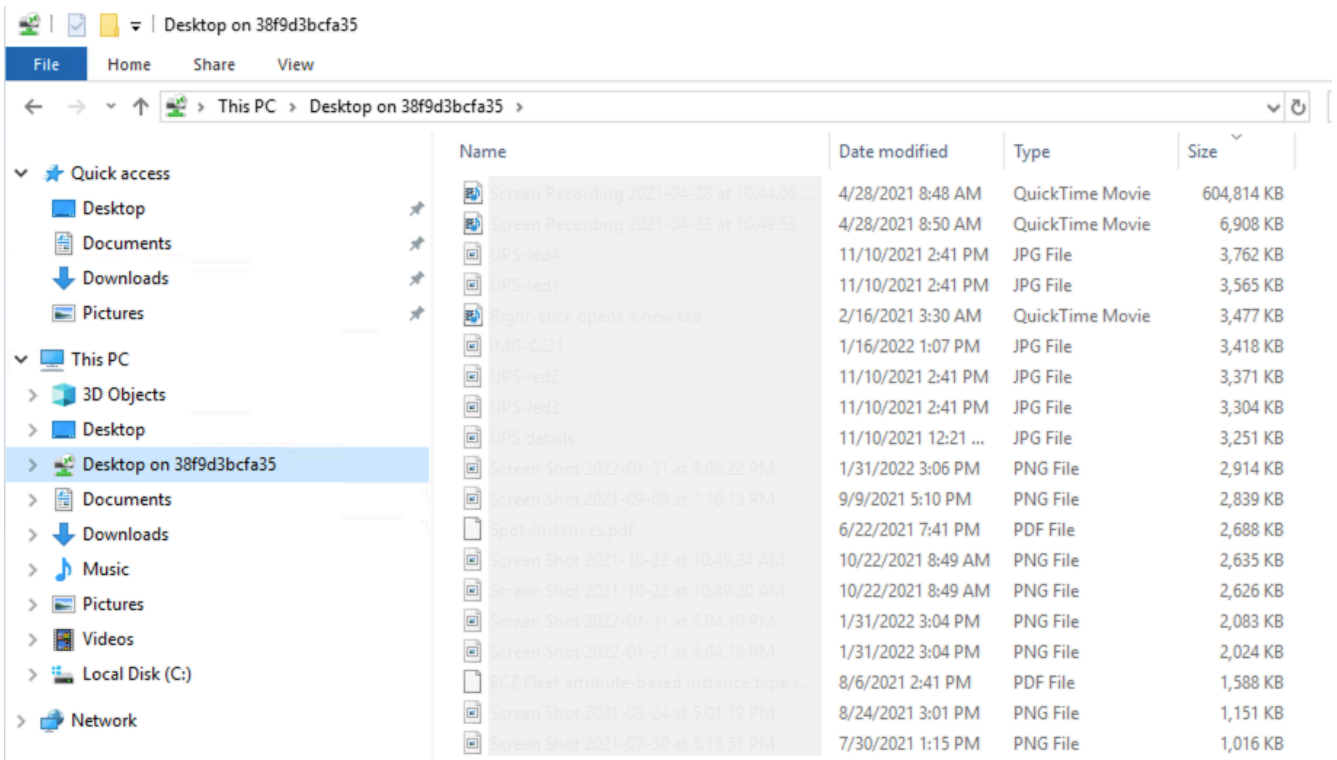
## macOS X

如要將遠端工作階段磁碟機映射至本機 macOS X 電腦上的本機資料夾

1. 開啟遠端桌面連線用戶端。
2. 瀏覽至您從 Amazon EC2 主控台下載的 RDP 檔案 (在您最初連接到執行個體時)，然後將其拖曳至遠端桌面連線用戶端。
3. 以滑鼠右鍵按一下 RDP 檔案，選擇 Edit (編輯)。
4. 選取 Folders (資料夾) 索引標籤，然後選取 Redirect folders (重新引導資料夾) 核取方塊。



5. 選擇左下方的 + 圖示，瀏覽至要映射的資料夾，然後選擇 Open (開啟)。為每個要映射的資料夾重複此步驟。
6. 選擇儲存。
7. 選擇 Connect (連接) 以連接到您的 Windows 執行個體。系統會提示您輸入密碼。
8. 在執行個體上的檔案總管中，展開 This PC (此 PC)，然後找出您可從中存取本機檔案的共用資料夾。於下列螢幕擷取畫面中，本機電腦上的 Desktop (桌面) 資料夾已映射至執行個體上的遠端工作階段磁碟機。



如需在 Mac 電腦上使用本機裝置可用於遠端工作階段的詳細資訊，請參閱 [Mac 用戶端入門](#)。

## 使用工作階段管理員進行連接

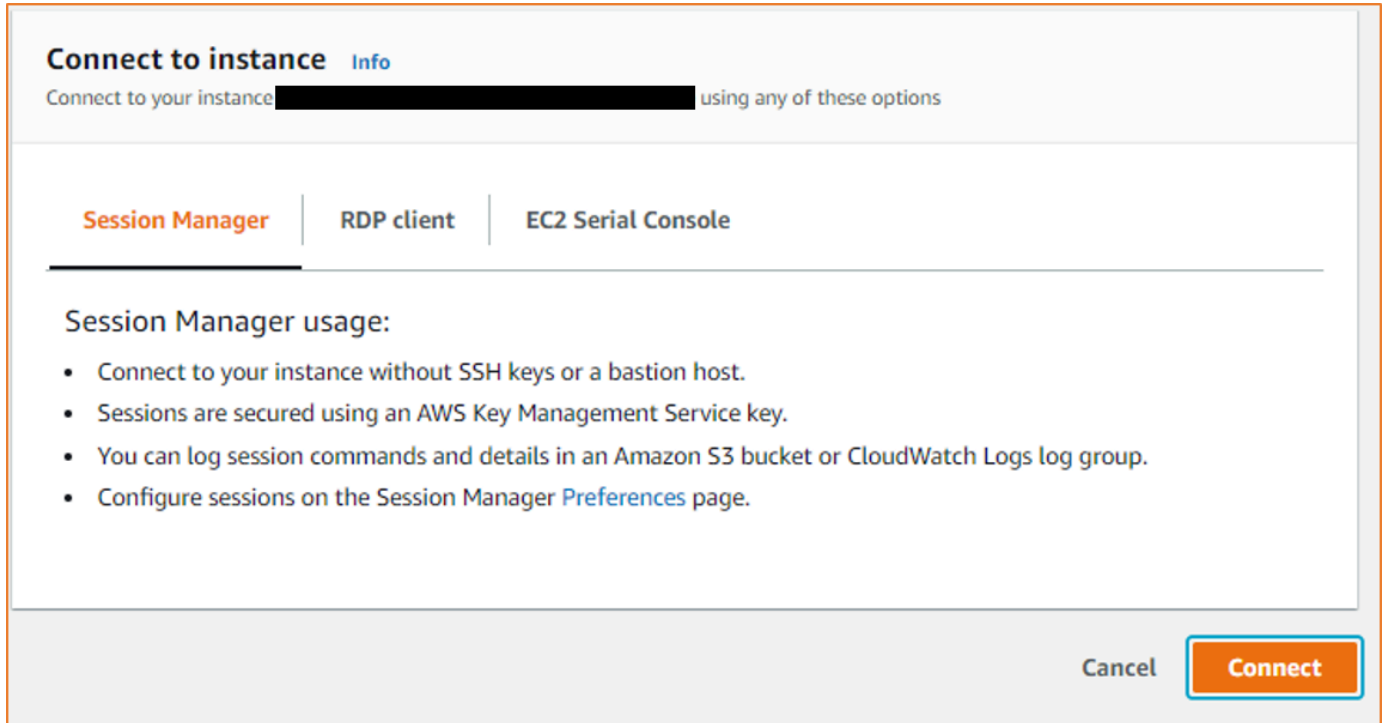
工作階段管理員是一項全受管 AWS Systems Manager 功能 Amazon EC2 可透過互動式、按一下瀏覽器型殼層或透過 AWS CLI 您可以使用工作階段管理員，搭配您帳戶中的執行個體啟動工作階段。工作階段啟動之後，您可以在執行個體上執行互動式指令，就像對任何其他連線類型執行一樣。如需工作階段管理員的詳細資訊，請參閱《AWS Systems Manager 使用者指南》中的 [AWS Systems Manager 工作階段管理員](#)。

嘗試使用工作階段管理員連接到執行個體之前，請確保必要的設定步驟已經完成。如需詳細資訊，請參閱 [設定工作階段管理員](#)。

使用 Amazon EC2 主控台上的工作階段管理員連線到 Amazon EC2 執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選取 Connect (連線)。
4. 對於 Connection method (連線方法)，選擇 Session Manager (工作階段管理員)。

## 5. 選擇 Connect (連線)。



### Tip

如果您收到未獲授權執行一或多個 Systems Manager 動作 (`ssm:command-name`) 的錯誤，則必須更新政策以允許您從 Amazon EC2 主控台啟動工作階段。如需詳細資訊和指示，請參閱《AWS Systems Manager 使用者指南》中的 [工作階段管理員的預設 IAM 政策快速入門](#)。

## 使用 EC2 執行個體 Connect 端點 Connect 至執行個體

EC2 執行個體 Connect Endpoint 可讓您從網際網路安全地連線至執行個體，而無需使用防禦主機，或要求您的虛擬私有雲端 (VPC) 具有直接的網際網路連線能力。

### 優勢

- 您可以連線至執行個體，而不需要執行個體擁有公用 IPv4 位址。AWS 所有公用 IPv4 位址的費用，包括與執行中執行個體和彈性 IP 位址相關聯的公用 IPv4 位址。如需詳細資訊，請參閱 [Amazon VPC 定價頁面](#) 中的公有 IPv4 地址。
- 您可以從網際網路連線至執行個體，而不需要 VPC 透過網際網路 [閘道直接連線網際網路](#)。



- 您可以控制對 EC2 執行個體 Connect 端點的建立和使用的存取，以使用 [IAM 政策和許可](#) 連接到執行個體。
- 連線至執行個體的所有嘗試 (無論是成功還是失敗) 都會記錄到 [CloudTrail](#)。

## 定價

使用 EC2 執行個體 Connect 端點無需額外費用。如果您使用 EC2 執行個體 Connect Endpoint 連接到不同可用區域中的執行個體，則跨可用區域的資料傳輸需額外付費。

## 目錄

- [運作方式](#)
- [考量事項](#)
- [授予使用 EC2 執行個體 Connect 端點的許可](#)
- [EC2 Instance Connect Endpoint 的安全群組](#)
- [建立 EC2 Instance Connect Endpoint](#)
- [使用 EC2 執行個體 Connect 端點 Connect 到亞馬遜 EC2 執行個體](#)
- [記錄透過 EC2 Instance Connect Endpoint 建立的連線](#)
- [刪除 EC2 執行個體 Connect 端點](#)
- [EC2 Instance Connect Endpoint 的服務連結角色](#)
- [EC2 執行個體 Connect 端點的配額](#)

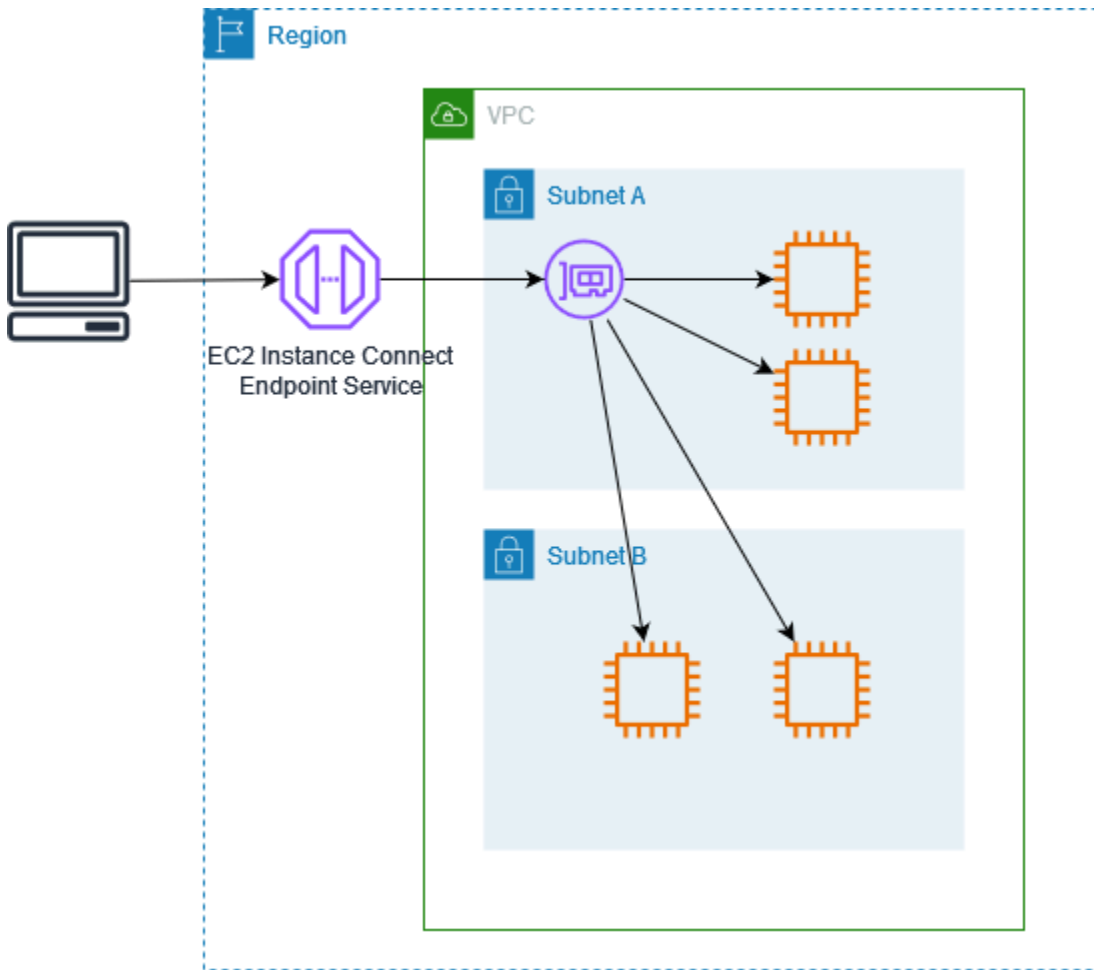
## 運作方式

EC2 執行個體 Connect 端點是具有身分識別的 TCP 代理。EC2 執行個體 Connect 端點服務會使用 IAM 實體的登入資料建立從電腦到端點的私有通道。流量在到達您的 VPC 之前已經過驗證和授權。

您可以 [設定其他安全群組規則](#)，以限制執行個體的輸入流量。例如，您可以使用輸入規則，僅允許來自 EC2 執行個體 Connect 端點的管理連接埠上的流量。

您可以設定路由表規則，以允許端點連線至 VPC 任何子網路中的任何執行個體。

下圖顯示使用者如何使用 EC2 執行個體 Connect 端點從網際網路連線到其執行個體。首先，在子網路 A 中建立 EC2 執行個體 Connect Endpoint。我們會在子網路中為端點建立網路介面，該介面可做為目的地到 VPC 中執行個體的流量的進入點。如果子網路 B 的路由表允許來自子網路 A 的流量，則您可以使用端點連線子網路 B 中的執行個體。



## 考量事項

在開始之前，請考慮下列事項。

- EC2 執行個體 Connect 端點專門用於管理流量使用案例，而不適用於大量資料傳輸。系統會限流大量資料傳輸。
- 您的執行個體必須具有 IPv4 地址 (私有或公有)。EC2 執行個體 Connect 端點不支援使用 IPv6 地址連線至執行個體。
- (Linux 執行個體) 如果您使用自己的 key pair 合，您可以使用任何 Linux AMI。否則，您的執行個體必須安裝 EC2 執行個體 Connect。如需有關哪些 AMI 包含 EC2 執行個體 Connect 以及如何在其他支援的 AMI 上安裝的詳細資訊，請參閱 [安裝 EC2 Instance Connect](#)
- 您可以在建立 EC2 執行個體 Connect 端點時將安全群組指派給該群組。否則，我們會使用 VPC 的預設安全性群組。EC2 執行個體 Connect 端點的安全群組必須允許傳出流量到目的地執行個體。如需詳細資訊，請參閱 [EC2 Instance Connect Endpoint 的安全群組](#)。

- 您可以設定 EC2 執行個體 Connect 端點，以便在將請求路由到執行個體時保留用戶端的來源 IP 地址。否則，網路介面的 IP 位址會變成所有傳入流量的用戶端 IP 位址。
- 如果您開啟用戶端 IP 保留，執行個體的安全群組必須允許來自用戶端的流量。此外，執行個體必須與 EC2 執行個體 Connect 端點位於相同的 VPC 中。
- 如果您關閉用戶端 IP 保留，執行個體的安全性群組必須允許來自 VPC 的流量。此為預設值。
- 下列執行個體類型不支援用戶端 IP 保留：C1、CG1、CG2、G1、HI1、M1、M2、M3 和 T1。如果您開啟用戶端 IP 保留功能，並嘗試使用 EC2 執行個體 Connect Endpoint 連線到具有上述其中一種執行個體類型的執行個體，則連線會失敗。
- 當流量透過傳輸閘道路由時，不支援用戶端 IP 保留。
- 當您建立 EC2 執行個體 Connect 端點時，會在 AWS Identity and Access Management (IAM) 中為 Amazon EC2 服務自動建立服務連結角色。Amazon EC2 使用服務連結角色在網路介面中佈建帳戶，這在建立 EC2 Instance Connect Endpoint 時是必需的。如需詳細資訊，請參閱 [EC2 Instance Connect Endpoint 的服務連結角色](#)。
- 每個 EC2 執行個體連線端點最多可支援 20 個並行連線。
- 建立的 TCP 連線的最長持續時間為 1 小時 (3,600 秒)。您可以在 IAM 政策中指定允許的最長持續時間，最長可達 3,600 秒。如需詳細資訊，請參閱 [使用 EC2 執行個體連線端點連線至執行個體的許可](#)。
- 加拿大西部 (卡加利) 不支援 EC2 執行個體 Connect 端點。

## 授予使用 EC2 執行個體 Connect 端點的許可

依預設，IAM 實體沒有建立、描述或修改 EC2 執行個體 Connect 端點的權限。IAM 管理員可以建立 IAM 政策，以授與對所需資源執行特定動作所需的權限。

如需有關建立和編輯 IAM 政策的資訊，請參閱《IAM 使用者指南》中的 [建立 IAM 政策](#)。

下列範例政策顯示您可以控制使用者對 EC2 執行個體 Connect 端點所擁有的許可。

### 範例

- [建立、描述和刪除 EC2 執行個體 Connect 端點的許可](#)
- [使用 EC2 執行個體連線端點連線至執行個體的許可](#)
- [僅從特定 IP 位址範圍連線的權限](#)

### 建立、描述和刪除 EC2 執行個體 Connect 端點的許可

若要建立 EC2 Instance Connect Endpoint，使用者需要下列動作的許可：

- `ec2:CreateInstanceConnectEndpoint`
- `ec2:CreateNetworkInterface`
- `ec2:CreateTags`
- `iam:CreateServiceLinkedRole`

若要描述和刪除 EC2 Instance Connect Endpoint，使用者需要下列動作的許可：

- `ec2:DescribeInstanceConnectEndpoints`
- `ec2>DeleteInstanceConnectEndpoint`

您可以建立一個政策，以准許在所有子網路中建立、描述和刪除 EC2 Instance Connect Endpoint。或者，僅將子網路 ARN 指定為允許的 Resource 或使用 `ec2:SubnetID` 條件索引鍵，來限制指定子網路的動作。您也可以使用 `aws:ResourceTag` 條件索引鍵來明確允許或拒絕具有特定標籤的端點建立。如需詳細資訊，請參閱《IAM 使用者指南》中的 [IAM 中的政策和許可](#)。

## IAM 政策範例

在下列 IAM 政策範例中，Resource 區段准許在所有子網路中建立和刪除端點，以星號 (\*) 指定。 `ec2:Describe*` API 動作不支援資源層級許可。因此，在 Resource 元素中必須包含 \* 萬用字元。

```
{
 "Version": "2012-10-17",
 "Statement": [{
 "Sid": "GrantAllActionsInAllSubnets",
 "Action": [
 "ec2:CreateInstanceConnectEndpoint",
 "ec2>DeleteInstanceConnectEndpoint",
 "ec2:CreateNetworkInterface",
 "ec2:CreateTags",
 "iam:CreateServiceLinkedRole"
],
 "Effect": "Allow",
 "Resource": "arn:aws:ec2:region:account-id:subnet/*"
 }],
 {
 "Action": [
 "ec2:CreateNetworkInterface"
],
 }
}
```

```
 "Effect": "Allow",
 "Resource": "arn:aws:ec2::security-group/*"
 },
 {
 "Sid": "DescribeInstanceConnectEndpoints",
 "Action": [
 "ec2:DescribeInstanceConnectEndpoints"
],
 "Effect": "Allow",
 "Resource": "*"
 }
]
```

## 使用 EC2 執行個體連線端點連線至執行個體的許可

`ec2-instance-connect:OpenTunnel` 動作准許建立執行個體的 TCP 連線，以便透過 EC2 Instance Connect Endpoint 進行連線。您可以指定要使用的 EC2 Instance Connect Endpoint。或者，帶有星號 (\*) 的 Resource 允許使用者使用任何可用的 EC2 Instance Connect Endpoint。您也可以根據是否存在作為條件索引鍵的資源標籤來限制執行個體的存取。

### 條件

- `ec2-instance-connect:remotePort`— 執行個體上可用來建立 TCP 連線的連接埠。使用此條件索引鍵時，嘗試連線到政策中指定的連接埠以外的任何其他連接埠上的執行個體會導致失敗。
- `ec2-instance-connect:privateIpAddress`— 與您要建立 TCP 連線的執行個體相關聯的目標私有 IP 位址。您可以指定單一 IP 地址 (例如 `10.0.0.1/32`) 或透過 CIDR 指定 IP 範圍 (例如 `10.0.1.0/28`)。使用此條件索引鍵時，嘗試連線至具有不同私有 IP 地址或超出 CIDR 範圍的執行個體會導致失敗。
- `ec2-instance-connect:maxTunnelDuration`— 已建立 TCP 連線的最長持續時間。單位為秒，持續時間範圍為至少 1 秒至最多 3,600 秒 (1 小時)。如果未指定條件，則預設持續時間會設為 3,600 秒 (1 小時)。嘗試連線至執行個體的時間超過 IAM 政策中指定的持續時間，或超過預設最大值時，會導致失敗。連線會在指定的持續時間後中斷。

如果在 IAM 政策中指定 `maxTunnelDuration`，且指定的值少於 3,600 秒 (預設值)，則您必須在連線至執行個體時在命令中指定 `--max-tunnel-duration`。如需有關如何連線至執行個體的資訊，請參閱 [使用 EC2 執行個體 Connect 端點 Connect 到亞馬遜 EC2 執行個體](#)。

您也可以授予使用者存取權，以根據 EC2 執行個體 Connect 端點上的資源標籤存在建立與執行個體的連線。如需詳細資訊，請參閱《IAM 使用者指南》中的 [IAM 中的政策和許可](#)。

對於 Linux 執行個體，此 `ec2-instance-connect:SendSSHPublicKey` 動作會授與將公開金鑰推送至執行個體的權限。`ec2:osuser` 條件指定可將公有金鑰推送至執行個體的 OS (作業系統) 使用者名稱。使用您用來啟動執行個體之 AMI 的預設使用者名稱。如需詳細資訊，請參閱 [為 EC2 Instance Connect 授予 IAM 許可](#)。

## IAM 政策範例

以下 IAM 政策範例允許 IAM 主體僅使用指定的 EC2 執行個體 Connect 端點 (由指定的端點 ID 識別) 連接到執行個體 `eice-123456789abcdef`。只有在滿足所有條件時，才能成功建立連接。

### Note

`ec2:Describe*` API 動作不支援資源層級許可。因此，在 Resource 元素中必須包含 \* 萬用字元。

## Linux

本範例會評估執行個體的連線是否在 `port 22` (SSH) 上建立，執行個體的私有 IP 位址是否位於 `10.0.1.0/31` (介於 `10.0.1.0` 和之間 `10.0.1.1`) 的範圍內，且小 `maxTunnelDuration` 於或等於 `3600` 秒。連線會在 `3600` 秒 (1 小時) 後中斷。

```
{
 "Version": "2012-10-17",
 "Statement": [{
 "Sid": "EC2InstanceConnect",
 "Action": "ec2-instance-connect:OpenTunnel",
 "Effect": "Allow",
 "Resource": "arn:aws:ec2:region:account-id:instance-connect-endpoint/eice-123456789abcdef",
 "Condition": {
 "NumericEquals": {
 "ec2-instance-connect:remotePort": "22"
 },
 "IpAddress": {
 "ec2-instance-connect:privateIpAddress": "10.0.1.0/31"
 },
 "NumericLessThanEquals": {
 "ec2-instance-connect:maxTunnelDuration": "3600"
 }
 }
]
}
```

```

 }
 },
 {
 "Sid": "SSHPublicKey",
 "Effect": "Allow",
 "Action": "ec2-instance-connect:SendSSHPublicKey",
 "Resource": "*",
 "Condition": {
 "StringEquals": {
 "ec2:osuser": "ami-username"
 }
 }
 },
 {
 "Sid": "Describe",
 "Action": [
 "ec2:DescribeInstances",
 "ec2:DescribeInstanceConnectEndpoints"
],
 "Effect": "Allow",
 "Resource": "*"
 }
]
}

```

## Windows

此範例會評估執行個體的連線是否在連接埠 3389 (RDP) 上建立，執行個體的私有 IP 位址是否位於 10.0.1.0/31 (介於 10.0.1.0 和之間 10.0.1.1) 的範圍內，且小 `maxTunnelDuration` 於或等於 3600 秒。連線會在 3600 秒 (1 小時) 後中斷。

```

{
 "Version": "2012-10-17",
 "Statement": [{
 "Sid": "EC2InstanceConnect",
 "Action": "ec2-instance-connect:OpenTunnel",
 "Effect": "Allow",
 "Resource": "arn:aws:ec2:region:account-id:instance-connect-endpoint/eice-123456789abcdef",
 "Condition": {
 "NumericEquals": {
 "ec2-instance-connect:remotePort": "3389"
 }
 }
 }]
}

```

```

 },
 "IpAddress": {
 "ec2-instance-connect:privateIpAddress": "10.0.1.0/31"
 },
 "NumericLessThanEquals": {
 "ec2-instance-connect:maxTunnelDuration": "3600"
 }
 }
},
{
 "Sid": "Describe",
 "Action": [
 "ec2:DescribeInstances",
 "ec2:DescribeInstanceConnectEndpoints"
],
 "Effect": "Allow",
 "Resource": "*"
}
]
}

```

### 僅從特定 IP 位址範圍連線的權限

下列 IAM 政策範例允許 IAM 主體連線到執行個體，條件是它們從政策中指定的 IP 地址範圍內的 IP 地址進行連線。如果 IAM 主體呼叫 `OpenTunnel` 的 IP 位址不在範圍內 `192.0.2.0/24` (此政策中的範例 IP 位址範圍)，則回應為 `Access Denied`。如需詳細資訊，請參閱《IAM 使用者指南》中的 [aws:SourceIp](#)。

```

{
 "Version": "2012-10-17",
 "Statement": [{
 "Effect": "Allow",
 "Action": "ec2-instance-connect:OpenTunnel",
 "Resource": "arn:aws:ec2:region:account-id:instance-connect-endpoint/eice-123456789abcdef",
 "Condition": {
 "IpAddress": {
 "aws:SourceIp": "192.0.2.0/24"
 },
 "NumericEquals": {
 "ec2-instance-connect:remotePort": "22"
 }
 }
]
}

```



```

 }
 },
 {
 "Sid": "SSHPublicKey",
 "Effect": "Allow",
 "Action": "ec2-instance-connect:SendSSHPublicKey",
 "Resource": "*",
 "Condition": {
 "StringEquals": {
 "ec2:osuser": "ami-username"
 }
 }
 },
 {
 "Effect": "Allow",
 "Action": [
 "ec2:DescribeInstances",
 "ec2:DescribeInstanceConnectEndpoints"
],
 "Resource": "*"
 }
]
}

```

## EC2 Instance Connect Endpoint 的安全群組

安全群組負責控制允許到達和離開其關聯資源的流量。例如，除非與執行個體相關聯的安全群組特別允許，否則我們拒絕傳入和傳出 Amazon EC2 執行個體的流量。

下列範例說明如何為 EC2 執行個體 Connect 端點和目標執行個體設定安全群組規則。

### 範例

- [EC2 執行個體 Connect 端點安全群組規則](#)
- [目標例項安全性群組規則](#)

### EC2 執行個體 Connect 端點安全群組規則

EC2 執行個體 Connect 端點的安全群組規則必須允許目標執行個體的輸出流量離開端點。您可以將執行個體安全性群組或 VPC 的 IPv4 位址範圍指定為目的地。

端點的流量源自 EC2 執行個體 Connect 端點服務，無論端點安全群組的輸入規則為何，都允許此流量。若要控制誰可以使用 EC2 執行個體 Connect 線端點連線到執行個體，請使用 IAM 政策。如需詳細資訊，請參閱 [使用 EC2 執行個體連線端點連線至執行個體的許可](#)。

#### 輸出規則範例：安全性群組參照

下列範例使用安全性群組參考，這表示目的地是與目標執行個體相關聯的安全性群組。此規則允許從端點傳輸到使用此安全性群組的所有執行個體的輸出流量。

通訊協定	目的地	連接埠範圍	註解
TCP	<i>##### ID</i>	22	允許與執行個體安全性群組相關聯的所有執行個體輸出 SSH 流量

#### 輸出規則範例：IPv4 位址範圍

下列範例允許輸出流量傳送至指定 IPv4 位址範圍。執行個體的 IPv4 位址是從其子網路指派的，因此您可以使用 VPC 的 IPv4 位址範圍。

通訊協定	目的地	連接埠範圍	註解
TCP	<i>VPC A IPv4 CIDR</i>	22	允許對外 SSH 流量傳送至 VPC

#### 目標例項安全性群組規則

目標執行個體的安全性群組規則必須允許來自 EC2 執行個體 Connect 端點的輸入流量。您可以指定端點安全性群組或 IPv4 位址範圍做為來源。如果您指定 IPv4 位址範圍，則來源會視用戶端 IP 保留是關閉還是開啟而定。如需詳細資訊，請參閱 [考量事項](#)。

由於安全群組是可設定狀態的，因此無論執行個體安全性群組的輸出規則為何，都允許回應流量離開 VPC。

#### 輸入規則範例：安全性群組參考

下列範例使用安全性群組參考，這表示來源是與端點相關聯的安全性群組。此規則允許從端點傳入 SSH 流量到使用此安全性群組的所有執行個體，無論用戶端 IP 保留是開啟還是關閉。如果 SSH 沒有其他輸入安全群組規則，則執行個體只接受來自端點的 SSH 流量。

通訊協定	來源	連接埠範圍	註解
TCP	#####	22	允許來自與端點安全性群組相關聯之資源的輸入 SSH 流量

輸入規則範例：用戶端 IP 保留關閉

下列範例允許來自指定 IPv4 位址範圍的輸入 SSH 流量。由於用戶端 IP 保留已關閉，因此來源 IPv4 位址是端點網路介面的位址。端點網路介面的位址是從其子網路指派的，因此您可以使用 VPC 的 IPv4 位址範圍來允許連線至 VPC 中的所有執行個體。

通訊協定	來源	連接埠範圍	註解
TCP	VPC A IPv4 CIDR	22	允許來自 VPC 雲端的入埠 SSH 流量

輸入規則範例：用戶端 IP 保留

下列範例允許來自指定 IPv4 位址範圍的輸入 SSH 流量。由於用戶端 IP 保留已開啟，因此來源 IPv4 位址是用戶端的位址。

通訊協定	來源	連接埠範圍	註解
TCP	## IPv4 #####	22	允許來自指定用戶端 IPv4 位址範圍的輸入流量

## 建立 EC2 Instance Connect Endpoint

您可以建立 EC2 執行個體 Connect 端點，以允許與執行個體進行安全連線。

建立 EC2 執行個體連線端點後，就無法修改它。相反地，您必須刪除 EC2 執行個體 Connect 端點，然後使用您需要的設定建立新的執行個體。

### 必要條件

您必須具有必要的 IAM 許可才能建立 EC2 Instance Connect Endpoint。如需詳細資訊，請參閱 [建立、描述和刪除 EC2 執行個體 Connect 端點的許可](#)。

## 共用子網路

您可以在與您共用的子網路中建立 EC2 執行個體 Connect 端點。您無法使用 VPC 擁有者在與您共用的子網路中建立的 EC2 執行個體 Connect 端點。

### 使用主控台建立端點

使用下列程序建立 EC2 執行個體 Connect 端點。

#### 建立 EC2 Instance Connect Endpoint

1. 前往 <https://console.aws.amazon.com/vpc/> 開啟 Amazon VPC 主控台。
2. 在左側導覽窗格中選擇 Endpoints (端點)。
3. 選擇 [建立端點]，然後依下列方式指定端點設定：
  - a. (選用) 針對名稱標籤，請輸入端點的名稱。
  - b. 針對服務類別，請選擇 EC2 Instance Connect Endpoint。
  - c. 對於 VPC，請選取具有目標執行個體的 VPC。
  - d. (選擇性) 若要保留用戶端 IP 位址，請展開其他設定，然後選取核取方塊。否則，預設會使用端點網路介面做為用戶端 IP 位址。
  - e. (選用) 針對安全群組，選取要與端點產生關聯的安全群組。否則，預設為使用 VPC 的預設安全性群組。如需詳細資訊，請參閱 [EC2 Instance Connect Endpoint 的安全群組](#)。
  - f. 針對子網路，請選取要在其中建立端點的子網路。
  - g. (選用) 若要新增標籤，請選擇 Add new tag (新增標籤)，然後輸入標籤的鍵和值。
4. 檢閱您的設定，然後選擇 [建立端點]。

端點的初始狀態為「擱置中」。您必須等到端點狀態為「可用」，才能使用此端點連線至執行個體。這可能需要幾分鐘的時間。

5. 若要使用端點連線至執行個體，請參閱 [連接至執行個體](#)。

### 使用建立端點 AWS CLI

使用 [create-instance-connect-endpoint](#) 命令。AWS CLI

### 必要條件

安裝 AWS CLI 版本 2 並使用您的憑據進行配置。若要取得[更多資訊](#)，請參閱《[AWS Command Line Interface 使用者指南](#)》AWS CLI 中的〈[安裝或更新至最新版本](#)〉AWS CLI 和〈[規劃](#)〉。或者，在其預先驗證的 AWS CLI shell 中打開 AWS CloudShell 並運行命令。

## 建立端點

使用以下命令為指定子網路中的 EC2 執行個體 Connect 端點建立端點網路界面。

```
aws ec2 create-instance-connect-endpoint --subnet-id subnet-0123456789example
```

下列為範例輸出。

```
{
 "OwnerId": "111111111111",
 "InstanceConnectEndpointId": "eice-0123456789example",
 "InstanceConnectEndpointArn": "arn:aws:ec2:us-east-1:111111111111:instance-connect-endpoint/eice-0123456789example",
 "State": "create-complete",
 "StateMessage": "",
 "DnsName": "eice-0123456789example.0123abcd.ec2-instance-connect-endpoint.us-east-1.amazonaws.com",
 "FipsDnsName": "eice-0123456789example.0123abcd.fips.ec2-instance-connect-endpoint.us-east-1.amazonaws.com",
 "NetworkInterfaceIds": [
 "eni-0123abcd"
],
 "VpcId": "vpc-0123abcd",
 "AvailabilityZone": "us-east-1a",
 "CreatedAt": "2023-04-07T15:43:53.000Z",
 "SubnetId": "subnet-0123abcd",
 "PreserveClientIp": false,
 "SecurityGroupIds": [
 "sg-0123abcd"
],
 "Tags": []
}
```

## 監視建立狀態

State 欄位的初始值為 create-in-progress。在您可使用此端點連線至執行個體之前，請等待狀態變為 create-complete。使用命令[describe-instance-connect-endpoints](#) AWS CLI 令監控 EC2 執行個體 Connect 端點的狀態。-query 參數會將結果篩選至欄位State。

```
aws ec2 describe-instance-connect-endpoints --instance-connect-endpoint-ids eice-0123456789example --query InstanceConnectEndpoints[*].State --output text
```

下列為範例輸出。

```
create-complete
```

## 使用 EC2 執行個體 Connect 端點 Connect 到亞馬遜 EC2 執行個體

您可以使用 EC2 執行個體 Connect 端點連接到支援安全殼層或 RDP 的 Amazon EC2 執行個體。

### 目錄

- [必要條件](#)
- [疑難排解](#)

### 必要條件

- 您必須具有必要的 IAM 許可才能連線至 EC2 Instance Connect Endpoint。如需詳細資訊，請參閱 [使用 EC2 執行個體連線端點連線至執行個體的許可](#)。
- EC2 Instance Connect Endpoint 必須為可用 (主控台) 或 create-complete (AWS CLI) 狀態。如果您的 VPC 沒有 EC2 執行個體 Connect 端點，則可以建立一個。如需詳細資訊，請參閱 [建立 EC2 Instance Connect Endpoint](#)。
- (Linux 執行個體) 若要使用 EC2 主控台連線至執行個體，或使用 CLI Connect 並讓 EC2 執行個體 Connect 處理暫時金鑰，您的執行個體必須已安裝 EC2 執行個體連接。如需詳細資訊，請參閱 [安裝 EC2 Instance Connect](#)。
- 確保執行個體的安全群組允許來自 EC2 執行個體 Connect 端點的傳入 SSH 流量。如需詳細資訊，請參閱 [目標例項安全性群組規則](#)。

### 使用 Amazon EC2 主控台連線至 Linux 執行個體

您可以使用 Amazon EC2 主控台連線到執行個體，如下所示。

### 使用以瀏覽器為基礎的用戶端連線至執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體，選擇 [Connect]。

4. 選擇 EC2 Instance Connect 索引標籤。
5. 針對連線類型，請選擇使用 EC2 Instance Connect Endpoint 進行連線。
6. 對於 EC2 執行個體 Connect 端點，請選擇 EC2 執行個體 Connect 端點的 ID。
7. 針對使用者名稱，如果您用來啟動執行個體的 AMI 使用的使用者名稱不是 `ec2-user`，請輸入正確的使用者名稱。
8. 針對通道持續時間上限 (秒)，輸入允許的 SSH 連線持續時間上限。

持續時間必須符合 IAM 政策中指定的任何 `maxTunnelDuration` 條件。如果您無法存取 IAM 政策，請聯絡您的管理員。

9. 選擇連線。這會為您的執行個體開啟終端機視窗。

### 使用 SSH 連線至您的 Linux 執行個體

您可以使用 SSH 連線到 Linux 執行個體，並使用 `open-tunnel` 命令建立私有通道。可以在單一連線或多重連線模式下使用 `open-tunnel`。

如需使用安全殼層連線 AWS CLI 至執行個體的詳細資訊，請參閱 [使用 Connect AWS CLI](#)。

以下範例使用 [OpenSSH](#)。您可以使用支援代理模式的任何其他 SSH 用戶端。

#### 單一連接

使用 SSH 和 `open-tunnel` 命令僅允許單一連線至執行個體

使用 `ssh` 和 `open-tunnel` AWS CLI 命令如下。 `-o` 代理命令包含 `open-tunnel` 命令，它可建立執行個體的私有通道。

```
ssh -i my-key-pair.pem ec2-user@i-0123456789example \
-o ProxyCommand='aws ec2-instance-connect open-tunnel --instance-
id i-0123456789example'
```

對於：

- `-i` – 指定用來啟動執行個體的金鑰對。
- `ec2-user@i-0123456789example` – 指定用來啟動執行個體的 AMI 使用者名稱，以及執行個體 ID。
- `--instance-id` – 指定要連線之執行個體的 ID。或者，指定 `%h`，其可從使用者中擷取執行個體 ID。

## 多重連線

若要允許多個連線至執行個體，請先執行[open-tunnel](#) AWS CLI 命令以開始偵聽新的 TCP 連線，然後使ssh用建立新的 TCP 連線和私人通道至您的執行個體。

允許使用 SSH 和 **open-tunnel** 命令多重連線至執行個體

1. 執行下列命令來開始偵聽本機電腦上指定連接埠上的新 TCP 連線。

```
aws ec2-instance-connect open-tunnel \
 --instance-id i-0123456789example \
 --local-port 8888
```

### 預期的輸出結果

```
Listening for connections on port 8888.
```

2. 在新終端視窗中，執行下列 ssh 命令，為執行個體建立新的 TCP 連線和私有通道。

```
ssh -i my-key-pair.pem ec2-user@localhost -p 8888
```

預期輸出 – 在第一個終端視窗中，您會看到以下內容：

```
[1] Accepted new tcp connection, opening websocket tunnel.
```

您可能會看到以下內容：

```
[1] Closing tcp connection.
```

## 使用 Connect 到您的 Linux 執行個體 AWS CLI

如果您只知道執行個體 ID，則可以使用 [ec2 執行個體 connect AWS CLI 指令](#)，透過安全殼層用戶端連線至執行個體。若要取得有關使用 [ec2 執行個體連線指令](#) 的更多資訊，請參閱 [〈〉](#)。[使用 Connect AWS CLI](#)

## 必要條件



安裝 AWS CLI 版本 2 並使用您的憑據進行配置。若要取得[更多資訊](#)，請參閱《[AWS Command Line Interface 使用者指南](#)》AWS CLI 中的〈[安裝或更新至最新版本](#)〉AWS CLI 和〈[規劃](#)〉。或者，在其預先驗證的 AWS CLI shell 中打開 AWS CloudShell 並運行命令。

使用執行個體 ID 和 EC2 Instance Connect Endpoint 連線至執行個體

如果您只知道執行個體識別碼，請使用 [ec2 執行個體連線](#) CLI 指令，並指定指 ssh 令、執行個體識別碼和具有 `--connection-type` 值的參數。eice

```
aws ec2-instance-connect ssh --instance-id i-1234567890example --connection-type eice
```

### Tip

如果使用此命令時出現錯誤，請確定您使用的是 AWS CLI 版本 2。該 ssh 參數僅在 AWS CLI 版本 2 中可用。如需詳細資訊，請參閱《[使用指南](#)》中的 AWS Command Line Interface 〈[關於 AWS CLI 版本 2](#)〉。

使用 EC2 執行個體 Connect 端點 Connect 至您的 Windows 執行個體

您可以透過 EC2 Instance Connect Endpoint，使用遠端桌面通訊協定 (RDP)，在沒有公有 IPv4 地址或公有 DNS 名稱的情況下連線到 Windows 執行個體。

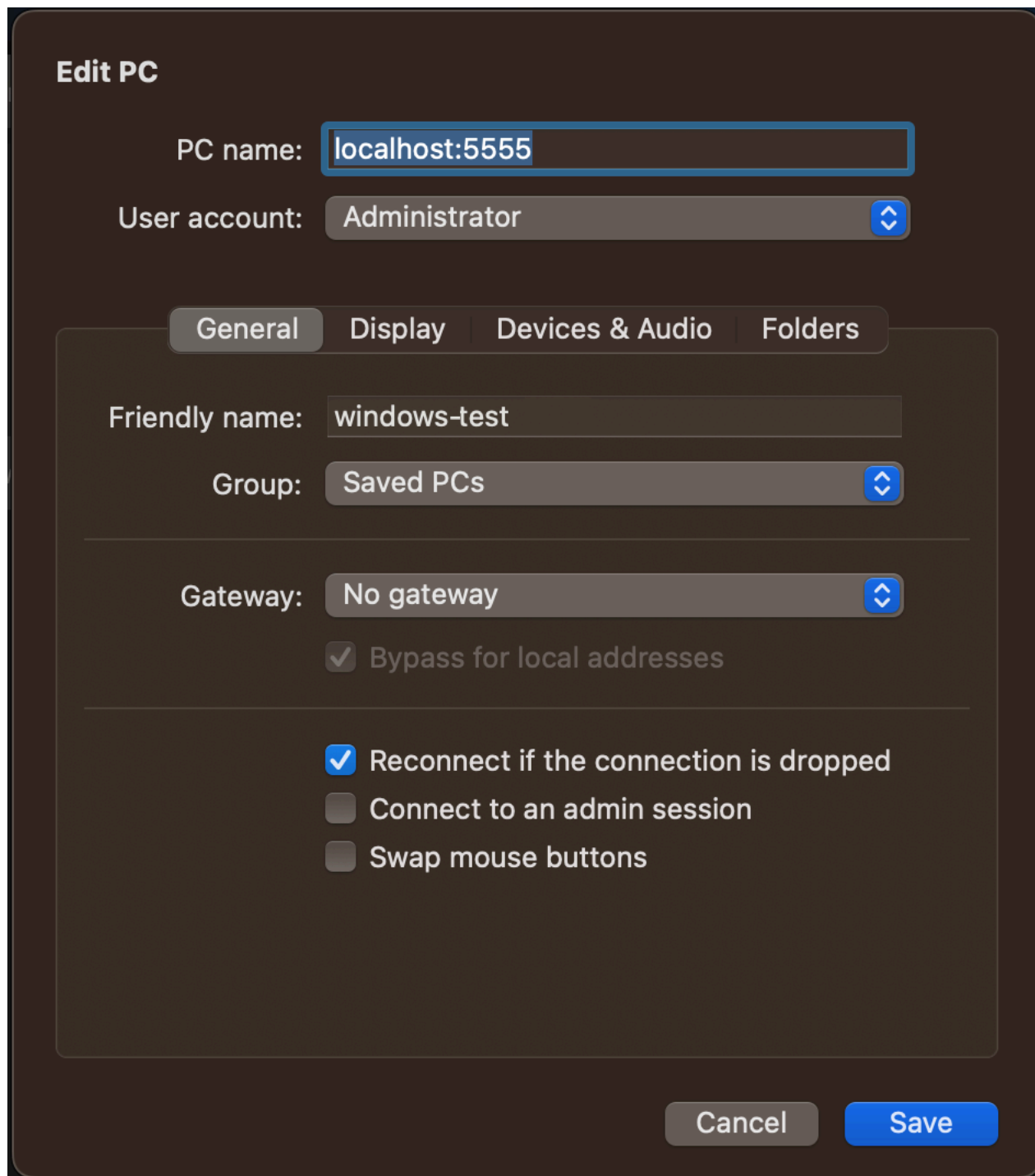
使用 RDP 用戶端連線至您的 Windows 執行個體。

1. 完成[使用 RDP Connect 到您的 Windows 執行個體](#)中的步驟 1 — 8。在步驟 8 下載 RDP 桌面檔案之後，您會收到「無法連線」訊息，這是預期的訊息，因為您的執行個體沒有公用 IP 位址。
2. 執行下列命令，為執行個體所在的 VPC 建立私有通道。--remote-port 必須為 3389，因為 RDP 預設會使用連接埠 3389。

```
aws ec2-instance-connect open-tunnel \
 --instance-id i-0123456789example \
 --remote-port 3389 \
 --local-port any-port
```

3. 在下載資料夾中，尋找您下載的 RDP 桌面檔案，然後將其拖曳至 RDP 用戶端視窗。
4. 以滑鼠右鍵按一下 RDP 桌面檔案，並選擇編輯。
5. 在 [編輯電腦] 視窗中，針對電腦名稱 (要連線的執行個體) 輸入 `localhost:local-port`，其中 `local-port` 使用與您在步驟 2 中指定的相同值，然後選擇 [儲存]。

請注意，編輯電腦視窗的以下螢幕截取畫面來自 Mac 中的 Microsoft 遠端桌面。如果使用 Windows 用戶端，則該視窗可能會有所不同。



6. 在 RDP 用戶端中，以滑鼠右鍵按一下電腦 (您剛設定)，然後選擇連線以連線至您的執行個體。
7. 出現提示時，輸入管理員帳戶的解密密碼。

## 疑難排解

請使用以下資訊來協助您診斷並修正使用 EC2 Instance Connect Endpoint 連線執行個體時可能遇到的問題。

### 無法連線至執行個體

以下是您可能無法連線至執行個體的常見原因。

- **安全群組**：檢查指派給 EC2 Instance Connect Endpoint 和執行個體的安全群組。有關所需安全群組規則的詳細資訊，請參閱 [EC2 Instance Connect Endpoint 的安全群組](#)。
- **執行個體狀態**：驗證您的執行個體處於 running 狀態。
- **金鑰對**：如果用來連線的命令需要私有金鑰，請確認執行個體具有公有金鑰，且您擁有對應的私有金鑰。
- **IAM 許可**：確認您具有所需的 IAM 許可。如需詳細資訊，請參閱 [授予使用 EC2 執行個體 Connect 端點的許可](#)。

如需 Linux 執行個體的詳細疑難排解秘訣，請參閱[連線到 Linux 執行個體的疑難排解](#)。如需 Windows 執行個體的疑難排解提示，請參閱[the section called “連線至 Windows 執行個體”](#)。

### ErrorCode: AccessDeniedException

如果您收到 AccessDeniedException 錯誤，且 IAM 政策中已指定 maxTunnelDuration 條件，則請務必在連線至執行個體時指定 --max-tunnel-duration 參數。如需有關此參數的詳細資訊，請參閱《AWS CLI 命令參考》中的 [open-tunnel](#)。

## 記錄透過 EC2 Instance Connect Endpoint 建立的連線

您可以使用 AWS CloudTrail 日誌記錄資源操作和稽核透過 EC2 執行個體 Connect 端點建立的連線。

如需 AWS CloudTrail 搭配 Amazon EC2 搭配使用的詳細資訊，請參閱[使用記錄 Amazon EC2 API 呼叫 AWS CloudTrail](#)。

### 記錄 EC2 執行個體使用 Connect 端點 API 呼叫 AWS CloudTrail

EC2 執行個體 Connect 端點資源作業會記錄 CloudTrail 為管理事件。進行下列 API 呼叫時，活動會記錄為 CloudTrail 事件記錄中的事件：

- CreateInstanceConnectEndpoint
- DescribeInstanceConnectEndpoints

- DeleteInstanceConnectEndpoint

您可以查看，搜索和下載最近的事件在您的 AWS 帳戶。如需詳細資訊，請參閱AWS CloudTrail 使用指南中的[檢視具有 CloudTrail 事件歷程記錄](#)的事件。

使用 AWS CloudTrail 來稽核使用 EC2 Instance Connect Endpoint 連線至執行個體的使用者

透過 EC2 執行個體連線端點對執行個體的連線嘗試會記錄 CloudTrail 在事件歷史記錄中。透過 EC2 執行個體連線端點啟動與執行個體的連線時，連線會記錄為與OpenTunnel. CloudTrail eventName

您可以建立將 CloudTrail 事件路由到目標的 Amazon EventBridge 規則。如需詳細資訊，請參閱[Amazon EventBridge 使用者指南](#)。

以下是已登入之OpenTunnel管理事件的範例 CloudTrail。

```
{
 "eventVersion": "1.08",
 "userIdentity": {
 "type": "IAMUser",
 "principalId": "ABCDEFGONGNOM00CB6XYTQEXAMPLE",
 "arn": "arn:aws:iam::1234567890120:user/IAM-friendly-name",
 "accountId": "123456789012",
 "accessKeyId": "ABCDEFGUKZHNAW40SN2AEXAMPLE",
 "userName": "IAM-friendly-name"
 },
 "eventTime": "2023-04-11T23:50:40Z",
 "eventSource": "ec2-instance-connect.amazonaws.com",
 "eventName": "OpenTunnel",
 "awsRegion": "us-east-1",
 "sourceIPAddress": "1.2.3.4",
 "userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
 "requestParameters": {
 "instanceConnectEndpointId": "eici-0123456789EXAMPLE",
 "maxTunnelDuration": "3600",
 "remotePort": "22",
 "privateIpAddress": "10.0.1.1"
 },
 "responseElements": null,
 "requestID": "98deb2c6-3b3a-437c-a680-03c4207b6650",
 "eventID": "bbba272c-8777-43ad-91f6-c4ab1c7f96fd",
 "readOnly": false,
 "resources": [{
 "accountId": "123456789012",
```

```
 "type": "AWS::EC2::InstanceConnectEndpoint",
 "ARN": "arn:aws:ec2:us-east-1:123456789012:instance-connect-endpoint/
eici-0123456789EXAMPLE"
 }],
 "eventType": "AwsApiCall",
 "managementEvent": true,
 "recipientAccountId": "123456789012",
 "eventCategory": "Management"
}
```

## 刪除 EC2 執行個體 Connect 端點

完成 EC2 執行個體 Connect 端點後，您可以將其刪除。

您必須具有必要的 IAM 許可才能建立 EC2 Instance Connect Endpoint。如需詳細資訊，請參閱 [建立、描述和刪除 EC2 執行個體 Connect 端點的許可](#)。

當您使用主控台刪除 EC2 執行個體 Connect 端點時，它會進入「刪除」狀態。如果刪除成功，刪除的端點將不再顯示。如果刪除失敗，狀態為delete-failed，狀態訊息會提供失敗原因。

當您使用刪除 EC2 執行個體 Connect 端點時 AWS CLI，它會進入delete-in-progress狀態。如果刪除成功，它將進入delete-complete狀態。如果刪除失敗，則狀態為delete-failed並StateMessage提供失敗原因。

### Console

#### 刪除 EC2 Instance Connect Endpoint

1. 前往 <https://console.aws.amazon.com/vpc/> 開啟 Amazon VPC 主控台。
2. 在左側導覽窗格中選擇 Endpoints (端點)。
3. 選取端點。
4. 選擇 Actions (動作)、Delete VPC endpoints (刪除 VPC 端點)。
5. 出現確認提示時，請按一下 **delete**。
6. 選擇刪除。

### AWS CLI

#### 刪除 EC2 Instance Connect Endpoint

使用命 [delete-instance-connect-endpoints](#) AWS CLI 令並指定要刪除的 EC2 執行個體 Connect 端點的 ID。

```
aws ec2 delete-instance-connect-endpoint --instance-connect-endpoint-id eice-03f5e49b83924bbc7
```

### 範例輸出

```
{
 "InstanceConnectEndpoint": {
 "OwnerId": "111111111111",
 "InstanceConnectEndpointId": "eice-0123456789example",
 "InstanceConnectEndpointArn": "arn:aws:ec2:us-east-1:111111111111:instance-connect-endpoint/eice-0123456789example",
 "State": "delete-in-progress",
 "StateMessage": "",
 "NetworkInterfaceIds": [],
 "VpcId": "vpc-0123abcd",
 "AvailabilityZone": "us-east-1d",
 "CreatedAt": "2023-02-07T12:05:37+00:00",
 "SubnetId": "subnet-0123abcd"
 }
}
```

## EC2 Instance Connect Endpoint 的服務連結角色

Amazon EC2 使用 AWS Identity and Access Management (IAM) [服務連結角色](#)。服務連結角色是直接連結至 Amazon EC2 的一種特殊 IAM 角色類型。服務連結角色由 Amazon EC2 預先定義，並包含所有必要的許可，以便 Amazon EC2 可以代表您呼叫其他 AWS 服務角色。如需詳細資訊，請參閱《IAM 使用者指南》中的 [使用服務連結角色](#)。

### EC2 執行個體連線端點的服務連結角色許可

Amazon EC2 用 `AWSServiceRoleForEC2InstanceConnect` 於在您的帳戶中建立和管理 EC2 執行個體 Connect 端點所需的網路界面。

服 `AWSServiceRoleForEC2InstanceConnect` 務連結角色會信任下列服務擔任該角色：

- `ec2-instance-connect.amazonaws.com`

AWSServiceRoleForEC2InstanceConnect服務連結角色使用受管政策 EC2 InstanceConnect 端點。若要檢視此政策的許可，請參閱AWS 受管政策參考中的 [EC2 InstanceConnect 端點](#)。

您必須設定許可，IAM 實體 (如使用者、群組或角色) 才可建立、編輯或刪除服務連結角色。如需詳細資訊，請參閱 IAM 使用者指南中的 [服務連結角色許可](#)。

為 EC2 執行個體 Connect 端點建立服務連結角色

您不需要手動建立 服務連結角色。當您建立 EC2 執行個體 Connect 端點時，Amazon EC2 會為您建立服務連結角色。

編輯 EC2 執行個體連線端點的服務連結角色

EC2 執行個體 Connect 端點不允許您編輯AWSServiceRoleForEC2InstanceConnect服務連結角色。

刪除 EC2 執行個體連線端點的服務連結角色

如果您不再需要使用 EC2 執行個體 Connect 端點，建議您刪除AWSServiceRoleForEC2InstanceConnect服務連結角色。

您必須先刪除所有 EC2 執行個體 Connect 端點資源，才能刪除服務連結角色。

若要刪除服務連結角色，請參閱《IAM 使用者指南》中的 [刪除服務連結角色](#)。

EC2 執行個體 Connect 端點的配額

您的每項 AWS 服務都 AWS 帳戶 有預設配額 (先前稱為限制)。除非另有說明，否則每個配額都是區域特定規定。

您 AWS 帳戶 具有以下與 EC2 實例 Connect 端點相關的配額。

描述	配額
每個 EC2 執行個體 Connect 端點的最大數 AWS 帳戶 量 AWS 區域	5
每個 VPC 的 EC2 Instance Connect Endpoints 數目上限	1
每個子網路的 EC2 Instance Connect Endpoints 數目上限	1

描述	配額
每個 EC2 執行個體連線端點的同時 Connect 線數目上限	20

## 將 EC2 執行個體連線至 AWS 資源

啟動執行個體之後，您可以將其連線到一或多個 AWS 資源。

此部分說明如何將 Amazon EC2 執行個體自動連線至 Amazon RDS 資料庫。

### 自動將 EC2 執行個體連線至 RDS 資料庫

您可以使用 Amazon EC2 主控台內的自動連線功能，將一個或多個 EC2 執行個體快速連線至 RDS 資料庫，以允許它們之間的流量。

如需詳細資訊，請參閱 [連線是如何自動設定的](#)。如需詳細的逐步解說，請參閱 [教學課程：將 Amazon EC2 執行個體連線至 Amazon RDS 資料庫](#)，其中包括連線 EC2 執行個體和 RDS 資料庫的其他方法。

#### 主題

- [成本](#)
- [必要條件](#)
- [自動連線執行個體和資料庫](#)
- [連線是如何自動設定的](#)

#### 成本

雖然自動將 EC2 執行個體連線到 RDS 資料庫無需付費，但是您需要支付基礎服務的費用。如果您的 EC2 執行個體和 RDS 資料庫位於不同的可用區域，則需支付資料傳輸費用。如需有關資料傳輸費用的資訊，請參閱「Amazon EC2 隨需定價」頁面中的 [資料傳輸](#)。

#### 必要條件

在自動將 EC2 執行個體連線至 RDS 資料庫之前，請檢查下列各項：

- EC2 執行個體必須處於 Running (執行中) 狀態。如果 EC2 執行個體處於另一個狀態，則無法連線。
- EC2 執行個體和 RDS 資料庫必須位於相同的虛擬私有雲端 (VPC) 中。如果 EC2 執行個體和 RDS 資料庫位於不同的 VPC 中，則不支援自動連線功能。



## 自動連線執行個體和資料庫

您可以在啟動執行個體後立即自動將 EC2 執行個體連線至 RDS 資料庫，或稍後進行連接。

### 啟動後立即自動連線

請遵循下列步驟，在啟動 EC2 執行個體後，立即自動將 EC2 執行個體連線至 RDS 資料庫。

若要檢視這些步驟的動畫，請參閱 [檢視動畫：自動將新啟動的 EC2 執行個體連線至 RDS 資料庫](#)。

若要使用 EC2 主控台將新啟動的 EC2 執行個體自動連線至 RDS 資料庫

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 從主控台儀表板選擇 Launch instance (啟動執行個體)，然後遵循步驟來[啟動執行個體](#)。
3. 在執行個體啟動確認頁面中，選擇 Connect an RDS database (連線 RDS 資料庫)。
4. 在 Connect RDS Database (連線 RDS 資料庫) 對話方塊中，請執行下列動作：
  - a. 對於 Database role (資料庫角色)，請選擇 Cluster (叢集) 或 Instance (執行個體)。
  - b. 對於 RDS database (RDS 資料庫)，請選取要連線的資料庫。

#### Note

EC2 執行個體和 RDS 資料庫必須位於相同的 VPC 中，才能彼此連線。

- c. 選擇連線。

## 檢視動畫：自動將新啟動的 EC2 執行個體連線至 RDS 資料庫

The screenshot displays the Amazon EC2 console interface. On the left is a navigation sidebar with categories like EC2 Dashboard, Instances, Images, Elastic Block Store, and Network & Security. The main content area is titled 'Resources' and shows a summary of EC2 resources in the Europe (Stockholm) Region, including 1 running instance, 1 key pair, 9 security groups, and 2 volumes. Below this is a 'Launch instance' section with a 'Launch instance' button and a 'Migrate a server' link. To the right, the 'Service health' section indicates that the EC2 service is operating normally in the Europe (Stockholm) region. Below that, a 'Zones' table lists three availability zones: eu-north-1a, eu-north-1b, and eu-north-1c, each with its corresponding Zone ID.

Zone name	Zone ID
eu-north-1a	eun1-az1
eu-north-1b	eun1-az2
eu-north-1c	eun1-az3

### 自動連線現有執行個體

使用下列步驟自動將現有 EC2 執行個體連線至 RDS 資料庫。

若要檢視這些步驟的動畫，請參閱 [檢視動畫：自動將現有 EC2 執行個體連線至 RDS 資料庫](#)。

若要使用 EC2 主控台自動將現有 EC2 執行個體連線至 RDS 資料庫

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取一個或多個要連線至 RDS 資料庫的 EC2 執行個體，然後選取 Actions (動作)、Networking (聯網)、Connect RDS database (連線 RDS 資料庫)。

如果無法使用 Connect RDS database (連線 RDS 資料庫)，請檢查 EC2 執行個體是否處於 Running (執行中) 狀態，以及其是否處於相同的 VPC。

4. 在 Connect RDS Database (連線 RDS 資料庫) 對話方塊中，請執行下列動作：

- a. 對於 Database role (資料庫角色)，請選擇 Cluster (叢集) 或 Instance (執行個體)。
- b. 對於 RDS database (RDS 資料庫)，請選取要連線的資料庫。

**Note**

EC2 執行個體和 RDS 資料庫必須位於相同的 VPC 中，才能彼此連線。

- c. 選擇連線。

**檢視動畫：自動將現有 EC2 執行個體連線至 RDS 資料庫**

The screenshot displays the Amazon EC2 console interface. On the left is a navigation sidebar with categories like EC2 Dashboard, Instances, Images, and Network & Security. The main content area is divided into several panels:

- Resources:** A table showing EC2 resource usage in the Europe (Stockholm) Region. It includes metrics for Instances (running), Dedicated Hosts, Elastic IPs, Key pairs, Load balancers, Placement groups, Security groups, Snapshots, and Volumes.
- Launch instance:** A section with a 'Launch instance' button and a 'Migrate a server' link. A note indicates that instances will launch in the Europe (Stockholm) Region.
- Scheduled events:** A section showing 'No scheduled events' for the Europe (Stockholm) region.
- Migrate a server:** A section with a link to 'Use AWS Application Migration Service to simplify and expedite migration'.
- Service health:** A section showing the status of the service in the Europe (Stockholm) region, indicating it is 'operating normally'. It also lists available zones (eu-north-1a, eu-north-1b, eu-north-1c).
- Account attributes:** A section on the right showing supported platforms, VPC, Default VPC, and other account settings.
- Explore AWS:** A section on the right with promotional cards for Amazon GuardDuty Malware Protection, AWS Graviton2, and T4g instances.

如需有關如何使用 Amazon RDS 主控台將 EC2 執行個體自動連線至 RDS 資料庫的資訊，請參閱《Amazon RDS 使用者指南》中的[設定與 EC2 執行個體的自動網路連線](#)。

連線是如何自動設定的

當您使用 EC2 主控台自動設定 EC2 執行個體和 RDS 資料庫之間的連線以允許它們之間的流量時，連線會由[安全群組](#)設定。

系統會自動建立安全群組並新增至 EC2 執行個體和 RDS 資料庫，如下所示：

- Amazon EC2 建立了一個名為 `ec2-rds-x` 的安全群組，並將其新增至 EC2 執行個體。它有一個傳出規則，透過將 `rds-ec2-x` (資料庫安全群組) 指定為其目的地，從而允許流量進入資料庫。

- Amazon RDS 建立了一個名為 rds-ec2-**x** 的安全群組，並將其新增至資料庫。它有一個傳入規則，透過將 ec2-rds-**x** (EC2 執行個體安全群組) 指定為其來源，從而允許來自 EC2 執行個體的流量。

安全群組會彼此參考作為目的地和來源，而且只允許資料庫連接埠上的流量。您可以重複使用這些安全群組，以便任何具有 rds-ec2-**x** 安全群組的資料庫都可以與任何具有 ec2-rds-**x** 安全群組的 EC2 執行個體進行通訊。

安全群組名稱遵循一種模式。對於 Amazon EC2 建立的安全群組，該模式為 ec2-rds-**x**，而對於 Amazon RDS 建立的安全群組，該模式為 rds-ec2-**x**。**x** 是一個數字，每次自動建立新的安全群組時，該數字就會增加 1。

## 教學課程：將 Amazon EC2 執行個體連線至 Amazon RDS 資料庫

### 教學課程目標

本教學課程的目的是了解如何使用 AWS Management Console 來設定 Amazon EC2 執行個體和 Amazon RDS 資料庫之間的連線。

設定連線有不同的選項。在本教學課程中，我們將探討以下三個選項：

- [選項 1：使用 EC2 主控台自動將 EC2 執行個體連線至 RDS 資料庫](#)

使用 EC2 主控台自動連線功能，自動設定 EC2 執行個體和 RDS 資料庫之間的連線，以允許 EC2 執行個體和 RDS 資料庫之間的流量。

- [選項 2：使用 RDS 主控台自動將 EC2 執行個體連線至 RDS 資料庫](#)

使用 RDS 主控台自動連線功能，自動設定 EC2 執行個體和 RDS 資料庫之間的連線，以允許 EC2 執行個體和 RDS 資料庫之間的流量。

- [選項 3：透過模仿自動連線功能，手動將 EC2 執行個體連線至 RDS 資料庫](#)

透過手動設定和指派安全群組來複製選項 1 和選項 2 中自動連線功能自動建立的組態，以設定 EC2 執行個體與 RDS 資料庫之間的連線。

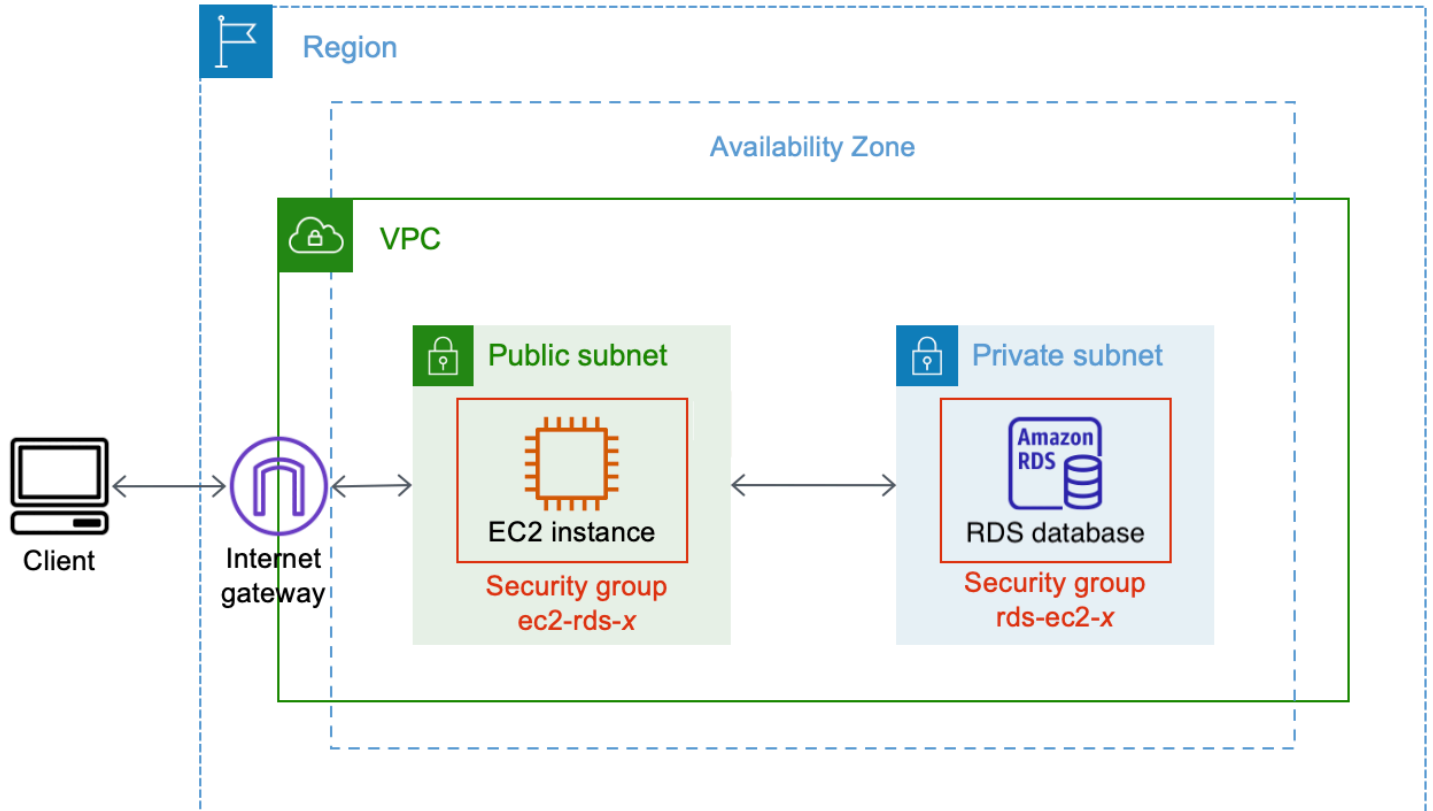
### Context

至於為什麼要在 EC2 執行個體和 RDS 資料庫之間設定連線，讓我們一起思考下面這種情況：您的網站向使用者提供表單以供填寫。您需要在資料庫中擷取表單資料。您可以在已被設定為 Web 伺服器的 EC2 執行個體上託管您的網站，並且可以在 RDS 資料庫中擷取表單資料。EC2 執行個體和 RDS 資料

庫必須彼此相連，以便表單資料可以從 EC2 執行個體傳送到 RDS 資料庫。本教學課程會說明如何設定該連線。請注意，這只是連線 EC2 執行個體和 RDS 資料庫的其中一個使用案例範例。

## 架構

下圖展示了所建立的資源，以及完成本教學課程中的所有步驟後產生的架構組態。



此圖展示了您將建立的下列資源：

- 您將在相同 AWS 區域的 VPC 和可用區域中建立 EC2 執行個體和 RDS 資料庫。
- 您將在公有子網路中建立 EC2 執行個體。
- 您將在私有子網路中建立 RDS 資料庫。

當您使用 RDS 主控台建立 RDS 資料庫並自動連線 EC2 執行個體時，會自動選取資料庫的 VPC、資料庫子網路群組和公開存取設定。在與 EC2 執行個體相同之 VPC 內的私有子網路中自動建立 RDS 資料庫。

- 網際網路使用者可以透過網際網路閘道使用 SSH 或 HTTP/HTTPS 連線至 EC2 執行個體。
- 網際網路使用者無法直接連線至 RDS 資料庫；只有 EC2 執行個體可連線至 RDS 資料庫。
- 當您使用自動連線功能允許 EC2 執行個體和 RDS 資料庫之間的流量時，會自動建立並新增下列安全群組：

- 建立安全群組 `ec2-rds-x` 並新增至 EC2 執行個體。它有一個傳出規則，該規則引用 `rds-ec2-x` 安全群組作為其目的地。這允許來自 EC2 執行個體的流量透過 `rds-ec2-x` 安全群組到達 RDS 資料庫。
- 建立安全群組 `rds-ec2-x` 並新增至 RDS 資料庫。它有一個傳入規則，該規則引用 `ec2-rds-x` 安全群組作為其來源。這允許來自 EC2 執行個體的流量透過 `ec2-rds-x` 安全群組到達 RDS 資料庫。

透過使用單獨的安全群組 (一個用於 EC2 執行個體，一個用於 RDS 資料庫)，您可以更好地控制執行個體和資料庫的安全性。如果您要在執行個體和資料庫上使用相同的安全群組，然後修改安全群組以適應資料庫，則修改會同時影響執行個體和資料庫。換句話說，如果您使用一個安全群組，您可能會無意中修改資源 (執行個體或資料庫) 的安全性，因為您忘記了安全群組已連接至該資源。

自動建立的安全群組也會遵守最低權限，因為其只允許透過建立特定於工作負載的安全群組對，在資料庫連接埠上進行此工作負載的相互連線。

## 考量事項

完成本教學課程中的任務後，請考量以下內容：

- 兩個主控台 - 在本教學課程中，您將使用下列兩個主控台：
  - Amazon EC2 主控台 – 您將使用 EC2 主控台啟動執行個體、自動將 EC2 執行個體連線到 RDS 資料庫，以及使用手動選項，透過建立安全群組來設定連線。
  - Amazon RDS 主控台 – 您將使用 RDS 主控台建立 RDS 資料庫，並自動將 EC2 執行個體連線到 RDS 資料庫。
- 一個 VPC - 若要使用自動連線功能，EC2 執行個體和 RDS 資料庫必須位於相同的 VPC 中。

如果您要手動設定 EC2 執行個體和 RDS 資料庫之間的連線，則可以在一個 VPC 中啟動 EC2 執行個體，並在另一個 VPC 中啟動 RDS 資料庫；不過，您需要設定其他路由和 VPC 組態。這是本教學課程中涵蓋的情況。

- 第一 AWS 區域 — EC2 執行個體和 RDS 資料庫必須位於同一個區域。
- 兩個安全群組 - EC2 執行個體與 RDS 資料庫之間的連線由兩個安全群組設定 - 針對 EC2 執行個體的安全群組，以及針對 RDS 資料庫的安全群組。

當您使用 EC2 主控台或 RDS 主控台自動連線功能來設定連線 (本教學課程的選項 1 和選項 2) 時，會自動建立安全群組並指派給 EC2 執行個體和 RDS 資料庫。

如果您不使用自動連線功能，則需要手動建立並指派安全群組。您可以在本教學課程的選項 3 中執行此操作。

## 教學課程所需時間

30 分鐘

您可以一次性完成整個教學課程，也可以一次完成一個任務。

## 成本

完成此教學課程後，您可能會為所建立的 AWS 資源產生費用。

您可以在[免費方案](#)下使用 Amazon EC2，前提是您的 AWS 帳戶年齡少於 12 個月，並且您可以根據免費方案要求設定資源。

如果您的 EC2 執行個體和 RDS 資料庫位於不同的可用區域，則需支付資料傳輸費用。若要避免產生這些費用，EC2 執行個體和 RDS 資料庫必須位於相同的可用區域。如需有關資料傳輸費用的資訊，請參閱「Amazon EC2 隨需定價」頁面中的[資料傳輸](#)。

若要避免在完成教學課程後產生費用，請務必刪除不再需要的資源。如需刪除資源的步驟，請參閱[清除](#)。

選項 1：使用 EC2 主控台自動將 EC2 執行個體連線至 RDS 資料庫

## 目標

選項 1 的目的是探索 EC2 主控台自動連線功能，其可自動設定 EC2 執行個體和 RDS 資料庫之間的連線，以允許從 EC2 執行個體到 RDS 資料庫的流量。在選項 3 中，您將了解如何手動設定連線。

## 開始之前

完成此教學課程需要以下項目：

- 與 EC2 執行個體處於相同 VPC 的 RDS 資料庫。您可以使用現有的 RDS 資料庫或遵循任務 1 中的步驟來建立新的 RDS 資料庫。
- 與 RDS 資料庫處於相同 VPC 的 EC2 執行個體。您可以使用現有的 EC2 執行個體或遵循任務 2 中的步驟來建立新的 EC2 執行個體。
- 呼叫以下操作的許可：
  - `ec2:AssociateRouteTable`
  - `ec2:AuthorizeSecurityGroupEgress`
  - `ec2:CreateRouteTable`

- `ec2:CreateSecurityGroup`
- `ec2:CreateSubnet`
- `ec2:DescribeInstances`
- `ec2:DescribeNetworkInterfaces`
- `ec2:DescribeRouteTables`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeSubnets`
- `ec2:ModifyNetworkInterfaceAttribute`
- `ec2:RevokeSecurityGroupEgress`

### 完成選項 1 的任務

- [任務 1：建立 RDS 資料庫 - 選用](#)
- [任務 2：啟動 EC2 執行個體 - 選用](#)
- [任務 3：自動將 EC2 執行個體連線到 RDS 資料庫](#)
- [任務 4：驗證連線組態](#)

### 任務 1：建立 RDS 資料庫 - 選用

#### Note

建立 Amazon RDS 資料庫並不是本教學課程的重點。如果您已有 RDS 資料庫，並且您想在本教學課程中使用該資料庫，則可跳過此任務。

### 任務目標

此任務的目標是建立 RDS 資料庫，以便您可以完成任務 3，您可以在其中設定 EC2 執行個體和 RDS 資料庫之間的連線。如果您有 RDS 資料庫可使用，則可跳過此任務。

#### Important

如果您使用現有的 RDS 資料庫，請確保其與 EC2 執行個體位於相同的 VPC 中，以便您可以使用自動連線功能。



## 建立 RDS 資料庫的步驟

使用下列步驟建立 RDS 資料庫。

若要檢視這些步驟的動畫，請參閱 [檢視動畫：建立 RDS 資料庫](#)。

### RDS 資料庫組態

此任務中的步驟會設定 RDS 資料庫，如下所示：

- 引擎類型：MySQL
- 範本：免費方案
- 資料庫執行個體識別符：**tutorial-database-1**
- 資料庫執行個體類別：db.t3.micro

#### Important

在生產環境中，您應設定資料庫以滿足您的特定需求。

### 若要建立 MySQL RDS 資料庫

1. 前往 <https://console.aws.amazon.com/rds/>，開啟 Amazon RDS 主控台。
2. 從區域選取器 (位於右上角) 中選取 AWS 區域。資料庫和 EC2 執行個體必須位於相同區域，才能使用 EC2 主控台內的自動連線功能。
3. 在儀表板中，選擇 Create database (建立資料庫)。
4. 在 Choose a database creation method (選取資料庫建立方法) 中，確認選取 Standard Create (標準建立)。如果您選取 Easy create (輕鬆建立)，則無法使用 VPC 選取器。您必須確保資料庫與 EC2 執行個體位於相同的 VPC 中，才能使用 EC2 主控台內的自動連線功能。
5. 在 Engine options (引擎選項) 中，針對 Engine type (引擎類型) 選擇 MySQL。
6. 在 Templates (範本) 中，選擇符合您需求的範例範本。在本教學課程中，選擇 Free tier (免費方案) 免費建立資料庫。但請注意，只有在您的帳戶使用未滿 12 個月時，才能使用免費方案。其他限制適用。在 Free tier (免費方案) 方塊中選擇 Info (資訊) 連結，可閱讀更多資訊。
7. 在 Settings (設定) 下，執行下列動作：
  - a. 在 DB instance identifier (資料庫執行個體識別符) 中，輸入資料庫的名稱。針對本教學，輸入 **tutorial-database-1**。

- b. 對於 Master username (主要使用者名稱)，請保留預設名稱，即 **admin**。
  - c. 對於 Master password (主要密碼)，請輸入您在本教學課程中可以記住的密碼，然後在 Confirm password (確認密碼) 中再次輸入密碼。
8. 在 Instance configuration (執行個體組態) 中，將 DB instance class (資料庫執行個體類別) 保留為預設值，即 db.t3.micro。如果您的帳戶使用未滿 12 個月，則可以免費使用此資料庫類別。其他限制適用。如需詳細資訊，請參閱 [AWS 免費方案](#)。
  9. 在 Connectivity (連線) 中，對於 Compute resource (運算資源)，選擇 Don't connect to an EC2 compute resource (不連線至 EC2 運算資源)，因為稍後會在任務 3 中連線 EC2 執行個體和 RDS 資料庫。

(稍後，在本教學課程的選項 2 中，您將透過選擇 Connect to an EC2 compute resource (連線至 EC2 運算資源) 來嘗試 RDS 主控台內的自動連線功能。)

10. 對於 Virtual private cloud (VPC) (虛擬私有雲端 (VPC))，選擇 VPC。VPC 必須具備資料庫子網路群組。若要使用自動連線功能，EC2 執行個體和 RDS 資料庫必須位於相同的 VPC 中。
11. 請保留本頁面中其他欄位的所有預設值。
12. 選擇建立資料庫。

在 Databases (資料庫) 畫面中，新資料庫的 Status (狀態) 為 Creating (正在建立)，直到資料庫準備就緒可供使用為止。狀態變更為 Available (可用) 時，您便可連線至資料庫。視資料庫類別和儲存體數量而定，可能需要最多 20 分鐘的時間，新資料庫才會可用。

## 檢視動畫：建立 RDS 資料庫

The screenshot shows the Amazon RDS console dashboard. On the left is a navigation menu with options like Dashboard, Databases, Performance insights, Snapshots, Automated backups, Reserved instances, Proxies, Subnet groups, Parameter groups, Option groups, Custom engine versions, Events, Event subscriptions, and Certificate update. The main content area features a top banner for Multi-AZ deployment, a 'Resources' section listing usage for DB Instances (3/40), DB Clusters (1/40), Reserved instances (0/40), Snapshots (1), Manual (DB Cluster 0/100, DB Instance 0/100), Automated (DB Cluster 1, DB Instance 0), Recent events (5), and Event subscriptions (0/20). A 'Create database' button is highlighted with a mouse cursor. Below the resources section is another 'Create database' heading and a partial sentence: 'Amazon Relational Database Service (RDS) makes it easy to set up, operate, and scale a relational database i'.

您現在已準備好使用 [任務 2：啟動 EC2 執行個體 - 選用](#)。

### 任務 2：啟動 EC2 執行個體 - 選用

#### Note

啟動執行個體並不是本教學課程的重點。如果您已有 Amazon EC2 執行個體，並且您想在本教學課程中使用該執行個體，則可跳過此任務。

## 任務目標

此任務的目標是啟動 EC2 執行個體，以便您可以完成任務 3，您可以在其中設定 EC2 執行個體和 Amazon RDS 資料庫之間的連線。如果您有 EC2 執行個體可使用，則可跳過此任務。

### Important

如果您使用現有的 EC2 執行個體，則請確保其與 RDS 資料庫位於相同的 VPC 中，以便您可以使用自動連線功能。

啟動 EC2 執行個體的步驟：

請遵循下列步驟來啟動本教學課程的 EC2 執行個體。

若要檢視這些步驟的動畫，請參閱 [檢視動畫：啟動 EC2 執行個體](#)。

## EC2 執行個體組態

此任務中的步驟會設定 EC2 執行個體，如下所示：

- 執行個體名稱：**tutorial-instance-1**
- AMI：Amazon Linux 2
- 執行個體類型：t2.micro
- 自動指派公有 IP：已啟用
- 具有以下三個規則的安全群組：
  - 允許來自您 IP 地址的 SSH
  - 允許來自任何地方的 HTTPS 流量
  - 允許來自任何地方的 HTTP 流量


### Important

在生產環境中，您應設定執行個體以滿足您的特定需求。

## 啟動 EC2 執行個體

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。

2. 從區域選取器 (位於右上角) 中選取 AWS 區域。執行個體和 RDS 資料庫必須位於相同區域，才能使用 EC2 主控台內的自動連線功能。
3. 在 EC2 Dashboard (EC2 儀表板) 中，選擇 Launch Instance (啟動執行個體)。
4. 在 Name and tags (名稱和標籤) 下，針對 Name (名稱)，輸入用於識別執行個體的名稱。針對本教學課程，將執行個體命名為 **tutorial-instance-1**。雖然執行個體名稱不是強制性的，但是當您在 EC2 主控台中選取執行個體時，名稱可協助您輕鬆識別其。
5. 在 Application and OS Images (應用程式和作業系統映像) 中，選擇符合您 Web 伺服器需求的 AMI。本教學課程使用 Amazon Linux 2。
6. 在 Instance type (執行個體類型) 中，針對 Instance type (執行個體類型)，選取符合您 Web 伺服器需求的執行個體類型。本教學課程使用的是 t2.micro。

 Note

您可以在[免費方t2.micro案](#)下使用 Amazon EC2，前提是您的 AWS 帳戶年齡少於 12 個月，而且您可以選擇 t2.micro 執行個體類型 (或 t3.micro 在無法使用的區域)。

7. 在 key pair (login) (金鑰對 (登入)) 中，針對 Key pair name (金鑰對名稱) 選擇您的金鑰對。
8. 在 Network settings (網路設定) 中，執行下列操作：
  - a. 對於 Network (網路) 和 Subnet (子網路)，如果您尚未變更預設 VPC 或子網路，則可以保留預設設定。

如果您已變更預設 VPC 或子網路，則請檢查下列項目：

- i. 執行個體和 RDS 資料庫必須位於相同的 VPC 中，才能使用自動連線功能。依預設，您只有一個 VPC。
  - ii. 您要在其中啟動執行個體的 VPC 必須連接網際網路閘道，以便您可以從網際網路存取 Web 伺服器。您的預設 VPC 會與網際網路閘道一起自動設定。
  - iii. 若要確保您的執行個體接收公有 IP 地址，針對 Auto-assign public IP (自動指派公有 IP)，檢查是否已選取 Enable (啟用)。如果選取 Disable (停用)，則請選取 Edit (編輯) (位於 Network Settings (網路設定) 右側)，然後針對 Auto-assign public IP (自動指派公有 IP) 選取 Enable (啟用)。
- b. 若要使用 SSH 連線至執行個體，您需要一個安全群組規則來授權來自您電腦公有 IPv4 地址的 SSH (Linux) 或 RDP (Windows) 流量。根據預設，當您啟動執行個體時，會建立新的安全群組，其中包含允許從任何地方傳入 SSH 流量的規則。

若要確保只有您的 IP 地址可以連線到您的執行個體，請在 Firewall (security groups) (防火牆 (安全群組)) 下，從 Allow SSH traffic from (允許 SSH 流量來源) 核取方塊旁的下拉式清單中選擇 My IP (我的 IP)。

c. 若要允許流量從網際網路傳輸到您的執行個體，請選取下列核取方塊：

- Allow HTTPs traffic from the internet (允許來自網際網路的 HTTPS 流量)
- Allow HTTP traffic from the internet (允許來自網際網路的 HTTP 流量)

9. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。

10. 保持確認頁面開啟。當您自動將執行個體連線至資料庫時，下一個任務會需要它。

如果執行個體無法啟動或狀態立即進入 terminated 而非 running，請參閱[針對執行個體啟動問題進行故障診斷](#)。

如需有關啟動執行個體的詳細資訊，請參閱[使用新的啟動執行個體精靈啟動執行個體](#)。

## 檢視動畫：啟動 EC2 執行個體

The screenshot displays the AWS Management Console interface for the EC2 service in the Europe (Stockholm) region. The left sidebar contains navigation options like EC2 Dashboard, Instances, Images, and Elastic Block Store. The main content area is divided into several sections:

- Resources:** A summary of EC2 resources in use, including 2 running instances, 2 instances, 0 dedicated hosts, 0 elastic IPs, 1 key pair, 0 load balancers, 0 placement groups, 10 security groups, 1 snapshot, and 3 volumes.
- Launch instance:** A section with a prominent orange 'Launch instance' button and a 'Migrate a server' link. A note indicates that instances will launch in the Europe (Stockholm) region.
- Service health:** Shows the region status as 'This service is operating normally'.
- Zones:** A table listing available availability zones: eu-north-1a (eun1-az1), eu-north-1b (eun1-az2), and eu-north-1c (eun1-az3).
- Scheduled events:** Shows 'No scheduled events' for the Europe (Stockholm) region.

您現在已準備好使用 [任務 3：自動將 EC2 執行個體連線到 RDS 資料庫](#)。

## 任務 3：自動將 EC2 執行個體連線到 RDS 資料庫

### 任務目標

此任務的目標是使用 EC2 主控台中的自動連線功能，自動設定 EC2 執行個體與 RDS 資料庫之間的連線。

### 連線 EC2 執行個體和 RDS 資料庫的步驟

使用 EC2 主控台中的自動功能，透過下列步驟連線 EC2 執行個體和 RDS 資料庫。

若要檢視這些步驟的動畫，請參閱 [檢視動畫：自動將新啟動的 EC2 執行個體連線至 RDS 資料庫](#)。

若要使用 EC2 主控台自動將 EC2 執行個體連線至 RDS 資料庫

1. 在執行個體啟動確認頁面 (應該從先前的任務中開啟) 中，選擇 Connect an RDS database (連線 RDS 資料庫)。

如果已關閉確認頁面，則請按照下列步驟操作：

- a. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
- b. 在導覽窗格中，選擇執行個體。
- c. 選取您剛建立的 EC2 執行個體，然後選取 Actions (動作)、Networking (聯網)、Connect RDS database (連線 RDS 資料庫)。

如果無法使用 Connect RDS database (連線 RDS 資料庫)，則請檢查 EC2 執行個體是否處於 Running (執行中) 狀態。

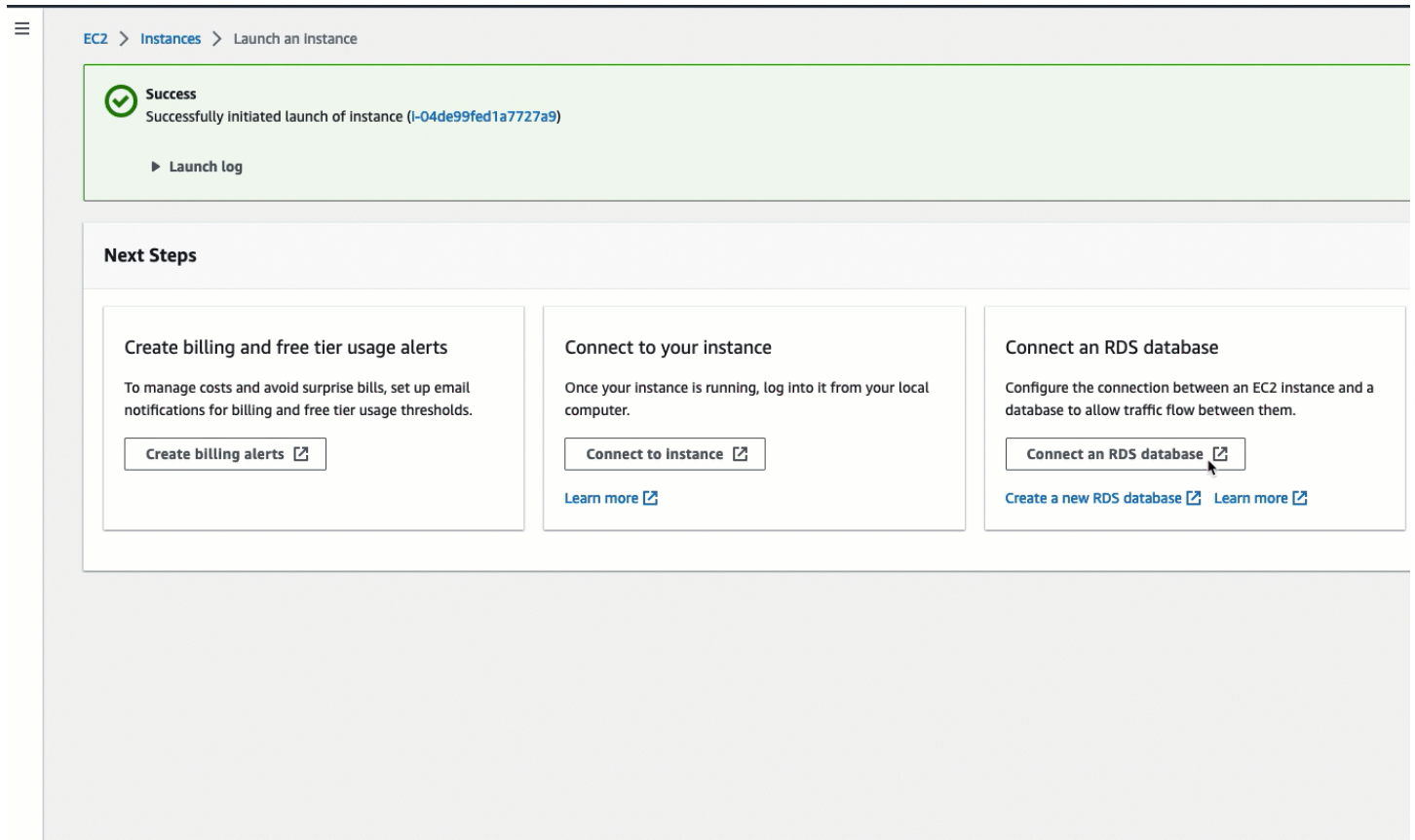
2. 對於 Database role (資料庫角色)，請選擇 Instance (執行個體)。在這種情況下，Instance (執行個體) 是指資料庫執行個體。
3. 對於 RDS database (RDS 資料庫)，請選擇您在任務 1 中建立的 RDS 資料庫。

#### Note

EC2 執行個體和 RDS 資料庫必須位於相同的 VPC 中，才能彼此連線。

4. 選擇連線。

## 檢視動畫：自動將新啟動的 EC2 執行個體連線至 RDS 資料庫



EC2 > Instances > Launch an Instance

**Success**  
Successfully initiated launch of instance (i-04de99fed1a7727a9)

▶ Launch log

### Next Steps

**Create billing and free tier usage alerts**

To manage costs and avoid surprise bills, set up email notifications for billing and free tier usage thresholds.

Create billing alerts [↗](#)

**Connect to your instance**

Once your instance is running, log into it from your local computer.

Connect to instance [↗](#)

[Learn more](#) [↗](#)

**Connect an RDS database**

Configure the connection between an EC2 Instance and a database to allow traffic flow between them.

Connect an RDS database [↗](#)

[Create a new RDS database](#) [↗](#) [Learn more](#) [↗](#)

您現在已準備好使用 [任務 4：驗證連線組態](#)。

### 任務 4：驗證連線組態

#### 任務目標

此任務的目標是確認已建立兩個安全群組並指派給執行個體和資料庫。

當您使用 EC2 主控台內的自動連線功能來設定連線時，會自動建立安全群組並指派給執行個體和資料庫，如下所示：

- 建立安全群組 `rds-ec2-x` 並新增至 RDS 資料庫。它有一個傳入規則，該規則引用 `ec2-rds-x` 安全群組作為其來源。這允許來自 EC2 執行個體的流量透過 `ec2-rds-x` 安全群組到達 RDS 資料庫。
- 建立安全群組 `ec2-rds-x` 並新增至 EC2 執行個體。它有一個傳出規則，該規則引用 `rds-ec2-x` 安全群組作為其目的地。這允許來自 EC2 執行個體的流量透過 `rds-ec2-x` 安全群組到達 RDS 資料庫。

#### 驗證連線組態的步驟

使用以下步驟驗證連線組態。



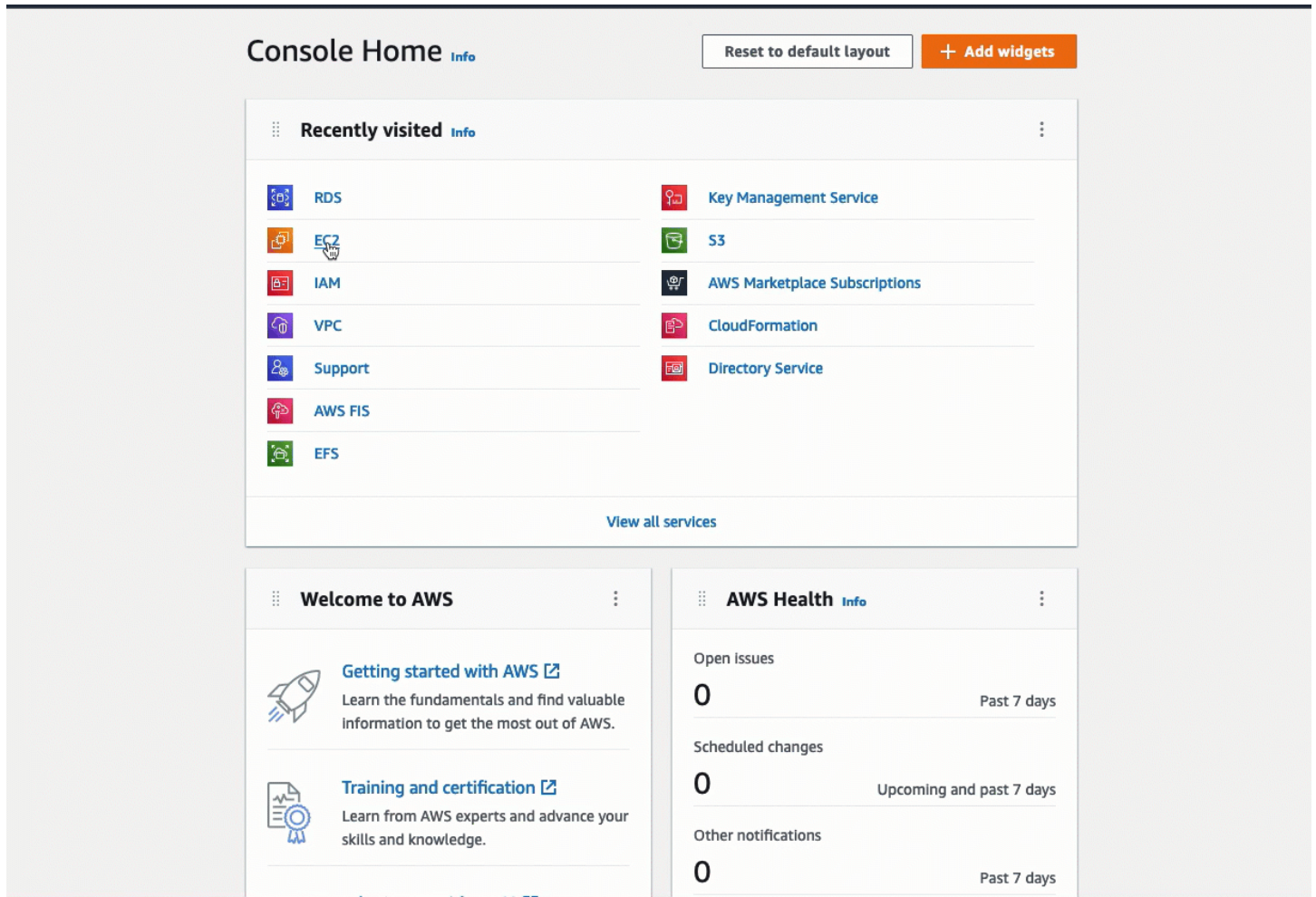
若要檢視這些步驟的動畫，請參閱 [檢視動畫：驗證連線組態](#)。

若要使用主控台驗證連線組態

1. 前往 <https://console.aws.amazon.com/rds/>，開啟 Amazon RDS 主控台。
2. 在導覽頁面中，選擇 Databases (資料庫)。
3. 選擇您為此教學課程建立的 RDS 資料庫。
4. 在 Connectivity & security (連線與安全) 索引標籤的 Security (安全性)、VPC security groups (VPC 安全群組) 中，確認會顯示名為 rds-ec2-**x** 的安全群組。
5. 選擇 rds-ec2-**x** 安全群組。EC2 主控台內的 Security Groups (安全群組) 畫面開啟。
6. 選擇 rds-ec2-**x** 安全群組，將其開啟。
7. 選擇 Inbound Rules (傳入規則) 索引標籤。
8. 確認下列安全群組規則存在，如下所示：
  - 類型：MYSQL/Aurora
  - 連接埠範圍：3306
  - 來源：**sg-0987654321example** / ec2-rds-**x** – 這是指派給您在上述步驟中驗證的 EC2 執行個體的安全群組。
  - 說明：允許來自 EC2 執行個體 (連接了 **sg-1234567890example**) 之連線的規則
9. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
10. 在導覽窗格中，選擇執行個體。
11. 選取您在上一個任務中選取要連線到 RDS 資料庫的 EC2 執行個體，然後選取 Security (安全) 索引標籤。
12. 在 Security details (安全詳細資訊)、Security groups (安全群組) 中，確認名為 ec2-rds-**x** 的安全群組在清單中。**x** 是一個數字。
13. 選擇 ec2-rds-**x** 安全群組，將其開啟。
14. 選擇 Outbound rules (傳出規則) 索引標籤。
15. 確認下列安全群組規則存在，如下所示：
  - 類型：MYSQL/Aurora
  - 連接埠範圍：3306
  - 目的地：**sg-1234567890example** / rds-ec2-**x**
  - 說明：允許從此安全群組連接的任何執行個體到 **database-tutorial** 之連線的規則

透過驗證這些安全群組和安全群組規則是否存在，並按照本程序所述將其指派給 RDS 資料庫和 EC2 執行個體，您可以使用自動連線功能驗證連線是否已自動設定。

### 檢視動畫：驗證連線組態



您已完成本教學課程的選項 1。您現在可以完成選項 2，會教您如何使用 RDS 主控台將 EC2 執行個體自動連線到 RDS 資料庫，或者您可完成選項 3，會教您如何手動設定在選項 1 中自動建立的安全群組。

選項 2：使用 RDS 主控台自動將 EC2 執行個體連線至 RDS 資料庫

#### 目標

選項 2 的目的是探索 RDS 主控台內的自動連線功能，其可自動設定 EC2 執行個體和 RDS 資料庫之間的連線，以允許從 EC2 執行個體到 RDS 資料庫的流量。在選項 3 中，您將了解如何手動設定連線。

#### 開始之前

完成此教學課程需要以下項目：

- 與 RDS 資料庫處於相同 VPC 的 EC2 執行個體。您可以使用現有的 EC2 執行個體或遵循任務 1 中的步驟來建立新執行個體。
- 呼叫以下操作的許可：
  - `ec2:AssociateRouteTable`
  - `ec2:AuthorizeSecurityGroupEgress`
  - `ec2:CreateRouteTable`
  - `ec2:CreateSecurityGroup`
  - `ec2:CreateSubnet`
  - `ec2:DescribeInstances`
  - `ec2:DescribeNetworkInterfaces`
  - `ec2:DescribeRouteTables`
  - `ec2:DescribeSecurityGroups`
  - `ec2:DescribeSubnets`
  - `ec2:ModifyNetworkInterfaceAttribute`
  - `ec2:RevokeSecurityGroupEgress`

## 完成選項 2 的任務

- [任務 1：啟動 EC2 執行個體 - 選用](#)
- [任務 2：建立 RDS 資料庫並自動將其連線到 EC2 執行個體](#)
- [任務 3：驗證連線組態](#)

## 任務 1：啟動 EC2 執行個體 - 選用

### Note

啟動執行個體並不是本教學課程的重點。如果您已有 Amazon EC2 執行個體，並且您想在本教學課程中使用該執行個體，則可跳過此任務。

## 任務目標

此任務的目標是啟動 EC2 執行個體，以便您可以完成任務 2，您可以在其中設定 EC2 執行個體和 Amazon RDS 資料庫之間的連線。如果您有 EC2 執行個體可使用，則可跳過此任務。

啟動 EC2 執行個體的步驟：

請遵循下列步驟來啟動本教學課程的 EC2 執行個體。

若要檢視這些步驟的動畫，請參閱 [檢視動畫：啟動 EC2 執行個體](#)。

## EC2 執行個體組態

此任務中的步驟會設定 EC2 執行個體，如下所示：

- 執行個體名稱：**tutorial-instance-2**
- AMI：Amazon Linux 2
- 執行個體類型：t2.micro
- 自動指派公有 IP：已啟用
- 具有以下三個規則的安全群組：
  - 允許來自您 IP 地址的 SSH
  - 允許來自任何地方的 HTTPS 流量
  - 允許來自任何地方的 HTTP 流量

### Important

在生產環境中，您應設定執行個體以滿足您的特定需求。

## 啟動 EC2 執行個體

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在 EC2 Dashboard (EC2 儀表板) 中，選擇 Launch Instance (啟動執行個體)。
3. 在 Name and tags (名稱和標籤) 下，針對 Name (名稱)，輸入用於識別執行個體的名稱。針對本教學課程，將執行個體命名為 **tutorial-instance-2**。雖然執行個體名稱並非必要名稱，但是當您在 RDS 主控台中選取執行個體時，名稱可協助您輕鬆識別其。
4. 在 Application and OS Images (應用程式和作業系統映像) 中，選擇符合您 Web 伺服器需求的 AMI。本教學課程使用 Amazon Linux。
5. 在 Instance type (執行個體類型) 中，針對 Instance type (執行個體類型)，選取符合您 Web 伺服器需求的執行個體類型。本教學課程使用的是 t2.micro。

**Note**

您可以在[免費方t2.micro案](#)下使用 Amazon EC2，前提是您的 AWS 帳戶年齡少於 12 個月，而且您可以選擇t2.micro執行個體類型 (或t3.micro在無法使用的區域)。

6. 在 key pair (login) (金鑰對 (登入)) 中，針對 Key pair name (金鑰對名稱) 選擇您的金鑰對。
7. 在Network settings (網路設定) 中，執行下列操作：
  - a. 對於 Network (網路) 和 Subnet (子網路)，如果您尚未變更預設 VPC 或子網路，則可以保留預設設定。

如果您已變更預設 VPC 或子網路，則請檢查下列項目：

- i. 執行個體必須位於與 RDS 資料庫相同的 VPC 中，才能使用自動連線組態。依預設，您只有一個 VPC。
  - ii. 您要在其中啟動執行個體的 VPC 必須連接網際網路閘道，以便您可以從網際網路存取 Web 伺服器。您的預設 VPC 會與網際網路閘道一起自動設定。
  - iii. 若要確保您的執行個體接收公有 IP 地址，針對 Auto-assign public IP (自動指派公有 IP)，檢查是否已選取 Enable (啟用)。如果選取 Disable (停用)，則請選取 Edit (編輯) (位於 Network Settings (網路設定) 右側)，然後針對 Auto-assign public IP (自動指派公有 IP) 選取 Enable (啟用)。
- b. 若要使用 SSH 連線至執行個體，您需要一個安全群組規則來授權來自您電腦公有 IPv4 地址的 SSH (Linux) 或 RDP (Windows) 流量。根據預設，當您啟動執行個體時，會建立新的安全群組，其中包含允許從任何地方傳入 SSH 流量的規則。

若要確保只有您的 IP 地址可以連線到您的執行個體，請在 Firewall (security groups) (防火牆 (安全群組)) 下，從 Allow SSH traffic from (允許 SSH 流量來源) 核取方塊旁的下拉式清單中選擇 My IP (我的 IP)。

- c. 若要允許流量從網際網路傳輸到您的執行個體，請選取下列核取方塊：
  - Allow HTTPs traffic from the internet (允許來自網際網路的 HTTPS 流量)
  - Allow HTTP traffic from the internet (允許來自網際網路的 HTTP 流量)
8. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。
9. 選擇 View all instances (檢視所有執行個體)，以關閉確認頁面並返回主控台。您的執行個體首先會處於 pending 狀態，然後進入 running 狀態。

如果執行個體無法啟動或狀態立即進入 `terminated` 而非 `running`，請參閱[針對執行個體啟動問題進行故障診斷](#)。

如需有關啟動執行個體的詳細資訊，請參閱[使用新的啟動執行個體精靈啟動執行個體](#)。

## 檢視動畫：啟動 EC2 執行個體

The screenshot displays the Amazon EC2 console interface. On the left is a navigation sidebar with categories like EC2 Dashboard, Instances, Images, Elastic Block Store, and Network & Security. The main content area is titled 'Resources' and shows a summary of EC2 resources in the Europe (Stockholm) Region. Below this, there are sections for 'Launch instance' with a 'Launch instance' button, 'Scheduled events' showing no events, and 'Service health' indicating the service is operating normally. A 'Zones' table lists available availability zones.

Resource	Count	Resource	Count	Resource	Count
Instances (running)	2	Dedicated Hosts	0	Elastic IPs	0
Instances	2	Key pairs	1	Load balancers	0
Placement groups	0	Security groups	10	Snapshots	1
Volumes	3				

Zone name	Zone ID
eu-north-1a	eun1-az1
eu-north-1b	eun1-az2
eu-north-1c	eun1-az3

您現在已準備好使用[任務 2：建立 RDS 資料庫並自動將其連線到 EC2 執行個體](#)。

## 任務 2：建立 RDS 資料庫並自動將其連線到 EC2 執行個體

### 任務目標

此任務的目標是建立 RDS 資料庫，並使用 RDS 主控台中的自動連線功能，自動設定 EC2 執行個體與 RDS 資料庫之間的連線。

### 建立 RDS 資料庫的步驟

使用 RDS 主控台中的自動功能，透過下列步驟建立 RDS 資料庫並將其連線至 EC2 執行個體。

若要檢視這些步驟的動畫，請參閱 [檢視動畫：建立 RDS 資料庫並自動將其連線至 EC2 執行個體](#)。

## 資料庫執行個體組態

此任務中的步驟會設定資料庫執行個體，如下所示：

- 引擎類型：MySQL
- 範本：免費方案
- 資料庫執行個體識別符：**tutorial-database**
- 資料庫執行個體類別：db.t3.micro

### Important

在生產環境中，您應設定執行個體以滿足您的特定需求。

若要建立 RDS 資料庫並自動將其連線至 EC2 執行個體

1. 前往 <https://console.aws.amazon.com/rds/>，開啟 Amazon RDS 主控台。
2. 從區域選取器 (位於右上角) 中，選擇您 AWS 區域 在其中建立 EC2 執行個體的執行個體。EC2 執行個體和 RDS 資料庫必須位於相同的區域中。
3. 在儀表板中，選擇 Create database (建立資料庫)。
4. 在 Choose a database creation method (選取資料庫建立方法) 中，確認選取 Standard Create (標準建立)。如果您選擇 Easy create (輕鬆建立)，則無法使用自動連線功能。
5. 在 Engine options (引擎選項) 中，針對 Engine type (引擎類型) 選擇 MySQL。
6. 在 Templates (範本) 中，選擇符合您需求的範例範本。在本教學課程中，選擇 Free tier (免費方案) 免費建立 RDS 資料庫。但請注意，只有在您的帳戶使用未滿 12 個月時，才能使用免費方案。其他限制適用。在 Free tier (免費方案) 方塊中選擇 Info (資訊) 連結，可閱讀更多資訊。
7. 在 Settings (設定) 下，執行下列動作：
  - a. 在 DB instance identifier (資料庫執行個體識別符) 中，輸入資料庫的名稱。針對本教學，輸入 **tutorial-database**。
  - b. 對於 Master username (主要使用者名稱)，請保留預設名稱，即 **admin**。
  - c. 對於 Master password (主要密碼)，請輸入您在本教學課程中可以記住的密碼，然後在 Confirm password (確認密碼) 中再次輸入密碼。

8. 在 Instance configuration (執行個體組態) 中，將 DB instance class (資料庫執行個體類別) 保留為預設值，即 db.t3.micro。如果您的帳戶少於 12 個月，則可以免費使用此執行個體。其他限制適用。如需詳細資訊，請參閱 [AWS 免費方案](#)。
9. 在 Connectivity (連線) 中，對於 Compute resource (運算資源)，選擇 Connect to an EC2 compute resource (連線至 EC2 運算資源)。這是 RDS 主控台內的自動連線功能。
10. 針對 EC2 instance (EC2 執行個體)，選擇您想要連線的 EC2 執行個體。在本教學課程中，您可以選擇在上一個任務中建立的執行個體 (您已命名為 **tutorial-instance**)，或選擇另一個現有執行個體。如果在清單中沒有看到您的執行個體，則請選擇 Connectivity (連線) 右側的重新整理圖示。

使用自動連線功能時，會將安全群組新增至此 EC2 執行個體，並將另一個安全群組新增至 RDS 資料庫。系統會自動設定安全群組，以允許 EC2 執行個體和 RDS 資料庫之間的流量。在下一個任務中，您將確認安全群組是否已建立並指派給 EC2 執行個體和 RDS 資料庫。

11. 選擇建立資料庫。

在 Databases (資料庫) 畫面中，新資料庫的 Status (狀態) 為 Creating (正在建立)，直到資料庫準備就緒可供使用為止。狀態變更為 Available (可用) 時，您便可連線至資料庫。視資料庫類別和儲存體數量而定，可能需要最多 20 分鐘的時間，新資料庫才會可用。

若要進一步了解，請參閱《Amazon RDS 使用者指南》中的設定與 EC2 執行個體的自動網路連線。



## 檢視動畫：建立 RDS 資料庫並自動將其連線至 EC2 執行個體

The screenshot shows the Amazon RDS console interface. On the left is a navigation sidebar with the following items: Dashboard (highlighted), Databases, Performance Insights, Snapshots, Automated backups, Reserved instances, Proxies, Subnet groups, Parameter groups, Option groups, Custom engine versions, Events, Event subscriptions, and Certificate update. The main content area features a top banner with an information icon and text: "Try the new Amazon RDS Multi-AZ deployment option for MySQL and PostgreSQL. For your Amazon RDS for MySQL and PostgreSQL workloads, improve transactional instances by deploying the Multi-AZ DB cluster. Learn more". Below this is a prominent orange "Create database" button with a mouse cursor hovering over it. Underneath the button, it says "Or, Restore Multi-AZ DB Cluster from Snapshot". Below the banner is a "Resources" section titled "Resources" which lists various RDS resources in the EU (Stockholm) region, including DB Instances (5/40), DB Clusters (1/40), Reserved instances (0/40), and Snapshots (2). The "Snapshots" section is further divided into Manual and Automated categories. At the bottom of the console, there is a "Create database" section with the text: "Amazon Relational Database Service (RDS) makes it easy to set up, operate, and scale a rel".

您現在已準備好使用 [任務 3：驗證連線組態](#)。

### 任務 3：驗證連線組態

#### 任務目標

此任務的目標是確認已建立兩個安全群組並指派給執行個體和資料庫。

當您使用 RDS 主控台內的自動連線功能來設定連線時，會自動建立安全群組並指派給執行個體和資料庫，如下所示：

- 建立安全群組 `rds-ec2-x` 並新增至 RDS 資料庫。它有一個傳入規則，該規則引用 `ec2-rds-x` 安全群組作為其來源。這允許來自 EC2 執行個體的流量透過 `ec2-rds-x` 安全群組到達 RDS 資料庫。
- 建立安全群組 `ec2-rds-x` 並新增至 EC2 執行個體。它有一個傳出規則，該規則引用 `rds-ec2-x` 安全群組作為其目的地。這允許來自 EC2 執行個體的流量透過 `rds-ec2-x` 安全群組到達 RDS 資料庫。

## 驗證連線組態的步驟

使用以下步驟驗證連線組態。

若要檢視這些步驟的動畫，請參閱 [檢視動畫：驗證連線組態](#)。

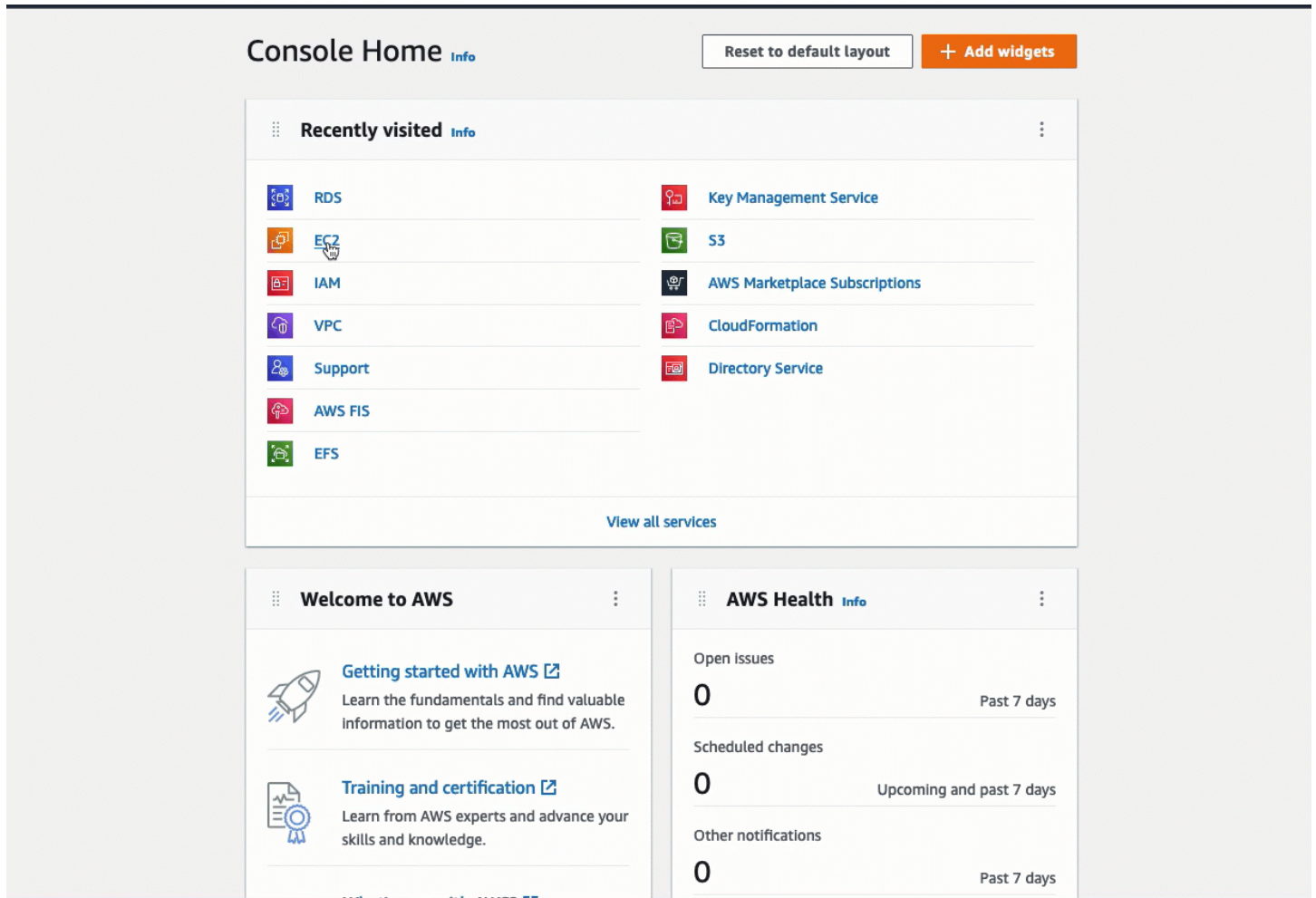
若要使用主控台驗證連線組態

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取您在上一個任務中選取要連線到 RDS 資料庫的 EC2 執行個體，然後選取 Security (安全) 索引標籤。
4. 在 Security details (安全詳細資訊)、Security groups (安全群組) 中，確認名為 `ec2-rds-x` 的安全群組在清單中。`x` 是一個數字。
5. 選擇 `ec2-rds-x` 安全群組，將其開啟。
6. 選擇 Outbound rules (傳出規則) 索引標籤。
7. 確認下列安全群組規則存在，如下所示：
  - 類型：MySQL/Aurora
  - 連接埠範圍：3306
  - 目的地：**`sg-1234567890example`** / `rds-ec2-x`
  - 說明：允許從此安全群組連接的任何執行個體到 **database-tutorial** 之連線的規則
8. 前往 <https://console.aws.amazon.com/rds/>，開啟 Amazon RDS 主控台。
9. 在導覽頁面中，選擇 Databases (資料庫)。
10. 選擇您為此教學課程建立的 RDS 資料庫。
11. 在 Connectivity & security (連線與安全) 索引標籤的 Security (安全性)、VPC security groups (VPC 安全群組) 中，確認會顯示名為 `rds-ec2-x` 的安全群組。

12. 選擇 `rds-ec2-x` 安全群組。EC2 主控台中的 Security Groups (安全群組) 畫面開啟。
13. 選擇 `rds-ec2-x` 安全群組，將其開啟。
14. 選擇 Inbound Rules (傳入規則) 索引標籤。
15. 確認下列安全群組規則存在，如下所示：
  - 類型：MYSQL/Aurora
  - 連接埠範圍：3306
  - 來源：***sg-0987654321example*** / `ec2-rds-x` – 這是指派給您在上述步驟中驗證的 EC2 執行個體的安全群組。
  - 說明：允許來自 EC2 執行個體 (連接了 ***sg-1234567890example***) 之連線的規則

透過驗證這些安全群組和安全群組規則是否存在，並按照本程序所述將其指派給 EC2 執行個體和 RDS 資料庫，您可以使用自動連線功能驗證連線是否已自動設定。

## 檢視動畫：驗證連線組態



您已完成本教學課程的選項 2。您現在可以完成選項 3，其會教導您如何手動設定在選項 2 中自動建立的安全群組。

選項 3：透過模仿自動連線功能，手動將 EC2 執行個體連線至 RDS 資料庫

### 目標

選項 3 的目標是了解如何透過手動複製自動連線功能的組態來手動設定 EC2 執行個體和 RDS 資料庫之間的連線。

### 開始之前

完成此教學課程需要以下項目：

- 與 RDS 資料庫處於相同 VPC 的 EC2 執行個體。您可以使用現有的 EC2 執行個體或遵循任務 1 中的步驟來建立新執行個體。

- 與 EC2 執行個體處於相同 VPC 的 RDS 資料庫。您可以使用現有的 RDS 資料庫或遵循任務 2 中的步驟來建立新資料庫。
- 呼叫以下操作的許可。如果您已完成本教學課程的選項 1，則已經擁有這些許可。
  - `ec2:AssociateRouteTable`
  - `ec2:AuthorizeSecurityGroupEgress`
  - `ec2:CreateRouteTable`
  - `ec2:CreateSecurityGroup`
  - `ec2:CreateSubnet`
  - `ec2:DescribeInstances`
  - `ec2:DescribeNetworkInterfaces`
  - `ec2:DescribeRouteTables`
  - `ec2:DescribeSecurityGroups`
  - `ec2:DescribeSubnets`
  - `ec2:ModifyNetworkInterfaceAttribute`
  - `ec2:RevokeSecurityGroupEgress`

### 完成選項 3 的任務

- [任務 1：啟動 EC2 執行個體 - 選用](#)
- [任務 2：建立 RDS 資料庫 - 選用](#)
- [任務 3：透過建立安全群組並將其指派給執行個體，手動將 EC2 執行個體連線至 RDS 資料庫](#)

### 任務 1：啟動 EC2 執行個體 - 選用

#### Note

啟動執行個體並不是本教學課程的重點。如果您已有 Amazon EC2 執行個體，並且您想在本教學課程中使用該執行個體，則可跳過此任務。

### 任務目標

此任務的目標是啟動 EC2 執行個體，以便您可以完成任務 3，您可以在其中設定 EC2 執行個體和 Amazon RDS 資料庫之間的連線。

啟動 EC2 執行個體的步驟：

請遵循下列步驟來啟動本教學課程的 EC2 執行個體。

若要檢視這些步驟的動畫，請參閱 [檢視動畫：啟動 EC2 執行個體](#)。

## EC2 執行個體組態

此任務中的步驟會設定 EC2 執行個體，如下所示：

- 執行個體名稱：**tutorial-instance**
- AMI：Amazon Linux 2
- 執行個體類型：t2.micro
- 自動指派公有 IP：已啟用
- 具有以下三個規則的安全群組：
  - 允許來自您 IP 地址的 SSH
  - 允許來自任何地方的 HTTPS 流量
  - 允許來自任何地方的 HTTP 流量

### Important

在生產環境中，您應設定執行個體以滿足您的特定需求。

## 啟動 EC2 執行個體

1. 登入 AWS Management Console 並開啟 Amazon EC2 主控台，網址為 <https://console.aws.amazon.com/ec2/>。
2. 在 EC2 Dashboard (EC2 儀表板) 中，選擇 Launch Instance (啟動執行個體)。
3. 在 Name and tags (名稱和標籤) 下，針對 Name (名稱)，輸入用於識別執行個體的名稱。針對本教學課程，將執行個體命名為 **tutorial-instance-manual-1**。雖然執行個體名稱不是強制性的，但名稱可協助您輕鬆識別其。
4. 在 Application and OS Images (應用程式和作業系統映像) 中，選擇符合您 Web 伺服器需求的 AMI。本教學課程使用 Amazon Linux。
5. 在 Instance type (執行個體類型) 中，針對 Instance type (執行個體類型)，選取符合您 Web 伺服器需求的執行個體類型。本教學課程使用的是 t2.micro。

**Note**

您可以在[免費方t2.micro案](#)下使用 Amazon EC2，前提是您的 AWS 帳戶年齡少於 12 個月，而且您可以選擇t2.micro執行個體類型 (或t3.micro在無法使用的區域)。

6. 在 key pair (login) (金鑰對 (登入)) 中，針對 Key pair name (金鑰對名稱) 選擇您的金鑰對。
7. 在 Network settings (網路設定) 中，執行下列操作：
  - a. 對於 Network (網路) 和 Subnet (子網路)，如果您尚未變更預設 VPC 或子網路，則可以保留預設設定。

如果您已變更預設 VPC 或子網路，則請檢查下列項目：

- i. 執行個體必須與 RDS 資料庫處於相同的 VPC 中。依預設，您只有一個 VPC。
  - ii. 您要在其中啟動執行個體的 VPC 必須連接網際網路閘道，以便您可以從網際網路存取 Web 伺服器。您的預設 VPC 會與網際網路閘道一起自動設定。
  - iii. 若要確保您的執行個體接收公有 IP 地址，針對 Auto-assign public IP (自動指派公有 IP)，檢查是否已選取 Enable (啟用)。如果選取 Disable (停用)，則請選取 Edit (編輯) (位於 Network Settings (網路設定) 右側)，然後針對 Auto-assign public IP (自動指派公有 IP) 選取 Enable (啟用)。
- b. 若要使用 SSH 連線至執行個體，您需要一個安全群組規則來授權來自您電腦公有 IPv4 地址的 SSH (Linux) 或 RDP (Windows) 流量。根據預設，當您啟動執行個體時，會建立新的安全群組，其中包含允許從任何地方傳入 SSH 流量的規則。

若要確保只有您的 IP 地址可以連線到您的執行個體，請在 Firewall (security groups) (防火牆 (安全群組)) 下，從 Allow SSH traffic from (允許 SSH 流量來源) 核取方塊旁的下拉式清單中選擇 My IP (我的 IP)。

- c. 若要允許流量從網際網路傳輸到您的執行個體，請選取下列核取方塊：
  - Allow HTTPs traffic from the internet (允許來自網際網路的 HTTPS 流量)
  - Allow HTTP traffic from the internet (允許來自網際網路的 HTTP 流量)
8. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。
9. 選擇 View all instances (檢視所有執行個體)，以關閉確認頁面並返回主控台。您的執行個體首先會處於 pending 狀態，然後進入 running 狀態。

如果執行個體無法啟動或狀態立即進入 `terminated` 而非 `running`，請參閱[針對執行個體啟動問題進行故障診斷](#)。

如需有關啟動執行個體的詳細資訊，請參閱[使用新的啟動執行個體精靈啟動執行個體](#)。

## 檢視動畫：啟動 EC2 執行個體

The screenshot shows the AWS Management Console interface for the EC2 Dashboard in the Europe (Stockholm) Region. The left sidebar contains navigation options like 'EC2 Dashboard', 'Instances', 'Images', and 'Elastic Block Store'. The main content area is divided into several sections:

- Resources:** A summary table showing the number of resources in use. A notification banner below it says 'Learn more about the latest in AWS Compute from AWS re:Invent by viewing the EC2 Videos.'
- Launch instance:** A section with a 'Launch instance' button and a 'Migrate a server' link. A note below states: 'Note: Your instances will launch in the Europe (Stockholm) Region.'
- Scheduled events:** A section showing 'Europe (Stockholm)' with 'No scheduled events'.
- Service health:** A section showing the status of the service as 'This service is operating normally'.
- Zones:** A table listing available zones in the region.

Resource	Count
Instances (running)	2
Instances	2
Placement groups	0
Volumes	3
Dedicated Hosts	0
Key pairs	1
Security groups	10
Elastic IPs	0
Load balancers	0
Snapshots	1

Zone name	Zone ID
eu-north-1a	eun1-az1
eu-north-1b	eun1-az2
eu-north-1c	eun1-az3

您現在已準備好使用 [任務 2：建立 RDS 資料庫 - 選用](#)。

## 任務 2：建立 RDS 資料庫 - 選用

### Note

建立 RDS 資料庫並不是教學課程中本部分的重點。如果您已有 RDS 資料庫，並且您想在本教學課程中使用該資料庫，則可跳過此任務。



## 任務目標

此任務的目標是建立 RDS 資料庫。將此執行個體連線至 EC2 執行個體時，您會在任務 3 中使用此執行個體。

## 建立 RDS 資料庫的步驟

請使用下列步驟為本教學課程的選項 3 建立 RDS 資料庫。

若要檢視這些步驟的動畫，請參閱 [檢視動畫：建立資料庫執行個體](#)。

## RDS 資料庫組態

此任務中的步驟會設定 RDS 資料庫，如下所示：

- 引擎類型：MySQL
- 範本：免費方案
- 資料庫執行個體識別符：**tutorial-database-manual**
- 資料庫執行個體類別：db.t3.micro

### Important

在生產環境中，您應設定執行個體以滿足您的特定需求。

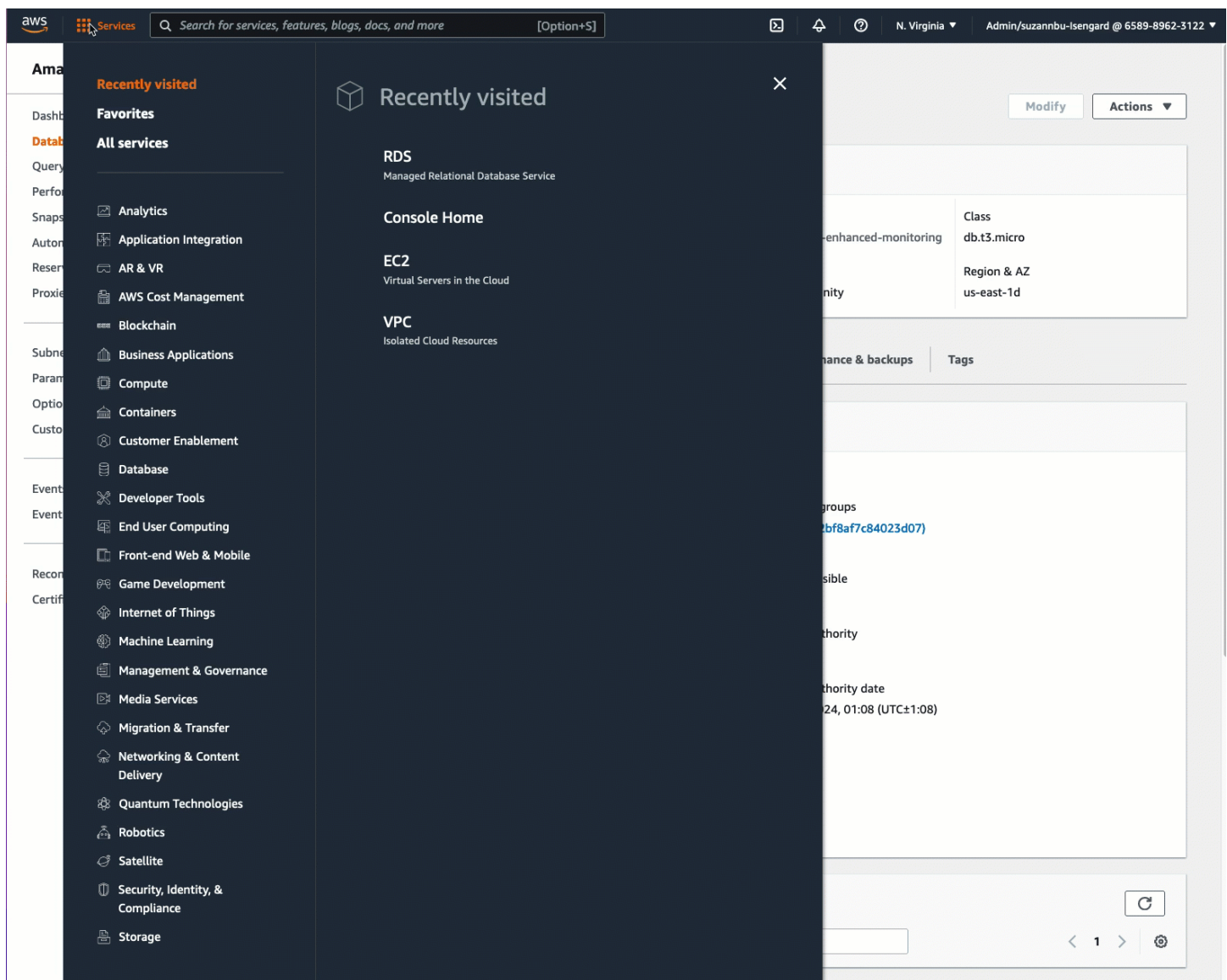
## 建立 MySQL 資料庫執行個體

1. 前往 <https://console.aws.amazon.com/rds/>，開啟 Amazon RDS 主控台。
2. 從區域選取器 (位於右上角) 中，選擇您 AWS 區域 在其中建立 EC2 執行個體的執行個體。EC2 執行個體與資料庫執行個體必須處於相同區域。
3. 在儀表板中，選擇 Create database (建立資料庫)。
4. 在 Choose a database creation method (選擇資料庫建立方法) 中，選擇 Easy create (輕鬆建立)。當您選擇此選項時，無法使用自動設定連線的自動連線功能。
5. 在 Engine options (引擎選項) 中，針對 Engine type (引擎類型) 選擇 MySQL。
6. 在 DB instance size (資料庫執行個體大小) 中，選擇 Free tier (免費方案)。
7. 在 DB instance identifier (資料庫執行個體識別符) 中，輸入 RDS 資料庫的名稱。針對本教學，輸入 **tutorial-database-manual**。

- 對於 Master username (主要使用者名稱)，請保留預設名稱，即 **admin**。
- 對於 Master password (主要密碼)，請輸入您在本教學課程中可以記住的密碼，然後在 Confirm password (確認密碼) 中再次輸入密碼。
- 選擇建立資料庫。

在 Databases (資料庫) 畫面中，新資料庫執行個體的 Status (狀態) 為 Creating (正在建立)，直到資料庫執行個體準備就緒可供使用為止。狀態變更為 Available (可用) 時，您便能連線至資料庫執行個體。視資料庫執行個體類別和儲存體數量而定，可能需要最多 20 分鐘的時間，新執行個體才會可用。

### 檢視動畫：建立資料庫執行個體



您現在已準備好使用 [任務 3：透過建立安全群組並將其指派給執行個體，手動將 EC2 執行個體連線至 RDS 資料庫](#)。

任務 3：透過建立安全群組並將其指派給執行個體，手動將 EC2 執行個體連線至 RDS 資料庫

### 任務目標

此任務的目標是透過手動執行下列動作來複製自動連線功能的連線組態：您建立兩個新的安全群組，然後將每個安全群組新增至 EC2 執行個體和 RDS 資料庫。

### 建立新安全群組並將其新增至執行個體的步驟

使用下列步驟，透過建立兩個新的安全群組，將 EC2 執行個體連線至 RDS 資料庫。然後，您可以將每個安全群組新增到 EC2 執行個體和 RDS 資料庫。

建立兩個新的安全群組，並分別指派給 EC2 執行個體和 RDS 資料庫

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 首先建立要新增至 EC2 執行個體的安全群組，如下所示：
  - a. 在導覽窗格中，選擇 Security Groups (安全群組)。
  - b. 選擇 Create Security Group (建立安全群組)。
  - c. 對於 Security group name (安全群組名稱)，輸入安全群組的描述性名稱。針對本教學，輸入 **ec2-rds-manual-configuration**。
  - d. 對於 Description (描述)，輸入簡短描述。針對本教學，輸入 **EC2 instance security group to allow EC2 instance to securely connect to RDS database**。
  - e. 選擇建立安全群組。建立 RDS 資料庫安全群組之後，您將回到此安全群組以新增傳出規則。
3. 現在，建立要新增至 RDS 資料庫的安全群組，如下所示：
  - a. 在導覽窗格中，選擇 Security Groups (安全群組)。
  - b. 選擇 Create Security Group (建立安全群組)。
  - c. 對於 Security group name (安全群組名稱)，輸入安全群組的描述性名稱。針對本教學，輸入 **rds-ec2-manual-configuration**。
  - d. 對於 Description (描述)，輸入簡短描述。針對本教學，輸入 **RDS database security group to allow EC2 instance to securely connect to RDS database**。
  - e. 在 Inbound rules (傳入規則) 中，選擇 Add rule (新增規則)，然後執行下列動作：
    - i. 對於 Type (類型)，選擇 MYSQL/Aurora。

- ii. 對於 Source (來源), 選擇您在此程序的步驟 2 中建立的 EC2 執行個體安全群組 `ec2-rds-manual-configuration`。
  - f. 選擇建立安全群組。
4. 編輯 EC2 執行個體安全群組以新增傳出規則, 如下所示:
  - a. 在導覽窗格中, 選擇安全群組。
  - b. 選取 EC2 執行個體安全群組 (已命名為 `ec2-rds-manual-configuration`), 然後選取 Outbound rules (傳出規則) 索引標籤。
  - c. 選擇 Edit outbound rules (編輯傳出規則)。
  - d. 選擇 Add rule (新增規則) 並執行下列動作:
    - i. 對於 Type (類型), 選擇 MYSQL/Aurora。
    - ii. 對於 Source (來源), 選擇您在此程序的步驟 3 中建立的 RDS 資料庫安全群組 `rds-ec2-manual-configuration`。
    - iii. 選擇儲存規則。
5. 將 EC2 執行個體安全群組新增至 EC2 執行個體, 如下所示:
  - a. 在導覽窗格中, 選擇執行個體。
  - b. 選取 EC2 執行個體, 然後選取 Actions (動作)、Security (安全性)、Change security groups (變更安全群組)。
  - c. 在 Associated security groups (關聯的安全群組) 中, 選取 Select security groups (選取安全群組) 欄位, 選取您先前建立的 `ec2-rds-manual-configuration`, 然後選取 Add security group (新增安全群組)。
  - d. 選擇儲存。
6. 將 RDS 資料庫安全群組新增至 RDS 資料庫, 如下所示:
  - a. 前往 <https://console.aws.amazon.com/rds/>, 開啟 Amazon RDS 主控台。
  - b. 在導覽窗格中, 選取 Databases (資料庫), 然後選取您的資料庫。
  - c. 選擇 Modify (修改)。
  - d. 在 Connectivity (連線) 中, 針對 Security group (安全群組) 選擇您先前建立的 `rds-ec2-manual-configuration`, 然後選擇 Continue (繼續)。
  - e. 在 Scheduling of modifications (修改排程) 中, 選擇 Apply immediately (立即套用)。
  - f. 選擇 Modify DB instance (修改資料庫執行個體)。

您現在已完成手動步驟，這些步驟會模擬當使用自動連線功能時所發生的自動步驟。

您已完成本教學課程的選項 3。如果您已完成選項 1、2 和 3，且不再需要本教學課程中建立的資源，您應該將其刪除，以避免產生不必要的費用。如需詳細資訊，請參閱 [清除](#)。

## 清除

現在您已經完成教學課程，最好清理 (刪除) 您不再想要使用的任何資源。清理 AWS 資源可防止您的帳戶產生任何進一步的費用。

## 主題

- [終止您的 EC2 執行個體](#)
- [刪除 RDS 資料庫](#)

## 終止您的 EC2 執行個體

如果您針對本教學課程專門啟動 EC2 執行個體，則可以將其終止以停止產生與其相關的任何費用。

### 欲使用主控台來終止執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取您為此教學課程建立的執行個體，然後選取 Instance state (執行個體狀態)、Terminate instance (終止執行個體)。
4. 出現確認提示時，請選擇 Terminate (終止)。

## 刪除 RDS 資料庫

如果您針對本教學課程專門建立了 RDS 資料庫，則可以將其刪除以停止產生與其相關的任何費用。

### 若要使用主控台刪除 RDS 資料庫

1. 前往 <https://console.aws.amazon.com/rds/>，開啟 Amazon RDS 主控台。
2. 在導覽窗格中，選擇 Databases (資料庫)。
3. 選取您為此教學課程建立的 RDS 資料庫，然後選取 Actions (動作)、Delete (刪除)。
4. 在方塊中輸入 **delete me**，然後選擇 Delete (刪除)。

## 識別您的 EC2 執行個體

您可能需要判斷您的應用程式是否在 EC2 執行個體上執行，特別是如果您有混合運算環境。每個執行個體都有簽署的執行個體身分識別文件，您可以透過密碼方式 您可以在下列本地、不可路由的地址 <http://169.254.169.254/latest/dynamic/instance-identity/> 找到這些文件。如需詳細資訊，請參閱 [執行個體身分文件](#)。

### 檢查系統 UUID

您可以獲取系統 UUID 並查看 UUID 的開頭八位字節 EC2 (在 Linux 中，這可能是小寫字母)。ec2 此方法很快，但可能不準確，因為不是 EC2 執行個體的系統很可能擁有以這些字元開頭的 UUID。此外，某些版本的 SMBIOS 使用小端格式，該格式不包括 EC2 在 UUID 的開頭。對於使用 SMBIOS 2.4 適用於視窗的 EC2 執行個體，或具有自己實作 SMBIOS 的 Amazon Linux 2 以外的 Linux 發行版，可能就是這種情況。

Linux 範例：從 DMI 取得 UUID (僅限 HVM AMI)

運用下列命令來透過桌面管理介面 (DMI) 取得 UUID：

```
[ec2-user ~]$ sudo dmidecode --string system-uuid
```

在下列的範例輸出中，UUID 是以「EC2」開頭，這代表系統可能是 EC2 執行個體。

```
EC2E1916-9099-7CAF-FD21-012345ABCDEF
```

在下列的範例輸出中，是以小端序格式來表示 UUID。

```
45E12AEC-DCD1-B213-94ED-012345ABCDEF
```

或者，對於建立在 Nitro 系統上的執行個體，您可以使用下列命令：

```
[ec2-user ~]$ cat /sys/devices/virtual/dmi/id/board_asset_tag
```

如果輸出為執行個體 ID，如下列範例輸出，則系統為 EC2 執行個體：

```
i-0af01c0123456789a
```

Linux 範例：從虛擬機器管理程式取得 UUID (僅限 PV AMI)

使用下列命令從 hypervisor 取得 UUID：

```
[ec2-user ~]$ cat /sys/hypervisor/uuid
```

在下列的範例輸出中，UUID 是以「ec2」開頭，這代表系統可能是 EC2 執行個體。

```
ec2e1916-9099-7caf-fd21-012345abcdef
```

視窗範例：使用 WMI 或視窗取得 UUID PowerShell

使用 Windows Management Instrumentation (Windows 管理規範) 命令列 (WMIC)，如下所示：

```
wmic path win32_computersystemproduct get uuid
```

或者，如果您使用的是 Windows PowerShell，請依照下列方式使用 Get-WmiObject 指令程式：

```
PS C:\> Get-WmiObject -query "select uuid from Win32_ComputerSystemProduct" | Select
UUID
```

在下列的範例輸出中，UUID 是以「EC2」開頭，這代表系統可能是 EC2 執行個體。

```
EC2AE145-D1DC-13B2-94ED-012345ABCDEF
```

如果是使用 SMBIOS 2.4 的執行個體，可能會以小端序格式來表示 UUID，例如：

```
45E12AEC-DCD1-B213-94ED-012345ABCDEF
```

## 檢查系統虛擬機器產生識別符

虛擬機器產生識別符由 128 位元的專屬緩衝區組成，該緩衝區解譯為加密隨機整數識別符。您可以擷取虛擬機器產生識別符以識別您的 Amazon Elastic Compute Cloud 執行個體。產生識別符透過 ACPI 表項目在執行個體的訪客作業系統中公開。若您的電腦被複製或匯入到 AWS(例如使用 [VM Import/Export](#))，那麼值就會改變。

範例：從 Linux 擷取虛擬機器產生識別碼

您可使用以下命令從執行 Linux 的執行個體中擷取虛擬機器產生識別符。

### Amazon Linux 2

1. 根據需要，使用以下命令更新現有軟體套件：

```
sudo yum update
```

2. 如有必要，使用下列命令以 busybox 套件作為來源：

```
sudo curl https://www.rpmfind.net/linux/epel/next/8/Everything/x86_64/Packages/b/busybox-1.35.0-2.el8.next.x86_64.rpm --output busybox.rpm
```

3. 如有必要，使用下列命令安裝必要條件套件：

```
sudo yum install busybox.rpm iasl -y
```

4. 執行下列 iasl 命令從 ACPI 表產生輸出結果：

```
sudo iasl -p ./SSDT2 -d /sys/firmware/acpi/tables/SSDT2
```

5. 執行下列命令，以檢閱 iasl 命令的輸出結果：

```
cat SSDT2.dsl
```

輸出結果應產生擷取虛擬機器產生識別符所需的地址空間：

```
Intel ACPI Component Architecture
ASL+ Optimizing Compiler/Disassembler version 20190509
Copyright (c) 2000 - 2019 Intel Corporation

File appears to be binary: found 32 non-ASCII characters, disassembling
Binary file appears to be a valid ACPI table, disassembling
Input file /sys/firmware/acpi/tables/SSDT2, Length 0x7B (123) bytes
ACPI: SSDT 0x0000000000000000 00007B (v01 AMAZON AMZNSSDT 00000001 AMZN
00000001)
Pass 1 parse of [SSDT]
Pass 2 parse of [SSDT]
Parsing Deferred Opcodes (Methods/Buffers/Packages/Regions)

Parsing completed
Disassembly completed
ASL Output: ./SSDT2.dsl - 1065 bytes
$
/*
* Intel ACPI Component Architecture
* AML/ASL+ Disassembler version 20190509 (64-bit version)
```



```

* Copyright (c) 2000 - 2019 Intel Corporation
*
* Disassembling to symbolic ASL+ operators
*
* Disassembly of /sys/firmware/acpi/tables/SSDT2, Tue Mar 29 16:15:14 2022
*
* Original Table Header:
* Signature "SSDT"
* Length 0x0000007B (123)
* Revision 0x01
* Checksum 0xB8
* OEM ID "AMAZON"
* OEM Table ID "AMZNSSDT"
* OEM Revision 0x00000001 (1)
* Compiler ID "AMZN"
* Compiler Version 0x00000001 (1)
*/
DefinitionBlock ("", "SSDT", 1, "AMAZON", "AMZNSSDT", 0x00000001)
{
 Scope (_SB)
 {
 Device (VMGN)
 {
 Name (_CID, "VM_Gen_Counter") // _CID: Compatible ID
 Name (_DDN, "VM_Gen_Counter") // _DDN: DOS Device Name
 Name (_HID, "AMZN0000") // _HID: Hardware ID
 Name (ADDR, Package (0x02)
 {
 0xFED01000,
 Zero
 })
 }
 }
}

```

6. (選用) 使用下列命令提升其餘步驟的終端許可：

```
sudo -s
```

7. 使用下列命令存放先前收集的地址空間：

```
VMGN_ADDR=0xFED01000
```

8. 使用下列命令重複執行地址空間並建置虛擬機器產生識別符：

```
for offset in 0x0 0x4 0x8 0xc; do busybox devmem $((VMGN_ADDR + $offset)) | sed 's/0x//' | sed -z '$ s/\n$//' >> vmgenid; done
```

9. 使用下列命令從輸出檔案中擷取虛擬機器產生識別符：

```
cat vmgenid ; echo
```

您的輸出應類似以下內容：

```
EC2F335D979132C4165896753E72BD1C
```

## Ubuntu

1. 根據需要，使用以下命令更新現有軟體套件：

```
sudo apt update
```

2. 如有必要，使用下列命令安裝必要條件套件：

```
sudo apt install busybox iasl -y
```

3. 執行下列 `iasl` 命令從 ACPI 表產生輸出結果：

```
sudo iasl -p ./SSDT2 -d /sys/firmware/acpi/tables/SSDT2
```

4. 執行下列命令，以檢閱 `iasl` 命令的輸出結果：

```
cat SSDT2.dsl
```

輸出結果應產生擷取虛擬機器產生識別符所需的地址空間：

```
Intel ACPI Component Architecture
ASL+ Optimizing Compiler/Disassembler version 20190509
Copyright (c) 2000 - 2019 Intel Corporation

File appears to be binary: found 32 non-ASCII characters, disassembling
Binary file appears to be a valid ACPI table, disassembling
```

```
Input file /sys/firmware/acpi/tables/SSDT2, Length 0x7B (123) bytes
ACPI: SSDT 0x0000000000000000 00007B (v01 AMAZON AMZNSSDT 00000001 AMZN
00000001)
Pass 1 parse of [SSDT]
Pass 2 parse of [SSDT]
Parsing Deferred Opcodes (Methods/Buffers/Packages/Regions)

Parsing completed
Disassembly completed
ASL Output: ./SSDT2.dsl - 1065 bytes
$
/*
 * Intel ACPI Component Architecture
 * AML/ASL+ Disassembler version 20190509 (64-bit version)
 * Copyright (c) 2000 - 2019 Intel Corporation
 *
 * Disassembling to symbolic ASL+ operators
 *
 * Disassembly of /sys/firmware/acpi/tables/SSDT2, Tue Mar 29 16:15:14 2022
 *
 * Original Table Header:
 * Signature "SSDT"
 * Length 0x0000007B (123)
 * Revision 0x01
 * Checksum 0xB8
 * OEM ID "AMAZON"
 * OEM Table ID "AMZNSSDT"
 * OEM Revision 0x00000001 (1)
 * Compiler ID "AMZN"
 * Compiler Version 0x00000001 (1)
 */
DefinitionBlock ("", "SSDT", 1, "AMAZON", "AMZNSSDT", 0x00000001)
{
Scope (_SB)
{
 Device (VMGN)
 {
 Name (_CID, "VM_Gen_Counter") // _CID: Compatible ID
 Name (_DDN, "VM_Gen_Counter") // _DDN: DOS Device Name
 Name (_HID, "AMZN0000") // _HID: Hardware ID
 Name (ADDR, Package (0x02)
 {
 0xFED01000,
 Zero
 }
 }
}
}
```

```
 })
 }
}
}
```

5. (選用) 使用下列命令提升其餘步驟的終端許可：

```
sudo -s
```

6. 使用下列命令存放先前收集的地址空間：

```
VMGN_ADDR=0xFED01000
```

7. 使用下列命令重複執行地址空間並建置虛擬機器產生識別符：

```
for offset in 0x0 0x4 0x8 0xc; do busybox devmem $((VMGN_ADDR + $offset)) | sed
's/0x//' | sed -z '$ s/\n$//' >> vmgenid; done
```

8. 使用下列命令從輸出檔案中擷取虛擬機器產生識別符：

```
cat vmgenid ; echo
```

您的輸出應類似以下內容：

```
EC2F335D979132C4165896753E72BD1C
```

範例：從 Windows 擷取虛擬機器產生識別碼

您可建立範例應用程式，從執行 Windows 的執行個體中擷取虛擬機器產生識別符。如需詳細資訊，請參閱 Microsoft 文件集中的 [《取得虛擬機器產生識別符》](#)。

## 管理亞馬遜 EC2 執行個體的系統設定

啟動執行個體之後，您可以以系統管理員身分登入以進行變更。本節著重於管理執行個體的系統設定。

### 目錄

- [設定亞馬遜 EC2 執行個體的時間](#)
- [Amazon EC2 Linux 執行個體的處理器狀態控制](#)

- [最佳化 CPU 選項](#)
- [Amazon EC2 上的 AMD SEV-SNP](#)
- [使用安裝媒體新增 Windows 系統元件](#)
- [管理 Linux 執行個體上的系統使用者](#)
- [為您的執行個體設定 Windows 管理員密碼](#)

## 設定亞馬遜 EC2 執行個體的時間

Amazon EC2 執行個體上一致且準確的時間參考對於許多伺服器任務和程序而言至關重要。在識別問題發生的時間和事件的時間順序方面，系統日誌中的時間戳記扮演著至關重要的角色。當您使用 AWS CLI 或 AWS SDK 向執行個體提出要求時，這些工具會代表您簽署要求。如果執行個體的日期和時間設定不正確，可能會導致簽名日期和請求日期之間出現差異，導致 AWS 拒絕您的請求。

為了解決這個重要層面，Amazon 提供 Amazon Time Sync Service，此服務可從所有 EC2 執行個體存取和提供不同 AWS 服務使用。該服務使用衛星連接和原子參考時鐘的隊列，以提供協調世界時 (UTC) 全球標準的準確和當前時間讀數。AWS 區域

Amazon Time Sync Service 可以使用網路時間通訊協定 (NTP)，或在[支援](#)的執行個體上提供本機精確時間協定 (PTP) 硬體時鐘。PTP 硬體時鐘支援 NTP 或直接 PTP 連線。NTP 和直接 PTP 連線使用的是相同高度精確時間來源，但直接 PTP 連線比 NTP 連線更精確。與 Amazon Time Sync Service 的 NTP 連線支援飛躍塗抹，而 PTP 與 PTP 硬體時鐘的連線不會塗抹時間。如需詳細資訊，請參閱 [閏秒](#)。

為了獲得最佳效能，我們建議您在 EC2 執行個體上使用本機 Amazon 時間同步服務。若要備份到執行個體上的本機 Amazon 時間同步服務，並將 Amazon EC2 外部的資源連接到 Amazon 時間同步服務，您可以使用位於的公有 Amazon 時間同步服務 `time.aws.com`。公有 Amazon Time Sync Service (例如本機 Amazon Time Sync Service)，會自動消彌加至 UTC 的任何閏秒。公用 Amazon 時間同步服務由我們的衛星連接和原子參考時鐘在全球範圍內提供支援。AWS 區域

### 主題

- [設定您的執行個體使用本機 Amazon Time Sync Service](#)
- [將您的執行個體或任何網際網路連線裝置設定為使用公用的 Amazon Time Sync Service](#)
- [比較 Linux 執行個體的時間戳記](#)
- [變更執行個體的時區](#)
- [閏秒](#)
- [相關資源](#)

## 設定您的執行個體使用本機 Amazon Time Sync Service

您的執行個體可以存取本機 Amazon Time Sync Service，如下所示：

- 透過位於下列 IP 地址端點的 NTP：
  - IPv4：169.254.169.123
  - IPv6：fd00:ec2::123(只有在[AWS 硝基系統上建置的執行個體](#)才能存取。)
- (僅限 Linux) 透過直接 PTP 連線來連線至本機 PTP 硬體時鐘：
  - PHC0

Amazon 的 AMI、視窗 AMI 和大多數合作夥伴 AMI 將您的執行個體設定為預設使用 NTP IPv4 端點。這是大多數客戶工作負載的建議設定。除非您要使用 IPv6 端點或直接連接到 PTP 硬體時鐘，否則從這些 AMI 啟動的執行個體不需要進一步的設定。

NTP 和 PTP 連線不需要變更任何 VPC 組態，而且執行個體不需要存取網際網路。

### Note

只有 Linux 執行個體可以使用直接 PTP 連線來連線到本機 PTP 硬體時鐘。Windows 執行個體會使用 NTP 連線到本機 PTP 硬體時鐘。

### 主題

- [連接至 Amazon Time Sync Service 的 IPv4 端點](#)
- [連接至 Amazon Time Sync Service 的 IPv6 端點](#)
- [連接至 PTP 硬體時鐘](#)

### 連接至 Amazon Time Sync Service 的 IPv4 端點

本節說明如何將執行個體設定為透過 IPv4 端點使用本機 Amazon Time Sync Service。

請遵循於執行個體作業系統的指示。

### Linux

預設情況下，AL2023 和最新版的 Amazon Linux 2 與 Amazon Linux AMI 設定為使用 Amazon Time Sync Service IPv4 端點。從這些 AMI 啟動的執行個體不需要進一步的組態，您可以略過下列程序。

如果您使用的 AMI 未依預設設定 Amazon Time Sync Service，請按照下列其中一個程序，使用 chrony 用戶端在執行個體上設定 Amazon Time Sync Service。須編輯 chrony 組態檔，新增 Amazon Time Sync Service 的伺服器項目。

請遵循於執行個體作業系統的指示。

## Amazon Linux

使用 chrony 連接至 Amazon Linux 上 Amazon Time Sync Service 的 IPv4 端點

1. 連線到您的執行個體，然後解除安裝 NTP 服務。

```
[ec2-user ~]$ sudo yum erase 'ntp*'
```

2. 安裝 chrony 套裝服務。

```
[ec2-user ~]$ sudo yum install chrony
```

3. 使用文字編輯器 (例如 /etc/chrony.conf 或 vim) 開啟 nano 檔案。確認檔案是否包含下列程式碼行：

```
server 169.254.169.123 prefer iburst minpoll 4 maxpoll 4
```

如果有這行內容，表示 Amazon Time Sync Service 已設定為使用 Amazon Time Sync Service IPv4 端點，您可前往下一個步驟。如果沒有，請將該行程式碼新增在檔案中已有之任何其他 server 或 pool 陳述式之後，然後儲存您的變更。

4. 重新啟動 chrony 協助程式 (chronyd)。

```
[ec2-user ~]$ sudo service chronyd restart
```

```
Starting chronyd: [OK]
```

### Note

在 RHEL 和 CentOS (最新為 6 版) 上，服務名稱為 chrony，而不是 chronyd。

5. 使用 chkconfig 命令設定 chronyd 在每次系統開機時啟動。

```
[ec2-user ~]$ sudo chkconfig chronyd on
```

## 6. 確認 chrony 是否使用 169.254.169.123 IPv4 端點來同步時間。

```
[ec2-user ~]$ chronyc sources -v
```

```
210 Number of sources = 7
```

```

 .-- Source mode '^' = server, '=' = peer, '#' = local clock.
 / .- Source state '*' = current synced, '+' = combined , '-' = not
combined,
 | / '?' = unreachable, 'x' = time may be in error, '~' = time too
variable.
 || .- xxxx [yyyy] +/-
zzzz
 || Reachability register (octal) -. | xxxx = adjusted
offset,
 || Log2(Polling interval) --. | | yyyy = measured
offset,
 || \ | | zzzz = estimated
error.
 || | | \
 MS Name/IP address Stratum Poll Reach LastRx Last sample
=====
 ^* 169.254.169.123 3 6 17 43 -30us[-226us] +/-
287us
 ^- ec2-12-34-231-12.eu-west> 2 6 17 43 -388us[-388us] +/-
11ms
 ^- tshirt.heanet.ie 1 6 17 44 +178us[+25us] +/-
1959us
 ^? tbag.heanet.ie 0 6 0 - +0ns[+0ns] +/-
0ns
 ^? bray.walcz.net 0 6 0 - +0ns[+0ns] +/-
0ns
 ^? 2a05:d018:c43:e312:ce77:> 0 6 0 - +0ns[+0ns] +/-
0ns
 ^? 2a05:d018:dab:2701:b70:b> 0 6 0 - +0ns[+0ns] +/-
0ns

```



在傳回的輸出中，`^*` 表示偏好的時間來源。

## 7. 確認時間同步指標是否由 `chrony` 回報。

```
[ec2-user ~]$ chronyc tracking
```

```
Reference ID : A9FEA97B (169.254.169.123)
 Stratum : 4
 Ref time (UTC) : Wed Nov 22 13:18:34 2017
 System time : 0.000000626 seconds slow of NTP time
 Last offset : +0.002852759 seconds
 RMS offset : 0.002852759 seconds
 Frequency : 1.187 ppm fast
 Residual freq : +0.020 ppm
 Skew : 24.388 ppm
 Root delay : 0.000504752 seconds
 Root dispersion : 0.001112565 seconds
 Update interval : 64.4 seconds
 Leap status : Normal
```

## Ubuntu

使用 `chrony` 連接至 Ubuntu 上 Amazon Time Sync Service 的 IPv4 端點

### 1. 連線至您的執行個體，然後使用 `apt` 來安裝 `chrony` 套件。

```
ubuntu:~$ sudo apt install chrony
```

#### Note

如有必要，請執行 `sudo apt update` 先更新您的執行個體。

### 2. 使用文字編輯器 (例如 `/etc/chrony/chrony.conf` 或 `vim`) 開啟 `nano` 檔案。將下列程式碼新增在檔案中已有之任何其他 `server` 或 `pool` 的陳述式之前，然後儲存您的變更：

```
server 169.254.169.123 prefer iburst minpoll 4 maxpoll 4
```

### 3. 重新啟動 `chrony` 服務。

```
ubuntu:~$ sudo /etc/init.d/chrony restart
```

```
Restarting chrony (via systemctl): chrony.service.
```

#### 4. 確認 chrony 是使用 169.254.169.123 IPv4 端點來同步時間。

```
ubuntu:~$ chronyc sources -v
```

```
210 Number of sources = 7

 .-- Source mode '^' = server, '=' = peer, '#' = local clock.
 / .- Source state '*' = current synced, '+' = combined , '-' = not
combined,
 | / '?' = unreachable, 'x' = time may be in error, '~' = time too
variable.
 || .- xxxx [yyyy]
+/- zzzz
 || Reachability register (octal) -. | xxxx =
adjusted offset,
 || Log2(Polling interval) --. | | yyyy =
measured offset,
 || \ | | zzzz =
estimated error.
 || | | \
 MS Name/IP address Stratum Poll Reach LastRx Last sample
=====
 ^* 169.254.169.123 3 6 17 12 +15us[+57us]
+/- 320us
 ^- tbag.heanet.ie 1 6 17 13 -3488us[-3446us]
+/- 1779us
 ^- ec2-12-34-231-12.eu-west- 2 6 17 13 +893us[+935us]
+/- 7710us
 ^? 2a05:d018:c43:e312:ce77:6 0 6 0 10y +0ns[+0ns]
+/- 0ns
 ^? 2a05:d018:d34:9000:d8c6:5 0 6 0 10y +0ns[+0ns]
+/- 0ns
 ^? tshirt.heanet.ie 0 6 0 10y +0ns[+0ns]
+/- 0ns
```

```

^? bray.walcz.net 0 6 0 10y +0ns[+0ns]
+/- 0ns

```

在傳回的輸出中，以 `^*` 開頭的行表示偏好的時間來源。

#### 5. 確認時間同步指標是否由 `chrony` 回報。

```
ubuntu:~$ chronyc tracking
```

```

Reference ID : 169.254.169.123 (169.254.169.123)
Stratum : 4
Ref time (UTC) : Wed Nov 29 07:41:57 2017
System time : 0.000000011 seconds slow of NTP time
Last offset : +0.000041659 seconds
RMS offset : 0.000041659 seconds
Frequency : 10.141 ppm slow
Residual freq : +7.557 ppm
Skew : 2.329 ppm
Root delay : 0.000544 seconds
Root dispersion : 0.000631 seconds
Update interval : 2.0 seconds
Leap status : Normal

```

## SUSE Linux

從 SUSE Linux Enterprise Server 15 開始，`chrony` 是 NTP 的預設實作。

使用 `chrony` 連接至 SUSE Linux 上 Amazon Time Sync Service 的 IPv4 端點

1. 使用文字編輯器 (例如 `/etc/chrony.conf` 或 `vim`) 開啟 `nano` 檔案。
2. 確認檔案是否包含下列程式碼行：

```
server 169.254.169.123 prefer iburst minpoll 4 maxpoll 4
```

如果此程式碼不存在，請新增它。

3. 將任何其他伺服器或集區的程式碼行變更為註解。
4. 開啟 `yast` 並啟用 `chrony` 服務。

## Windows

從 2018 年 8 月版本開始，Windows AMI 預設會使用 Amazon Time Sync Service。從這些 AMI 啟動的執行個體不需要進一步的設定，您可以略過下列程序。

如果您使用的 AMI 並未依預設設定 Amazon 時間同步服務，請先驗證您目前的 NTP 組態。如果您的執行個體已在使用 Amazon Time Sync Service 的 IPv4 端點，則無需進一步設定。如果您的執行個體不是使用 Amazon Time Sync Service，請完成程序，將 NTP 伺服器變更為使用 Amazon Time Sync Service。

### 確認 NTP 組態

1. 從執行個體中，開啟命令提示視窗。
2. 輸入下列命令，以取得目前 NTP 組態：

```
w32tm /query /configuration
```

此命令會傳回 Windows 執行個體目前的組態設定，並顯示是否已連接至 Amazon Time Sync Service。

3. (選用) 輸入下列命令，以取得目前組態的狀態：

```
w32tm /query /status
```

此命令會傳回資訊，例如執行個體最後一次與 NTP 伺服器同步的時間以及輪詢間隔。

### 將 NTP 伺服器改為使用 Amazon Time Sync Service

1. 從命令提示視窗，執行下列命令：

```
w32tm /config /manualpeerlist:169.254.169.123 /syncfromflags:manual /update
```

2. 使用下列命令，確認新設定：

```
w32tm /query /configuration
```

在傳回的輸出中，確認 NtpServer 顯示 169.254.169.123 IPv4 端點。

## Amazon Windows AMI 的預設網路時間通訊協定 (NTP) 設定

Amazon 機器映像 (AMI) 通常會遵守 out-of-the-box 預設值，除非需要變更才能在 EC2 基礎設施上運作。下列設定已確認在虛擬化環境中運作良好，以及可將任何時鐘漂移保持在一秒的準確度內：

- 更新間隔 — 控制時間服務調整系統時間以達到精確度的頻率。AWS 設定每兩分鐘進行一次更新間隔。
- NTP 伺服器：從 2018 年 8 月發行版本開始，AMI 預設使用 Amazon Time Sync Service。您可以從任何 AWS 區域 位於 IPv4 端點上的任何一個端點存取此次服務。此外，0x9 標記表示時間服務正作為用戶端，並使用 SpecialPollInterval 來判定以設定的時間伺服器進行查入的頻率。
- 類型 – “NTP” 表示服務將充當獨立式 NTP 用戶端，而不是充當網域的一部分。
- [啟用] 和 InputProvider — 啟用時間服務，並提供作業系統的時間。
- 特殊輪詢間隔 — 每 900 秒 (15 分鐘) 檢查一次設定的 NTP 伺服器。

登錄檔路徑	金鑰名稱	資料
港島樓 M:\System\ CurrentControl 設置\ 服務\ w32 時間\ Config	UpdateInterval	120
港島樓 M:\System\ CurrentControl 設置\ 服務\ w32 時間\ 參數	NtpServer	169.254.169.123,0x9
港島樓 M:\System\ CurrentControl 設置\ 服務\ w32 時間\ 參數	Type	NTP
港島樓 M:\System\ CurrentControl 設置\ 服務\ w32 時間\ TimeProviders NtpClient	已啟用	1
港島樓 M:\System\ CurrentControl 設置\ 服務\ w32 時間\ TimeProviders NtpClient	InputProvider	1

登錄檔路徑	金鑰名稱	資料
港島樓 M:\System\ CurrentControl 設置\ 服務\ w32 時間\ TimeProviders NtpClient	SpecialPoll間隔	900

連接至 Amazon Time Sync Service 的 IPv6 端點

本節說明如要設定執行個體以透過 IPv6 端點使用本機 Amazon Time Sync Service，步驟與 [連接至 Amazon Time Sync Service 的 IPv4 端點](#) 所述有何不同。它不會說明整個 Amazon Time Sync Service 組態程序。

IPv6 端點只能在 [AWS Nitro 系統上建置的執行個體上](#) 存取。

#### Note

我們不建議同時使用 IPv4 和 IPv6 端點項目。IPv4 和 IPv6 NTP 封包來自您的執行個體的相同本機伺服器。同時設定 IPv4 和 IPv6 端點是不必要的做法，而且不會改善執行個體的時間準確性。

請遵循於執行個體作業系統的指示。

## Linux

根據您使用的 Linux 發行版，當您到達編輯時間 .conf 檔案的步驟時，您將使用 Amazon 時間同步服務 (fd00:ec2::123) 的 IPv6 端點，而不是 IPv4 端點 ( ) : 169.254.169.123

```
server fd00:ec2::123 prefer iburst minpoll 4 maxpoll 4
```

儲存檔案，並確認 chrony 是使用 fd00:ec2::123 IPv6 端點來同步時間：

```
[ec2-user ~]$ chronyc sources -v
```

在輸出中，如果有看到 fd00:ec2::123 IPv6 端點，表示組態已完成。

## Windows

當您到達將 NTP 伺服器變更為使用 Amazon 時間同步服務的步驟時，您將使用 Amazon 時間同步服務 (fd00:ec2::123) 的 IPv6 端點，而不是 IPv4 端點 (169.254.169.123)：

```
w32tm /config /manualpeerlist:fd00:ec2::123 /syncfromflags:manual /update
```

確認您的新設定正在使用 fd00:ec2::123 IPv6 端點同步處理時間：

```
w32tm /query /configuration
```

在輸出中，確認是否NtpServer顯示 fd00:ec2::123 IPv6 端點。

## 連接至 PTP 硬體時鐘

PTP 硬體時鐘是 [AWS Nitro 系統](#) 的一部分，因此可在 [支援的裸機和虛擬化 EC2 執行個體](#) 上直接存取，不會使用任何客戶資源。

PTP 硬體時鐘的 NTP 端點與透過 IPv4 或 IPv6 進行的一般 Amazon Time Sync Service 連線相同。如果軟體設定為 NTP 端點，且在具有 PTP 硬體時鐘的執行個體上執行，便會透過 NTP 自動連接到 PTP 硬體時鐘。

## 要求

符合下列需求才能在執行個體上使用 PTP 硬體時鐘：

- 支援 AWS 區域：美國東部 (維吉尼亞北部) 和亞太區域 (東京)
- 支援的執行個體系列：
  - 一般用途：兆 7 安培，兆 7 克，兆平方米，M7
  - 計算最佳化：C7 安培、C7gD、C7i
  - 最佳化記憶體：7 安培、R7 克、R7 公升
- (僅限 Linux) ENA 驅動程式 2.10.0 版或更新版本安裝在支援的作業系統上。如需有關支援作業系統的詳細資訊，請參閱的驅動程式[先決條件GitHub](#)。

請遵循於執行個體作業系統的指示。

## Linux

本節說明如何使用直接 PTP 連線，將執行個體設定為透過 PTP 硬體時鐘使用本機 Amazon Time Sync Service。需將 PTP 硬體時鐘的伺服器項目新增至 chrony 組態檔。

如果您的執行個體具有 PTP 硬體時鐘，而且您已設定 NTP 連線 (連接 IPv4 或 IPv6 端點)，執行個體時間便會自動使用 PTP 硬體時鐘。以下步驟設定直接 PTP 連線，可提供比 NTP 連線更準確的時間。

## 連接至 PTP 硬體時鐘

1. 連接至執行個體，並安裝適用於彈性網路介面卡 (ENA) 2.10.0 版或更新版本的 Linux 核心驅動程式。如需安裝指示，請參閱上的[彈性網路介面卡 \(ENA\) 系列的 Linux 核心驅動程式GitHub](#)。
2. 確認 `/dev/ptp0` 裝置顯示在執行個體上。

```
[ec2-user ~]$ ls /dev/ptp0
```

預期的輸出如下：如果 `/dev/ptp0` 不在輸出中，表示 ENA 驅動程式未正確安裝。請參閱此程序中的步驟 1 以安裝驅動程式。

```
/dev/ptp0
```

3. 使用文字編輯器編輯 `/etc/chrony.conf`，然後在檔案中的任何位置新增以下行。

```
refclock PHC /dev/ptp0 poll 0 delay 0.000010 prefer
```

4. 使用以下命令重新啟動 `chrony`。

```
[ec2-user ~]$ sudo systemctl restart chronyd
```

5. 確認 `chrony` 是使用 PTP 硬體時鐘來同步此執行個體上的時間。

```
[ec2-user ~]$ chronyc sources
```

預期的輸出結果

```
MS Name/IP address Stratum Poll Reach LastRx Last sample
=====
#* PHC0 0 0 377 1 +2ns[+1ns] +/- 5031ns
```

在傳回的輸出中，\* 表示偏好的時間來源。PHC0 對應 PTP 硬體時鐘。重新啟動 `chrony` 之後，可能需要等待幾秒鐘，星號才會出現。

## Windows

Windows 執行個體僅支援與本機 PTP 硬體時鐘的 NTP 連線。



PTP 硬體時鐘的 NTP 端點與透過 IPv4 或 IPv6 連接的一般 Amazon Time Sync Service 連線相同。如果軟體設定為連接到 NTP 端點，且在具有 PTP 硬體時鐘的執行個體上執行，便會透過 NTP 自動連接到 PTP 硬體時鐘。

將您的執行個體或任何網際網路連線裝置設定為使用公用的 Amazon Time Sync Service

您可以將執行個體或任何連接網際網路的裝置 (例如本機電腦或內部部署伺服器) 設定為使用可透過網際網路從 `time.aws.com` 存取的公有 Amazon Time Sync Service。您可以使用公有的 Amazon 時間同步服務做為本機 Amazon 時間同步服務的備份，以及將資源連接 AWS 到 Amazon 時間同步服務以外的資源。

### Note

為了獲得最佳效能，我們建議您在執行個體上使用本機 Amazon 時間同步服務，並且僅使用公有 Amazon 時間同步服務做為備份。

使用執行個體或裝置作業系統的指示。

## Linux

若要將 Linux 執行個體或裝置設定為使用公有 Amazon Time Sync Service，請使用 `chrony` 或 `ntpd`

1. 使用文字編輯器編輯 `/etc/chrony.conf` (若使用 `chrony`) 或 `/etc/ntp.conf` (若使用 `ntpd`)，如下所示：
  - a. 若要防止執行個體或裝置嘗試混合已消彌和未消彌的伺服器，請刪除以 `server` 開頭的行或變成註解，但連接本機 Amazon Time Sync Service 的任何現有連線除外。

### Important

如果要將 EC2 執行個體設定為連接到公有 Amazon Time Sync Service，請勿移除以下行，這一行會將執行個體設定為連接到本機 Amazon Time Sync Service。本機 Amazon Time Sync Service 是更直接的連線，可提供更高的時鐘準確度。公有 Amazon Time Sync Service 只能當成備案。

```
server 169.254.169.123 prefer iburst minpoll 4 maxpoll 4
```

- b. 新增以下行以連接至公有 Amazon Time Sync Service。

```
pool time.aws.com iburst
```

2. 使用以下其中一個命令重新啟動常駐程式。

- chrony

```
sudo service chronyd force-reload
```

- ntpd

```
sudo service ntp reload
```

## macOS

將您的 macOS 執行個體或裝置設定為使用公有 Amazon Time Sync Service

1. 開啟 System Preferences (系統偏好設定)。
2. 選擇 Date & Time (日期與時間)，然後選擇 Date & Time (日期與時間) 標籤。
3. 若要進行變更，請選擇鎖定圖示，然後在出現提示時輸入您的密碼。
4. 對於 Set date and time automatically (自動設定日期與時間)，輸入 **time.aws.com**。

## Windows

將您的 Windows 執行個體或裝置設定為使用公有 Amazon Time Sync Service

1. 開啟 Control Panel (主控台)。
2. 選擇 Date and Time (日期與時間) 圖示。
3. 選擇 Internet Time (網際網路時間) 標籤。如果您的 PC 是網域的一部分，就無法使用此索引標籤。在這種情況下，系統會與網域控制器同步時間。您可以將控制器設定為使用公有 Amazon Time Sync Service。
4. 選擇 Change settings (變更設定)。
5. 選取 Synchronize with an Internet time server (與網際網路時間伺服器同步) 的核取方塊。
6. 在 Server (伺服器) 旁，輸入 **time.aws.com**。

## 將您的 Windows Server 執行個體或裝置設定為使用公有 Amazon Time Sync Service

- 遵照 [Microsoft 的指示](#)，更新您的登錄檔。

## 比較 Linux 執行個體的時間戳記

如果您使用的是 Amazon 時間同步服務，則可以將 Amazon EC2 Linux 執行個體上的時間戳記與進行比較，ClockBound 以判斷事件的真實時間。ClockBound 測量 EC2 執行個體的時鐘準確性，並允許您檢查指定的時間戳記是否處於過去或 future 與執行個體的當前時鐘相關。此資訊對於判斷 EC2 執行個體之間事件和交易的順序和一致性非常有用，而且與每個執行個體的地理位置無關。

ClockBound 是一個開源守護進程和庫。要了解更多內容 ClockBound，包括安裝說明，敬請參閱 [ClockBound](#) (詳見) GitHub。

ClockBound 僅支援 Linux 執行個體。

如果您正在使用連接 PTP 硬體時鐘的直接 PTP 連線，您的時間常駐程式 (例如 chrony) 將低估時鐘誤差範圍。這是因為 PTP 硬體時鐘不會像 NTP 那樣傳遞正確的誤差範圍資訊至 chrony。因此，時鐘同步處理常駐程式會假設時鐘是根據 UTC 調整準確度，因此會有誤差範圍 0。為了測量完整的錯誤限制，Nitro 系統會計算 PTP 硬體時鐘的錯誤限制，並透過 ENA 驅動程式sysfs檔案系統將其提供給 EC2 執行個體。您可以直接將其作為一個值讀取，以納秒為單位。

### 若要擷取 PTP 硬體時脈錯誤繫結

1. 首先使用下列指令之一，取得 PTP 硬體時鐘裝置的正確位置。根據用來啟動執行個體的 AMI，命令中的路徑會有所不同。
  - 針對 Amazon Linux 2 :

```
cat /sys/class/net/eth0/device/uevent | grep PCI_SLOT_NAME
```

- 對於 Amazon

```
cat /sys/class/net/ens5/device/uevent | grep PCI_SLOT_NAME
```

輸出是 PCI 插槽名稱，也就是 PTP 硬體時鐘裝置的位置。在此範例中，位置為0000:00:03.0。

```
PCI_SLOT_NAME=0000:00:03.0
```

- 若要擷取 PTP 硬體時鐘錯誤繫結，請執行下列命令。包括上一個步驟中的 PCI 插槽名稱。

```
cat /sys/bus/pci/devices/0000:00:03.0/phc_error_bound
```

輸出即為 PTP 硬體時鐘的時鐘誤差範圍 (以奈秒為單位)。

若要計算在使用 PTP 硬體時脈的直接 PTP 連線時，在特定時間點的正確時脈誤差，您必須新增查詢 PTP 硬體時脈的時脈錯誤，chrony或是 ClockBound 在chrony輸出 PTP 硬體時脈時間的時脈錯誤。如需有關測量和監控時脈準確度的詳細資訊，請參閱[使用 Amazon 時間同步服務和 Amazon 管理 Amazon 執行個體時鐘準確性 CloudWatch — 第 1 部分](#)。

## 變更執行個體的時區

依預設，Amazon EC2 執行個體會設定為 UTC (國際標準時間) 的時區。您可以將執行個體上的時間變更為當地時區或網路中的另一個時區。

請遵循於執行個體作業系統的指示。

### Linux

#### Important

此資訊適用於 Amazon Linux。如需其他分發的詳細資訊，請參閱其特定文件。

## 變更 AL2023 或 Amazon Linux 2 執行個體上的時區

- 檢視系統目前的時區設定。

```
[ec2-user ~]$ timedatectl
```

- 列出可用的時區。

```
[ec2-user ~]$ timedatectl list-timezones
```

- 設定選擇的時區。

```
[ec2-user ~]$ sudo timedatectl set-timezone America/Vancouver
```

- (選用) 再次執行 timedatectl 命令確認目前的時區已更新為新的時區。

```
[ec2-user ~]$ timedatectl
```

## 變更 Amazon Linux 執行個體上的時區

1. 找出執行個體使用的時區。/usr/share/zoneinfo 目錄包含時區資料檔案的階層。瀏覽該位置的目錄結構尋找您時區的檔案。

```
[ec2-user ~]$ ls /usr/share/zoneinfo
Africa Chile GB Indian Mideast posixrules US
America CST6CDT GB-Eire Iran MST PRC UTC
Antarctica Cuba GMT iso3166.tab MST7MDT PST8PDT WET
Arctic EET GMT0 Israel Navajo right W-SU
...
```

此位置有些項目是目錄 (例如 America)，這些目錄包含特定城市的時區檔案。找出用於執行個體的所在城市 (或所在時區的城市)。

2. 使用新的時區更新 /etc/sysconfig/clock 檔案。在此範例中，我們使用洛杉磯的時區資料檔案：/usr/share/zoneinfo/America/Los\_Angeles。
  - a. 使用文字編輯器 (例如 vim 或 nano) 開啟 /etc/sysconfig/clock 檔案。您需要使用 sudo 搭配編輯器命令，因為 /etc/sysconfig/clock 的擁有者是 root。

```
[ec2-user ~]$ sudo nano /etc/sysconfig/clock
```

- b. 找到 ZONE 項目，將它變更為時區檔案 (省略路徑的 /usr/share/zoneinfo 區段)。例如，若要變更至洛杉磯，請將 ZONE 項目變更成：

```
ZONE="America/Los_Angeles"
```

### Note

UTC=true 項目不要變更成其他值。此項目用於硬體時鐘，當您在執行個體上設定不同時區時不需要調整。

- c. 儲存檔案，然後結束文字編輯器。
3. 建立 /etc/localtime 和您時區檔案之間的符號連結，以便執行個體參考本機時間資訊時尋找時間檔案。

```
[ec2-user ~]$ sudo ln -sf /usr/share/zoneinfo/America/Los_Angeles /etc/localtime
```

4. 重新開機系統以在所有服務和應用程式中挑選新的時區資訊。

```
[ec2-user ~]$ sudo reboot
```

5. (選用) 使用 `date` 命令確認目前的時區已更新為新的時區。目前的時區會出現在輸出中。在下列範例中，目前的時區是 PDT，它指的是洛杉磯時區。

```
[ec2-user ~]$ date
Sun Aug 16 05:45:16 PDT 2020
```

## Windows

### 變更 Windows 執行個體的時區

1. 從執行個體中，開啟命令提示視窗。
2. 找出執行個體使用的時區。若要取得時區清單，請使用下列命令：

```
tzutil /l
```

此命令會傳回所有可用時區的清單，格式如下：

```
display name
time zone ID
```

3. 找出要指派給執行個體的時區 ID。
4. 使用下列命令指派其他時區：

```
tzutil /s "Pacific Standard Time"
```

新的時區應該會立即生效。

#### Note

您可以使用下列命令來指派 UTC 時區：

```
tzutil /s "UTC"
```

為了防止您的時區在您為 Windows 伺服器設定之後變更

當您變更 Windows 執行個體上的時區時，必須確定在系統重新啟動之後仍然會持續保留時區。否則，執行個體重新啟動時，會回復成使用 UTC 時間。您可以透過新增 RealTimeIsUniversal 登錄機碼來保留您的時區設定。依預設，此機碼會在所有最新一代的執行個體上設定。若要確認是否已設定 RealTimeIsUniversal 登錄機碼，請參閱下列程序中的步驟 4。如果未設定機碼，請從頭開始執行下列步驟。

若要設定 RealTimeIsUniversal 登錄機碼

1. 從執行個體中，開啟命令提示視窗。
2. 使用下列命令，新增登錄機碼：

```
reg add "HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\TimeZoneInformation" /v RealTimeIsUniversal /d 1 /t REG_DWORD /f
```

3. 如果您使用的是在 2013 年 2 月 22 日之前建立的視窗伺服器 2008 AMI (不是視窗伺服器 2008 R2)，我們建議您更新至最新的 AWS 視窗 AMI。如果您使用的是執行 Windows Server 2008 R2 (「不是」 Windows Server 2008) 的 AMI，您必須確認 Microsoft Hotfix [KB2922223](#) 已確實安裝。如果未安裝此修補程式，我們建議您更新至最新的 AWS Windows AMI。
4. (選用) 使用下列命令，確認執行個體已順利儲存機碼：

```
reg query "HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\TimeZoneInformation" /s
```

此命令會傳回 TimeZoneInformation 登錄機碼的子機碼。您應該會在清單底端看到 RealTimeIsUniversal 機碼，類似以下：

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\TimeZoneInformation
 Bias REG_DWORD 0x1e0
 DaylightBias REG_DWORD 0xffffffffc4
 DaylightName REG_SZ @tzres.dll,-211
 DaylightStart REG_BINARY 00000300020002000000000000000000
 StandardBias REG_DWORD 0x0
 StandardName REG_SZ @tzres.dll,-212
```

StandardStart	REG_BINARY	00000B00010002000000000000000000
TimeZoneKeyName	REG_SZ	Pacific Standard Time
DynamicDaylightTimeDisabled	REG_DWORD	0x0
ActiveTimeBias	REG_DWORD	0x1a4
RealTimeIsUniversal	REG_DWORD	0x1

## 閏秒

1972 年推出的閏秒是為了消彌由於地球自轉速度不穩定，導致國際原子時間 (TAI) 與太陽時間 (Ut1) 之間產生的差異，而不定期校正 UTC 時間的一秒鐘。為了代表客戶管理閏秒，我們在 Amazon Time Sync Service 中設計了閏秒消彌機制。如需詳細資訊，請參閱[閏秒搶先報：即將到來的閏秒和 AWS](#)。

[國際度量衡局第 27 屆大會決議，將在 2035 年之前取消閏秒](#)，我們全力支持這項決議。

為了支援這次轉變，我們仍計劃透過本機 NTP 連線或公有 NTP 集區 (time.aws.com) 存取 Amazon Time Sync Service 時，在閏秒事件期間消彌時間。不過 PTP 硬體時鐘不提供消彌時間選項。如果發生閏秒，PTP 硬體時鐘會依照 UTC 標準加上閏秒。在大多數情況下，消彌的閏秒和閏秒時間來源相同。但由於閏秒事件期間會有所不同，所以不建議在閏秒事件期間，在您的時間用戶端組態中同時使用消彌和未消彌的時間來源。

## 相關資源

- AWS 運算部落格：[時間到了：Amazon EC2 執行個體上的微秒精確時鐘](#)
- (Linux) <https://chrony-project.org/>
- (視窗) [視窗時間服務如何運作](#) (Microsoft)
- (視窗) [W32TM \(Microsoft\)](#)
- (視窗) [視窗時間服務如何處理閏秒](#) (Microsoft)
- (視窗) [關於閏秒和視窗的故事：它很可能不是 Y2K](#) (Microsoft)

## Amazon EC2 Linux 執行個體的處理器狀態控制

C-state 可控制核心在閒置狀態要進入的休眠等級。C-state 從 C0 (最淺閒置狀態，此時核心處於喚醒狀態並執行指令) 開始編號，最高可到 C6 (最深閒置狀態，此時核心會關閉)。

P-state 則可控制所需的核能效能 (CPU 頻率)。P-state 從 P0 (最高效能設定，此時核心可視需要使用 Intel Turbo Boost 技術來提高頻率) 開始編號，之後則從 P1 (請求最大基準頻率的 P-state) 到 P15 (最低適用頻率)。



## C-state 和 P-state

下列執行個體類型可供作業系統控制處理器的 C-state 和 P-state：

- 一般用途：m4.10xlarge|m4.16xlarge|m5.metal|m5d.metal|m5n.metal|m5zn.metal|m6i.metal|m6id.metal|m7a.metal-48x1|m7i.metal-24x1|m7i.metal-48x1
- 運算最佳化：c4.8xlarge|c5.metal|c5an.metal|c5adn.metal|c5n.metal|c6i.metal|c6id.metal|c7a.metal-48x1|c7i.metal-24x1|c7i.metal-48x1
- 記憶體最佳化：r4.8xlarge|r4.16xlarge|r5.metal|r5b.metal|r5d.metal|r6i.metal|r7a.metal-48x1|r7i.metal-24x1|r7i.metal-48x1|r7iz.metal-16x1|r7iz.metal-32x1|u-6tb1.metal|u-9tb1.metal|u-12tb1.metal|u-18tb1.metal|u-24tb1.metal|x1.16xlarge|x1.32xlarge|x1e.8xlarge|x1e.16xlarge|x1e.32xlarge|z1d.metal
- 儲存最佳化：d2.8xlarge|d3.metal|d3en.metal|i3.8xlarge|i3.16xlarge|i3.metal|i3en.metal|h1.8xlarge|h1.16xlarge
- 加速運算：f1.16xlarge|g3.16xlarge|g4dn.metal|p2.16xlarge|p3.16xlarge

### 僅 C-state

下列執行個體類型可供作業系統控制處理器的 C-state：

- 一般用途：m5.12xlarge|m5.24xlarge|m5d.12xlarge|m5d.24xlarge|m5n.12xlarge|m5n.24xlarge|m5dn.12xlarge|m5dn.24xlarge|m6a.24xlarge|m6a.48xlarge|m6ad.metal|m6i.16xlarge|m6i.32xlarge|m7a.medium|m7a.large|m7a.xlarge|m7a.2xlarge|m7a.4xlarge|m7a.8xlarge|m7a.12xlarge|m7a.16xlarge|m7a.24xlarge|m7a.32xlarge|m7a.48xlarge|m7i.large|m7i.xlarge|m7i.2xlarge|m7i.4xlarge|m7i.8xlarge|m7i.12xlarge|m7i.16xlarge|m7i.24xlarge|m7i.48xlarge
- 運算最佳化：c5.9xlarge|c5.12xlarge|c5.18xlarge|c5.24xlarge|c5a.24xlarge|c5ad.24xlarge|c5d.9xlarge|c5d.12xlarge|c5d.18xlarge|c5d.24xlarge|c5n.9xlarge|c5n.18xlarge|c6a.24xlarge|c6a.32xlarge|c6a.48xlarge|c6i.16xlarge|c6i.32xlarge|c7a.medium|c7a.large|c7a.xlarge|c7a.2xlarge|c7a.4xlarge|c7a.8xlarge|c7a.12xlarge|c7a.16xlarge|c7a.24xlarge|c7a.32xlarge|c7a.48xlarge|c7i.large|c7i.xlarge|c7i.2xlarge|c7i.4xlarge|c7i.8xlarge|c7i.12xlarge|c7i.16xlarge|c7i.24xlarge|c7i.48xlarge

- 記憶體最佳

化 : r5.12xlarge r5.24xlarge r5d.12xlarge r5d.24xlarge r5n.12xlarge r5n.24xlarge r5dn.12xlarge r5dn.24xlarge r6a.24xlarge | r6a.48xlarge | r6i.16xlarge | r6i.32xlarge r6id.32xlarge | r6in.32xlarge | r7a.medium | r7a.large r7a.xlarge | r7a.2xlarge | r7a.4xlarge | r7a.8xlarge | r7a.12xlarge r7a.16xlarge | r7a.24xlarge | r7a.32xlarge | r7a.48xlarge r7i.large | r7i.xlarge | r7i.2xlarge | r7i.4xlarge | r7i.8xlarge r7i.12xlarge | r7i.16xlarge | r7i.24xlarge | r7i.48xlarge r7iz.large | r7iz.xlarge | r7iz.2xlarge | r7iz.4xlarge | r7iz.8xlarge | r7iz.12xlarge | r7iz.16xlarge | r7iz.32xlarge | u-6tb1.56xlarge | u-6tb1.112xlarge | u-9tb1.112xlarge | u-12tb1.112xlarge | u-18tb1.112xlarge | u-24tb1.112xlarge | u7i-12tb.224xlarge | u7in-16tb.224xlarge | u7in-24tb.224xlarge | u7in-32tb.224xlarge | z1d.6xlarge | z1d.12xlarge

- 儲存最佳化 : d3en.12xlarge | dl1.24xlarge | i3en.12xlarge | i3en.24xlarge | i4i.metal | r5b.12xlarge | r5b.24xlarge | i4i.16xlarge

- 加速運

算 : dl1.24xlarge g5.24xlarge g5.48xlarge g6.24xlarge g6.48xlarge inf1.24xlarge p3dn.24xlarge p4de.24xlarge | vt1.24xlarge

AWS Graviton 處理器具有內建省電模式，並以固定頻率運作。因此，它們不能為作業系統提供控制 C-state 和 P-states 的能力。

建議您變更 C-state 或 P-state 設定來提高處理器效能穩定性、減少延遲或針對特定工作負載微調執行個體。預設 C-state 和 P-state 設定提供最大效能，適合多數工作負載使用。然而，若您的應用程式更適合犧牲較高的單核心或雙核心頻率以降低延遲，或在較低頻率更能展現穩定效能 (而不適合使用突增 Turbo Boost 頻率)，請考慮嘗試調整這些執行個體的 C-state 或 P-state 設定。

如需有關不同處理器組態以及如何監控 [Amazon Linux 組態效果的詳細資訊](#)，請參閱 [Amazon Linux 2 使用者指南](#) 中的 [Amazon Amazon EC2 Linux 執行個體的處理器狀態控制](#)。這些程序是針對 Amazon Linux 撰寫的，並適用於 Amazon Linux；不過，它們也可能適用於其他 Linux 核心為 3.9 或更新版本的 Linux 發行版。如需其他 Linux 發行版本和處理器狀態控制的詳細資訊，請參閱您的系統特定文件。

## 最佳化 CPU 選項

許多 Amazon EC2 執行個體支援同時多執行緒，此種技術可在單一 CPU 核心上同時執行多個執行緒。每個執行緒皆代表執行個體上的一個虛擬 CPU (vCPU)。執行個體有預設數量的 CPU 核心，該數量因執行個體類型而異。例如，m5.xlarge 執行個體類型有兩個 CPU 核心，每個核心有兩個執行緒—因此預設共有四個 vCPU。

**Note**

每個 vCPU 都是 CPU 核心的一個執行緒，但 T2 執行個體、M7a 執行個體、Apple Silicon Mac 執行個體和 64 位元 ARM 平台除外，例如採用 AWS Graviton 處理器技術的執行個體。

多數情況下都會有 Amazon EC2 執行個體類型，其有符合您工作負載的記憶體和 vCPU 數量組合。不過，您仍可指定下列的 CPU 選項，以針對特定的工作負載或商業需求將您的執行個體最佳化：

- **Number of CPU cores (CPU 核心數量)**：您可自訂執行個體的 CPU 核心數量。這種做法可針對有足夠 RAM 數量可執行記憶體密集型工作負載但 CPU 核心較少的執行個體，將軟體的授權成本最佳化。
- **Threads per core (每核心的執行緒數量)**：您可以為每個 CPU 核心指定單一執行緒，來停用多執行緒。這種做法適用於特定的工作負載，例如高效能運算 (HPC) 的工作負載。

您可在執行個體啟動期間指定這些 CPU 選項。指定 CPU 選項不會增減費用。您要支付的費用就跟使用預設 CPU 選項啟動的執行個體一樣。

## 內容

- [指定 CPU 選項的規則](#)
- [各執行個體類型的 CPU 核心和每 CPU 核心的執行緒](#)
- [為執行個體指定 CPU 選項](#)
- [檢視執行個體的 CPU 選項](#)

## 指定 CPU 選項的規則

若要為執行個體指定 CPU 選項，請注意下列規則：

- 您無法指定裸機執行個體的 CPU 選項。
- CPU 選項只能在執行個體啟動期間指定，啟動後即無法修改。
- 當您啟動執行個體時，您必須在要求中同時指定 CPU 核心和執行緒的數量。如需請求範例，請參閱 [為執行個體指定 CPU 選項](#)。
- 執行個體的 vCPU 數目是 CPU 核心數量乘以每核心的執行緒數量。若要指定自訂數量的 vCPU，您必須指定對該執行個體類型有效的 CPU 核心和每核心執行緒數量。不得超過執行個體預設的 vCPU 數量。如需詳細資訊，請參閱 [各執行個體類型的 CPU 核心和每 CPU 核心的執行緒](#)。

- 若要停用多執行緒，請針對每個核心指定一個執行緒。
- 當現有的執行個體變更執行個體類型時，CPU 選項將自動變更為該新執行個體類型的預設 CPU 選項。
- 您指定的 CPU 選項在執行個體停止、啟動或重新啟動後仍會保留。

## 各執行個體類型的 CPU 核心和每 CPU 核心的執行緒

下表列出支援指定 CPU 選項的執行個體類型。

### 目錄

- [一般用途執行個體](#)
- [運算最佳化執行個體](#)
- [記憶體最佳化執行個體](#)
- [儲存最佳化執行個體](#)
- [加速運算執行個體](#)
- [高效能運算執行個體](#)

### 一般用途執行個體

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m2.xlarge	2	2	1	1、2	1
m2.2xlarge	4	4	1	1、2、3、4	1
m2.4xlarge	8	8	1	1、2、3、4、5、6、7、8	1
m3.large	2	1	2	1	1、2
m3.xlarge	4	2	2	1、2	1、2
m3.2xlarge	8	4	2	1、2、3、4	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m4.large	2	1	2	1	1、2
m4.xlarge	4	2	2	1、2	1、2
m4.2xlarge	8	4	2	1、2、3、4	1、2
m4.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
m4.10xlarge	40	20	2	2、4、6、8、10、12、14、16、18、20	1、2
m4.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
m5.large	2	1	2	1	1、2
m5.xlarge	4	2	2	2	1、2
m5.2xlarge	8	4	2	2、4	1、2
m5.4xlarge	16	8	2	2、4、6、8	1、2
m5.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m5.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
m5.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
m5.24xlarge	96	48	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
m5a.large	2	1	2	1	1、2
m5a.xlarge	4	2	2	2	1、2
m5a.2xlarge	8	4	2	2、4	1、2
m5a.4xlarge	16	8	2	2、4、6、8	1、2
m5a.8xlarge	32	16	2	4、6、8、10、12、14、16	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m5a.12xlarge	48	24	2	6、12、18、24	1、2
m5a.16xlarge	64	32	2	8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
m5a.24xlarge	96	48	2	12、18、24、36、48	1、2
m5ad.large	2	1	2	1	1、2
m5ad.xlarge	4	2	2	2	1、2
m5ad.2xlarge	8	4	2	2、4	1、2
m5ad.4xlarge	16	8	2	2、4、6、8	1、2
m5ad.8xlarge	32	16	2	4、6、8、10、12、14、16	1、2
m5ad.12xlarge	48	24	2	6、12、18、24	1、2
m5ad.16xlarge	64	32	2	8、10、12、14、16、18、20、22、24、26、28、30、32	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m5ad.24xlarge	96	48	2	12、18、24、36、48	1、2
m5d.large	2	1	2	1	1、2
m5d.xlarge	4	2	2	2	1、2
m5d.2xlarge	8	4	2	2、4	1、2
m5d.4xlarge	16	8	2	2、4、6、8	1、2
m5d.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
m5d.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
m5d.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2



執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m5d.24xlarge	96	48	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
m5dn.large	2	1	2	1	1、2
m5dn.xlarge	4	2	2	1、2	1、2
m5dn.2xlarge	8	4	2	2、4	1、2
m5dn.4xlarge	16	8	2	2、4、6、8	1、2
m5dn.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
m5dn.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
m5dn.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m5dn.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
m5n.large	2	1	2	1	1、2
m5n.xlarge	4	2	2	1、2	1、2
m5n.2xlarge	8	4	2	2、4	1、2
m5n.4xlarge	16	8	2	2、4、6、8	1、2
m5n.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
m5n.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
m5n.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m5n.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
m5zn.large	2	1	2	1	1、2
m5zn.xlarge	4	2	2	1、2	1、2
m5zn.2xlarge	8	4	2	2、4	1、2
m5zn.3xlarge	12	6	2	2、4、6	1、2
m5zn.6xlarge	24	12	2	2、4、6、8、10、12	1、2
m5zn.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
m6a.large	2	1	2	1	1、2
m6a.xlarge	4	2	2	1、2	1、2
m6a.2xlarge	8	4	2	1、2、3、4	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m6a.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
m6a.8xlarge	32	16	2	4、6、8、10、12、14、16	1、2
m6a.12xlarge	48	24	2	一、二、三、四、五、六、七、八、十六、二十四	1、2
m6a.16xlarge	64	32	2	四、六、八、十、十、十、十四、十六、三十二	1、2
m6a.24xlarge	96	48	2	四、六、八、十、十、十、十四、十六、三十八	1、2
m6a.32xlarge	128	64	2	八、十六、二十、二十、二十、二十四	1、2
m6a.48xlarge	192	96	2	八、十六、二十、二十、二十、二、三十六	1、2
m6g.large	2	2	1	1、2	1
m6g.xlarge	4	4	1	1、2、3、4	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m6g.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
m6g.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
m6g.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m6g.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m6g.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
m6gd.large	2	2	1	1、2	1
m6gd.xlarge	4	4	1	1、2、3、4	1
m6gd.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m6gd.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
m6gd.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1
m6gd.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1



執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m6gd.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
m6i.large	2	1	2	1	1、2
m6i.xlarge	4	2	2	1、2	1、2
m6i.2xlarge	8	4	2	2、4	1、2
m6i.4xlarge	16	8	2	2、4、6、8	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m6i.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
m6i.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
m6i.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
m6i.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m6i.32xlarge	128	64	2	二、四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、十六、四十、四、五十五、五十五、五十五、五十六、六十六、六十六、64	1、2
m6id.large	2	1	2	1	1、2
m6id.xlarge	4	2	2	1、2	1、2
m6id.2xlarge	8	4	2	2、4	1、2
m6id.4xlarge	16	8	2	2、4、6、8	1、2
m6id.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m6id.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
m6id.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
m6id.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m6id.32xlarge	128	64	2	二、四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四十六、四十、四、五十五、五十五、五十五、五十六、六十六、六十六、64	1、2
m6idn.large	2	1	2	1	1、2
m6idn.xlarge	4	2	2	1、2	1、2
m6idn.2xlarge	8	4	2	1、2、3、4	1、2
m6idn.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
m6idn.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m6idn.12xlarge	48	24	2	一、二、三、四、五、六、七、八、九、十、十、十、十、十、十、十、十、十、十、十、二、二、二、二、二、二、二、24	1、2
m6idn.16xlarge	64	32	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1、2
m6idn.24xlarge	96	48	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m6idn.32xlarge	128	64	2	四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、四十、四十、四十、五十五、五十五、五十五、五十六、六、六、六、六、六、64	1、2
m6in.large	2	1	2	1	1、2
m6in.xlarge	4	2	2	1、2	1、2
m6in.2xlarge	8	4	2	1、2、3、4	1、2
m6in.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
m6in.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m6in.12xlarge	48	24	2	一、二、三、四、五、六、七、八、九、十、十、十、十、十、十、十、十、十、十、十、二、二、二、二、二、二、二、24	1、2
m6in.16xlarge	64	32	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1、2
m6in.24xlarge	96	48	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2



執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m6in.32xlarge	128	64	2	四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四十、四十、四十、五十五、五十五、五十五、五十六、六、六、六、六、六、64	1、2
m7a.large	2	2	1	1、2	1
m7a.xlarge	4	4	1	1、2、3、4	1
m7a.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
m7a.4xlarge	16	16	1	一、二、四、六、八、十、十、十、十四、十六	1
m7a.8xlarge	32	32	1	1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m7a.12xlarge	48	48	1	一、二、三、四、五、六、十二、十八、二、二、三十、四、四、四十八	1
m7a.16xlarge	64	64	1	一、二、三、四、五、六、七、八、十六、二、三十、四十、四十八、五十六	1
m7a.24xlarge	96	96	1	一、二、三、四、五、六、七、八、九、十、十、十、十、二、二、四、四、四、六、七、八四、九六	1
m7a.32xlarge	128	128	1	四、六、八、十、十、十、十四、十六、四、四、六、八、八	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m7a.48xlarge	192	192	1	四、六、八、十、十、十、十四、十六、十、二十、二十、二十、二、六、四、168、192	1
m7g.large	2	2	1	1、2	1
m7g.xlarge	4	4	1	1、2、3、4	1
m7g.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
m7g.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
m7g.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m7g.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m7g.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
m7gd.large	2	2	1	1、2	1
m7gd.xlarge	4	4	1	1、2、3、4	1
m7gd.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m7gd.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
m7gd.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1
m7gd.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m7gd.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
m7i.large	2	1	2	1	1、2
m7i.xlarge	4	2	2	1、2	1、2
m7i.2xlarge	8	4	2	1、2、3、4	1、2
m7i.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m7i.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2
m7i.12xlarge	48	24	2	一、二、三、四、五、六、七、八、九、十、十、十、十、十、十、十、十、十、十、十、十、十、二、二、二、二、二、二、二、24	1、2
m7i.16xlarge	64	32	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1、2



執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m7i.24xlarge	96	48	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m7i.48xlarge	192	96	2	四、六、八、十、十、十四、十六、二十、二十、二十、二十六、二十、三十、三十、三十、四、四、四、四、五十、五十五、五十六、六、六、六、六、六、六、六、六、六、六、七、七、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、六	1、2
m7i-flex.large	2	1	2	1	1、2
m7i-flex.xlarge	4	2	2	1、2	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
m7i-flex.2xlarge	8	4	2	1、2、3、4	1、2
m7i-flex.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
m7i-flex.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2
t3.nano	2	1	2	1	1、2
t3.micro	2	1	2	1	1、2
t3.small	2	1	2	1	1、2
t3.medium	2	1	2	1	1、2
t3.large	2	1	2	1	1、2
t3.xlarge	4	2	2	2	1、2
t3.2xlarge	8	4	2	2、4	1、2
t3a.nano	2	1	2	1	1、2
t3a.micro	2	1	2	1	1、2
t3a.small	2	1	2	1	1、2
t3a.medium	2	1	2	1	1、2
t3a.large	2	1	2	1	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
t3a.xlarge	4	2	2	2	1、2
t3a.2xlarge	8	4	2	2、4	1、2
t4g.nano	2	2	1	1、2	1
t4g.micro	2	2	1	1、2	1
t4g.small	2	2	1	1、2	1
t4g.medium	2	2	1	1、2	1
t4g.large	2	2	1	1、2	1
t4g.xlarge	4	4	1	1、2、3、4	1
t4g.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1

### 運算最佳化執行個體

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c3.large	2	1	2	1	1、2
c3.xlarge	4	2	2	1、2	1、2
c3.2xlarge	8	4	2	1、2、3、4	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c3.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
c3.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
c4.large	2	1	2	1	1、2
c4.xlarge	4	2	2	1、2	1、2
c4.2xlarge	8	4	2	1、2、3、4	1、2
c4.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
c4.8xlarge	36	18	2	2、4、6、8、10、12、14、16、18	1、2
c5.large	2	1	2	1	1、2
c5.xlarge	4	2	2	2	1、2
c5.2xlarge	8	4	2	2、4	1、2
c5.4xlarge	16	8	2	2、4、6、8	1、2
c5.9xlarge	36	18	2	2、4、6、8、10、12、14、16、18	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c5.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
c5.18xlarge	72	36	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36	1、2
c5.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
c5a.large	2	1	2	1	1、2
c5a.xlarge	4	2	2	1、2	1、2
c5a.2xlarge	8	4	2	1、2、3、4	1、2
c5a.4xlarge	16	8	2	1、2、3、4、8	1、2
c5a.8xlarge	32	16	2	1、2、3、4、8、12、16	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c5a.12xlarge	48	24	2	1, 2, 3, 4, 8, 12, 16, 20, 24	1、2
c5a.16xlarge	64	32	2	1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32	1、2
c5a.24xlarge	96	48	2	1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48	1、2
c5ad.large	2	1	2	1	1、2
c5ad.xlarge	4	2	2	1、2	1、2
c5ad.2xlarge	8	4	2	1、2、3、4	1、2
c5ad.4xlarge	16	8	2	1, 2, 3, 4, 8	1、2
c5ad.8xlarge	32	16	2	1, 2, 3, 4, 8, 12, 16	1、2
c5ad.12xlarge	48	24	2	1, 2, 3, 4, 8, 12, 16, 20, 24	1、2
c5ad.16xlarge	64	32	2	1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c5ad.24xlarge	96	48	2	1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48	1、2
c5d.large	2	1	2	1	1、2
c5d.xlarge	4	2	2	2	1、2
c5d.2xlarge	8	4	2	2、4	1、2
c5d.4xlarge	16	8	2	2、4、6、8	1、2
c5d.9xlarge	36	18	2	2、4、6、8、10、12、14、16、18	1、2
c5d.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
c5d.18xlarge	72	36	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36	1、2



執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c5d.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
c5n.large	2	1	2	1	1、2
c5n.xlarge	4	2	2	2	1、2
c5n.2xlarge	8	4	2	2、4	1、2
c5n.4xlarge	16	8	2	2、4、6、8	1、2
c5n.9xlarge	36	18	2	2、4、6、8、10、12、14、16、18	1、2
c5n.18xlarge	72	36	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36	1、2
c6a.large	2	1	2	1	1、2
c6a.xlarge	4	2	2	1、2	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c6a.2xlarge	8	4	2	1、2、3、4	1、2
c6a.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
c6a.8xlarge	32	16	2	4、6、8、10、12、14、16	1、2
c6a.12xlarge	48	24	2	一、二、三、四、五、六、七、八、十六、二十四	1、2
c6a.16xlarge	64	32	2	四、六、八、十、十、十、十四、十六、三十二	1、2
c6a.24xlarge	96	48	2	四、六、八、十、十、十、十四、十六、三十八	1、2
c6a.32xlarge	128	64	2	八、十六、二十、二十、二十、二十四	1、2
c6a.48xlarge	192	96	2	八、十六、二十、二十、二十、二、三十六	1、2
c6g.large	2	2	1	1、2	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c6g.xlarge	4	4	1	1、2、3、4	1
c6g.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
c6g.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
c6g.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c6g.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c6g.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
c6gd.large	2	2	1	1、2	1
c6gd.xlarge	4	4	1	1、2、3、4	1
c6gd.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c6gd.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
c6gd.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1
c6gd.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c6gd.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
c6gn.medium	1	1	1	1	1
c6gn.large	2	2	1	1、2	1
c6gn.xlarge	4	4	1	1、2、3、4	1
c6gn.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c6gn.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
c6gn.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1
c6gn.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1



執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c6gn.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
c6i.large	2	1	2	1	1、2
c6i.xlarge	4	2	2	1、2	1、2
c6i.2xlarge	8	4	2	2、4	1、2
c6i.4xlarge	16	8	2	2、4、6、8	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c6i.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
c6i.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
c6i.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
c6i.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c6i.32xlarge	128	64	2	二、四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、十六、四十、四、五十五、五十五、五十五、五十六、六十六、六十六、64	1、2
c6id.large	2	1	2	1	1、2
c6id.xlarge	4	2	2	1、2	1、2
c6id.2xlarge	8	4	2	2、4	1、2
c6id.4xlarge	16	8	2	2、4、6、8	1、2
c6id.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c6id.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
c6id.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
c6id.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c6id.32xlarge	128	64	2	二、四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四十六、四十、四、五十五、五十五、五十五、五十六、六十六、六十六、64	1、2
c6in.large	2	1	2	1	1、2
c6in.xlarge	4	2	2	1、2	1、2
c6in.2xlarge	8	4	2	1、2、3、4	1、2
c6in.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
c6in.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c6in.12xlarge	48	24	2	一、二、三、四、五、六、七、八、九、十、十、十、十、十、十、十、十、十、十、十、二、二、二、二、二、二、二、24	1、2
c6in.16xlarge	64	32	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1、2
c6in.24xlarge	96	48	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c6in.32xlarge	128	64	2	四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四十、四十、四十、五十五、五十五、五十五、五十六、六、六、六、六、六、64	1、2
c7a.large	2	2	1	1、2	1
c7a.xlarge	4	4	1	1、2、3、4	1
c7a.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
c7a.4xlarge	16	16	1	一、二、四、六、八、十、十、十、十四、十六	1
c7a.8xlarge	32	32	1	1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c7a.12xlarge	48	48	1	一、二、三、四、五、六、十二、十八、二、二、三十、四、四、四十八	1
c7a.16xlarge	64	64	1	一、二、三、四、五、六、七、八、十六、二、三十、四十、四十八、五十六	1
c7a.24xlarge	96	96	1	一、二、三、四、五、六、七、八、九、十、十、十、十、二、二、四、四、四、六、七、八四、九六	1
c7a.32xlarge	128	128	1	四、六、八、十、十、十、十四、十六、四、四、六、八、八	1



執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c7a.48xlarge	192	192	1	四、六、八、十、十、十四、十六、十、二十、二十、二十、二、六、四、168、192	1
c7g.large	2	2	1	1、2	1
c7g.xlarge	4	4	1	1、2、3、4	1
c7g.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
c7g.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
c7g.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c7g.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c7g.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
c7gd.large	2	2	1	1、2	1
c7gd.xlarge	4	4	1	1、2、3、4	1
c7gd.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c7gd.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
c7gd.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1
c7gd.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c7gd.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
c7gn.large	2	2	1	1、2	1
c7gn.xlarge	4	4	1	1、2、3、4	1
c7gn.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c7gn.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
c7gn.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1
c7gn.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c7gn.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
c7i.large	2	1	2	1	1、2
c7i.xlarge	4	2	2	1、2	1、2
c7i.2xlarge	8	4	2	1、2、3、4	1、2
c7i.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c7i.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2
c7i.12xlarge	48	24	2	一、二、三、四、五、六、七、八、九、十、十、十、十、十、十、十、十、十、十、十、十、十、二、二、二、二、二、二、二、24	1、2
c7i.16xlarge	64	32	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1、2



執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c7i.24xlarge	96	48	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1、2



執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
c7i-flex.2xlarge	8	4	2	1、2、3、4	1、2
c7i-flex.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
c7i-flex.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2

### 記憶體最佳化執行個體

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r3.large	2	1	2	1	1、2
r3.xlarge	4	2	2	1、2	1、2
r3.2xlarge	8	4	2	1、2、3、4	1、2
r3.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
r3.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
r4.large	2	1	2	1	1、2
r4.xlarge	4	2	2	1、2	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r4.2xlarge	8	4	2	1、2、3、4	1、2
r4.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
r4.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2
r4.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
r5.large	2	1	2	1	1、2
r5.xlarge	4	2	2	2	1、2
r5.2xlarge	8	4	2	2、4	1、2
r5.4xlarge	16	8	2	2、4、6、8	1、2
r5.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
r5.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r5.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
r5.24xlarge	96	48	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
r5a.large	2	1	2	1	1、2
r5a.xlarge	4	2	2	2	1、2
r5a.2xlarge	8	4	2	2、4	1、2
r5a.4xlarge	16	8	2	2、4、6、8	1、2
r5a.8xlarge	32	16	2	4、6、8、10、12、14、16	1、2
r5a.12xlarge	48	24	2	6、12、18、24	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r5a.16xlarge	64	32	2	8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
r5a.24xlarge	96	48	2	12、18、24、36、48	1、2
r5ad.large	2	1	2	1	1、2
r5ad.xlarge	4	2	2	2	1、2
r5ad.2xlarge	8	4	2	2、4	1、2
r5ad.4xlarge	16	8	2	2、4、6、8	1、2
r5ad.8xlarge	32	16	2	4、6、8、10、12、14、16	1、2
r5ad.12xlarge	48	24	2	6、12、18、24	1、2
r5ad.16xlarge	64	32	2	8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
r5ad.24xlarge	96	48	2	12、18、24、36、48	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r5b.large	2	1	2	1	1、2
r5b.xlarge	4	2	2	1、2	1、2
r5b.2xlarge	8	4	2	2、4	1、2
r5b.4xlarge	16	8	2	2、4、6、8	1、2
r5b.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
r5b.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
r5b.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
r5b.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r5d.large	2	1	2	1	1、2
r5d.xlarge	4	2	2	2	1、2
r5d.2xlarge	8	4	2	2、4	1、2
r5d.4xlarge	16	8	2	2、4、6、8	1、2
r5d.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
r5d.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
r5d.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
r5d.24xlarge	96	48	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2



執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r5dn.large	2	1	2	1	1、2
r5dn.xlarge	4	2	2	1、2	1、2
r5dn.2xlarge	8	4	2	2、4	1、2
r5dn.4xlarge	16	8	2	2、4、6、8	1、2
r5dn.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
r5dn.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
r5dn.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
r5dn.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r5n.large	2	1	2	1	1、2
r5n.xlarge	4	2	2	1、2	1、2
r5n.2xlarge	8	4	2	2、4	1、2
r5n.4xlarge	16	8	2	2、4、6、8	1、2
r5n.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
r5n.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
r5n.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
r5n.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r6a.large	2	1	2	1	1、2
r6a.xlarge	4	2	2	1、2	1、2
r6a.2xlarge	8	4	2	1、2、3、4	1、2
r6a.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
r6a.8xlarge	32	16	2	4、6、8、10、12、14、16	1、2
r6a.12xlarge	48	24	2	一、二、三、四、五、六、七、八、十六	1、2
r6a.16xlarge	64	32	2	四、六、八、十、十、十、十四、十六、三	1、2
r6a.24xlarge	96	48	2	四、六、八、十、十、十、十四、十六、三十八	1、2
r6a.32xlarge	128	64	2	八、十六、二十、二十、二十、二十四	1、2
r6a.48xlarge	192	96	2	八、十六、二十、二十、二十、二、三十六	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r6g.large	2	2	1	1、2	1
r6g.xlarge	4	4	1	1、2、3、4	1
r6g.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
r6g.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
r6g.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r6g.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r6g.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
r6gd.large	2	2	1	1、2	1
r6gd.xlarge	4	4	1	1、2、3、4	1
r6gd.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r6gd.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
r6gd.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1
r6gd.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r6gd.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
r6i.large	2	1	2	1	1、2
r6i.xlarge	4	2	2	1、2	1、2
r6i.2xlarge	8	4	2	2、4	1、2
r6i.4xlarge	16	8	2	2、4、6、8	1、2



執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r6i.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
r6i.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
r6i.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
r6i.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPU	預設 CPU 核 心	預設每核心執 行緒	有效的 CPU 核心	每個核心的有 效執行緒
r6i.32xlarge	128	64	2	二、四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、五十五、五十五、五十五、五十六、六十六、六十六、六、六、64	1、2
r6idn.large	2	1	2	1	1、2
r6idn.xlarge	4	2	2	1、2	1、2
r6idn.2xlarge	8	4	2	1、2、3、4	1、2
r6idn.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
r6idn.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r6idn.12xlarge	48	24	2	一、二、三、四、五、六、七、八、九、十、十、十、十、十、十、十、十、十、十、十、二、二、二、二、二、二、二、24	1、2
r6idn.16xlarge	64	32	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1、2
r6idn.24xlarge	96	48	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r6idn.32xlarge	128	64	2	四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四十、四十、四十、四十、五十五、五十五、五十五、五十六、六、六、六、六、六、六、六、六、64	1、2
r6in.large	2	1	2	1	1、2
r6in.xlarge	4	2	2	1、2	1、2
r6in.2xlarge	8	4	2	1、2、3、4	1、2
r6in.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
r6in.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r6in.12xlarge	48	24	2	一、二、三、四、五、六、七、八、九、十、十、十、十、十、十、十、十、十、十、十、二、二、二、二、二、二、二、24	1、2
r6in.16xlarge	64	32	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1、2
r6in.24xlarge	96	48	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r6in.32xlarge	128	64	2	四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四十、四十、四十、四十、五十五、五十五、五十五、五十六、六、六、六、六、六、六、六、六、64	1、2
r6id.large	2	1	2	1	1、2
r6id.xlarge	4	2	2	1、2	1、2
r6id.2xlarge	8	4	2	2、4	1、2
r6id.4xlarge	16	8	2	2、4、6、8	1、2
r6id.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r6id.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
r6id.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
r6id.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r6id.32xlarge	128	64	2	二、四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、五十五、五十五、五十六、六十六、六十六、六、六、64	1、2
r7a.large	2	2	1	1、2	1
r7a.xlarge	4	4	1	1、2、3、4	1
r7a.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
r7a.4xlarge	16	16	1	一、二、四、六、八、十、十、十、十四、十六	1
r7a.8xlarge	32	32	1	1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32	1



執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r7a.12xlarge	48	48	1	一、二、三、四、五、六、十二、二、二、二、三十、四、四、四十八	1
r7a.16xlarge	64	64	1	一、二、三、四、五、六、七、八、十六、二、三十、四十、四十八、五十六	1
r7a.24xlarge	96	96	1	一、二、三、四、五、六、七、八、九、十、十、十、十、二、二、二、三十六、六、七、八四、九六	1
r7a.32xlarge	128	128	1	四、六、八、十、十、十、十四、十六、四、四、六、八、八	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r7a.48xlarge	192	192	1	四、六、八、十、十、十四、十六、十、二十、二十、二十、二、六、四、168、192	1
r7g.large	2	2	1	1、2	1
r7g.xlarge	4	4	1	1、2、3、4	1
r7g.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
r7g.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
r7g.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r7g.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r7g.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
r7gd.large	2	2	1	1、2	1
r7gd.xlarge	4	4	1	1、2、3、4	1
r7gd.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r7gd.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
r7gd.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1
r7gd.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r7gd.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
r7i.large	2	1	2	1	1、2
r7i.xlarge	4	2	2	1、2	1、2
r7i.2xlarge	8	4	2	1、2、3、4	1、2
r7i.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r7i.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2
r7i.12xlarge	48	24	2	一、二、三、四、五、六、七、八、九、十、十、十、十、十、十、十、十、十、十、十、十、十、二、二、二、二、二、二、二、24	1、2
r7i.16xlarge	64	32	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r7i.24xlarge	96	48	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1、2





執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r7iz.2xlarge	8	4	2	1、2、3、4	1、2
r7iz.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
r7iz.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2
r7iz.12xlarge	48	24	2	一、二、三、四、五、六、七、八、九、十、十、十、十、十、十、十、十、十、十、二、二、二、二、二、二、二、24	1、2
r7iz.16xlarge	64	32	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
r7iz.32xlarge	128	64	2	四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、四十、四十、四十、四十、五十五、五十五、五十五、五十六、六、六、六、六、六、六、六	1、2
u-3tb1.56xlarge	224	112	2	四、八、十二、二十、二十、二、二、三、三十、四十、四十、四十、五十六、六十六、六十、七、七、八、八、八、八、九二、九六、百、一零八、112	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
u-6tb1.56 xlarge	224	224	1	八、十六、二十四、四十、五十六、六十二、八十八、九六、九六、十二、一二、一二、一百二十、一六、六、六、六、六、六、八、二千八、二千二、二千二、二十八、二十六、二	1
u-6tb1.11 2xlarge	448	224	2	八、十六、二十四、四十、五十六、六十二、八十八、九六、九六、十二、一二、一二、一百二十、一六、六、六、六、六、六、八、二千八、二千二、二千二、二十八、二十六、二	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
u-9tb1.11 2xlarge	448	224	2	八、十六、二十四、四十、五十六、六十二、八十八、九六、九六、十二、一二、一二、一百二十、一六、六、六、六、六、六、八、二千八、二千二、二千二、二十八、二十六、二	1、2
u-12tb1.1 12xlarge	448	224	2	八、十六、二十四、四十、五十六、六十二、八十八、九六、九六、十二、一二、一二、一百二十、一六、六、六、六、六、六、八、二千八、二千二、二千二、二十八、二十六、二	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
u-18tb1.1 12xlarge	448	224	2	八、十六、二十四、四十、五十六、六十二、八十八、九六、九六、十二、一二、一二、一百二十、一六、六、六、六、六、六、八、二千八、二千二、二千二、二十八、二十六、二	1、2
u-24tb1.1 12xlarge	448	224	2	八、十六、二十四、四十、五十六、六十二、八十八、九六、九六、十二、一二、一二、一百二十、一六、六、六、六、六、六、八、二千八、二千二、二千二、二十八、二十六、二	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
u7i-12tb. 224xlarge	896	448	2	十六、二十、三十、四十、五十六、七、八、八、九六、九六、十二、一二、十二、十二、六、六、六、六、八、八、二、二、二、三六、352、352、2、352、3652、262、262、282、282、384、392、四、百、408、416、424、432、440、448	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
u7in-16tb .224xlarge	896	448	2	十六、二十、三十、四十、五十六、七、八、八、九六、九六、十二、一二、十二、十二、六、六、六、六、八、八、二、二、二、三六、352、352、2、352、3652、262、262、282、282、384、392、四、百、408、416、424、432、440、448	1、2



執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
u7in-24tb .224xlarge	896	448	2	十六、二十、三十、四十、五十六、七、八、八、九六、九六、十二、一二、十二、十二、六、六、六、六、八、八、二、二、二、三六、352、352、2、352、3652、262、262、282、282、384、392、四、百、408、416、424、432、440、448	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
u7in-32tb .224xlarge	896	448	2	十六、二十、三十、四十、五十六、七、八、八、九六、九六、十二、一二、十二、十二、六、六、六、六、八、八、二、二、二、三六、352、352、2、352、3652、262、262、282、282、384、392、四 百、408、416、424、432、440、448	1、2
x1.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
x1.32xlarge	128	64	2	4、8、12、16、20、24、28、32、36、40、44、48、52、56、60、64	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
x2gd.large	2	2	1	1、2	1
x2gd.xlarge	4	4	1	1、2、3、4	1
x2gd.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
x2gd.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
x2gd.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
x2gd.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
x2gd.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
x2idn.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2





執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
x2iezn.2xlarge	8	4	2	2、4	1、2
x2iezn.4xlarge	16	8	2	2、4、6、8	1、2
x2iezn.6xlarge	24	12	2	2、4、6、8、10、12	1、2
x2iezn.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
x2iezn.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
x1e.xlarge	4	2	2	1、2	1、2
x1e.2xlarge	8	4	2	1、2、3、4	1、2
x1e.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
x1e.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2



執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
x1e.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
x1e.32xlarge	128	64	2	4、8、12、16、20、24、28、32、36、40、44、48、52、56、60、64	1、2
z1d.large	2	1	2	1	1、2
z1d.xlarge	4	2	2	1、2	1、2
z1d.2xlarge	8	4	2	2、4	1、2
z1d.3xlarge	12	6	2	2、4、6	1、2
z1d.6xlarge	24	12	2	2、4、6、8、10、12	1、2
z1d.12xlarge	48	24	2	4、6、8、10、12、14、16、18、20、22、24	1、2

## 儲存最佳化執行個體

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
d2.xlarge	4	2	2	1、2	1、2
d2.2xlarge	8	4	2	1、2、3、4	1、2
d2.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
d2.8xlarge	36	18	2	2、4、6、8、10、12、14、16、18	1、2
d3.xlarge	4	2	2	1、2	1、2
d3.2xlarge	8	4	2	2、4	1、2
d3.4xlarge	16	8	2	2、4、6、8	1、2
d3.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
d3en.xlarge	4	2	2	1、2	1、2
d3en.2xlarge	8	4	2	2、4	1、2
d3en.4xlarge	16	8	2	2、4、6、8	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
d3en.6xlarge	24	12	2	2、4、6、8、10、12	1、2
d3en.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
d3en.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
h1.2xlarge	8	4	2	1、2、3、4	1、2
h1.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
h1.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2
h1.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
i2.xlarge	4	2	2	1、2	1、2
i2.2xlarge	8	4	2	1、2、3、4	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
i2.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
i2.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
i3.large	2	1	2	1	1、2
i3.xlarge	4	2	2	1、2	1、2
i3.2xlarge	8	4	2	1、2、3、4	1、2
i3.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
i3.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2
i3.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
i3en.large	2	1	2	1	1、2
i3en.xlarge	4	2	2	1、2	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
i3en.2xlarge	8	4	2	2、4	1、2
i3en.3xlarge	12	6	2	2、4、6	1、2
i3en.6xlarge	24	12	2	2、4、6、8、10、12	1、2
i3en.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
i3en.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
i4g.large	2	2	1	1、2	1
i4g.xlarge	4	4	1	1、2、3、4	1
i4g.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
i4g.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
i4g.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
i4g.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
i4i.large	2	1	2	1	1、2
i4i.xlarge	4	2	2	1、2	1、2
i4i.2xlarge	8	4	2	1、2、3、4	1、2
i4i.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
i4i.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2
i4i.12xlarge	48	24	2	一、二、三、四、五、六、七、八、九、十、十、十、十、十、十、十、十、十、十、十、十、十、二、二、二、二、二、二、二、24	1、2
i4i.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
i4i.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2





執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
im4gn.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1
im4gn.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
is4gen.medium	1	1	1	1	1
is4gen.large	2	2	1	1、2	1
is4gen.xlarge	4	4	1	1、2、3、4	1
is4gen.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
is4gen.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
is4gen.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1

## 加速運算執行個體

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
dl1.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
dl2q.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
f1.2xlarge	8	4	2	1、2、3、4	1、2
f1.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
f1.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
g3.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
g3.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2
g3.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
g4ad.xlarge	4	2	2	2	1、2
g4ad.2xlarge	8	4	2	2、4	1、2
g4ad.4xlarge	16	8	2	2、4、8	1、2
g4ad.8xlarge	32	16	2	2、4、8、16	1、2
g4ad.16xlarge	64	32	2	2、4、8、16、32	1、2
g4dn.xlarge	4	2	2	2	1、2
g4dn.2xlarge	8	4	2	2、4	1、2
g4dn.4xlarge	16	8	2	2、4、6、8	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
g4dn.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
g4dn.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
g4dn.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
g5g.xlarge	4	4	1	1、2、3、4	1
g5g.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
g5g.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
g5g.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1
g5g.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
g6.xlarge	4	2	2	1、2	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
g6.2xlarge	8	4	2	1、2、3、4	1、2
g6.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
g6.8xlarge	32	16	2	一、二、四、六、八、十、十、十、十四、十六	1、2
g6.12xlarge	48	24	2	一、二、三、六、九、十二、十五、十八、二十一、二十四	1、2
g6.16xlarge	64	32	2	1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32	1、2
g6.24xlarge	96	48	2	一、二、三、四、五、六、十二、二、二、二、三十、四、四、四十八	1、2
g6.48xlarge	192	96	2	四、六、八、十、十、二、二十四、四十、六十、七、八四、九六	1、2



執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
gr6.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
gr6.8xlarge	32	16	2	一、二、四、六、八、十、十、十、十四、十六	1、2
inf1.xlarge	4	2	2	2	1、2
inf1.2xlarge	8	4	2	2、4	1、2
inf1.6xlarge	24	12	2	2、4、6、8、10、12	1、2
inf1.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
inf2.xlarge	4	2	2	1、2	1、2
inf2.8xlarge	32	16	2	4、6、8、10、12、14、16	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
inf2.24xlarge	96	48	2	四、六、八、十、十、十四、十六、三十八	1、2
inf2.48xlarge	192	96	2	四、八、十二、十六、二十、二十二、六	1、2
p2.xlarge	4	2	2	1、2	1、2
p2.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2
p2.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
p3.2xlarge	8	4	2	1、2、3、4	1、2
p3.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2

執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
p3.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
p3dn.24xlarge	96	48	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
p4d.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
p4de.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2





執行個體類型	預設 vCPU	預設 CPU 核心	預設每核心執行緒	有效的 CPU 核心	每個核心的有效執行緒
				四、四、四、 四、四、四、 四、四、四、 四、四、四 十、四十、四 十、四十、五 十五、五十 五、五十五、 五十六、六、 六、六、六、 六、六、64	

## 為執行個體指定 CPU 選項

您可在執行個體啟動期間指定 CPU 選項。

下列範例說明在 EC2 主控台中使用啟動執行個體精靈和[執行個體](#) AWS CLI 命令時，如何指定 CPU 選項，以及 EC2 主控台中之建立啟動範本頁面和建立 [AWS CLI](#) 啟動範本命令。對於 EC2 機群或 Spot 機群，您必須於啟動範本中指定 CPU 選項。

以下範圍適用於 r5.4xlarge 執行個體類型，其[預設值](#)如下：

- 預設 CPU 核心：8
- 預設每核心執行緒：2
- 預設 vCPU：16 (8 \* 2)
- CPU 核心的有效數目：2、4、6、8
- 有效的每核心執行緒數量：1、2

## 停用多執行緒

若要停用多執行緒，請針對每個核心指定 1 個執行緒。

## New console

在執行個體啟動期間停用多執行緒

1. 遵照 [快速啟動執行個體](#) 程序並視需要設定執行個體。
2. 展開進階詳細資訊，然後選取指定 CPU 選項核取方塊。
3. 針對 Core count (核心計數)，選擇所需的 CPU 核心數量。在本範例中，若要指定 r5.4xlarge 執行個體的預設 CPU 核心計數，請選擇 8。
4. 若要停用多執行緒，請針對 Threads per core (每核心執行緒)，選擇 1。
5. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。如需詳細資訊，請參閱 [使用新的啟動執行個體精靈啟動執行個體](#)。

## Old console

在執行個體啟動期間停用多執行緒

1. 遵循[使用舊版啟動執行個體精靈啟動執行個體](#)程序。
2. 在 Configure Instance Details (設定執行個體詳細資訊) 頁面上，針對 CPU options (CPU 選項)，選擇 Specify CPU options (指定 CPU 選項)。
3. 針對 Core count (核心計數)，選擇所需的 CPU 核心數量。在本範例中，若要指定 r5.4xlarge 執行個體的預設 CPU 核心計數，請選擇 8。
4. 若要停用多執行緒，請針對 Threads per core (每核心執行緒)，選擇 1。
5. 依精靈提示繼續作業。檢閱完您在 Review Instance Launch (檢閱執行個體啟動) 頁面的選項後，請選擇 Launch (啟動)。如需詳細資訊，請參閱 [使用舊版啟動執行個體精靈啟動執行個體](#)。

## AWS CLI

在執行個體啟動期間停用多執行緒

使用 [run-instances](#) AWS CLI 命令，為 1 參數的 ThreadsPerCore 指定值 `--cpu-options`。在 CoreCount 指定 CPU 核心數目。在本範例中，若要指定 r5.4xlarge 執行個體的預設 CPU 核心計數，請指定值 8。

```
aws ec2 run-instances \
 --image-id ami-1a2b3c4d \
 --cpu-options
```

```
--instance-type r5.4xlarge \
--cpu-options "CoreCount=8,ThreadsPerCore=1" \
--key-name MyKeyPair
```

## 啟動時指定 vCPU 的自訂數量

您可自訂執行個體的 CPU 核心數量和每個核心的執行緒數量。

下列範例會啟動具有 4 個 vCPUs 的 r5.4xlarge 執行個體。

### New console

在執行個體啟動期間指定 vCPU 的自訂數量

1. 遵照 [快速啟動執行個體](#) 程序並視需要設定執行個體。
2. 展開進階詳細資訊，然後選取指定 CPU 選項核取方塊。
3. 若要取得 4 個 vCPUs，請為每個核心指定 2 個 CPU 核心和 2 個執行緒，如下所示：
  - 針對「核心數」，選擇「2」。
  - 針對 Threads per core (每核心執行緒)，選擇 2。
4. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。如需詳細資訊，請參閱 [使用新的啟動執行個體精靈啟動執行個體](#)。

### Old console

在執行個體啟動期間指定 vCPU 的自訂數量

1. 遵循 [使用舊版啟動執行個體精靈啟動執行個體](#) 程序。
2. 在 Configure Instance Details (設定執行個體詳細資訊) 頁面上，針對 CPU options (CPU 選項)，選擇 Specify CPU options (指定 CPU 選項)。
3. 若要取得 4 個 vCPUs，請為每個核心指定 2 個 CPU 核心和 2 個執行緒，如下所示：
  - 針對「核心數」，選擇「2」。
  - 針對 Threads per core (每核心執行緒)，選擇 2。
4. 依精靈提示繼續作業。檢閱完您在 Review Instance Launch (檢閱執行個體啟動) 頁面的選項後，請選擇 Launch (啟動)。如需詳細資訊，請參閱 [使用舊版啟動執行個體精靈啟動執行個體](#)。



## AWS CLI

在執行個體啟動期間指定 vCPU 的自訂數量

使用[執行執行個體](#)指 AWS CLI 令，並指定參數中 CPU 核心的數目和執行緒數目 `--cpu-options`。您可以為每個核心指定 2 個 CPU 核心和 2 個執行緒，以取得 4 個 vCPUs。

```
aws ec2 run-instances \
 --image-id ami-1a2b3c4d \
 --instance-type r5.4xlarge \
 --cpu-options "CoreCount=2,ThreadsPerCore=2" \
 --key-name MyKeyPair
```

或者，每個核心指定 4 個 CPU 核心和 1 個執行緒 (停用多執行緒) 以取得 4 個 vCPUs：

```
aws ec2 run-instances \
 --image-id ami-1a2b3c4d \
 --instance-type r5.4xlarge \
 --cpu-options "CoreCount=4,ThreadsPerCore=1" \
 --key-name MyKeyPair
```

在啟動範本中指定 vCPU 的自訂數量

可在啟動範本中自訂執行個體的 CPU 核心數量和每個核心的執行緒數量。

下列範例會建立啟動範本，以指定具有 4 個 vCPUs 之 `r5.4xlarge` 執行個體的組態。

## Console

在啟動範本中指定 vCPU 的自訂數量

1. 遵照 [從參數建立啟動範本](#) 程序並視需要設定啟動範本。
2. 展開進階詳細資訊，然後選取指定 CPU 選項核取方塊。
3. 若要取得 4 個 vCPUs，請為每個核心指定 2 個 CPU 核心和 2 個執行緒，如下所示：
  - 針對「核心數」，選擇「2」。
  - 針對 Threads per core (每核心執行緒)，選擇 2。
4. 在摘要面板中，檢閱執行個體組態，然後選擇建立啟動範本。如需詳細資訊，請參閱 [從啟動範本啟動執行個體](#)。

## AWS CLI

在啟動範本中指定 vCPU 的自訂數量

使用「[建立啟動範本](#)」指 AWS CLI 令，並指定參數中 CPU 核心的數目和執行緒數目。CpuOptions 您可以為每個核心指定 2 個 CPU 核心和 2 個執行緒，以取得 4 個 vCPUs。

```
aws ec2 create-launch-template \
 --launch-template-name TemplateForCPUOptions \
 --version-description CPUOptionsVersion1 \
 --launch-template-data file://template-data.json
```

以下是包含啟動範本資料的 JSON 檔案範例，其中包括此範例的執行個體組態的 CPU 選項。

```
{
 "NetworkInterfaces": [{
 "AssociatePublicIpAddress": true,
 "DeviceIndex": 0,
 "Ipv6AddressCount": 1,
 "SubnetId": "subnet-7b16de0c"
 }],
 "ImageId": "ami-8c1be5f6",
 "InstanceType": "r5.4xlarge",
 "TagSpecifications": [{
 "ResourceType": "instance",
 "Tags": [{
 "Key": "Name",
 "Value": "webserver"
 }]
 }],
 "CpuOptions": {
 "CoreCount": 2,
 "ThreadsPerCore": 2
 }
}
```

或者，每個核心指定 4 個 CPU 核心和 1 個執行緒 (停用多執行緒) 以取得 4 個 vCPUs：

```
{
 "NetworkInterfaces": [{
 "AssociatePublicIpAddress": true,
 "DeviceIndex": 0,
 "Ipv6AddressCount": 1,
```

```
 "SubnetId": "subnet-7b16de0c"
]],
 "ImageId": "ami-8c1be5f6",
 "InstanceType": "r5.4xlarge",
 "TagSpecifications": [{
 "ResourceType": "instance",
 "Tags": [{
 "Key": "Name",
 "Value": "webserver"
 }]
 }],
 "CpuOptions": {
 "CoreCount": 4,
 "ThreadsPerCore": 1
 }
}
```

## 檢視執行個體的 CPU 選項

您可在 Amazon EC2 主控台中檢視現有執行個體的 CPU 選項，或使用 AWS CLI 描述執行個體來執行此作業。

### Console

使用主控台檢視執行個體的 CPU 選項

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側的導覽窗格中，選取 Instances (執行個體)，然後選取執行個體。
3. 在 Details (詳細資料) 標籤的 Host and placement group (主機與置放群組) 下，尋找 Number of vCPUs (vCPU 的數目)。

### AWS CLI

檢視執行個體的 CPU 選項 (AWS CLI)

使用 [describe-instances](#) 命令。

```
aws ec2 describe-instances --instance-ids i-123456789abcde123
```

```
...
```

```
"Instances": [
 {
 "Monitoring": {
 "State": "disabled"
 },
 "PublicDnsName": "ec2-198-51-100-5.eu-central-1.compute.amazonaws.com",
 "State": {
 "Code": 16,
 "Name": "running"
 },
 "EbsOptimized": false,
 "LaunchTime": "2018-05-08T13:40:33.000Z",
 "PublicIpAddress": "198.51.100.5",
 "PrivateIpAddress": "172.31.2.206",
 "ProductCodes": [],
 "VpcId": "vpc-1a2b3c4d",
 "CpuOptions": {
 "CoreCount": 34,
 "ThreadsPerCore": 1
 },
 "StateTransitionReason": "",
 ...
 }
]
...
```

在傳回的輸出中，CoreCount 欄位顯示執行個體的核心數量。ThreadsPerCore 欄位顯示每核心的執行緒數量。

或者，若要檢視 CPU 資訊，您可以連線至執行個體，並使用下列其中一種系統工具：

- 視窗執行個體 Task Manager 上的視窗
- Linux 執行個體上的 lscpu 指令

您可以用 AWS Config 來記錄、評估、稽核和評估執行個體的組態變更，包括已終止的執行個體。如需詳細資訊，請參閱《AWS Config 開發人員指南》中的 [開始使用 AWS Config](#)。

## Amazon EC2 上的 AMD SEV-SNP

AMD Secure Encrypted Virtualization-Secure Nested Paging (AMD SEV-SNP) 是一項 CPU 功能，可提供下列屬性：

- 認證 – AMD SEV-SNP 可讓您擷取已簽署的認證報告，其中包含可用來驗證執行個體的狀態和身分的加密措施，以及是否在正版 AMD 硬體上執行。如需詳細資訊，請參閱 [使用 AMD SEV-SNP 進行認證](#)。
- 記憶體加密 — 從 AMD EPYC (米蘭)、AWS 引力通 2 和英特爾至強可擴展 (冰湖) 處理器開始，執行個體記憶體始終是加密的。為 AMD SEV-SNP 啟用的執行個體會使用執行個體特定金鑰進行記憶體加密。

## 概念和術語

在開始使用 AMD SEV-SNP 之前，請確保您已熟悉下列概念和術語。

### AMD SEV-SNP 認證報告

AMD SEV-SNP 認證報告是執行個體可向 CPU 請求的文件。AMD SEV-SNP 認證報告可用於驗證執行個體的狀態和身分，以及確認執行個體是否在准許的 AMD 環境中執行。該報告包含啟動測量，這是執行個體初始啟動狀態的密碼編譯雜湊，包含其初始執行個體記憶體內容和 vCPU 的初始狀態。AMD SEV-SNP 認證報告採用 VLEK 簽章進行簽署，可以鏈接回 AMD 信任根源。

### VLEK

Versioned Loaded Endorsement Key (VLEK) 是由 AMD 認證的版本化簽署金鑰，並由 AMD CPU 用來簽署 AMD SEV-SNP 認證報告。可以使用 AMD 提供的憑證來驗證 VLEK 簽章。

### OVMF 二進位

Open Virtual Machine Firmware (OVMF) 是用來為執行個體提供 UEFI 環境的早期開機程式碼。早期開機程式碼在 AMI 中的程式碼啟動之前執行。OVMF 也會尋找並執行 AMI 中提供的開機載入器。更多詳細資訊，請參閱 [OVMF 儲存庫](#)。

## 要求

若要使用 AMD SEV-SNP，必須執行下列動作：

- 使用下列其中一個支援的執行個體類型：
  - 一般用途：m6a.large | m6a.xlarge | m6a.2xlarge | m6a.4xlarge | m6a.8xlarge
  - 運算優化：c6a.large | c6a.xlarge | c6a.2xlarge | c6a.4xlarge | c6a.8xlarge | c6a.12xlarge | c6a.16xlarge
  - 記憶體優化：r6a.large | r6a.xlarge | r6a.2xlarge | r6a.4xlarge
- 在支援的情況下啟動執行個體 AWS 區域。目前僅支援美國東部 (俄亥俄) 和歐洲 (愛爾蘭)。

- 使用具有 uefi 或 uefi-preferred 開機模式的 AMI，以及支援 AMD SEV-SNP 的作業系統。如需有關作業系統上 AMD SEV-SNP 支援的詳細資訊，請參閱各自的作業系統文件。對於 AWS，AMD SEV-SNP 在 AL2023、RHEL 9.3、SLES 15 SP4 和 Ubuntu 23.04 及更高版本上受到支援。

## 考量事項

您只能在啟動執行個體時開啟 AMD SEV-SNP。當您的執行個體啟動時開啟 AMD SEV-SNP，則適用下列規則。

- 無法關閉 AMD SEV-SNP。在整個執行個體生命週期中，它會保持
- 您只能將[執行個體類型變更](#)為支援 AMD SEV-SNP 的其他執行個體類型。
- 不支援休眠和硝基飛地。
- 不支援專用主機。
- 如果執行個體的基礎主機已排程進行維護，您將在事件發生前 14 天收到排定的事件通知。您必須手動停止或重新啟動執行個體，才能將其移至新主機。

## 定價

在開啟 AMD SEV-SNP 的情形下啟動 Amazon EC2 執行個體時，需支付額外的小時用量費，相當於所選執行個體類型[隨需小時費率](#)的 10%。

此 AMD SEV-SNP 用量費是針對 Amazon EC2 執行個體之用量而單獨收取的費用。預留執行個體、Savings Plans 和作業系統用量不會影響此費用。

如果將 Spot 執行個體設定為在開啟 [AMD SEV-SNP](#) 的情形下啟動，則需要支付額外的小時用量費，相當於所選執行個體類型[隨需小時費率](#)的 10%。如果配置策略使用價格作為輸入，則 Spot 機群不包含此額外費用；僅使用 Spot 價格。

## 在 Amazon EC2 上使用 AMD SEV-SNP

若要在 Amazon EC2 上使用 AMD SEV-SNP，請完成下列任務。

### 任務

- [尋找支援的執行個體類型](#)
- [啟動時開啟 AMD SEV-SNP](#)
- [檢查 AMD SEV-SNP 狀態](#)

## 尋找支援的執行個體類型

您可以使用 AWS CLI 來尋找支援 AMD SEV-SNP 的執行個體類型。

若要使用尋找支援 AMD SEV-SNP 的執行個體類型 AWS CLI，請使用下列命令。[describe-instance-types](#)

```
$ C:\> aws ec2 describe-instance-types \
--filters Name=processor-info.supported-features,Values=amd-sev-snp \
--query 'InstanceTypes[*].InstanceType'
```

範例輸出。

```
[
 "r6a.2xlarge",
 "m6a.large",
 "m6a.2xlarge",
 "r6a.xlarge",
 "c6a.16xlarge",
 "c6a.8xlarge",
 "m6a.4xlarge",
 "c6a.12xlarge",
 "r6a.4xlarge",
 "c6a.xlarge",
 "c6a.4xlarge",
 "c6a.2xlarge",
 "m6a.xlarge",
 "c6a.large",
 "r6a.large",
 "m6a.8xlarge"
]
```

## 啟動時開啟 AMD SEV-SNP

您可以使用在已 AWS CLI 開啟 AMD SEV-SNP 的情況下啟動執行個體。

若要使用啟動 AMD SEV-SNP 的情況下啟動執行個體 AWS CLI，請使用[run-instances](#)指令並包含選項。--cpu-options AmdSevSnp=enabled對於 --image-id，指定具有 uefi 或 uefi-preferred 開機模式的 AMI，以及支援 AMD SEV-SNP 的作業系統。對於 --instance-type，指定支援的執行個體類型。

```
$ C:\> aws ec2 run-instances \
```

```
--image-id supported_ami_id \
--instance-type supported_instance_type \
--key-name key_pair_name \
--subnet-id subnet_id \
--cpu-options AmdSevSnp=enabled
```

## 檢查 AMD SEV-SNP 狀態

您可以使用下列其中一種方法來檢查 AMD SEV-SNP 的狀態。

### AWS CLI

若要使用檢查執行個體是否已開啟 AMD SEV-SNP AWS CLI，請使用指令。[describe-instances](#)對於 `--instance-ids`，指定要檢查的執行個體 ID。

```
$ C:\> aws ec2 describe-instances --instance-ids instance_id
```

在命令輸出中，`CpuOptions` 中的 `AmdSevSnp` 值表示 AMD SEV-SNP 是開啟還是關閉。

### AWS CloudTrail

在執行個體啟動要求的情況下，值 `"cpuOptions": {"AmdSevSnp": enabled}` 表示執行個體的 AMD SEV-SNP 已開啟。AWS CloudTrail

## 使用 AMD SEV-SNP 進行認證

認證是一種程序，可讓您的執行個體證明其狀態和身分。當您為執行個體開啟 AMD SEV-SNP 時，可以向基礎處理器請求 AMD SEV-SNP 認證報告。AMD SEV-SNP 認證報告包含初始訪客記憶體內容的密碼編譯雜湊 (稱為啟動測量) 和初始 vCPU 狀態。認證報告採用 VLEK 簽章進行簽署，可以鏈接回 AMD 信任根源。您可以使用認證報告中包含的啟動測量來驗證執行個體是否在正版 AMD 環境中執行，並驗證用於啟動執行個體的初始啟動程式碼。

若要使用 AMD SEV-SNP 執行認證，請完成下列步驟。

### 步驟 1：取得認證報告

在此步驟中，您要安裝並建置公用 `snpghost` 程式，然後使用它來要求 AMD SEV-SNP 證明報告和憑證。

1. 執行下列命令以從建置 `snpghost` 公用程式 [snpghost repository](#)。

```
$ C:\> git clone https://github.com/virtee/snpghost.git
```



```
$ C:\> cd snpguest
$ C:\> cargo build -r
$ C:\> cd target/release
```

2. 產生驗證報告的請求。公用程式會向主機要求驗證報告，並將其寫入包含所提供請求資料的二進位檔案。

下列範例會建立隨機要求字串，並使用它做為要求檔案 (`request-file.txt`)。當指令傳回驗證報告時，它會儲存在您指定的檔案路徑 (`report.bin`) 中。在此情況下，公用程式會將報告儲存在目前目錄中。

```
$ C:\> ./snpguest report report.bin request-file.txt --random
```

3. 從主機記憶體要求憑證，並將其儲存為 PEM 檔案。下列範例會將檔案儲存在與 `snpguest` 公用程式相同的目錄中。如果指定目錄中已存在憑證，則會覆寫這些憑證。

```
$ C:\> ./snpguest certificates PEM ./
```

## 步驟 2：驗證報告簽名

驗證報告使用稱為版本化的載入背書金鑰 (VLEK) 簽署的憑證，此憑證由 AMD 發行。AWS 在此步驟中，您可以驗證 VLEK 憑證是由 AMD 核發，並且驗證報告是由該 VLEK 憑證簽署。

1. 從 AMD 官方網站下載信任證書的 VLEK 根目錄到當前目錄。

```
$ C:\> sudo curl --proto '=https' --tlsv1.2 -sSf https://kdsintf.amd.com/vlek/v1/Milan/cert_chain -o ./cert_chain.pem
```

2. 使用 `openssl` 來驗證 VLEK 憑證是否由 AMD 信任根憑證簽署。

```
$ C:\> sudo openssl verify --CAfile ./cert_chain.pem vlek.pem
```

預期的輸出結果：

```
certs/vcek.pem: OK
```

3. 使用 `snpguest` 公用程式來驗證認證報告由 VLEK 憑證簽署。

```
$ C:\> ./snpguest verify attestation ./ report.bin
```

預期的輸出結果。

```
Reported TCB Boot Loader from certificate matches the attestation report.
Reported TCB TEE from certificate matches the attestation report.
Reported TCB SNP from certificate matches the attestation report.
Reported TCB Microcode from certificate matches the attestation report.
VEK signed the Attestation Report!
```

## 使用安裝媒體新增 Windows 系統元件

Windows Server 作業系統包含許多選用元件。在每個 Amazon EC2 Windows Server AMI 中包含所有選用元件不太實際。相反的，我們提供您安裝媒體 EBS 快照，其中具備在您的 Windows 執行個體上設定或安裝元件所需要的檔案。

若要存取及安裝選用元件，您必須尋找屬於您版本 Windows Server 的正確 EBS 快照，從快照建立磁碟區，然後將磁碟區連接到您的執行個體。

### 開始之前

使用 AWS Management Console 或命令列工具取得執行個體的執行個體 ID 和可用區域。您必須在與您執行個體相同的可用區域中建立您的 EBS 磁碟區。

### 使用主控台新增 Windows 元件

使用下列程序來使用將 Windows 元件 AWS Management Console 加入您的執行個體。

使用主控台為您的執行個體新增 Windows 元件

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Snapshots (快照)。
3. 從 Filter (篩選條件) 列中，選擇 Public Snapshots (公有快照)。
4. 新增 Owner Alias (擁有者別名) 篩選條件，然後選擇 amazon。
5. 新增 Description (描述) 篩選條件，然後輸入 **Windows**。
6. 按 Enter
7. 選取符合您系統架構及語言偏好設定的快照。例如，若執行個體執行 Windows Server 2019，請選取 Windows 2019 English Installation Media (Windows 2019 英文版安裝媒體)。

- 選擇 Actions (動作)、Create volume from snapshot (從快照建立磁碟區)。
- 針對可用區域，選取符合您 Windows 執行個體的可用區域。選擇 Add tag (新增標籤)，然後為標籤索引鍵指定 **Name**，並為標籤值指定描述性名稱。選擇建立磁碟區。
- 在 Volume Successfully Created (磁碟區建立成功) 訊息 (綠色橫幅) 中，選擇您剛剛建立的磁碟區。
- 選擇 Actions (動作)、Attach Volume (連接磁碟區)。
- 從 Instance (執行個體) 中，選取執行個體 ID。
- 對於 Device name (裝置名稱)，輸入配件的裝置名稱。如需裝置名稱的說明，請參閱 [Amazon EC2 實例上的設備名稱](#)。
- 選擇 Attach volume (連接磁碟區)。
- 連線到您的執行個體，然後讓磁碟區可用。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的讓 Amazon EBS 磁碟區可供使用](#)。

 Important

請不要初始化磁碟區。

- 開啟 Control Panel (控制面板)，Programs and Features (程式和功能)。選擇 Turn Windows features on or off (開啟或關閉 Windows 功能)。若您收到提示要求安裝媒體，請指定使用安裝媒體的 EBS 磁碟區。
- (選用) 完成媒體安裝後，您可以分離磁碟區。分離磁碟區後，即可將它刪除。

## 使用視窗工具加入視窗元件 PowerShell

請使用下列程序，使用 Windows 專用工具將 Windows 元件新增 PowerShell 至您的執行個體。

使用視窗適用的工具將 Windows 元件新增至您的執行個體 PowerShell

- 搭配 Owner 和 description 篩選器使用 [Get-EC2Snapshot](#) 指令程式，以取得可用安裝媒體快照集的清單。

```
PS C:\> Get-EC2Snapshot -Owner amazon -Filter @{ Name="description";
Values="Windows*" }
```

- 在輸出中，注意符合您系統架構和語言偏好設定的快照 ID。例如：

...

```

DataEncryptionKeyId :
Description : Windows 2019 English Installation Media
Encrypted : False
KmsKeyId :
OwnerAlias : amazon
OwnerId : 123456789012
Progress : 100%
SnapshotId : snap-22da283e
StartTime : 10/25/2019 8:00:47 PM
State : completed
StateMessage :
Tags : {}
VolumeId : vol-be5eafcb
VolumeSize : 6
...

```

3. 使用指 [New-EC2Volume](#) 令程式從快照建立磁碟區。指定與您執行個體相同的可用區域。

```

PS C:\> New-EC2Volume -AvailabilityZone us-east-1a -VolumeType gp2 -
SnapshotId snap-22da283e

```

4. 在輸出中，記下磁碟區 ID。

```

Attachments : {}
AvailabilityZone : us-east-1a
CreateTime : 4/18/2017 10:50:25 AM
Encrypted : False
Iops : 100
KmsKeyId :
Size : 6
SnapshotId : snap-22da283e
State : creating
Tags : {}
VolumeId : vol-06aa9e1fbf8b82ed1
VolumeType : gp2

```

5. 使用指 [Add-EC2Volume](#) 令程式將磁碟區附加至您的執行個體。

```

PS C:\> Add-EC2Volume -InstanceId i-087711ddaf98f9489 -
VolumeId vol-06aa9e1fbf8b82ed1 -Device xvdh

```

6. 連線到您的執行個體，然後讓磁碟區可用。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的讓 Amazon EBS 磁碟區可供使用](#)。

**⚠ Important**

請不要初始化磁碟區。

7. 開啟 Control Panel (控制面板), Programs and Features (程式和功能)。選擇 Turn Windows features on or off (開啟或關閉 Windows 功能)。若您收到提示要求安裝媒體, 請指定使用安裝媒體的 EBS 磁碟區。
8. (選擇性) 當您完成安裝媒體時, 請使用 [Dismount-EC2Volume](#) 指令程式將磁碟區與執行個體中斷連結。卸離磁碟區之後, 您可以使用指 [Remove-EC2Volume](#) 指令程式刪除磁碟區。

## 使用新增視窗元件 AWS CLI

使用下列程序來使用將 Windows 元件 AWS CLI 加入您的執行個體。

若要使用將 Windows 元件新增至您的執行個體 AWS CLI

1. 使用 [describe-snapshots](#) 命令, 搭配 `owner-ids` 參數和 `description` 篩選條件, 取得可用安裝媒體快照的清單。

```
aws ec2 describe-snapshots --owner-ids amazon --filters
Name=description,Values=Windows*
```

2. 在輸出中, 注意符合您系統架構和語言偏好設定的快照 ID。例如:

```
{
 "Snapshots": [
 ...
 {
 "OwnerAlias": "amazon",
 "Description": "Windows 2019 English Installation Media",
 "Encrypted": false,
 "VolumeId": "vol-be5eafcb",
 "State": "completed",
 "VolumeSize": 6,
 "Progress": "100%",
 "StartTime": "2019-10-25T20:00:47.000Z",
 "SnapshotId": "snap-22da283e",
 "OwnerId": "123456789012"
 },
 ...
]
}
```

```
]
}
```

3. 使用 [create-volume](#) 命令從快照建立磁碟區。指定與您執行個體相同的可用區域。

```
aws ec2 create-volume --snapshot-id snap-22da283e --volume-type gp2 --availability-
zone us-east-1a
```

4. 在輸出中，記下磁碟區 ID。

```
{
 "AvailabilityZone": "us-east-1a",
 "Encrypted": false,
 "VolumeType": "gp2",
 "VolumeId": "vol-0c98b37f30bcbc290",
 "State": "creating",
 "Iops": 100,
 "SnapshotId": "snap-22da283e",
 "CreateTime": "2017-04-18T10:33:10.940Z",
 "Size": 6
}
```

5. 使用 [attach-volume](#) 命令將磁碟區連接到您的執行個體。

```
aws ec2 attach-volume --volume-id vol-0c98b37f30bcbc290 --instance-
id i-01474ef662b89480 --device xvdg
```

6. 連線到您的執行個體，然後讓磁碟區可用。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的讓 Amazon EBS 磁碟區可供使用](#)。

#### Important

請不要初始化磁碟區。

7. 開啟 Control Panel (控制面板)，Programs and Features (程式和功能)。選擇 Turn Windows features on or off (開啟或關閉 Windows 功能)。若您收到提示要求安裝媒體，請指定使用安裝媒體的 EBS 磁碟區。
8. (選用) 完成媒體安裝後，請使用 [detach-volume](#) 命令將磁碟區與您的執行個體分離。分離磁碟區後，您可以使用 [delete-volume](#) 命令來刪除磁碟區。

## 管理 Linux 執行個體上的系統使用者

每個 Linux 執行個體都會使用預設 Linux 系統使用者來啟動。您可以將使用者新增至執行個體，並刪除使用者。

對於預設使用者，[預設使用者名稱](#)會取決於您啟動執行個體時指定的 AMI。

### Note

預設情況下，密碼驗證和根登入會停用，而 sudo 會啟用。若要登入執行個體，您必須建立金鑰對。如需有關登入的詳細資訊，請參閱[連接至您的 Linux 執行個體](#)。

您可以允許執行個體的密碼驗證和根登入。如需詳細資訊，請參閱適用於您作業系統的文件。

### Note

Linux 系統使用者不應和 IAM 使用者混淆。如需詳細資訊，請參閱《IAM 使用者指南》中的[IAM 使用者](#)。

## 目錄

- [預設使用者名稱](#)
- [考量事項](#)
- [建立使用者](#)
- [移除使用者](#)

## 預設使用者名稱

EC2 執行個體的預設使用者名稱取決於您啟動執行個體時指定的 AMI。

預設使用者名稱為：

- 對於 AL2023、Amazon Linux 2 或 Amazon Linux AMI，使用者名稱為 ec2-user。
- 若使用 CentOS AMI，使用者名稱為 centos 或 ec2-user。
- 若為 Debian AMI，使用者名稱為 admin。
- 對於 Fedora AMI，使用者名稱為 fedora 或 ec2-user。

- 對於 RHEL AMI，使用者名稱為 `ec2-user` 或 `root`。
- 針對 SUSE AMI，使用者名稱為 `ec2-user` 或 `root`。
- 若為 Ubuntu AMI，使用者名稱為 `ubuntu`。
- 若使用 Oracle AMI，使用者名稱為 `ec2-user`。
- 若使用 Bitnami AMI，使用者名稱為 `bitnami`。

### Note

若要尋找其他 Linux 發行版的預設使用者名稱，請洽詢 AMI 提供者。

## 考量事項

許多應用程式都適合使用預設使用者。不過，您可以選擇新增使用者，讓個人能有自己的檔案和工作空間。此外，為新使用者建立使用者也會比授予多名 (可能缺乏經驗) 預設使用者的存取權更安全，因為若預設使用者使用不當，可能會對系統造成重大傷害。如需詳細資訊，請參閱[保護 EC2 執行個體的要訣](#)。

若要讓使用者能夠使用 Linux 系統使用者，透過 SSH 存取您的 EC2 執行個體，您必須與使用者共用 SSH 金鑰。或者，您可以使用 EC2 Instance Connect 為使用者提供存取權限，如此就不需要共用和管理 SSH 金鑰。如需詳細資訊，請參閱[使用 EC2 Instance Connect 連線至 Linux 執行個體](#)。

## 建立使用者

首先建立使用者，然後新增允許使用者連線至並登入執行個體的 SSH 公有金鑰。

### 建立使用者

1. [建立新的金鑰對](#)。您必須將 `.pem` 檔案提供給您要為其建立使用者的使用者。他們必須使用此檔案來連接到執行個體。
2. 從您在上一個步驟中建立的金鑰對中，擷取公開金鑰。

```
$ C:\> ssh-keygen -y -f /path_to_key_pair/key-pair-name.pem
```

此命令會傳回公有金鑰，如下列範例所示。

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQClKsfkNkuSevGj3eYhCe53pcjqP3maAhDFcvBS706Vhz2ItxCih
```



```
+PnDSUaw+WNQn/mZphTk/a/gU8jEzo0WbkM4yxyb/wB96xbiFveSFJu0p/
d6RJhJ0I0iBxrlsLnBItnctkiJ7FbtXJMXLvvwJryDUilBMTjYtwB+QhYXUM0zce5Pjz5/
i8SeJtjnV3iAoG/cQk+0FzZqaeJAAHco
+CY/5WrUBkrHmFJr6HcXkvJdWPkYQS3xqC0+FmUZofz221CBt5IMucxXPkX4rWi
+z7wB3RbBQoQzd8v7yeb70z1PnW0yN0qFU0XA246RA8QFYiCNYwI3f05p6KLxEXAMPLE
```

3. 連線到執行個體。
4. 使用 `adduser` 命令建立使用者並將其新增到系統 (在 `/etc/passwd` 檔案中具有項目)。此命令也會為使用者建立群組和主目錄。在此範例中，使用者命名為 *newuser*。

- Amazon Linux 和 Amazon Linux 2

如果使用 Amazon Linux 和 Amazon Linux 2，則在預設情況下，建立的使用者已停用密碼驗證。

```
[ec2-user ~]$ sudo adduser newuser
```

- Ubuntu

包含 `--disabled-password` 參數，以建立已停用密碼驗證的使用者。

```
[ubuntu ~]$ sudo adduser newuser --disabled-password
```

5. 切換到新的使用者，以便您建立的目錄和檔案可擁有適當的所有權。

```
[ec2-user ~]$ sudo su - newuser
```

此提示會從 `ec2-user` 變更為 *newuser*，表示您已將 shell 工作階段切換至新使用者。

6. 將 SSH 公有金鑰新增到使用者。首先在使用者的主目錄中建立 SSH 金鑰檔案的目錄、然後建立金鑰檔案，最後將公有金鑰貼到金鑰檔案中，如下列子步驟中所示。
  - a. 在 `.ssh` 主目錄中建立 *newuser* 目錄，然後將其檔案許可變更為 `700` (只有擁有者能讀取、寫入和開啟目錄)。

```
[newuser ~]$ mkdir .ssh
```

```
[newuser ~]$ chmod 700 .ssh
```

**⚠ Important**

若沒有這些明確的檔案許可，使用者將無法登入。

- b. 在 `.ssh` 目錄中建立名為 `authorized_keys` 的檔案，然後將其檔案許可變更為 `600` (只有擁有者能讀取和寫入檔案)。

```
[newuser ~]$ touch .ssh/authorized_keys
```

```
[newuser ~]$ chmod 600 .ssh/authorized_keys
```

**⚠ Important**

若沒有這些明確的檔案許可，使用者將無法登入。

- c. 使用您喜愛的文字編輯器 (例如 `vim` 或 `nano`) 來開啟 `authorized_keys` 檔案。

```
[newuser ~]$ nano .ssh/authorized_keys
```

將您在步驟 2 中擷取到的公開金鑰貼到檔案中，並儲存這些變更。

**⚠ Important**

請確定將此公開金鑰貼成連續一行。此公開金鑰絕對不可分成多行。

使用者現在應該能使用對應到您為 `authorized_keys` 檔案新增之公有金鑰的私有金鑰，在您的執行個體上登入 `newuser` 使用者。如需連接至 Linux 執行個體之不同方法的詳細資訊，請參閱[連接至您的 Linux 執行個體](#)。

## 移除使用者

若不再需要使用者，您可以移除該使用者，使該使用者無法再被使用。

使用 `userdel` 命令來移除系統的使用者。當指定 `-r` 參數時，也會同時刪除使用者的主目錄和郵件多工緩衝處理。若要保留使用者的主目錄和郵件多工緩衝處理，請忽略 `-r` 參數。

```
[ec2-user ~]$ sudo userdel -r olduser
```

## 為您的執行個體設定 Windows 管理員密碼

當您連線至 Windows 執行個體時，您必須指定使用者帳戶和有存取執行個體許可的密碼。第一次連線至執行個體時，系統會提示您指定管理員帳戶和預設密碼。

使用 AWS 視窗 AMI 適用於視窗伺服器 2012 R2 及更早版本，[使用 EC2Config 服務 \(舊版\) 設定 Windows 執行個體](#) 會產生預設密碼。使用 AWS 視窗 AMI 適用於視窗伺服器 2016 和 2019，[使用 EC2Launch 設定 Windows 執行個體](#) 會產生預設密碼。使用適用於 AWS 視窗伺服器 2022 及更新版本的 AMI，[使用 EC2Launch v2 設定 Windows 執行個體](#) 會產生預設密碼。

### Note

使用 Windows Server 2016 和更新版本，會停用本機管理員的 Password never expires。使用 Windows Server 2012 R2 及舊版，啟用本機管理員的 Password never expires。

## 連線後變更管理員密碼

當您第一次連線至執行個體時，建議您變更預設的管理員密碼。使用下列程序變更 Windows 執行個體的管理員密碼。

### Important

將新密碼存放於安全處。您無法使用 Amazon EC2 主控台擷取新密碼。主控台只能擷取預設的密碼。如果您在變更密碼後嘗試使用預設密碼連線執行個體，您會收到 "Your credentials did not work" (您的登入資料無法作用) 錯誤。

## 變更本機管理員密碼

1. 連線至執行個體並開啟命令提示。
2. 執行下列命令。如果您的新密碼包含特殊字元，請務必以雙引號括住密碼。

```
net user Administrator "new_password"
```

### 3. 將新密碼存放於安全處。

## 變更遺失或過期的密碼

如果您的密碼遺失或過期，您可產生新的密碼。如需密碼重設程序，請參閱[重設遺失或過期的 Windows 管理員密碼](#)。

## 管理 Amazon EC2 執行個體的裝置驅動程式

您從中啟動的 EC2 AMI 上未預先安裝某些驅動程式。其他人可能需要更新才能利用擴充功能。下列主題涵蓋部分連接至 EC2 執行個體之裝置驅動程式的安裝、更新和組態。

### 目錄

- [在您的 Amazon EC2 實例上安裝 NVIDIA 驅動程式](#)
- [在您的 Amazon EC2 實例上安裝 AMD 驅動程式](#)
- [Windows 執行個體的全虛擬驅動程式](#)
- [AWS 適用於視窗執行個體的 NVMe](#)

## 在您的 Amazon EC2 實例上安裝 NVIDIA 驅動程式

已連接 NVIDIA GPU 的執行個體 (例如 P3 或 G4dn 執行個體) 必須安裝適當的 NVIDIA 驅動程式。根據執行個體類型，您可以下載公有 NVIDIA 驅動程式、從僅適用於 AWS 客戶的 Simple Storage Service (Amazon S3) 下載驅動程式，或將 AMI 與預先安裝的驅動程式搭配使用。

若要在具有連接 AMD GPU 的執行個體 (例如 G4ad 執行個體) 上安裝 AMD 驅動程式，請參閱。[安裝 AMD 驅動程式](#)若要安裝 NVIDIA 驅動程式，請參閱[安裝 NVIDIA 驅動程式](#)。

### 內容

- [NVIDIA 驅動程式類型](#)
- [依執行個體類型分類的可用驅動程式](#)
- [安裝選項](#)
  - [選項 1：已安裝 NVIDIA 驅動程式的 AMI](#)
  - [選項 2：公有 NVIDIA 驅動程式](#)
  - [選項 3：網格驅動程式 \(G6、G6、G5、G4DN 和 G3 執行個體\)](#)

- [選項 4：NVIDIA 遊戲驅動程式 \(G5 和 G4dn 執行個體\)](#)
- [安裝 CUDA 附加版本](#)

## NVIDIA 驅動程式類型

以下是可與以 GPU 為基礎之執行個體搭配使用的主要 NVIDIA 驅動程式類型。

### Tesla 驅動程式

這些驅動程式主要用於運算工作負載，將 GPU 用於平行化浮點計算等運算任務，以進行機器學習，以及用於高效能運算應用程式的快速 Fourier 轉換。

### GRID 驅動程式

這些驅動程式經過認證，可為轉譯 3D 模型或高解析度影片等內容的專業視覺化應用程式提供最佳效能。您可以將 GRID 驅動程式設定為支援兩種模式。Quadro 虛擬工作站可提供每個 GPU 四個 4K 顯示器的存取權限。GRID vApp 可提供 RDSH 應用程式託管功能。

### 遊戲驅動程式

這些驅動程式包含遊戲的最佳化功能，而且經常更新，可提供效能增強功能。它們支援每個 GPU 一個 4K 顯示器。

## 設定的模式

在 Windows 上，Tesla 驅動程式是設定為在 Tesla Compute Cluster (TCC) 模式下執行。GRID 和遊戲驅動程式是設定為在 Windows Display Driver Model (WDDM) 模式下執行。在 TCC 模式下，卡片可供運算工作負載專用。在 WDDM 模式下，卡片可同時支援運算和圖形工作負載。

## NVIDIA 控制面板

GRID 和遊戲驅動程式不支援 NVIDIA 控制面板。Tesla 驅動程式不支援此面板。

## Tesla、GRID 和遊戲驅動程式支援的 API

- OpenCL、OpenGL 和 Vulkan
- NVIDIA CUDA 和相關程式庫 (例如 cuDNN、TensorRT、nvJPEG 和 cuBLAS)
- 適用於影片編碼的 NVENC 以及適用於影片解碼的 NVDEC
- 僅限視窗的應用程式介面：DirectX 接輸入、DirectX 像加速、DirectX 線追蹤

## 依執行個體類型分類的可用驅動程式

下表摘述每種 GPU 執行個體類型的支援 NVIDIA 驅動程式。

執行個體類型	Tesla 驅動程式	GRID 驅動程式	遊戲驅動程式
G3	是	是	否
G4dn	是	是	是
G5	是	是	是
G5g	是 <sup>1</sup>	否	否
G6	是	是	否
GR6	是	是	否
P2	是	否	否
P3	是	否	否
P4d	是	否	否
P4de	是	否	否

<sup>1</sup> 此 Tesla 驅動程式也支援 ARM64 平台專屬的最佳化圖形應用程式。

<sup>2</sup> 僅使用 Marketplace AMI

### 安裝選項

使用下列其中一個選項，取得 GPU 執行個體所需的 NVIDIA 驅動程式。

#### 選項

- [選項 1：已安裝 NVIDIA 驅動程式的 AMI](#)
- [選項 2：公有 NVIDIA 驅動程式](#)
- [選項 3：網格驅動程式 \(G6、G6、G5、G4DN 和 G3 執行個體\)](#)
- [選項 4：NVIDIA 遊戲驅動程式 \(G5 和 G4dn 執行個體\)](#)

## 選項 1：已安裝 NVIDIA 驅動程式的 AMI

AWS 和 NVIDIA 提供了不同的 Amazon 機器映像 (AMI) 隨附的 NVIDIA 驅動程序安裝。

- [具有 Tesla 驅動程式的 Marketplace 產品](#)
- [具有 GRID 驅動程式的 Marketplace 產品](#)
- [具有遊戲驅動程式的 Marketplace 產品](#)

若要檢閱依賴於您的作業系統 (OS) 平台的考量事項，請選擇適用於 AMI 的索引標籤。

### Linux

若要使用這些 AMI 的其中之一來更新安裝的驅動程式版本，您必須從執行個體中解除安裝 NVIDIA 套件，以避免發生版本衝突。使用此命令解除安裝 NVIDIA 套件：

```
[ec2-user ~]$ sudo yum erase nvidia cuda
```

CUDA 工具組套件與 NVIDIA 驅動程式具有相依性。解除安裝 NVIDIA 套件會清除 CUDA 工具組。在安裝 NVIDIA 驅動程式之後，您必須解除安裝 CUDA 工具組。

### Windows

如果您使用其中一個供應項目建立自訂的 Windows AMI AWS Marketplace I，則 AMI 必須是使用 Windows Sysprep 建立的標準化映像，以確保 GRID 驅動程式可正常運作。如需詳細資訊，請參閱 [使用視窗系統建立 AMI](#)。

## 選項 2：公有 NVIDIA 驅動程式

提供的選項 AWS 隨驅動程序必要的許可證一起提供。或者，您可以安裝公用驅動程式並攜帶自己的授權。若要安裝公用驅動程式，請依照此處所述從 NVIDIA 網站下載。

或者，您可以使用提供的選項，AWS 而不是公共驅動程序。若要在 P3 執行個體上使用 GRID 驅動程式，請依照 [選項 1](#) 中所述使用 AWS Marketplace AMI。若要在 G6、G6、G5、G4dn 或 G3 執行個體上使用網格驅動程式，請依照 [選項 1](#) 中所述使用 AWS Marketplace AMI，或依照中所述安裝所提供的 NVIDIA 驅動程式。AWS [選項 3：網格驅動程式 \(G6、G6、G5、G4DN 和 G3 執行個體\)](#)

### 下載公用 NVIDIA 驅動程式

登入您的執行個體，然後從 <http://www.nvidia.com/Download/Find.aspx> 下載適用於執行個體類型的 64 位元 NVIDIA 驅動程式。對於產品類型、產品系列及產品，請使用下表中的選項。

執行個體	產品類型	產品系列	產品
G3	Tesla	M-Class	M60
G4dn	Tesla	T 系列	T4
G5 <sup>1</sup>	Tesla	A 系列	A10
G5g <sup>2</sup>	Tesla	T 系列	NVIDIA T4G
G6 <sup>3</sup>	Tesla	L 型系列	L4
GR6 <sup>3</sup>	Tesla	L 型系列	L4
P2	Tesla	K 系列	K80
P3	Tesla	V 系列	V100
P4d	Tesla	A 系列	A100
P4de	Tesla	A 系列	A100
四季	Tesla	H 系列	H100

<sup>1</sup> G5 執行個體需要驅動程式版本 470.00 或更新版本。

<sup>2</sup> G5g 執行個體需要驅動程式版本 470.82.01 或更新版本。作業系統是 Linux aarch64

<sup>3</sup> G6 和 Gr6 執行個體需要驅動程式版本 525.0 或更新版本。

<sup>4</sup> 個 P5 執行個體需要 530 或更新版本的驅動程式。

若要在 Linux 作業系統上安裝 NVIDIA 驅動程式，請參閱 [NVIDIA 驅動程式安裝快速入門指南](#)。

若要在視窗上安裝 NVIDIA 驅動程式，請依照下列步驟執行：

1. 開啟您下載驅動程式的所在資料夾，並啟動安裝檔案。依照指示安裝驅動程式，並視需要將執行個體重新開機。
2. 使用 Device Manager 停用標有警告圖示的名為 Microsoft Basic Display Adapter 的顯示轉接器。安裝這些 Windows 功能：Media Foundation 和 Quality Windows Audio Video Experience (高品質 Windows 音訊/視訊體驗)。



**⚠ Important**

請勿停用名為 Microsoft Remote Display Adapter 的顯示轉接器。如果停用 Microsoft Remote Display Adapter，您的連線可能會中斷，並在重新開機後會嘗試連線至執行個體可能會失敗。

3. 檢查裝置管理員，確認 GPU 是否正常運作。
4. 若要達到 GPU 的最佳效能，請完成[優化 Amazon EC2 執行個體上的 GPU 設定](#)中的最佳化步驟。

**選項 3：網格驅動程式 (G6、G6、G5、G4DN 和 G3 執行個體)**

這些下載僅供 AWS 客戶使用。下載後，為了遵守 NVIDIA GRID 雲端使用者授權合約 (EULA) 中提到的 AWS 解決方案要求，您同意使用下載的軟體僅用於開發 AMI，以搭配 NVIDIA L4、NVIDIA T10G、NVIDIA 特斯拉 T4 或 NVIDIA 特斯拉 M60 硬體使用。安裝軟體時，您必須遵循 [NVIDIA GRID 雲端最終使用者授權合約](#) 的條款。如需作業系統之 NVIDIA GRID 驅動程式版本的詳細資訊，請參閱 NVIDIA 網站上的 [NVIDIA® Virtual GPU \(vGPU\) 軟體文件](#)。

**考量事項**

- G6 和 Gr6 執行個體需要 GRID 17 或更新版本。
- G5 執行個體需要 GRID 13.1 或更新版本 (或 GRID 12.4 或更新版本)。
- G3 執行個體需要 AWS 提供 DNS 解析，GRID 授權才能運作。
- 僅 14.0 版或更高版本的 NVIDIA 驅動程式支援 [IMDSv2](#)。
- 對於 Windows 執行個體，如果您從自訂的 Windows AMI 啟動執行個體，則 AMI 必須是使用 Windows Sysprep 建立的標準化映像檔，以確保 GRID 驅動程式可正常運作。如需詳細資訊，請參閱 [使用視窗系統建立 AMI](#)。
- 網格 17.0 及更新版本不支援視窗伺服器 2019。
- 網格 14.2 及更新版本不支援視窗伺服器 2016。
- G3 執行個體不支援 GRID 17.0 及更新版本。

**Amazon Linux 和 Amazon Linux 2****在執行個體上安裝 NVIDIA GRID 驅動程式**

1. 連線至您的 Linux 執行個體。

2. AWS CLI 在您的 Linux 執行個體上安裝並設定預設認證。如需詳細資訊，請參閱《AWS Command Line Interface 使用者指南》中的[安裝 AWS CLI](#)。

**⚠ Important**

您的使用者或角色必須擁有包含 AmazonS3 ReadOnly 存取政策的授與權限。如需詳細資訊，請參閱 [AWSAmazon 簡單儲存服務使用者指南中的受管政策：AmazonS3 存 ReadOnly 取](#)。

3. 如果尚未安裝 gcc 和 make，請進行安裝。

```
[ec2-user ~]$ sudo yum install gcc make
```

4. 更新套件快取，並取得執行個體的套件更新。

```
[ec2-user ~]$ sudo yum update -y
```

5. 將執行個體重新開機以載入最新的核心版本。

```
[ec2-user ~]$ sudo reboot
```

6. 重新開機後，請重新連線至您的執行個體。
7. 針對您目前執行的核心版本，安裝 gcc 編譯器與核心標頭套件。

```
[ec2-user ~]$ sudo yum install -y gcc kernel-devel-$(uname -r)
```

8. 使用下列命令下載 GRID 驅動程式安裝公用程式：

```
[ec2-user ~]$ aws s3 cp --recursive s3://ec2-linux-nvidia-drivers/latest/ .
```

此儲存貯體會保留 GRID 驅動程式的多個版本。您可以使用下列命令查看所有可用版本。

```
[ec2-user ~]$ aws s3 ls --recursive s3://ec2-linux-nvidia-drivers/
```

9. 請使用以下命令新增許可來執行驅動程式安裝公用程式。

```
[ec2-user ~]$ chmod +x NVIDIA-Linux-x86_64*.run
```

10. 如下所述執行自我安裝指令碼，以安裝您已下載的 GRID 驅動程式。例如：

```
[ec2-user ~]$ sudo /bin/sh ./NVIDIA-Linux-x86_64*.run
```

**Note**

如果您是使用 Amazon Linux 2 搭配核心版本 5.10，請使用下列命令來安裝 GRID 驅動程式。

```
[ec2-user ~]$ sudo CC=/usr/bin/gcc10-cc ./NVIDIA-Linux-x86_64*.run
```

出現提示時，請接受授權合約，並依需求指定安裝選項 (您可以接受預設選項)。

11. 確認驅動程式可正常運作。下列命令的回應會列出已安裝的 NVIDIA 驅動程式版本和 GPU 詳細資料。

```
[ec2-user ~]$ nvidia-smi -q | head
```

12. 如果您在 G4dn、G5 或 G5g 執行個體上使用 14.x 版或更高版本的 NVIDIA vGPU 軟體，請使用以下命令來停用 GSP。如需有關為何需要執行此程序的詳細資訊，請瀏覽 [NVIDIA 的說明文件](#)。

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /etc/modprobe.d/nvidia.conf
```

13. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

14. (選用) 視您的使用案例而定，您可能會完成下列選用步驟。如果您不需要這項功能，請勿完成這些步驟。
  - a. 為了協助善加利用最高 4K 解析度的四個顯示器，請設定高效能顯示器通訊協定 [NICE DCV](#)。
  - b. 預設會啟用 NVIDIA Quadro 虛擬工作站。若要針對主控功能的 RDSH 應用程式啟動 GRID 虛擬應用程式，請完成在您的 [Amazon EC2 GPU 執行個體上啟用 NVIDIA 網格虛擬應用程式](#) 中的 GRID 虛擬應用程式啟動步驟。

## CentOS 7 和 Red Hat Enterprise Linux 7

### 在執行個體上安裝 NVIDIA GRID 驅動程式

1. 連線至您的 Linux 執行個體。如果尚未安裝 gcc 和 make，請進行安裝。
2. 更新套件快取，並取得執行個體的套件更新。

```
[ec2-user ~]$ sudo yum update -y
```

3. 將執行個體重新開機以載入最新的核心版本。

```
[ec2-user ~]$ sudo reboot
```

4. 重新開機後，請重新連線至您的執行個體。
5. 針對您目前執行的核心版本，安裝 gcc 編譯器與核心標頭套件。

```
[ec2-user ~]$ sudo yum install -y gcc kernel-devel-$(uname -r)
```

6. 停用 NVIDIA 圖形卡的 nouveau 開放原始碼驅動程式。
  - a. 將 nouveau 新增到 /etc/modprobe.d/blacklist.conf 黑名單檔案。複製下列程式碼區塊並貼至終端機。

```
[ec2-user ~]$ cat << EOF | sudo tee --append /etc/modprobe.d/blacklist.conf
blacklist vga16fb
blacklist nouveau
blacklist rivafb
blacklist nvidiafb
blacklist rivatv
EOF
```

- b. 編輯 /etc/default/grub 檔案並新增下行：

```
GRUB_CMDLINE_LINUX="rdblacklist=nouveau"
```

- c. 重建 Grub 組態。

```
[ec2-user ~]$ sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

7. 使用下列命令下載 GRID 驅動程式安裝公用程式：

```
[ec2-user ~]$ aws s3 cp --recursive s3://ec2-linux-nvidia-drivers/latest/ .
```

此儲存貯體會保留 GRID 驅動程式的多個版本。您可以使用下列命令查看所有可用版本。

```
[ec2-user ~]$ aws s3 ls --recursive s3://ec2-linux-nvidia-drivers/
```

- 請使用以下命令新增許可來執行驅動程式安裝公用程式。

```
[ec2-user ~]$ chmod +x NVIDIA-Linux-x86_64*.run
```

- 如下所述執行自我安裝指令碼，以安裝您已下載的 GRID 驅動程式。例如：

```
[ec2-user ~]$ sudo /bin/sh ./NVIDIA-Linux-x86_64*.run
```

出現提示時，請接受授權合約，並依需求指定安裝選項 (您可以接受預設選項)。

- 確認驅動程式可正常運作。下列命令的回應會列出已安裝的 NVIDIA 驅動程式版本和 GPU 詳細資料。

```
[ec2-user ~]$ nvidia-smi -q | head
```

- 如果您在 G4dn、G5 或 G5g 執行個體上使用 14.x 版或更高版本的 NVIDIA vGPU 軟體，請使用以下命令來停用 GSP。如需有關為何需要執行此程序的詳細資訊，請瀏覽 [NVIDIA 的說明文件](#)。

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /etc/modprobe.d/nvidia.conf
```

- 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

- (選用) 視您的使用案例而定，您可能會完成下列選用步驟。如果您不需要這項功能，請勿完成這些步驟。

- 為了協助善加利用最高 4K 解析度的四個顯示器，請設定高效能顯示器通訊協定 [NICE DCV](#)。

- b. 預設會啟用 NVIDIA Quadro 虛擬工作站。若要針對主控功能的 RDSH 應用程式啟動 GRID 虛擬應用程式，請完成[在您的 Amazon EC2 GPU 執行個體上啟用 NVIDIA 網格虛擬應用程式](#)中的 GRID 虛擬應用程式啟動步驟。
- c. 安裝 GUI 桌上型電腦/工作站套件。

```
[ec2-user ~]$ sudo yum groupinstall -y "Server with GUI"
```

## CentOS Stream 8 和 Red Hat Enterprise Linux 8

### 在執行個體上安裝 NVIDIA GRID 驅動程式

1. 連線至您的 Linux 執行個體。如果尚未安裝 gcc 和 make，請進行安裝。
2. 更新套件快取，並取得執行個體的套件更新。

```
[ec2-user ~]$ sudo yum update -y
```

3. 將執行個體重新開機以載入最新的核心版本。

```
[ec2-user ~]$ sudo reboot
```

4. 重新開機後，請重新連線至您的執行個體。
5. 針對您目前執行的核心版本，安裝 gcc 編譯器與核心標頭套件。

```
[ec2-user ~]$ sudo dnf install -y make gcc elfutils-libelf-devel libglvnd-devel kernel-devel-$(uname -r)
```

6. 使用下列命令下載 GRID 驅動程式安裝公用程式：

```
[ec2-user ~]$ aws s3 cp --recursive s3://ec2-linux-nvidia-drivers/latest/ .
```

此儲存貯體會保留 GRID 驅動程式的多個版本。您可以使用下列命令查看所有可用版本。

```
[ec2-user ~]$ aws s3 ls --recursive s3://ec2-linux-nvidia-drivers/
```

7. 請使用以下命令新增許可來執行驅動程式安裝公用程式。

```
[ec2-user ~]$ chmod +x NVIDIA-Linux-x86_64*.run
```

- 如下所述執行自我安裝指令碼，以安裝您已下載的 GRID 驅動程式。例如：

```
[ec2-user ~]$ sudo /bin/sh ./NVIDIA-Linux-x86_64*.run
```

出現提示時，請接受授權合約，並依需求指定安裝選項 (您可以接受預設選項)。

- 確認驅動程式可正常運作。下列命令的回應會列出已安裝的 NVIDIA 驅動程式版本和 GPU 詳細資料。

```
[ec2-user ~]$ nvidia-smi -q | head
```

- 如果您在 G4dn、G5 或 G5g 執行個體上使用 14.x 版或更高版本的 NVIDIA vGPU 軟體，請使用以下命令來停用 GSP。如需有關為何需要執行此程序的詳細資訊，請瀏覽 [NVIDIA 的說明文件](#)。

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /etc/modprobe.d/nvidia.conf
```

- 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

- (選用) 視您的使用案例而定，您可能會完成下列選用步驟。如果您不需要這項功能，請勿完成這些步驟。
  - 為了協助善加利用最高 4K 解析度的四個顯示器，請設定高效能顯示器通訊協定 [NICE DCV](#)。
  - 預設會啟用 NVIDIA Quadro 虛擬工作站。若要針對主控功能的 RDSH 應用程式啟動 GRID 虛擬應用程式，請完成在您的 [Amazon EC2 GPU 執行個體上啟用 NVIDIA 網格虛擬應用程式](#) 中的 GRID 虛擬應用程式啟動步驟。
  - 安裝 GUI 工作站套件。

```
[ec2-user ~]$ sudo dnf groupinstall -y workstation
```

## Rocky Linux 8

### 在 Linux 執行個體上安裝 NVIDIA GRID 驅動程式

- 連線至您的 Linux 執行個體。如果尚未安裝 gcc 和 make，請進行安裝。

- 更新套件快取，並取得執行個體的套件更新。

```
[ec2-user ~]$ sudo yum update -y
```

- 將執行個體重新開機以載入最新的核心版本。

```
[ec2-user ~]$ sudo reboot
```

- 重新開機後，請重新連線至您的執行個體。
- 針對您目前執行的核心版本，安裝 gcc 編譯器與核心標頭套件。

```
[ec2-user ~]$ sudo dnf install -y make gcc elfutils-libelf-devel libglvnd-devel
kernel-devel-$(uname -r)
```

- 使用下列命令下載 GRID 驅動程式安裝公用程式：

```
[ec2-user ~]$ aws s3 cp --recursive s3://ec2-linux-nvidia-drivers/latest/ .
```

此儲存貯體會保留 GRID 驅動程式的多個版本。您可以使用下列命令查看所有可用版本。

```
[ec2-user ~]$ aws s3 ls --recursive s3://ec2-linux-nvidia-drivers/
```

- 請使用以下命令新增許可來執行驅動程式安裝公用程式。

```
[ec2-user ~]$ chmod +x NVIDIA-Linux-x86_64*.run
```

- 如下所述執行自我安裝指令碼，以安裝您已下載的 GRID 驅動程式。例如：

```
[ec2-user ~]$ sudo /bin/sh ./NVIDIA-Linux-x86_64*.run
```

出現提示時，請接受授權合約，並依需求指定安裝選項 (您可以接受預設選項)。

- 確認驅動程式可正常運作。下列命令的回應會列出已安裝的 NVIDIA 驅動程式版本和 GPU 詳細資料。

```
[ec2-user ~]$ nvidia-smi -q | head
```

- 如果您在 G4dn、G5 或 G5g 執行個體上使用 14.x 版或更高版本的 NVIDIA vGPU 軟體，請使用以下命令來停用 GSP。如需有關為何需要執行此程序的詳細資訊，請瀏覽 [NVIDIA 的說明文件](#)。



```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /
etc/modprobe.d/nvidia.conf
```

## 11. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

## 12. (選用) 視您的使用案例而定，您可能會完成下列選用步驟。如果您不需要這項功能，請勿完成這些步驟。

- a. 為了協助善加利用最高 4K 解析度的四個顯示器，請設定高效能顯示器通訊協定 [NICE DCV](#)。
- b. 預設會啟用 NVIDIA Quadro 虛擬工作站。若要針對主控功能的 RDSH 應用程式啟動 GRID 虛擬應用程式，請完成在您的 [Amazon EC2 GPU 執行個體上啟用 NVIDIA 網格虛擬應用程式](#) 中的 GRID 虛擬應用程式啟動步驟。

## Ubuntu 和 Debian

### 在執行個體上安裝 NVIDIA GRID 驅動程式

1. 連線至您的 Linux 執行個體。如果尚未安裝 gcc 和 make，請進行安裝。
2. 更新套件快取，並取得執行個體的套件更新。

```
$ sudo apt-get update -y
```

3. (Ubuntu) 升級 linux-aws 套件以接收最新版本。

```
$ sudo apt-get upgrade -y linux-aws
```

(Debian) 升級套件以接收最新版本。

```
$ sudo apt-get upgrade -y
```

4. 將執行個體重新開機以載入最新的核心版本。

```
$ sudo reboot
```

5. 重新開機後，請重新連線至您的執行個體。
6. 針對您目前執行的核心版本，安裝 gcc 編譯器與核心標頭套件。

```
$ sudo apt-get install -y gcc make linux-headers-$(uname -r)
```

7. 停用 NVIDIA 圖形卡的 nouveau 開放原始碼驅動程式。
  - a. 將 nouveau 新增到 /etc/modprobe.d/blacklist.conf 黑名單檔案。複製下列程式碼區塊並貼至終端機。

```
$ cat << EOF | sudo tee --append /etc/modprobe.d/blacklist.conf
blacklist vga16fb
blacklist nouveau
blacklist rivafb
blacklist nvidiafb
blacklist rivatv
EOF
```

- b. 編輯 /etc/default/grub 檔案並新增下行：

```
GRUB_CMDLINE_LINUX="rdblacklist=nouveau"
```

- c. 重建 Grub 組態。

```
$ sudo update-grub
```

8. 使用下列命令下載 GRID 驅動程式安裝公用程式：

```
$ aws s3 cp --recursive s3://ec2-linux-nvidia-drivers/latest/ .
```

此儲存貯體會保留 GRID 驅動程式的多個版本。您可以使用下列命令查看所有可用版本。

```
$ aws s3 ls --recursive s3://ec2-linux-nvidia-drivers/
```

9. 請使用以下命令新增許可來執行驅動程式安裝公用程式。

```
$ chmod +x NVIDIA-Linux-x86_64*.run
```

10. 如下所述執行自我安裝指令碼，以安裝您已下載的 GRID 驅動程式。例如：

```
$ sudo /bin/sh ./NVIDIA-Linux-x86_64*.run
```

出現提示時，請接受授權合約，並依需求指定安裝選項 (您可以接受預設選項)。

11. 確認驅動程式可正常運作。下列命令的回應會列出已安裝的 NVIDIA 驅動程式版本和 GPU 詳細資料。

```
$ nvidia-smi -q | head
```

12. 如果您在 G4dn、G5 或 G5g 執行個體上使用 14.x 版或更高版本的 NVIDIA vGPU 軟體，請使用以下命令來停用 GSP。如需有關為何需要執行此程序的詳細資訊，請瀏覽 [NVIDIA 的說明文件](#)。

```
$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /etc/modprobe.d/nvidia.conf
```

13. 重新啟動執行個體。

```
$ sudo reboot
```

14. (選用) 視您的使用案例而定，您可能會完成下列選用步驟。如果您不需要這項功能，請勿完成這些步驟。

- a. 為了協助善加利用最高 4K 解析度的四個顯示器，請設定高效能顯示器通訊協定 [NICE DCV](#)。
- b. 預設會啟用 NVIDIA Quadro 虛擬工作站。若要針對主控功能的 RDSH 應用程式啟動 GRID 虛擬應用程式，請完成[在您的 Amazon EC2 GPU 執行個體上啟用 NVIDIA 網格虛擬應用程式](#)中的 GRID 虛擬應用程式啟動步驟。
- c. 安裝 GUI 桌上型電腦/工作站套件。

```
$ sudo apt-get install -y lightdm ubuntu-desktop
```

## 視窗作業系統

### 在 Windows 執行個體上安裝 NVIDIA GRID 驅動程式

1. Connect 至您的 Windows 執行個體並開啟 PowerShell 視窗。

2. 設定 Windows 執行個體 AWS Tools for Windows PowerShell 上的預設認證。如需詳細資訊，請參閱 AWS Tools for Windows PowerShell 使用者指南中 [AWS Tools for Windows PowerShell 入門](#)。

### Important

您的使用者或角色必須擁有包含 AmazonS3 ReadOnly 存取政策的授與權限。如需詳細資訊，請參閱 [AWSAmazon 簡單儲存服務使用者指南中的受管政策：AmazonS3 存 ReadOnly 取](#)。

3. 使用下列 PowerShell 命令，將驅動程式和 [NVIDIA GRID 雲端使用者授權合約](#) 從 Amazon S3 下載至您的桌面。

```
$Bucket = "ec2-windows-nvidia-drivers"
$KeyPrefix = "latest"
$LocalPath = "$home\Desktop\NVIDIA"
$Objects = Get-S3Object -BucketName $Bucket -KeyPrefix $KeyPrefix -Region us-east-1
foreach ($Object in $Objects) {
 $LocalFileName = $Object.Key
 if ($LocalFileName -ne '' -and $Object.Size -ne 0) {
 $LocalFilePath = Join-Path $LocalPath $LocalFileName
 Copy-S3Object -BucketName $Bucket -Key $Object.Key -LocalFile $LocalFilePath -
Region us-east-1
 }
}
```

此儲存貯體會保留 NVIDIA GRID 驅動程式的多個版本。您可以移除 `-KeyPrefix $KeyPrefix` 選項，來下載儲存貯體中的所有可用 Windows 版本。如需作業系統之 NVIDIA GRID 驅動程式版本的詳細資訊，請參閱 NVIDIA 網站上的 [NVIDIA® Virtual GPU \(vGPU\) 軟體文件](#)。

從 GRID 版本 11.0 開始，您可以針對 G3 和 G4dn 執行個體使用 latest 下的驅動程式。我們不會將 11.0 之後的版本新增至 g4/latest，但會將版本 11.0 和 G4dn 專用的舊版保留在 g4/latest 下。

G5 執行個體需要 GRID 13.1 或更新版本 (或 GRID 12.4 或更新版本)。

4. 導覽至桌面並按兩下安裝檔案予以啟動 (選擇與您的執行個體 OS 版本對應的驅動程式版本)。依照指示安裝驅動程式，並視需要將執行個體重新開機。若要確認 GPU 是否正常運作，請檢查裝置管理員。

5. (選用) 使用下列命令在控制面板中停用授權頁面，以防止使用者意外變更產品類型 (NVIDIA GRID 虛擬工作站預設為啟用)。如需詳細資訊，請參閱 [《GRID 授權使用者指南》](#)。

## PowerShell

執行下列 PowerShell 命令以建立登錄值，以停用控制台中的授權頁面。AWS Windows AWS Tools for PowerShell 中的 AMI 預設為 32 位元版本，且此命令會失敗。請改用作業系統 PowerShell 隨附的 64 位元版本。

```
New-Item -Path "HKLM:\SOFTWARE\NVIDIA Corporation\Global" -Name GridLicensing
New-ItemProperty -Path "HKLM:\SOFTWARE\NVIDIA Corporation\Global\GridLicensing" -
Name "NvCplDisableManageLicensePage" -PropertyType "DWord" -Value "1"
```

## 命令提示

執行下列登錄命令以建立登錄值來停用控制面板中的授權頁面。您可以使用 [命令提示字元] 視窗或 64 位元版本的執行它 PowerShell。

```
reg add "HKLM\SOFTWARE\NVIDIA Corporation\Global\GridLicensing" /v
NvCplDisableManageLicensePage /t REG_DWORD /d 1
```

6. (選用) 視您的使用案例而定，您可能會完成下列選用步驟。如果您不需要這項功能，請勿完成這些步驟。
  - a. 為了協助善加利用最高 4K 解析度的四個顯示器，請設定高效能顯示器通訊協定 [NICE DCV](#)。
  - b. 預設會啟用 NVIDIA Quadro 虛擬工作站。若要針對主控功能的 RDSH 應用程式啟動 GRID 虛擬應用程式，請完成在您的 [Amazon EC2 GPU 執行個體上啟用 NVIDIA 網格虛擬應用程式](#) 中的 GRID 虛擬應用程式啟動步驟。

## 選項 4：NVIDIA 遊戲驅動程式 (G5 和 G4dn 執行個體)

這些驅動程式僅供 AWS 客戶使用。通過下載它們，您同意使用下載的軟件僅用於開發與 NVIDIA A10G 和 NVIDIA 特斯拉 T4 硬件一起使用的 AMI。安裝軟體時，您必須遵循 [NVIDIA GRID 雲端最終使用者授權合約](#) 的條款。

## 考量事項

- G3 執行個體需要 AWS 提供 DNS 解析，GRID 授權才能運作。
- 僅 495.x 版或更高版本的 NVIDIA 驅動程式支援 [IMDSv2](#)。

## Amazon Linux 和 Amazon Linux 2

### 在執行個體上安裝 NVIDIA 遊戲驅動程式

1. 連線至您的 Linux 執行個體。
2. AWS CLI 在您的 Linux 執行個體上安裝並設定預設認證。如需詳細資訊，請參閱《AWS Command Line Interface 使用者指南》中的[安裝 AWS CLI](#)。

#### Important

您的使用者或角色必須擁有包含 AmazonS3 ReadOnly 存取政策的授與權限。如需詳細資訊，請參閱 [AWSAmazon 簡單儲存服務使用者指南中的受管政策：AmazonS3 存 ReadOnly 取](#)。

3. 如果尚未安裝 gcc 和 make，請進行安裝。

```
[ec2-user ~]$ sudo yum install gcc make
```

4. 更新套件快取，並取得執行個體的套件更新。

```
[ec2-user ~]$ sudo yum update -y
```

5. 將執行個體重新開機以載入最新的核心版本。

```
[ec2-user ~]$ sudo reboot
```

6. 重新開機後，請重新連線至您的執行個體。
7. 針對您目前執行的核心版本，安裝 gcc 編譯器與核心標頭套件。

```
[ec2-user ~]$ sudo yum install -y gcc kernel-devel-$(uname -r)
```

8. 使用下列命令下載遊戲驅動程式安裝公用程式：

```
[ec2-user ~]$ aws s3 cp --recursive s3://nvidia-gaming/linux/latest/ .
```

此儲存貯體會保留遊戲驅動程式的多個版本。您可以使用下列命令查看所有可用版本：

```
[ec2-user ~]$ aws s3 ls --recursive s3://nvidia-gaming/linux/
```

9. 從下載的 .zip 封存中擷取遊戲驅動程序安裝公用程式。

```
[ec2-user ~]$ unzip latest-driver-name.zip -d nvidia-drivers
```

10. 請使用以下命令新增許可來執行驅動程式安裝公用程式：

```
[ec2-user ~]$ chmod +x nvidia-drivers/NVIDIA-Linux-x86_64*-grid.run
```

11. 使用以下命令執行安裝程式：

```
[ec2-user ~]$ sudo ./nvidia-drivers/NVIDIA-Linux-x86_64*.run
```

### Note

如果您是使用 Amazon Linux 2 搭配核心版本 5.10，請使用下列命令來安裝 NVIDIA 遊戲驅動程式。

```
[ec2-user ~]$ sudo CC=/usr/bin/gcc10-cc ./NVIDIA-Linux-x86_64*.run
```

出現提示時，請接受授權合約，並依需求指定安裝選項 (您可以接受預設選項)。

12. 使用以下命令建立所需的組態檔案。

```
[ec2-user ~]$ cat << EOF | sudo tee -a /etc/nvidia/gridd.conf
vGamingMarketplace=2
EOF
```

13. 使用以下命令下載認證檔案及重新命名。

- 對於 460.39 版本或更新版本：

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2023_9_22.cert"
```

- 對於 440.68 到 445.48 版本：

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2020_04.cert"
```

- 對於較早版本

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2019_09.cert"
```

14. 如果您在 G4dn、G5 或 G5g 執行個體上使用 510.x 版或更高版本的 NVIDIA 驅動程式，請使用以下命令來停用 GSP。如需有關為何需要執行此程序的詳細資訊，請瀏覽 [NVIDIA 的說明文件](#)。

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /etc/modprobe.d/nvidia.conf
```

15. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

16. (選用) 為協助善加利用最高 4K 解析度的單一顯示器，請設定高效能顯示器通訊協定 [NICE DCV](#)。

## CentOS 7 和 Red Hat Enterprise Linux 7

### 在執行個體上安裝 NVIDIA 遊戲驅動程式

1. 連線至您的 Linux 執行個體。如果尚未安裝 gcc 和 make，請進行安裝。
2. 更新套件快取，並取得執行個體的套件更新。

```
[ec2-user ~]$ sudo yum update -y
```

3. 將執行個體重新開機以載入最新的核心版本。

```
[ec2-user ~]$ sudo reboot
```

4. 重新開機後，請重新連線至您的執行個體。
5. 針對您目前執行的核心版本，安裝 gcc 編譯器與核心標頭套件。

```
[ec2-user ~]$ sudo yum install -y unzip gcc kernel-devel-$(uname -r)
```

6. 停用 NVIDIA 圖形卡的 nouveau 開放原始碼驅動程式。
  - a. 將 nouveau 新增到 /etc/modprobe.d/blacklist.conf 黑名單檔案。複製下列程式碼區塊並貼至終端機。



```
[ec2-user ~]$ cat << EOF | sudo tee --append /etc/modprobe.d/blacklist.conf
blacklist vga16fb
blacklist nouveau
blacklist rivafb
blacklist nvidiafb
blacklist rivatv
EOF
```

- b. 編輯 `/etc/default/grub` 檔案並新增下行：

```
GRUB_CMDLINE_LINUX="rdblacklist=nouveau"
```

- c. 重建 Grub 組態。

```
[ec2-user ~]$ sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

7. 使用下列命令下載遊戲驅動程式安裝公用程式：

```
[ec2-user ~]$ aws s3 cp --recursive s3://nvidia-gaming/linux/latest/ .
```

此儲存貯體會保留遊戲驅動程式的多個版本。您可以使用下列命令查看所有可用版本：

```
[ec2-user ~]$ aws s3 ls --recursive s3://nvidia-gaming/linux/
```

8. 從下載的 `.zip` 封存中擷取遊戲驅動程序安裝公用程式。

```
[ec2-user ~]$ unzip vGPU-SW-*vGaming-Linux-Guest-Drivers.zip -d nvidia-drivers
```

9. 請使用以下命令新增許可來執行驅動程式安裝公用程式：

```
[ec2-user ~]$ chmod +x nvidia-drivers/Linux/NVIDIA-Linux-x86_64*-grid.run
```

10. 使用以下命令執行安裝程式：

```
[ec2-user ~]$ sudo ./nvidia-drivers/Linux/NVIDIA-Linux-x86_64*.run
```

出現提示時，請接受授權合約，並依需求指定安裝選項 (您可以接受預設選項)。

11. 使用以下命令建立所需的組態檔案。

```
[ec2-user ~]$ cat << EOF | sudo tee -a /etc/nvidia/gridd.conf
vGamingMarketplace=2
EOF
```

12. 使用以下命令下載認證檔案及重新命名。

- 對於 460.39 版本或更新版本：

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2023_9_22.cert"
```

- 對於 440.68 到 445.48 版本：

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2020_04.cert"
```

- 對於較早版本

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2019_09.cert"
```

13. 如果您在 G4dn、G5 或 G5g 執行個體上使用 510.x 版或更高版本的 NVIDIA 驅動程式，請使用以下命令來停用 GSP。如需有關為何需要執行此程序的詳細資訊，請瀏覽 [NVIDIA 的說明文件](#)。

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /etc/modprobe.d/nvidia.conf
```

14. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

15. (選用) 為協助善加利用最高 4K 解析度的單一顯示器，請設定高效能顯示器通訊協定 [NICE DCV](#)。如果您不需要此功能，請勿完成此步驟。

## CentOS Stream 8 和 Red Hat Enterprise Linux 8

### 在執行個體上安裝 NVIDIA 遊戲驅動程式

1. 連線至您的 Linux 執行個體。如果尚未安裝 gcc 和 make，請進行安裝。
2. 更新套件快取，並取得執行個體的套件更新。

```
[ec2-user ~]$ sudo yum update -y
```

3. 將執行個體重新開機以載入最新的核心版本。

```
[ec2-user ~]$ sudo reboot
```

4. 重新開機後，請重新連線至您的執行個體。
5. 針對您目前執行的核心版本，安裝 gcc 編譯器與核心標頭套件。

```
[ec2-user ~]$ sudo yum install -y unzip gcc kernel-devel-$(uname -r)
```

6. 使用下列命令下載遊戲驅動程式安裝公用程式：

```
[ec2-user ~]$ aws s3 cp --recursive s3://nvidia-gaming/linux/latest/ .
```

此儲存貯體會保留遊戲驅動程式的多個版本。您可以使用下列命令查看所有可用版本：

```
[ec2-user ~]$ aws s3 ls --recursive s3://nvidia-gaming/linux/
```

7. 從下載的 .zip 封存中擷取遊戲驅動程序安裝公用程式。

```
[ec2-user ~]$ unzip vGPUSW-*vGaming-Linux-Guest-Drivers.zip -d nvidia-drivers
```

8. 請使用以下命令新增許可來執行驅動程式安裝公用程式：

```
[ec2-user ~]$ chmod +x nvidia-drivers/Linux/NVIDIA-Linux-x86_64*-grid.run
```

9. 使用以下命令執行安裝程式：

```
[ec2-user ~]$ sudo ./nvidia-drivers/Linux/NVIDIA-Linux-x86_64*.run
```

出現提示時，請接受授權合約，並依需求指定安裝選項 (您可以接受預設選項)。

10. 使用以下命令建立所需的組態檔案。

```
[ec2-user ~]$ cat << EOF | sudo tee -a /etc/nvidia/gridd.conf
vGamingMarketplace=2
EOF
```

11. 使用以下命令下載認證檔案及重新命名。

- 對於 460.39 版本或更新版本：

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2023_9_22.cert"
```

- 對於 440.68 到 445.48 版本：

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2020_04.cert"
```

- 對於較早版本

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2019_09.cert"
```

12. 如果您在 G4dn、G5 或 G5g 執行個體上使用 510.x 版或更高版本的 NVIDIA 驅動程式，請使用以下命令來停用 GSP。如需有關為何需要執行此程序的詳細資訊，請瀏覽 [NVIDIA 的說明文件](#)。

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /etc/modprobe.d/nvidia.conf
```

13. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

14. (選用) 為協助善加利用最高 4K 解析度的單一顯示器，請設定高效能顯示器通訊協定 [NICE DCV](#)。

## Rocky Linux 8

在執行個體上安裝 NVIDIA 遊戲驅動程式

1. 連線至您的 Linux 執行個體。如果尚未安裝 gcc 和 make，請進行安裝。

- 更新套件快取，並取得執行個體的套件更新。

```
[ec2-user ~]$ sudo yum update -y
```

- 將執行個體重新開機以載入最新的核心版本。

```
[ec2-user ~]$ sudo reboot
```

- 重新開機後，請重新連線至您的執行個體。
- 針對您目前執行的核心版本，安裝 gcc 編譯器與核心標頭套件。

```
[ec2-user ~]$ sudo dnf install -y unzip gcc make elfutils-libelf-devel libglvnd-devel kernel-devel-$(uname -r)
```

- 使用下列命令下載遊戲驅動程式安裝公用程式：

```
[ec2-user ~]$ aws s3 cp --recursive s3://nvidia-gaming/linux/latest/ .
```

此儲存貯體會保留遊戲驅動程式的多個版本。您可以使用下列命令查看所有可用版本：

```
[ec2-user ~]$ aws s3 ls --recursive s3://nvidia-gaming/linux/
```

- 從下載的 .zip 封存中擷取遊戲驅動程序安裝公用程式。

```
[ec2-user ~]$ unzip vGPUSW-*vGaming-Linux-Guest-Drivers.zip -d nvidia-drivers
```

- 請使用以下命令新增許可來執行驅動程式安裝公用程式：

```
[ec2-user ~]$ chmod +x nvidia-drivers/Linux/NVIDIA-Linux-x86_64*-grid.run
```

- 使用以下命令執行安裝程式：

```
[ec2-user ~]$ sudo ./nvidia-drivers/Linux/NVIDIA-Linux-x86_64*.run
```

出現提示時，請接受授權合約，並依需求指定安裝選項 (您可以接受預設選項)。

- 使用以下命令建立所需的組態檔案。

```
[ec2-user ~]$ cat << EOF | sudo tee -a /etc/nvidia/gridd.conf
vGamingMarketplace=2
```

## EOF

11. 使用以下命令下載認證檔案及重新命名。

- 對於 460.39 版本或更新版本：

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2023_9_22.cert"
```

- 對於 440.68 到 445.48 版本：

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2020_04.cert"
```

- 對於較早版本

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2019_09.cert"
```

12. 如果您在 G4dn、G5 或 G5g 執行個體上使用 510.x 版或更高版本的 NVIDIA 驅動程式，請使用以下命令來停用 GSP。如需有關為何需要執行此程序的詳細資訊，請瀏覽 [NVIDIA 的說明文件](#)。

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /etc/modprobe.d/nvidia.conf
```

13. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

14. (選用) 為協助善加利用最高 4K 解析度的單一顯示器，請設定高效能顯示器通訊協定 [NICE DCV](#)。

## Ubuntu 和 Debian

在執行個體上安裝 NVIDIA 遊戲驅動程式

- 連線至您的 Linux 執行個體。如果尚未安裝 gcc 和 make，請進行安裝。
- 更新套件快取，並取得執行個體的套件更新。

```
$ sudo apt-get update -y
```

3. 升級 linux-aws 套件以接收最新版本。

```
$ sudo apt-get upgrade -y linux-aws
```

4. 將執行個體重新開機以載入最新的核心版本。

```
$ sudo reboot
```

5. 重新開機後，請重新連線至您的執行個體。
6. 針對您目前執行的核心版本，安裝 gcc 編譯器與核心標頭套件。

```
$ sudo apt-get install -y unzip gcc make linux-headers-$(uname -r)
```

7. 停用 NVIDIA 圖形卡的 nouveau 開放原始碼驅動程式。
  - a. 將 nouveau 新增到 /etc/modprobe.d/blacklist.conf 黑名單檔案。複製下列程式碼區塊並貼至終端機。

```
$ cat << EOF | sudo tee --append /etc/modprobe.d/blacklist.conf
blacklist vga16fb
blacklist nouveau
blacklist rivafb
blacklist nvidiafb
blacklist rivatv
EOF
```

- b. 編輯 /etc/default/grub 檔案並新增下行：

```
GRUB_CMDLINE_LINUX="rdblacklist=nouveau"
```

- c. 重建 Grub 組態。

```
$ sudo update-grub
```

8. 使用下列命令下載遊戲驅動程式安裝公用程式：

```
$ aws s3 cp --recursive s3://nvidia-gaming/linux/latest/ .
```

此儲存貯體會保留遊戲驅動程式的多個版本。您可以使用下列命令查看所有可用版本：

```
$ aws s3 ls --recursive s3://nvidia-gaming/linux/
```

9. 從下載的 .zip 封存中擷取遊戲驅動程序安裝公用程式。

```
$ unzip vGPUSW-*vGaming-Linux-Guest-Drivers.zip -d nvidia-drivers
```

10. 請使用以下命令新增許可來執行驅動程式安裝公用程式：

```
$ chmod +x nvidia-drivers/Linux/NVIDIA-Linux-x86_64*-grid.run
```

11. 使用以下命令執行安裝程式：

```
$ sudo ./nvidia-drivers/Linux/NVIDIA-Linux-x86_64*.run
```

出現提示時，請接受授權合約，並依需求指定安裝選項 (您可以接受預設選項)。

12. 使用以下命令建立所需的組態檔案。

```
$ cat << EOF | sudo tee -a /etc/nvidia/gridd.conf
vGamingMarketplace=2
EOF
```

13. 使用以下命令下載認證檔案及重新命名。

- 對於 460.39 版本或更新版本：

```
$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2023_9_22.cert"
```

- 對於 440.68 到 445.48 版本：

```
$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2020_04.cert"
```

- 對於較早版本

```
$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2019_09.cert"
```



14. 如果您在 G4dn、G5 或 G5g 執行個體上使用 510.x 版或更高版本的 NVIDIA 驅動程式，請使用以下命令來停用 GSP。如需有關為何需要執行此程序的詳細資訊，請瀏覽 [NVIDIA 的說明文件](#)。

```
$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /etc/modprobe.d/nvidia.conf
```

15. 重新啟動執行個體。

```
$ sudo reboot
```

16. (選用) 為協助善加利用最高 4K 解析度的單一顯示器，請設定高效能顯示器通訊協定 [NICE DCV](#)。如果您不需要此功能，請勿完成此步驟。

## 視窗作業系統

在執行個體上安裝 NVIDIA 遊戲驅動程式之前，除了針對所有遊戲驅動程式提到的考量之外，您還必須確定符合下列先決條件。

- 如果您使用自訂的 Windows AMI 啟動 Windows 執行個體，則 AMI 必須是使用 Windows Sysprep 建立的標準化映像檔，以確保遊戲驅動程式正常運作。如需詳細資訊，請參閱 [使用視窗系統建立 AMI](#)。
- 設定 Windows 執行個體 AWS Tools for Windows PowerShell 上的預設認證。如需詳細資訊，請參閱 AWS Tools for Windows PowerShell 使用者指南中 [AWS Tools for Windows PowerShell 入門](#)。
- 您的使用者或角色必須擁有包含 AmazonS3 ReadOnly 存取政策的授與權限。如需詳細資訊，請參閱 [AWS Amazon 簡單儲存服務使用者指南中的受管政策：AmazonS3 存 ReadOnly 取](#)。

在 Windows 執行個體上安裝 NVIDIA 遊戲驅動程式

1. Connect 至您的 Windows 執行個體並開啟 PowerShell 視窗。
2. 使用下列 PowerShell 指令下載並安裝遊戲驅動程式。

```
$Bucket = "nvidia-gaming"
$KeyPrefix = "windows/latest"
$LocalPath = "$home\Desktop\NVIDIA"
$Objects = Get-S3Object -BucketName $Bucket -KeyPrefix $KeyPrefix -Region us-east-1
foreach ($Object in $Objects) {
```

```
$LocalFileName = $Object.Key
if ($LocalFileName -ne '' -and $Object.Size -ne 0) {
 $LocalFilePath = Join-Path $LocalPath $LocalFileName
 Copy-S3Object -BucketName $Bucket -Key $Object.Key -LocalFile $LocalFilePath -
 Region us-east-1
}
}
```

此 S3 儲存貯體會保留 NVIDIA GRID 驅動程式的多個版本。如果將 \$KeyPrefix 變數的值從 "windows/latest" 變更為 "windows"，則可以下載儲存貯體中的所有可用版本。

3. 導覽至桌面並按兩下安裝檔案予以啟動 (選擇與您的執行個體 OS 版本對應的驅動程式版本)。依照指示安裝驅動程式，並視需要將執行個體重新開機。若要確認 GPU 是否正常運作，請檢查裝置管理員。
4. 使用下列其中一種方法來註冊驅動程式。

Version 527.27 or above

使用 64 位元版本或 [命令提示字元] 視窗建立下列登錄機碼。 PowerShell

```
key : HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\nvlddmkm
\Global
```

名稱 : vGamingMarketplace

type (類型) : DWord

value (值) : 2

PowerShell

執行下列 PowerShell 命令以建立此登錄值。AWS Windows AWS Tools for PowerShell 中的 AMI 預設為 32 位元版本，且此命令會失敗。請改用作業系統 PowerShell 隨附的 64 位元版本。

```
New-ItemProperty -Path "HKLM:\SYSTEM\CurrentControlSet\Services\nvlddmkm\Global"
-Name "vGamingMarketplace" -PropertyType "DWord" -Value "2"
```

命令提示

執行下列登錄命令來建立此登錄值。您可以使用 [命令提示字元] 視窗或 64 位元版本的執行它 PowerShell。

```
reg add "HKLM\SYSTEM\CurrentControlSet\Services\nvlddmkm\Global" /v
vGamingMarketplace /t REG_DWORD /d 2
```

### Earlier versions

使用 64 位元版本或 [命令提示字元] 視窗建立下列登錄機碼。 PowerShell

key : HKEY\_LOCAL\_MACHINE\SOFTWARE\NVIDIA Corporation\Global

名稱 : vGamingMarketplace

type (類型) : DWord

value (值) : 2

### PowerShell

執行下列 PowerShell 命令以建立此登錄值。 AWS Windows AWS Tools for PowerShell 中的 AMI 預設為 32 位元版本，且此命令會失敗。請改用作業系統 PowerShell 隨附的 64 位元版本。

```
New-ItemProperty -Path "HKLM:\SOFTWARE\NVIDIA Corporation\Global" -Name
"vGamingMarketplace" -PropertyType "DWord" -Value "2"
```

### 命令提示

執行下列登錄命令，使用「命令提示字元」視窗來建立此登錄機碼。您也可以可以在 64 位元版本的中使用此命令 PowerShell。

```
reg add "HKLM\SOFTWARE\NVIDIA Corporation\Global" /v vGamingMarketplace /t
REG_DWORD /d 2
```

5. 在中執行下列命令 PowerShell。這會下載認證檔案、重新命名檔案 GridSwCert.txt，並將檔案移至系統磁碟機上的 Public Documents (公用文件) 資料夾。通常，資料夾路徑為 C:\Users\Public\Documents。

- 對於 461.40 版本或更新版本：

```
Invoke-WebRequest -Uri "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertWindows_2023_9_22.cert" -OutFile "$Env:PUBLIC\Documents\GridSwCert.txt"
```

- 對於 445.87 版本：

```
Invoke-WebRequest -Uri "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Windows_2020_04.cert" -OutFile "$Env:PUBLIC\Documents\GridSwCert.txt"
```

- 對於較早版本

```
Invoke-WebRequest -Uri "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Windows_2019_09.cert" -OutFile "$Env:PUBLIC\Documents\GridSwCert.txt"
```

### Note

如果您在下載檔案時收到錯誤訊息，而且您使用的是 Windows 伺服器 2016 或更早版本，則可能需要為您的 PowerShell 終端機啟用 TLS 1.2。您可以使用下列命令為目前 PowerShell 工作階段啟用 TLS 1.2，然後再試一次：

```
[Net.ServicePointManager]::SecurityProtocol =
[Net.SecurityProtocolType]::Tls12
```

6. 將執行個體重新開機。
7. 使用下列命令以確認 NVIDIA 遊戲授權。

```
C:\Windows\System32\DriverStore\FileRepository\nv_dispswi.inf_*\nvidia-smi.exe -q
```

輸出格式應類似以下內容。

```
vGPU Software Licensed Product
Product Name : NVIDIA Cloud Gaming
License Status : Licensed (Expiry: N/A)
```

8. (選用) 為協助善加利用最高 4K 解析度的單一顯示器，請設定高效能顯示器通訊協定 [NICE DCV](#)。如果您不需要此功能，請勿完成此步驟。

## 安裝 CUDA 附加版本

在執行個體上安裝 NVIDIA 圖形驅動程序後，您可以安裝與圖形驅動程序隨附版本以外的 CUDA 版本。下列程序會示範如何在執行個體上設定多個版本的 CUDA。

### 在 Linux 上安裝 CUDA 工具包

請依照下列步驟在 Linux 上安裝 CUDA 工具組：

1. 連線至您的 Linux 執行個體。
2. 開啟 [NVIDIA 網站](#) 並選取您需要的 CUDA 版本。
3. 選取執行個體上作業系統的架構、發行版本和版本。針對安裝程式類型，選取 runfile (local)。
4. 依照指示下載安裝指令碼。
5. 將執行權限新增至您使用下列命令下載的安裝指令碼。

```
[ec2-user ~]$ chmod +x downloaded_installer_file
```

6. 執行安裝指令碼，如下所示安裝 CUDA 工具套件，並將 CUDA 版本編號新增到工具套件路徑。

```
[ec2-user ~]$ sudo sh downloaded_installer_file --silent --override --toolkit --samples --toolkitpath=/usr/local/cuda-version --samplespath=/usr/local/cuda --no-opengl-libs
```

7. (選用) 設定預設 CUDA 版本，如下所示。

```
[ec2-user ~]$ sudo ln -s /usr/local/cuda-version /usr/local/cuda
```

### 在視窗上安裝 CUDA 工具包

請依照下列步驟在視窗上安裝 CUDA 工具組：

#### 若要安裝 CUDA 工具套件

1. 連接至 Windows 執行個體。
2. 開啟 [NVIDIA 網站](#) 並選取您需要的 CUDA 版本。
3. 在 Installer Type (安裝程式類型) 中，選取 exe (local)，然後選取 Download (下載)。
4. 使用瀏覽器執行下載的安裝檔案。依指示安裝 CUDA 工具套件。您可能需要重新啟動執行個體。

## 在您的 Amazon EC2 實例上安裝 AMD 驅動程序

已連接 AMD GPU 的執行個體 (例如 G4ad 執行個體) 必須安裝適當的 AMD 驅動程式。根據您的需求，您可以使用預先安裝驅動程式的 AMI，或從 Amazon S3 中下載驅動程式。

若要在已連接 NVIDIA GPU 的執行個體 (例如 G4dn 執行個體) 上安裝 NVIDIA 驅動程式，請參閱 [安裝 NVIDIA 驅動程式](#)。

### 內容

- [企業驅動程式的 AMD Radeon Pro 軟體](#)
- [安裝 AMD 驅動程式的 AMI](#)
- [AMD 驅動程式下載](#)
- [設定適用於 Linux 的互動式桌面](#)

### 企業驅動程式的 AMD Radeon Pro 軟體

企業驅動程式的 AMD Radeon Pro 軟體專為提供專業級圖形使用案例支援而打造。使用驅動程式，您可以為每個 GPU 設定兩個 4K 顯示執行個體。

### 支援的 API

- OpenGL、OpenCL
- Vulkan
- AMD 進階媒體架構
- 影像加速 API
- DirectX 9 和更新版本
- Microsoft 硬體媒體基礎轉換


### 安裝 AMD 驅動程式的 AMI

AWS 提供了安裝 AMD 驅動程序的不同 Amazon 機器映像 (AMI)。開啟 [具有 AMD 驅動程式的 Marketplace 產品](#)。

### AMD 驅動程式下載

如果使用的 AMI 沒有安裝 AMD 驅動程式，您可以下載 AMD 驅動程式並將其安裝在您的執行個體上。只有下列作業系統版本支援 AMD 驅動程式：

- Amazon Linux 2 與內核版本 4.14

 Note


AMD 驅動程式版本 amdgpu-pro-20.20-1184451 和更新的驅動程式版本需要核心版本 5.15 或更高版本。

- Windows Server 2016
- Windows Server 2019

這些下載僅供 AWS 客戶使用。下載即表示您同意下載的軟體，只會用來開發 AMIs，以搭配 AMD Radeon Pro V520 硬體使用，而不做其他用途。安裝軟體時，您必須遵循 [AMD Software 雲端最終使用者授權合約](#) 的條款。

在您的執行個體上安裝 AMD 驅動程式

1. 連線至您的 Linux 執行個體。
2. AWS CLI 在您的 Linux 執行個體上安裝並設定預設認證。如需詳細資訊，請參閱《AWS Command Line Interface 使用者指南》中的 [安裝 AWS CLI](#)。

 Important

您的使用者或角色必須擁有包含 AmazonS3 ReadOnly 存取政策的授與權限。如需詳細資訊，請參閱 [AWSAmazon 簡單儲存服務使用者指南中的受管政策：AmazonS3 存 ReadOnly 取](#)。

3. 如果尚未安裝 gcc 和 make，請進行安裝。

```
$ sudo yum install gcc make
```

4. 更新套件快取，並取得執行個體的套件更新。

- 針對 Amazon Linux 2：

```
$ sudo amazon-linux-extras install epel -y
$ sudo yum update -y
```

- 對於

```
$ wget https://repo.radeon.com/.preview/a0e4ef1dffbc95b4abb54e891f265e61/amdgpu-
install/5.5.02.05.2/ubuntu/jammy/amdgpu-install_5.5.02.05.50502-1_all.deb
$ sudo apt install ./amdgpu-install_5.5.02.05.50502-1_all.deb
$ sudo sed -i 's#repo.radeon.com#&/.preview/a0e4ef1dffbc95b4abb54e891f265e61#' /
etc/apt/sources.list.d/{amdgpu.list,rocm.list,amdgpu-proprietary.list}
```

- 對於其他版本：

```
$ sudo dpkg --add-architecture i386
$ sudo apt-get update -y && sudo apt upgrade -y
```

- CentOS 系統：

```
$ sudo yum install epel-release -y
$ sudo yum update -y
```

5. 重新啟動執行個體。

```
$ sudo reboot
```

6. 重新啟動後重新連線至執行個體。
7. 下載最新的 AMD 驅動程式。

#### Note

略過此步驟

```
$ aws s3 cp --recursive s3://ec2-amd-linux-drivers/latest/ .
```

8. 解壓縮檔案。

- Amazon Linux 2 和 CentOS：

```
$ tar -xf amdgpu-pro-*rhel*.tar.xz
```

- Ubuntu 系統：



**Note**

略過此步驟

```
$ tar -xf amdgpu-pro*ubuntu*.xz
```

- 變更為解壓縮驅動程式的資料夾。
- 新增缺少的用於驅動程式安裝的模組。

- Amazon Linux 2 和 CentOS :

跳過此步驟。

- Ubuntu 系統 :

**Note**

略過此步驟

```
$ sudo apt install linux-modules-extra-$(uname -r) -y
```

- 執行自主安裝指令碼以安裝完整的圖形堆疊。

- 對於

```
$ sudo amdgpu-install --usecase=workstation --vulkan=pro --opencl=rocr,legacy -y
```

- 對於 Amazon Linux 2 和 CentOS 和其他版本 :

```
$./amdgpu-pro-install -y --opencl=pal,legacy
```

- 重新啟動執行個體。

```
$ sudo reboot
```

- 確認驅動程式可正常運作。

```
$ dmesg | grep amdgpu
```

回應如下所示：

```
Initialized amdgpu
```

在您的執行個體上安裝 AMD 驅動程式

1. Connect 至您的 Windows 執行個體並開啟 PowerShell 視窗。
2. 設定 Windows 執行個體 AWS Tools for Windows PowerShell 上的預設認證。如需詳細資訊，請參閱 AWS Tools for Windows PowerShell 使用者指南中 [AWS Tools for Windows PowerShell 入門](#)。

#### Important

您的使用者或角色必須擁有包含 AmazonS3 ReadOnly 存取政策的授與權限。如需詳細資訊，請參閱 [AWSAmazon 簡單儲存服務使用者指南中的受管政策：AmazonS3 存 ReadOnly 取](#)。

3. 使用以下 PowerShell 命令將驅動程式從 Amazon S3 下載到您的桌面。

```
$Bucket = "ec2-amd-windows-drivers"
$KeyPrefix = "latest" # use "archives" for Windows Server 2016
$LocalPath = "$home\Desktop\AMD"
$Objects = Get-S3Object -BucketName $Bucket -KeyPrefix $KeyPrefix -Region us-east-1
foreach ($Object in $Objects) {
 $LocalFileName = $Object.Key
 if ($LocalFileName -ne '' -and $Object.Size -ne 0) {
 $LocalFilePath = Join-Path $LocalPath $LocalFileName
 Copy-S3Object -BucketName $Bucket -Key $Object.Key -LocalFile $LocalFilePath -
 Region us-east-1
 }
}
```

4. 解壓縮下載的驅動程式檔案，並使用下列 PowerShell 指令執行安裝程式。

```
Expand-Archive $LocalFilePath -DestinationPath "$home\Desktop\AMD\$KeyPrefix" -
Verbose
```

現在，請檢查新目錄的內容。可以使用 `Get-ChildItem PowerShell` 指令擷取目錄名稱。

```
Get-ChildItem "$home\Desktop\AMD\$KeyPrefix"
```

輸出格式應類似以下內容：

```
Directory: C:\Users\Administrator\Desktop\AMD\latest

Mode LastWriteTime Length Name
---- -
d----- 10/13/2021 12:52 AM 210414a-365562C-Retail_End_User.2
```

安裝驅動程式：

```
pnputil /add-driver $home\Desktop\AMD\$KeyPrefix*.inf /install /subdirs
```

5. 依照指示安裝驅動程式，並視需要將執行個體重新開機。
6. 若要確認 GPU 是否正常運作，請檢查裝置管理員。您應該會看到列為顯示轉接器的「AMD Radeon Pro V520 MxGPU」。
7. 為了協助善加利用最高 4K 解析度的四個顯示器，請設定高效能顯示器通訊協定 [NICE DCV](#)。

## 設定適用於 Linux 的互動式桌面

在您確認 Linux 執行個體已安裝 AMD GPU 驅動程式且使用 `amdgpu` 之後，您就可以安裝互動式桌面管理員。為了獲得最佳的相容性和效能，我們建議您使用 MATE 桌面環境。

先決條件

開啟文字編輯器，並將以下內容儲存為名為 `xorg.conf` 的檔案。你的執行個體需要這個檔案。

```
Section "ServerLayout"
Identifier "Layout0"
Screen 0 "Screen0"
InputDevice "Keyboard0" "CoreKeyboard"
InputDevice "Mouse0" "CorePointer"
EndSection
Section "Files"
ModulePath "/opt/amdgpu/lib64/xorg/modules/drivers"
ModulePath "/opt/amdgpu/lib/xorg/modules"
```

```
ModulePath "/opt/amdgpu-pro/lib/xorg/modules/extensions"
ModulePath "/opt/amdgpu-pro/lib64/xorg/modules/extensions"
ModulePath "/usr/lib64/xorg/modules"
ModulePath "/usr/lib/xorg/modules"
EndSection
Section "InputDevice"
generated from default
Identifier "Mouse0"
Driver "mouse"
Option "Protocol" "auto"
Option "Device" "/dev/psaux"
Option "Emulate3Buttons" "no"
Option "ZAxisMapping" "4 5"
EndSection
Section "InputDevice"
generated from default
Identifier "Keyboard0"
Driver "kbd"
EndSection
Section "Monitor"
Identifier "Monitor0"
VendorName "Unknown"
ModelName "Unknown"
EndSection
Section "Device"
Identifier "Device0"
Driver "amdgpu"
VendorName "AMD"
BoardName "Radeon MxGPU V520"
BusID "PCI:0:30:0"
EndSection
Section "Extensions"
Option "DPMS" "Disable"
EndSection
Section "Screen"
Identifier "Screen0"
Device "Device0"
Monitor "Monitor0"
DefaultDepth 24
Option "AllowEmptyInitialConfiguration" "True"
SubSection "Display"
 Virtual 3840 2160
 Depth 32
EndSubSection
```

```
EndSection
```

若要在 Amazon Linux 2 上設定互動式桌面

1. 安裝 EPEL 儲存庫。

```
$ C:\> sudo amazon-linux-extras install epel -y
```

2. 安裝 MATE 桌面。

```
$ C:\> sudo amazon-linux-extras install mate-desktop1.x -y
$ C:\> sudo yum groupinstall "MATE Desktop" -y
$ C:\> sudo systemctl disable firewalld
```

3. 將 xorg.conf 檔案複製至 /etc/X11/xorg.conf。
4. 重新啟動執行個體。

```
$ C:\> sudo reboot
```

5. (選用) [安裝 NICE DCV 伺服器](#)，使用 NICE DCV 做為高效能顯示通訊協定，然後使用偏好的用戶端[連線至 NICE DCV 工作階段](#)。

在 Ubuntu 上設定互動式桌面

1. 安裝 MATE 桌面。

```
$ sudo apt install xorg-dev ubuntu-mate-desktop -y
$ C:\> sudo apt purge ifupdown -y
```

2. 將 xorg.conf 檔案複製至 /etc/X11/xorg.conf。
3. 重新啟動執行個體。

```
$ sudo reboot
```

4. 為適當版本的 Ubuntu 安裝 AMF 編碼器。

```
$ sudo apt install ./amdgpu-pro-20.20-*/amf-amdgpu-pro_20.20-*_amd64.deb
```

5. (選用) [安裝 NICE DCV 伺服器](#)，使用 NICE DCV 做為高效能顯示通訊協定，然後使用偏好的用戶端[連線至 NICE DCV 工作階段](#)。

## 6. 安裝 DCV 之後給予 DCV 使用者視訊許可：

```
$ sudo usermod -aG video dcv
```

### 在 CentOS 上設定互動式桌面

#### 1. 安裝 EPEL 儲存庫。

```
$ sudo yum update -y
$ C:\> sudo yum install epel-release -y
```

#### 2. 安裝 MATE 桌面。

```
$ sudo yum groupinstall "MATE Desktop" -y
$ C:\> sudo systemctl disable firewalld
```

#### 3. 將 xorg.conf 檔案複製至 /etc/X11/xorg.conf。

#### 4. 重新啟動執行個體。

```
$ sudo reboot
```

#### 5. (選用) [安裝 NICE DCV 伺服器](#)，使用 NICE DCV 做為高效能顯示通訊協定，然後使用偏好的用戶端[連線至 NICE DCV 工作階段](#)。

## Windows 執行個體的全虛擬驅動程式

Windows AMI 包含一組驅動程式，可用來許可存取虛擬化硬體。Amazon EC2 會使用這些驅動程式將執行個體存放區和 Amazon EBS 磁碟區映射到他們的裝置。下表顯示不同驅動程式的關鍵差異。

	RedHat 光伏	Citrix PV	AWS 光伏
執行個體類型	不支援所有執行個體類型。 若您指定不支援的執行個體類型，執行個體便會故障。	支援 Xen 執行個體類型。	支援 Xen 執行個體類型。

	RedHat 光伏	Citrix PV	AWS 光伏
已連接磁碟區	支援最多 16 個已連接磁碟區。	支援超過 16 個已連接磁碟區。	支援超過 16 個已連接磁碟區。
網路	驅動程式有已知問題。當發生此問題時，網路連線會在高負載下重設 (例如快速 FTP 檔案傳輸)。		驅動程式會在位於相容的執行個體類型上時，自動在網路介面卡上設定 Jumbo Frame。當執行個體位於叢集置放群組中時，這會在叢集置放群組中的執行個體之間提供較佳的網路效能。如需詳細資訊，請參閱 <a href="#">置放群組</a> 。

下列資料表顯示您在 Amazon EC2 之每個版本的 Windows Server 上建議執行的 PV 驅動程式。

Windows Server 版本	PV 驅動程式版本
Windows Server 2022	AWS 光伏最新版
Windows Server 2019	AWS 光伏最新版
Windows Server 2016	AWS 光伏最新版

Windows Server 版本	PV 驅動程式版本
Windows Server 2012 R2	AWS 光伏最新版
Windows Server 2012	AWS 光伏最新版
Windows Server 2008 R2	AWS 光伏版本 8.3.5
Windows Server 2008	Citrix PV 5.9
Windows Server 2003	Citrix PV 5.9

## 目錄

- [AWS 光伏驅動器](#)
- [Citrix PV 驅動程式](#)
- [RedHat 光伏驅動器](#)
- [訂閱 通知](#)
- [升級 Windows 執行個體上的 PV 驅動程式](#)
- [排解 Windows 執行個體上的光伏驅動](#)

## AWS 光伏驅動器

P AWS V 驅動程式會儲存在目%ProgramFiles%\Amazon\Xentools錄中。此目錄也包含公用符號和指令行工具xenstore\_client.exe，可讓您存取中的項目 XenStore。例如，下列 PowerShell 命令會從 Hypervisor 傳回目前的時間：

```
PS C:\> [DateTime]::FromFileTimeUTC((gwmi -n root\wmi -cl
 AWSXenStoreBase).XenTime).ToString("hh:mm:ss")
11:17:00
```

AWS 光伏驅動程式元件列於下的 Windows 登錄中HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services。這些驅動程式元件如下：xenbus、xeniface、xennet、xenvbd 及 xenvif。

AWS 光伏驅動程式也有一個名為的 Windows 服務 LiteAgent，該服務以使用者模式執行。它會處理來自 Xen 產生執行個體上 AWS API 的關閉和重新啟動事件等工作。您可以藉由從命令列執行 Services.msc 來存取及管理服務。在 Nitro 產生執行個體上執行時，不會使用 AWS PV 驅動程式，



且 LiteAgent 服務會從驅動程式版本 8.2.4 開始自我停止。更新至最新的 AWS PV 驅動程式也會更新 LiteAgent 並改善所有執行個體世代的可靠性。

## 安裝最新的 AWS 光伏驅動程式

Amazon Windows AMI 包含一組驅動程式，可用來許可存取虛擬化硬體。Amazon EC2 會使用這些驅動程式將執行個體存放區和 Amazon EBS 磁碟區映射到他們的裝置。我們建議您安裝最新的驅動程式，改善您 EC2 Windows 執行個體的穩定性和效能。

### 安裝選項

- 您可以使用 AWS Systems Manager 自動更新 PV 驅動程式。如需詳細資訊，請參閱 AWS Systems Manager 使用者指南中的[演練：自動更新 EC2 Windows 執行個體上的 PV 驅動程式 \(主控台\)](#)。
- 您可以[下載](#)驅動程式套件並手動執行安裝程式。請務必檢查 readme.txt 檔案是否有系統需求。如需有關下載和安裝 AWS PV 驅動程式，或是升級網域控制器的資訊，請參閱[手動升級 Windows 伺服器執行個體 \(AWS PV 升級\)](#)。

## AWS 光伏驅動器封裝歷史

下表顯示每個驅動程式版本對 AWS PV 驅動程式的變更。

套件版本	詳細資訊	發行日期
<a href="#">8.4.3</a>	修正了套件安裝程式中的錯誤，以改善升級體驗。	2023 年 1 月 24 日
<a href="#">8.4.2</a>	解決爭用情況的穩定性修復。	2022 年 4 月 13 日
<a href="#">8.4.1</a>	改進的套件安裝程式。	2022 年 1 月 7 日
<a href="#">8.4.0</a>	<ul style="list-style-type: none"> <li>• 穩定性修正，以解決磁碟 IO 的罕見卡住情況。</li> <li>• 穩定性修正，以解決 EBS 磁碟區分離期間的罕見當機情況。</li> <li>• 新增了將負載分散至多個核心的功能，以便工作負載充分利用超過 20,000 個 IOPS，以及應對因瓶頸而遭受降級的情況。若要啟用此功能，請參閱<a href="#">工作負載充分利用超過 20,000 個磁碟 IOPS，以及因 CPU 瓶頸而遭受降級的情況</a>。</li> </ul>	2021 年 3 月 2 日

套件版本	詳細資訊	發行日期
	<ul style="list-style-type: none"> <li>AWS 視窗伺服器 2008 R2 上的 PV 8.4 安裝將會失敗。AWS 光伏版本 8.3.5 及更早版本在視窗伺服器 2008 R2 上受到支援。</li> </ul>	
<a href="#">8.3.5</a>	改進的套件安裝程式。	2022 年 1 月 7 日
<a href="#">8.3.4</a>	改善網路裝置連接的可靠性。	2020 年 8 月 4 日
<a href="#">8.3.3</a>	<ul style="list-style-type: none"> <li>更新為 XenStore-facing 元件，以防止錯誤處理路徑期間的錯誤檢查。</li> <li>更新為儲存元件，以防止提交無效的 SRB 時發生當機。</li> </ul> <p>若要在 Windows Server 2008 R2 執行個體上更新此驅動程式，您必須先驗證已安裝適當的修補程式來處理下列 Microsoft Security Advisory：<a href="#">Microsoft Security Advisory 3033929</a>。</p>	2020 年 2 月 4 日
<a href="#">8.3.2</a>	強化的聯網元件可靠性	2019 年 7 月 30 日
<a href="#">8.3.1</a>	改善儲存體元件的效能和健全性。	2019 年 6 月 12 日
<a href="#">8.2.7</a>	已改善效率，支援遷移至最新一代執行個體類型。	2019 年 5 月 20 日
<a href="#">8.2.6</a>	提升損毀傾印路徑的效率。	2019 年 1 月 15 日
<a href="#">8.2.5</a>	<p>其他安全強化功能。</p> <p>PowerShell 安裝程序現在可以在包。</p>	2018 年 12 月 12 日
<a href="#">8.2.4</a>	可靠性改進功能。	2018 年 10 月 2 日

套件版本	詳細資訊	發行日期
<a href="#">8.2.3</a>	<p>錯誤修正與效能改進。</p> <p>將 EBS 磁碟區 ID 回報為 EBS 磁碟區的磁碟序號。這可啟用如 S2D 等叢集情境。</p>	2018 年 5 月 29 日
<a href="#">8.2.1</a>	<p>網路及儲存體效能改善及多個強固性修正。</p> <p>若要確認已安裝此版本，請參閱下列 Windows 登錄值：HKLM \Software\Amazon\PVDriver\Version 8.2.1 。</p>	2018 年 3 月 8 日
<a href="#">7.4.6</a>	穩定性修復，使 AWS 光伏驅動器更具彈性。	2017 年 4 月 26 日
7.4.3	<p>新增 Windows Server 2016 支援。</p> <p>所有支援的 Windows OS 版本穩定性修正。</p> <p>*AWS 光伏驅動程式 7.4.3 版的簽章將於 2019 年 3 月 29 日到期。我們建議您更新至最新的 AWS PV 驅動程式。</p>	2016 年 11 月 18 日
7.4.2	X1 執行個體類型支援的穩定性修正。	2016 年 8 月 2 日
7.4.1	<ul style="list-style-type: none"> <li>• AWS 光伏存儲驅動程序的性能改進。</li> <li>• AWS PV 儲存驅動程式中的穩定性修正：修正執行個體以錯誤檢查碼 0x0000Dead 造成系統當機的問題。</li> <li>• AWS PV 網路驅動程式的穩定性修正。</li> <li>• 新增 Windows Server 2008R2 支援。</li> </ul>	2016 年 7 月 12 日
7.3.2	<ul style="list-style-type: none"> <li>• 改善記錄日誌及診斷。</li> <li>• AWS 光伏存儲驅動程序的穩定性修復。在某些案例中，磁碟在重新連接到執行個體之後，可能不會在 Windows 中出現。</li> <li>• 新增 Windows Server 2012 支援。</li> </ul>	2015 年 6 月 24 日
7.3.1	TRIM 更新：與 TRIM 請求的相關修正。此修正會穩定執行個體，並改善執行個體管理大量 TRIM 請求時的效能。	

套件版本	詳細資訊	發行日期
7.3.0	TRIM 支援：AWS PV 驅動程式現在會將 TRIM 要求傳送給虛擬機器管理程序。暫時性磁碟會在基礎儲存體支援 TRIM (SSD) 時適當處理 TRIM 請求。請注意，2015 年 3 月時 EBS 式儲存體尚不支援 TRIM。	
7.2.5	<ul style="list-style-type: none"> <li>• AWS PV 儲存驅動程式中的穩定性修正：在某些情況下，AWS 光伏驅動程式可能會解除參照無效記憶體並導致系統故障。</li> <li>• 生成崩潰轉儲時的穩定性修復：在某些情況下，AWS PV 驅動程序可能會在寫入崩潰轉儲時卡在競爭情況下。在此版本之前，僅能透過強制驅動程式停止並重新啟動，遺失記憶體傾印來解決。</li> </ul>	
7.2.4	裝置 ID 持久性：此驅動程式修正會遮罩平台 PCI 裝置 ID，強制系統一律呈現相同的裝置 ID，即使執行個體已移動也一樣。更一般的來說，修正會影響虛擬化處理程序呈現虛擬裝置的方式。此修正程式也包含對 AWS PV 驅動程式的共同安裝程式所做的修改，以便系統持續保留對應的虛擬裝置	
7.2.2	<ul style="list-style-type: none"> <li>• 以目錄服務還原模式 (DSRM) 模式載入 AWS PV 驅動程式：目錄服務還原模式是 Windows Server 網域控制站的安全模式開機選項。</li> <li>• 在重新連接虛擬網路介面卡裝置時持續裝置 ID：此修正會強制系統檢查 MAC 地址映射並持續裝置 ID。此修正會確保介面卡在重新連接時保留其靜態設定。</li> </ul>	
7.2.1	<ul style="list-style-type: none"> <li>• 於安全模式中執行：修正驅動程式無法在安全模式中載入的問題。之前，P AWS V 驅動程式只會在正常執行的系統中實例化。</li> <li>• 新增磁碟至 Microsoft Windows 儲存集區 (Storage Pools)：先前我們會合成 Page 83 查詢。修正會停用 Page 83 支援。請注意，這不會影響目前在叢集環境中使用的儲存集區，因為 PV 磁碟並非有效的叢集磁碟。</li> </ul>	
7.2.0	基本：P AWS V 基本版本。	

## Citrix PV 驅動程式

Citrix PV 驅動程式存放在 `%ProgramFiles%\Citrix\XenTools` (32 位元執行個體) 或 `%ProgramFiles(x86)%\Citrix\XenTools` (64 位元執行個體) 目錄。

Citrix PV 驅動程式元件會在 Windows 登錄中的 `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services` 下列出。這些驅動程式元件如下：  
`xenevtchn`、`xeniface`、`xennet`、`Xennet6`、`xensvc`、`xenvbd` 及 `xenvif`。

Citrix 也有一個名為的驅動程式元件 `XenGuestAgent`，該元件會以 Windows 服務的形式執行。它會處理像是 API 的關機及重新啟動事件等任務。您可以藉由從命令列執行 `Services.msc` 來存取及管理服務。

若您在執行特定工作負載時發生網路錯誤，您可能需要停用 Citrix PV 驅動程式的 TCP 卸載功能。如需詳細資訊，請參閱 [TCP 卸載](#)。

## RedHat 光伏驅動器

RedHat 舊版執行個體支援驅動程式，但由於驅動程式限制，因此不建議在 RAM 超過 12GB 的較新執行個體上使用。具有超過 12GB RAM 執行 RedHat 驅動程式的執行個體可能無法開機並變得無法存取。我們建議您將 RedHat 驅動程式升級到 Citrix 光伏驅動程式，然後將 Citrix 光伏驅動器升級為 AWS 光伏驅動器。

RedHat 驅動程式的來源檔案位於 `%ProgramFiles%\RedHat` (32 位元執行個體) 或 `%ProgramFiles(x86)%\RedHat` (64 位元執行個體) 目錄中。這兩個驅動程式是 `rhelnet` RedHat 半虛擬化網路驅動程式，以及 `rhelscsi` RedHat SCSI 微型連接埠驅動程式。

## 訂閱通知

Amazon SNS 會在 EC2 Windows 驅動程式發佈新版本時通知您。使用下列其中一種方法訂閱這些通知。

### Note

您必須為訂閱的 SNS 主題指定「區域」。

## 從主控台訂閱 EC2 通知

1. 在 <https://console.aws.amazon.com/sns/v3/home> 開啟 Amazon SNS 主控台。

2. 如有必要，請在導覽列中將「區域」變更為美國東部 (維吉尼亞北部)。您必須選取此區域，因為您要訂閱的 SNS 通知位於本區域中。
3. 在導覽窗格中，選擇 Subscriptions (訂閱)。
4. 選擇 Create subscription (建立訂閱)。
5. 在 Create subscription (建立訂閱) 對話方塊中，執行下列動作：
  - a. 針對 TopicARN，複製下列 Amazon Resource Name (ARN)：  
`arn:aws:sns:us-east-1:801119661308:ec2-windows-drivers`
  - b. 針對 Protocol (通訊協定)，選擇 Email。
  - c. 針對 Endpoint (端點)，輸入可用於接收通知的電子郵件地址。
  - d. 選擇 Create subscription (建立訂閱)。
6. 您將會收到確認電子郵件。開啟電子郵件並遵循指示完成訂閱。

## 訂閱 EC2 通知使用 AWS CLI

若要使用訂閱 EC2 通知 AWS CLI，請使用下列命令。

```
aws sns subscribe --topic-arn arn:aws:sns:us-east-1:801119661308:ec2-windows-drivers --region us-east-1 --protocol email --notification-endpoint YourUserName@YourDomainName.ext
```

## 訂閱 EC2 通知使用 AWS Tools for PowerShell

若要使用適用於 Windows 的工具訂閱 EC2 通知 PowerShell，請使用下列命令。

```
Connect-SNSNotification -TopicArn 'arn:aws:sns:us-east-1:801119661308:ec2-windows-drivers' -Region us-east-1 -Protocol email -Endpoint 'YourUserName@YourDomainName.ext'
```

只要有新的 EC2 Windows 驅動程式發行，我們就會向訂閱者傳送通知。如果您不想再接收這些通知，請使用下列程序來取消訂閱。

## 取消訂閱 Amazon EC2 Windows 驅動程式通知

1. 在 <https://console.aws.amazon.com/sns/v3/home> 開啟 Amazon SNS 主控台。
2. 在導覽窗格中，選擇 Subscriptions (訂閱)。
3. 選取訂閱的核取方塊，然後選取 Actions (動作)，Delete subscriptions (刪除訂閱)。出現確認提示時，請選擇 Delete (刪除)。

## 升級 Windows 執行個體上的 PV 驅動程式

我們建議您安裝最新的 PV 驅動程式，改善您 EC2 Windows 執行個體的穩定性和效能。此頁面上的指示可協助您下載驅動程式套件並執行安裝程式。

### 確認 Windows 執行個體使用的驅動程式

開啟 Control Panel (控制台) 中的 Network Connections (網路連線)，並檢視 Local Area Connection (區域連線)。檢查驅動程式是否為下列項目的其中之一：

- AWS 光伏網路裝置
- Citrix PV 乙太網路卡
- RedHat 光伏網卡驅動

或者，您可以檢查 `pnputil -e` 命令的輸出。

### 系統要求

請務必檢查下載中的 `readme.txt` 檔案是否有系統需求。

### 目錄

- [使用代理商升級 Windows 伺服器執行個體 \(AWS PV 升級\)](#)
- [手動升級 Windows 伺服器執行個體 \(AWS PV 升級\)](#)
- [升級網域控制站 \(AWS PV 升級\)](#)
- [升級 Windows Server 2008 及 2008 R2 執行個體 \(RedHat 升級至 Citrix PV\)](#)
- [升級您的 Citrix Xen 客體代理程式服務](#)

### 使用代理商升級 Windows 伺服器執行個體 (AWS PV 升級)

您可以使用「代理商」功能來安裝或升級 AWS PV 驅動程式套件。AWS Systems Manager 安裝或升級可以執行一次，也可以按照排程安裝或更新。此代理商套件不支援「安裝類型 **In-place update**」選項。

#### Important

如果您的執行個體為網域控制站，請參閱[升級網域控制站 \(AWS PV 升級\)](#)。網域控制器執行個體的升級程序與標準版本的 Windows 不同。

1. 我們建議您建立備份，以防您需要復原變更。

 Tip

您可以使用 Systems Manager 自動化來建立 AMI，而不是從 Amazon EC2 主控台建立 AWS-CreateImage 立 AMI。如需詳細資訊，請參閱 AWS Systems Manager 自動化工作流程簿參考使用者指南 [AWS-CreateImage](#) 中的。

- a. 停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。停止執行個體之前，請確認您已將所需的任何資料從執行個體存放區磁碟區複製到永久性儲存，例如 Amazon EBS 或 Amazon S3。
  - b. 在導覽窗格中，選擇 Instances (執行個體)。
  - c. 選取需要升級驅動程式的執行個體，並依序選取 Instance state (執行個體狀態)、Stop instance (停止執行個體)。
  - d. 停止執行個體後，請選取執行個體，依序選取 Actions (動作)、Image and templates (映像和範本)，然後選取 Create image (建立映像)。
  - e. 選擇 Instance state (執行個體狀態)、Start instance (啟動執行個體)。
2. 使用遠端桌面連線到執行個體。如需詳細資訊，請參閱 [the section called “使用 RDP 用戶端 Connect 至您的 Windows 執行個體”](#)。
  3. 建議您讓所有非系統磁碟離線，並且在磁碟管理中記下次要磁碟的任何磁碟機代號映射，之後再執行此升級。如果您要執行 AWS PV 驅動程式的就地更新，則不需要執行此步驟。我們也建議在服務主控台中將非必要的服務設為 Manual (手動) 啟動。
  4. 如需有關如何使用「代理商」安裝或升級 AWS PV 驅動程式套件的指示，請參閱「使用 AWS Systems Manager 者指南」中的「[安裝或更新套件](#)」中的程序。
  5. 對於「名稱」，選擇 AWSPVDriver。
  6. 對於安裝類型，選取解除安裝並重新安裝。
  7. 視需要設定套裝軟體的其他參數，並使用中參照的程序執行安裝或升級 [Step 4](#)。

執行散發程式套件之後，執行個體會自動重新開機，然後升級驅動程式。執行個體最多可能會在 15 分鐘內無法使用。

8. 升級完成後，執行個體在 Amazon EC2 主控台中通過兩項運作狀態檢查後，請使用遠端桌面連線至執行個體，確認已安裝新驅動程式。
9. 連線完成後，執行下列 PowerShell 命令：



```
Get-ItemProperty HKLM:\SOFTWARE\Amazon\PVDriver
```

10. 確認驅動程式版本與驅動程式版本歷史記錄表格中列出的最新版本相同。如需詳細資訊，請參閱[AWS 光伏驅動器封裝歷史](#)開啟磁碟管理以檢閱任何離線輔助磁碟區，並將其與中所述磁碟機代號相對應的線上[Step 3](#)。

如果您先前停[TCP 卸載](#)用 Netsh 的 Citrix 光伏驅動程式，我們建議您在升級到 AWS PV 驅動程式後重新啟用此功能。AWS 光伏驅動程式中不存在 Citrix 驅動程式的 TCP 卸載問題。因此，TCP 卸載可提供更好的 AWS PV 驅動程式效能。

如果您先前已將靜態 IP 位址或 DNS 組態套用至網路介面，則在升級 AWS PV 驅動程式之後，可能需要重新套用靜態 IP 位址或 DNS 組態。

### 手動升級 Windows 伺服器執行個體 (AWS PV 升級)

請使用下列程序在視窗伺服器 2008 R2、視窗伺服器 2012 年 R2、視窗伺服器 2012 年 R2、視窗伺服器 2012 年 R2、視窗伺服器 2012 年 R2、視窗伺服器 2016 年、視窗伺服器 2016 年、視窗伺服器 2019 或視窗伺服器 2022 上執行 AWS 光伏驅動程式的就地升級，或將光伏驅動程式 AWS 此升級不適用於 RedHat 驅動程式或其他版本的 Windows 伺服器。

某些舊版 Windows 伺服器無法使用最新的驅動程式。若要確認您的作業系統所使用的驅動程式版本，請參閱 [Windows 執行個體的全虛擬驅動程式](#) 頁面中的驅動程式版本表。

#### Important

如果您的執行個體為網域控制站，請參閱[升級網域控制站 \(AWS PV 升級\)](#)。網域控制器執行個體的升級程序與標準版本的 Windows 不同。

### 手動升級 AWS PV 驅動程式

1. 我們建議您建立備份，以防您需要復原變更。

#### Tip

您可以使用 Systems Manager 自動化來建立 AMI，而不是從 Amazon EC2 主控台建 AWS-CreateImage 立 AMI。如需詳細資訊，請參閱 AWS Systems Manager 自動化工作流程簿參考使用者指南[AWS-CreateImage](#)中的。

- a. 停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。停止執行個體之前，請確認您已將所需的任何資料從執行個體存放區磁碟區複製到永久性儲存，例如 Amazon EBS 或 Amazon S3。
  - b. 在導覽窗格中，選擇 Instances (執行個體)。
  - c. 選取需要升級驅動程式的執行個體，並依序選取 Instance state (執行個體狀態)、Stop instance (停止執行個體)。
  - d. 停止執行個體後，請選取執行個體，依序選取 Actions (動作)、Image and templates (映像和範本)，然後選取 Create image (建立映像)。
  - e. 選擇 Instance state (執行個體狀態)、Start instance (啟動執行個體)。
2. 使用遠端桌面連線到執行個體。
  3. 建議您讓所有非系統磁碟離線，並且在磁碟管理中記下次要磁碟的任何磁碟機代號映射，之後再執行此升級。如果您要執行 AWS PV 驅動程式的就地更新，則不需要執行此步驟。我們也建議在服務主控台中將非必要的服務設為 Manual (手動) 啟動。
  4. 將最新的驅動程式套件[下載](#)到執行個體。

或者，執行下列 PowerShell 命令：

```
Invoke-WebRequest https://s3.amazonaws.com/ec2-windows-drivers-downloads/AWSPV/Latest/AWSPVDriver.zip -outfile $env:USERPROFILE\pv_driver.zip
Expand-Archive $env:userprofile\pv_driver.zip -DestinationPath
$env:userprofile\pv_drivers
```

#### Note

如果您在下載檔案時收到錯誤訊息，而且您使用的是 Windows 伺服器 2016 或更早版本，則可能需要為您的 PowerShell 終端機啟用 TLS 1.2。您可以使用下列命令為目前 PowerShell 工作階段啟用 TLS 1.2，然後再試一次：

```
[Net.ServicePointManager]::SecurityProtocol =
[Net.SecurityProtocolType]::Tls12
```

5. 解壓縮資料夾的內容，然後執行 AWSPVDriverSetup.msi。

在執行 MSI 後，執行個體會自動重新開機並升級驅動程式。執行個體最多可能會在 15 分鐘內無法使用。升級完成且執行個體在 Amazon EC2 主控台中通過兩項運作狀態檢查之後，您可以使用遠端桌面連線至執行個體，然後執行下列 PowerShell 命令來驗證是否已安裝新驅動程式：

```
Get-ItemProperty HKLM:\SOFTWARE\Amazon\PVDriver
```

確認驅動程式版本與驅動程式版本歷史記錄表格中列出的最新版本相同。如需詳細資訊，請參閱[AWS 光伏驅動器封裝歷史](#)開啟磁碟管理以檢閱任何離線輔助磁碟區，並將其與中所述磁碟機代號相對應的線上[Step 3](#)。

如果您先前停[TCP 卸載](#)用 Netsh 的 Citrix 光伏驅動程式，我們建議您在升級到 AWS PV 驅動程式後重新啟用此功能。AWS 光伏驅動程式中不存在 Citrix 驅動程式的 TCP 卸載問題。因此，TCP 卸載可提供更好的 AWS PV 驅動程式效能。

如果您先前已將靜態 IP 位址或 DNS 組態套用至網路介面，則在升級 AWS PV 驅動程式之後，可能需要重新套用靜態 IP 位址或 DNS 組態。

### 升級網域控制站 (AWS PV 升級)

在網域控制站上使用下列程序來執行 PV 驅動程式的就地升級，或從 Citrix AWS PV 驅動程式升級為 AWS PV 驅動程式。

### 升級網域控制站

1. 我們建議您建立網域控制站的備份，以防您需要復原變更。不支援使用 AMI 作為備份。如需詳細資訊，請參閱 Microsoft 文件中的[虛擬網域控制站的備份和還原考慮](#)。
2. 執行下列命令設定 Windows 開機進入目錄服務還原模式 (DSRM)。

#### Warning

在執行此命令前，請確認您知道 DSRM 的密碼。您將需要此資訊，才能在升級完成且執行個體自動重新開機後登入您的執行個體。

```
bcdedit /set {default} safeboot dsrepair
```

PowerShell:

```
PS C:\> bcdedit /set "{default}" safeboot dsrepair
```

系統必須開機進入 DSRM，因為升級公用程式會移除 Citrix PV 儲存驅動程式，以便安裝 AWS PV 驅動程式。因此，建議您在磁碟管理中記下次要磁碟的任何磁碟機代號和資料夾映射。當 Citrix PV 儲存體驅動程式不存在時，將無法偵測到輔助磁碟機。在輔助磁碟機上使用 NTDS 資料夾的網域控制站將無法開機，因為無法偵測到輔助磁碟。

**Warning**

在您執行此命令後，請不要手動重新開機系統。否則將會無法觸及系統，因為 Citrix PV 驅動程式不支援 DSRM。

- 執行下列命令，將 **DisableDCCheck** 新增到登錄：

```
reg add HKLM\SOFTWARE\Wow6432Node\Amazon\AWSPVDriverSetup /v DisableDCCheck /t REG_SZ /d true
```

- 將最新的驅動程式套件[下載](#)到執行個體。
- 解壓縮資料夾的內容，然後執行 AWSPVDriverSetup.msi。

在執行 MSI 後，執行個體會自動重新開機並升級驅動程式。執行個體最多可能會在 15 分鐘內無法使用。

- 在升級完成且執行個體在 Amazon EC2 主控台中通過兩項運作狀態檢查後，使用遠端桌面連線到執行個體。開啟磁碟管理來檢閱任何離線的輔助磁碟區，並根據稍早記下的對應磁碟機代號和資料夾映射讓磁碟區回到線上狀態。

您必須以下列格式指定使用者名稱來連線到執行個體：hostname\administrator。例如，Win2K12TestBox\ 管理員。

- 執行下列命令移除 DSRM 開機組態：

```
bcdedit /deletevalue safeboot
```

- 重新啟動執行個體。
- 若要完成升級程序，請確認已安裝新的驅動程式。在裝置管理員中，於 Storage Controllers (儲存控制器) 下，尋找 AWS PV Storage Host Adapter (AWS PV 儲存體主機控制器)。確認驅動程式版本與驅動程式版本歷史記錄表格中列出的最新版本相同。如需詳細資訊，請參閱 [AWS 光伏驅動器封裝歷史](#)。
- 執行下列命令，從登錄刪除 **DisableDCCheck**：

```
reg delete HKLM\SOFTWARE\Wow6432Node\Amazon\AWSPVDriverSetup /v DisableDCCheck
```

### Note

如果您先前停 [TCP 卸載](#) 用 Netsh 的 Citrix 光伏驅動程式，我們建議您在升級到 AWS PV 驅動程式後重新啟用此功能。AWS 光伏驅動程式中不存在 Citrix 驅動程式的 TCP 卸載問題。因此，TCP 卸載可提供更好的 AWS PV 驅動程式效能。

## 升級 Windows Server 2008 及 2008 R2 執行個體 (RedHat 升級至 Citrix PV)

在開始將 RedHat 驅動程式升級到 Citrix PV 驅動程式之前，請務必執行以下操作：

- 安裝最新版的 EC2Config 服務。如需詳細資訊，請參閱 [安裝最新版本的 EC2Config](#)。
- 請確認您已安裝視窗 PowerShell 3.0。若要驗證您已安裝的版本，請在 PowerShell 視窗中執行下列命令：

```
PS C:\> $PSVersionTable.PSVersion
```

視窗 PowerShell 3.0 隨附於視窗管理架構 (WMF) 3.0 版安裝套件中。如果您需要安裝視窗 PowerShell 3.0，請參閱 Microsoft 下載中心的 [視窗管理架構 3.0](#)。

- 備份您在執行個體上的重要資訊，或是從執行個體建立 AMI。如需建立 AMI 的詳細資訊，請參閱 [創建一個 Amazon EBS-backed AMI](#)。

### Tip

您可以使用 Systems Manager 自動化來建立 AMI，而不是從 Amazon EC2 主控台建 `AWS-CreateImage` 立 AMI。如需詳細資訊，請參閱 AWS Systems Manager 自動化工作流程簿參考使用者指南 [AWS-CreateImage](#) 中的。

若您建立 AMI，請確認執行下列作業：

- 記下您的密碼。
- 請不要手動或使用 EC2Config 服務執行 Sysprep 工具。

- 設定您的乙太網路卡，以自動使用 DHCP 取得 IP 地址。如需詳細資訊，請參閱在 Microsoft TechNet 程式庫中[設定 TCP/IP 設定](#)。

## 升級 RedHat 驅動程式

1. 連線到您的執行個體，並以本機管理員登入。如需連線至執行個體的詳細資訊，請參閱[連線至 Windows 執行個體](#)。
2. 在您的執行個體中，[下載](#) Citrix PV 升級套件。
3. 將升級套件的內容解壓縮到您選擇的位置。
4. 按兩下 Upgrade.bat 檔案。若您收到安全警告，請選擇 Run (執行)。
5. 在 Upgrade Drivers (升級驅動程式) 對話方塊中，檢閱資訊並於準備好開始升級時選擇 Yes (是)。
6. 在適用於 Windows 解除安裝程式的 Red Hat 準虛擬化 Xen 驅動程式對話方塊中，選擇是以移除軟體。RedHat 您的執行個體將會重新開機。

### Note

若您沒有看到解除安裝程式對話方塊，請在 Windows 任務列中選擇 Red Hat Paravirtualize (Red Hat 全虛擬化)。



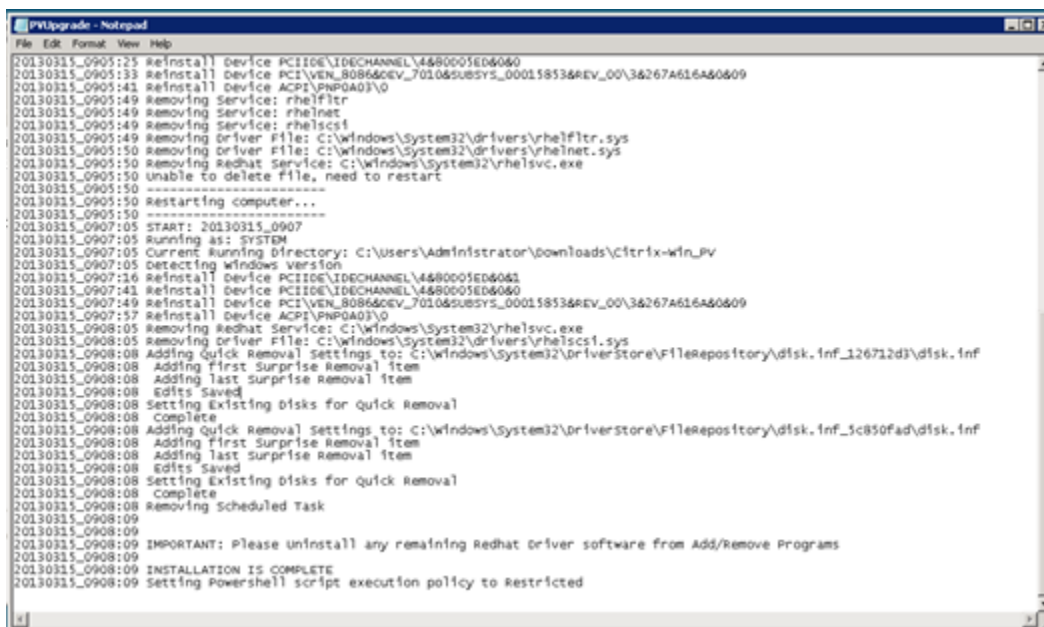
7. 檢查執行個體已重新開機並準備好可供使用。
  - a. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
  - b. 在 Instances (執行個體) 頁面上，依序選取 Actions (動作)、Monitor and troubleshoot (監視和故障診斷)，然後選取 Get system log (取得系統記錄檔)。
  - c. 升級操作應該會重新開機伺服器 3 或 4 次。您可以在日誌檔案中，查看顯示的 Windows is Ready to use 次數來了解。

```

Microsoft Windows NT 6.0.6002 Service Pack 2 (en-US)
Ec2Config service v2.1.9.0
RedHat PV NIC Driver v1.3.10.0
2013/03/15 17:11:01Z: Waiting for meta-data accessibility...
2013/03/15 17:11:02Z: Meta-data is now available.
<RDPCERTIFICATE>
<THUMBPRINT>D6BFD64F21359516C781CA7DF2821C5EFC35648A</THUMBPRINT>
</RDPCERTIFICATE>
<Username>Administrator</Username>
<Password>
L79ThJPF8LyIL38I2ht0FBrjet3vnT2csTiU/XGVMRCH7kQtBznAnXrKd1sirXlx19BwVMsd9b38jFJqv01IUpgNNJRZoCdc7IbUw
</Password>
2013/03/15 17:11:30Z: Product activation was successful.
2013/03/15 17:11:32Z: Message: Windows is Ready to use
Microsoft Windows NT 6.0.6002 Service Pack 2 (en-US)
Ec2Config service v2.1.9.0
2013/03/15 21:04:24Z: There was an exception writing driver information to console: System.Exception: U
at Ec2Config.Service1.Go()
2013/03/15 21:04:35Z: Waiting for meta-data accessibility...
2013/03/15 21:04:40Z: Meta-data is now available.
<RDPCERTIFICATE>
<THUMBPRINT>D6BFD64F21359516C781CA7DF2821C5EFC35648A</THUMBPRINT>
</RDPCERTIFICATE>
2013/03/15 21:05:08Z: Product activation was successful.
2013/03/15 21:05:09Z: Message: Windows is Ready to use
Microsoft Windows NT 6.0.6002 Service Pack 2 (en-US)
Ec2Config service v2.1.9.0
Citrix PV Ethernet Adapter v5.9.960.49119
2013/03/15 21:07:20Z: Waiting for meta-data accessibility...
2013/03/15 21:07:21Z: Meta-data is now available.
<RDPCERTIFICATE>
<THUMBPRINT>D6BFD64F21359516C781CA7DF2821C5EFC35648A</THUMBPRINT>
</RDPCERTIFICATE>
2013/03/15 21:07:27Z: Message: Windows is Ready to use

```

8. 連線到您的執行個體，並以本機管理員登入。
9. 關閉 Red Hat Paravirtualized Xen Drivers for Windows uninstaller (Windows Red Hat 全虛擬化 Xen 驅動程式解除安裝程式) 對話方塊。
10. 確認安裝已完成。導覽至您先前解壓縮的 Citrix-WIN\_PV 資料夾，開啟 PVUpgrade.log 檔案，然後檢查是否有 INSTALLATION IS COMPLETE 的文字。



```

PVUpgrade - Notepad
File Edit Format View Help
20130315_0905:125 Reinstall Device PCIIDE\IDECHANNEL\4480001ED6060
20130315_0905:133 Reinstall Device PCI\VEN_8086&DEV_7010&SUBSYS_00015853&REV_00\3&267A616A&0609
20130315_0905:143 Reinstall Device ACPI\PNP0A03\0
20130315_0905:149 Removing Service: rhelftr
20130315_0905:149 Removing Service: rhelnet
20130315_0905:149 Removing Service: rhelscs1
20130315_0905:149 Removing Driver File: C:\Windows\System32\drivers\rhelftr.sys
20130315_0905:150 Removing Driver File: C:\Windows\System32\drivers\rhelnet.sys
20130315_0905:150 Removing Redhat Service: C:\Windows\System32\rhelsvc.exe
20130315_0905:150 unable to delete file, need to restart
20130315_0905:150 Restarting computer...
20130315_0905:150 *****
20130315_0907:05 START: 20130315_0907
20130315_0907:05 Running as: SYSTEM
20130315_0907:05 Current Running Directory: C:\Users\Administrator\downloads\Citrix-Win_PV
20130315_0907:05 Detecting Windows version
20130315_0907:16 Reinstall Device PCIIDE\IDECHANNEL\4480001ED6060
20130315_0907:143 Reinstall Device PCIIDE\IDECHANNEL\4480001ED6060
20130315_0907:149 Reinstall Device PCI\VEN_8086&DEV_7010&SUBSYS_00015853&REV_00\3&267A616A&0609
20130315_0907:157 Reinstall Device ACPI\PNP0A03\0
20130315_0908:05 Removing Redhat Service: C:\Windows\System32\rhelsvc.exe
20130315_0908:05 Removing Driver File: C:\Windows\System32\drivers\rhelscs1.sys
20130315_0908:08 Adding Quick Removal Settings to: C:\Windows\System32\DriverStore\FileRepository\disk.inf_126712d3\disk.inf
20130315_0908:08 Adding first Surprise Removal Item
20130315_0908:08 Adding last surprise Removal Item
20130315_0908:08 Edits Saved
20130315_0908:08 Setting Existing disks for Quick Removal
20130315_0908:08 Complete
20130315_0908:08 Adding Quick Removal Settings to: C:\Windows\System32\DriverStore\FileRepository\disk.inf_5c850fad\disk.inf
20130315_0908:08 Adding first Surprise Removal Item
20130315_0908:08 Adding last surprise Removal Item
20130315_0908:08 Edits Saved
20130315_0908:08 Setting Existing disks for Quick Removal
20130315_0908:08 Complete
20130315_0908:08 Removing Scheduled Task
20130315_0908:09
20130315_0908:09 IMPORTANT: Please uninstall any remaining Redhat driver software from Add/Remove Programs
20130315_0908:09
20130315_0908:09 INSTALLATION IS COMPLETE
20130315_0908:09 Setting Powershell script execution policy to Restricted

```

## 升級您的 Citrix Xen 客體代理程式服務

若您在 Windows Server 上使用 Citrix PV 驅動程式，您可以升級 Citrix Xen 客體代理程式服務。此 Windows 服務會處理像是 API 的關機及重新啟動事件等任務。您可以在任何版本的 Windows Server 上執行此升級套件，只要執行個體是執行 Citrix PV 驅動程式即可。

### Important

對於 Windows 伺服器 2008 R2 及更新版本，我們建議您升級至包含客體代理程式更新的 AWS 光伏驅動程式。

在您開始升級您的驅動程式前，請確認您已備份您在執行個體上的重要資訊，或是從執行個體建立 AMI。如需建立 AMI 的詳細資訊，請參閱[創建一個 Amazon EBS-backed AMI](#)。

### Tip

您可以使用 Systems Manager 自動化來建立 AMI，而不是從 Amazon EC2 主控台建 AWS-CreateImage 立 AMI。如需詳細資訊，請參閱 AWS Systems Manager 自動化工作流程簿參考使用者指南 [AWS-CreateImage](#) 中的。

若您建立 AMI，請確認執行下列作業：

- 請不要在 EC2Config 服務中啟用 Sysprep 工具。
- 記下您的密碼。
- 將您的乙太網路卡設為 DHCP。

## 升級您的 Citrix Xen 客體代理程式服務

1. 連線到您的執行個體，並以本機管理員登入。如需連線至執行個體的詳細資訊，請參閱[連線至 Windows 執行個體](#)。
2. 在您的執行個體上，[下載](#) Citrix 升級套件。
3. 將升級套件的內容解壓縮到您選擇的位置。
4. 按兩下 Upgrade.bat 檔案。若您收到安全警告，請選擇 Run (執行)。
5. 在 Upgrade Drivers (升級驅動程式) 對話方塊中，檢閱資訊並於準備好開始升級時選擇 Yes (是)。



6. 當升級完成時，將會開啟 PVUpgrade.log 檔案，其中包含 UPGRADE IS COMPLETE 的文字。
7. 將執行個體重新開機。

## 排解 Windows 執行個體上的光伏驅動

對於您使用舊版 Amazon EC2 映像和 PV 驅動程式時可能發生問題，以下是解決方案。

### 內容

- [Windows Server 2012 R2 在執行個體重新開機後遺失網路及儲存體連線能力](#)
- [TCP 卸載](#)
- [時間同步](#)
- [工作負載充分利用超過 20,000 個磁碟 IOPS，以及因 CPU 瓶頸而遭受降級的情況](#)

### Windows Server 2012 R2 在執行個體重新開機後遺失網路及儲存體連線能力

#### Important

僅 2014 年 9 月以前推出的 AMI 才會發生此問題。

2014 年 9 月 10 日之前開放使用的 Windows Server 2012 R2 Amazon Machine Images (AMI) 可能會在執行個體重新開機後遺失網路及儲存體的連線能力。AWS Management Console 系統記錄檔中的錯誤指出：「無法偵測控制台輸出的 PV 驅動程式詳細資料」。連線能力遺失是由「隨插即用清理」功能造成的。此功能會每 30 天掃描並停用非使用中的系統裝置。功能會不正確的將 EC2 網路裝置識別為非使用中，並從系統移除它。當發生此狀況時，執行個體便會在重新開機後遺失網路連線能力。

針對您懷疑受到此問題影響的系統，您可以下載及執行就地驅動程式升級。若您無法執行就地驅動程式升級，您可以執行協助程式指令碼。指令碼會判斷您的執行個體是否受到影響。若其受到影響，但 Amazon EC2 網路裝置尚未遭到移除，指令碼會停用隨插即用清理掃描。若已移除網路裝置，指令碼會修復裝置，停用隨插即用清理掃描，讓您的執行個體重新開機並啟用網路連線能力。

### 內容

- [選擇如何修正問題](#)
- [方法 1 - 增強聯網](#)
- [方法 2 - 登錄組態](#)

- [執行補救指令碼](#)

## 選擇如何修正問題

有兩種方法可以還原受此問題影響執行個體的網路及儲存體連線能力。選擇下列其中一種方法：

方法	必要條件	程序概觀
方法 1 - 增強聯網	增強聯網僅在 Virtual Private Cloud (VPC) 中可用，且需要 C3 執行個體類型。若伺服器目前使用的並非 C3 執行個體類型，您必須暫時變更它。	您將伺服器執行個體類型變更為 C3 執行個體。增強聯網便可讓您連線到受影響的執行個體並修正問題。在修正問題之後，您再將執行個體變更回原始的執行個體類型。此方法通常比方法 2 更快，也較不容易導致使用者錯誤。執行 C3 執行個體時，您將需要支付額外的費用。
方法 2 - 登錄組態	建立或存取第二個伺服器的能力。變更登錄設定的能力。	您將根磁碟區從受影響的執行個體分離，再將其連接到不同的執行個體、連線，並在登錄中進行變更。執行額外的伺服器時，您將需要支付額外的費用。此方法比方法 1 慢，但此方法可在方法 1 無法解決問題時使用。

### 方法 1 - 增強聯網

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 尋找受影響的執行個體。選取執行個體，並依序選取 Instance state (執行個體狀態)、Stop instance (停止執行個體)。

**⚠ Warning**

停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區的資料，請將資料備份至持久性儲存。

4. 在執行個體停止後，建立備份。選取執行個體，依序選取 Actions (動作)、Image and templates (映像和範本)，然後選取 Create image (建立映像)。
5. 將執行個體類型**變更**為任何 C3 執行個體類型。
6. [啟動](#)實例。
7. 使用遠端桌面 Connect 至執行個體，然後將 AWS PV 驅動程式升級套件[下載](#)至執行個體。
8. 解壓縮資料夾的內容，然後執行 AWSPVDriverSetup.msi。

在執行 MSI 後，執行個體會自動重新開機並升級驅動程式。執行個體最多可能會在 15 分鐘內無法使用。

9. 在升級完成且執行個體在 Amazon EC2 主控台中通過兩項運作狀態檢查後，使用遠端桌面連線到執行個體，確認已安裝新的驅動程式。在裝置管理員中，於 Storage Controllers (儲存控制器) 下，尋找 AWS PV Storage Host Adapter (AWS PV 儲存體主機控制器)。確認驅動程式版本與驅動程式版本歷史記錄表格中列出的最新版本相同。如需詳細資訊，請參閱 [AWS 光伏驅動器封裝歷史](#)。
10. 停止執行個體，並將執行個體變更回原始的執行個體類型。
11. 啟動執行個體，繼續一般的使用。

## 方法 2 - 登錄組態

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 尋找受影響的執行個體。選取執行個體，並依序選取 Instance state (執行個體狀態)、Stop instance (停止執行個體)。

**⚠ Warning**

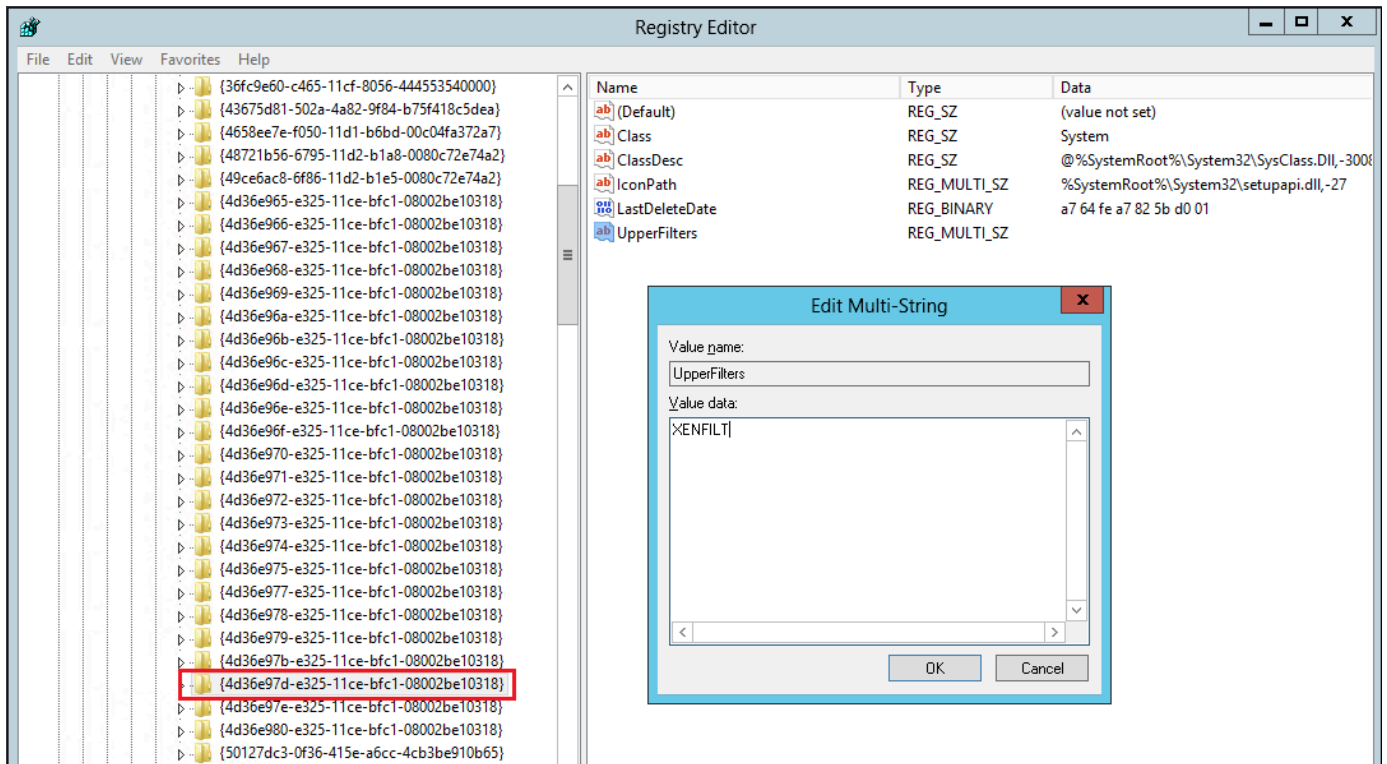
停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區的資料，請將資料備份至持久性儲存。

4. 選擇 Launch Instance (啟動執行個體)，在與受影響的執行個體相同的可用區域中建立暫時性的 Windows Server 2008 或 Windows Server 2012 執行個體。請不要建立 Windows Server 2012 R2 執行個體

**⚠ Important**

若您沒有在與受影響的執行個體相同的可用區域內建立執行個體，您將無法將受影響執行個體的根磁碟區連接到新的執行個體。

5. 在導覽窗格中，選擇 Volumes (磁碟區)。
6. 尋找受影響執行個體的根磁碟區。[分開磁碟區](#)，再[將磁碟區連接](#)至先前建立的暫時執行個體。使用預設裝置名稱 (xvdf) 連接它。
7. 使用遠端桌面連接至暫時性執行個體，然後使用 Disk Management 公用程式將[磁碟區變成可用](#)。
8. 在暫時性執行個體上，開啟 Run (執行) 對話方塊，輸入 **regedit**，然後按 Enter 鍵。
9. 在登錄編輯程式導覽窗格中，選擇 HKEY\_Local\_Machine，然後從 File (檔案) 選單選擇 Load Hive (載入 Hive)。
10. 在 Load Hive (載入 Hive) 對話方塊中，導覽至 Affected Volume (受影響的裝置)\Windows \System32\config\System 然後在 Key Name (索引鍵名稱) 對話方塊中輸入暫時性的名稱。例如，輸入 OldSys。
11. 在登錄編輯程式的導覽窗格中，尋找下列索引鍵：  
  
HKEY\_本地機器\#####\001\###\##\4 ControlSet  
  
HKEY\_本機電腦\#####\001\###\##\4 ControlSet
12. 對於每個按鍵，連按兩下 UpperFilters，輸入 XENFILT 的值，然後選擇 [確定]。



13. 尋找下列索引鍵：

HKEY\_本地機器\#####\001\##\####\## ControlSet

14. 使用名稱 ActiveDevice 和下列值建立新字串 (REG\_SZ)：

PCI\VEN\_5853&DEV\_0001&SUBSYS\_00015853&REV\_01

15. 尋找下列索引鍵：

HKEY\_本地機器\#####\001\##\XENBUS ControlSet

16. 將 Count 從 0 變更為 1。

17. 尋找並刪除下列索引鍵：

HKEY\_本地機器\#####密鑰\_名稱\001\服務\特別\ControlSet StartOverride

HKEY\_本地機器\#####\001\##\####件\ControlSet StartOverride

18. 在登錄編輯程式導覽窗格中，選擇一開始開啟登錄編輯程式時建立的暫時性機碼。

19. 從 File (檔案) 選單中，選擇 Unload Hive (解除載入 Hive)。

20. 在磁碟管理公用程式中，選擇先前連接的磁碟機，開啟內容 (按右鍵) 選單，然後選擇 Offline (離線)。

21. 在 Amazon EC2 主控台中，從暫時性的執行個體分離受影響的磁碟區，然後以 /dev/sda1 做為裝置名稱，重新連接到您的 Windows Server 2012 R2 執行個體。您必須指定此裝置名稱，將磁碟區指定為根磁碟區。
22. [啟動實例](#)。
23. 使用遠端桌面 Connect 至執行個體，然後將 AWS PV 驅動程式升級套件[下載](#)至執行個體。
24. 解壓縮資料夾的內容，然後執行 AWSPVDriverSetup.msi。

在執行 MSI 後，執行個體會自動重新開機並升級驅動程式。執行個體最多可能會在 15 分鐘內無法使用。

25. 在升級完成且執行個體在 Amazon EC2 主控台中通過兩項運作狀態檢查後，使用遠端桌面連線到執行個體，確認已安裝新的驅動程式。在裝置管理員中，於 Storage Controllers (儲存控制器) 下，尋找 AWS PV Storage Host Adapter (AWS PV 儲存體主機控制器)。確認驅動程式版本與驅動程式版本歷史記錄表格中列出的最新版本相同。如需詳細資訊，請參閱 [AWS 光伏驅動器封裝歷史](#)。
26. 刪除或停止您在此程序中建立的暫時性執行個體。

### 執行補救指令碼

若您無法執行就地驅動程式升級或遷移至較新的執行個體，您可以執行補救指令碼來修正隨插即用清理任務造成的問題。

### 執行補救指令碼

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取您要執行補救指令碼的執行個體。依序選擇 Instance state (執行個體狀態)、Stop instance (停止執行個體)。

#### Warning

停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區的資料，請將資料備份至持久性儲存。

4. 在執行個體停止後，建立備份。選取執行個體，依序選取 Actions (動作)、Image and templates (映像和範本)，然後選取 Create image (建立映像)。
5. 依序選擇 Instance state (執行個體狀態)、Start instance (啟動執行個體)。

6. 使用遠端桌面 Connect 至執行個體，然後將 RemediateDriverIssue .zip 資料夾[下載](#)至執行個體。
7. 解壓縮資料夾的內容。
8. 根據 Readme.txt 檔案中的說明執行補救指令碼。檔案位於您解 RemediateDriverIssue 壓縮 .zip 的資料夾中。

## TCP 卸載

### Important

此問題不適用於執行 AWS PV 或 Intel 網路驅動程式的執行個體。

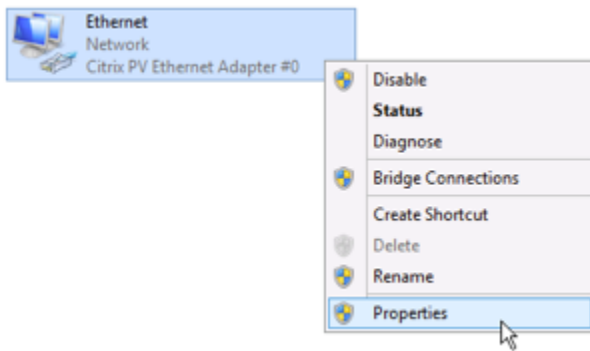
根據預設，Windows AMI 中的 Citrix PV 驅動程式會啟用 TCP 卸載。若您發生傳輸層級的錯誤或封包傳輸錯誤 (在 Windows 效能監視器中可見)—例如當您執行特定 SQL 工作負載時—您可能需要停用此功能。

### Warning

停用 TCP 卸載可能會降低您執行個體的網路效能。

## 停用 Windows Server 2012 和 2008 的 TCP 卸載

1. 連線到您的執行個體，並以本機管理員登入。
2. 若您使用 Windows Server 2012，請按 Ctrl+Esc 存取 Start (開始) 畫面，然後選擇 Control Panel (控制台)。若您使用 Windows Server 2008，請選取 Start (開始)，然後選取 Control Panel (控制台)。
3. 依序選擇 Network and Internet (網路和網際網路) 和 Network and Sharing Center (網路和共用中心)。
4. 選擇 Change adapter settings (變更介面卡設定)。
5. 以滑鼠右鍵按一下 Citrix PV Ethernet Adapter #0 (Citrix PV 乙太網路卡 #0)，然後選取 Properties (屬性)。



6. 在 Local Area Connection Properties (區域連線屬性) 對話方塊中，選擇 Configure (設定) 以開啟 Citrix PV Ethernet Adapter #0 Properties (Citrix PV 乙太網路卡 #0 屬性) 對話方塊。
7. 在 Advanced (進階) 標籤上，停用每個屬性，但 Correct TCP/UDP Checksum Value (修正 TCP/UDP 總和檢查碼值) 除外。若要停用屬性，請從 Property (屬性) 中選取，然後從 Value (值) 選取 Disabled (停用)。
8. 選擇 OK (確定)。
9. 從命令列視窗中執行下列命令。

```
netsh int ip set global taskoffload=disabled
netsh int tcp set global chimney=disabled
netsh int tcp set global rss=disabled
netsh int tcp set global netdma=disabled
```

10. 重新啟動執行個體。

## 時間同步

在 2013.02.13 Windows AMI 發行前，Citrix Xen 客體代理程式可能會不正確的設定系統時間。此可能導致您的 DHCP 租用到期。若您在連線到您的執行個體時發生問題，您可能需要更新代理程式。

若要判斷您是否已具有更新後的 Citrix Xen 客體代理程式，請檢查 C:\Program Files\Citrix\XenGuestAgent.exe 檔案是否來自 2013 年 3 月。若此檔案的日期在該日期之前，請更新 Citrix Xen 客體代理程式服務。如需詳細資訊，請參閱 [升級您的 Citrix Xen 客體代理程式服務](#)。

工作負載充分利用超過 20,000 個磁碟 IOPS，以及因 CPU 瓶頸而遭受降級的情況

如果您使用的 Windows 執行個體執行 AWS PV 驅動程式，其充分利用超過 20,000 個 IOPS，並且遇到錯誤檢查碼 0x9E: USER\_MODE\_HEALTH\_MONITOR，則可能會受此問題的影響。

AWS 光伏驅動程式中的磁碟讀取和寫入 (iOS) 分為兩個階段：IO 準備和 IO 完成。依預設，準備階段會在單一任意核心上執行。完成階段會在核心 0 上執行。處理 IO 所需的運算量因其大小和其他屬性而



有所差異。某些 iOS 在準備階段使用更多運算，其他則在完成階段如此。當執行個體驅動超過 20,000 個 IOPS 時，準備或完成階段可能會導致瓶頸，其執行所在的 CPU 佔用 100% 的容量。準備或完成階段是否成為瓶頸，取決於應用程式所使用的 IO 屬性。

從 AWS PV 驅動器 8.4.0 開始，準備階段和完成階段的負載可以分散到多個核心，從而消除瓶頸。每個應用程式使用不同的 IO 屬性。因此，套用下列其中一個組態可能會提升、降低或不會影響應用程式的效能。套用任何這些組態之後，請監控應用程式，以確認其符合您所需的效能。

## 1. 必要條件

開始此疑難排解程序之前，請先確認下列先決條件：

- 您的執行個體使用 AWS PV 驅動程式 8.4.0 或更新版本。若要升級，請參閱[升級 Windows 執行個體上的 PV 驅動程式](#)。
- 您擁有對執行個體的 RDP 存取權。如需了解使用 RDP 連線至 Windows 執行個體的步驟，請參閱[使用 RDP 用戶端 Connect 至您的 Windows 執行個體](#)。
- 您擁有執行處理的管理員存取權。

## 2. 觀察執行個體上的 CPU 負載

您可以使用 Windows 任務管理員來檢視每個 CPU 上的負載，以確定磁碟 IO 的潛在瓶頸。

1. 確認您的應用程式正在執行，並處理與生產工作負載類似的流量。
2. 使用 SSH 連線至您的執行個體。
3. 選擇執行個體上的 Start (開始) 功能表。
4. 請在 Start (開始) 功能表中輸入 Task Manager，以開啟 Task Manager (任務管理員)。
5. 如果 Task Manager (任務管理員) 顯示 Summary View (摘要檢視)，請選擇 More details (更多詳細資訊)，以展開詳細檢視。
6. 選擇 Performance (效能) 索引標籤。
7. 在左側窗格中選取 CPU。
8. 在主側窗格中的圖形上按一下滑鼠右鍵，然後選取 Change graph to (將圖形變更為) > Logical processors (邏輯處理器)，以顯示每個個別核心。
9. 根據您的執行個體上有多少核心，您可能會看到隨時間顯示 CPU 負載的行，或者可能只會看到一個數字。
  - 如果您看到隨時間顯示負載的圖形，請尋找方塊幾乎完全著色的 CPU。
  - 如果您在每個核心上看到一個數字，請尋找一致顯示 95% 以上的核心。

10 請注意核心 0 或不同的核心正在經歷繁重的負載。

### 3. 選擇要套用的組態

組態名稱	套用此組態的時間	備註
<a href="#">Default configuration</a>	工作負載驅動少於 20,000 個 IOPS，或其他組態並未改善效能或穩定性。	針對此組態，IO 會發生在幾個核心上，這可能會藉由增加快取區域性並減少內容切換，從較小工作負載獲益。
<a href="#">Allow driver to choose whether to distribute completion</a>	工作負載驅動超過 20,000 個 IOPS，核心 0 上觀察到中度或高負載。	建議使用 PV 8.4.0 或更新版本的所有 Xen 執行個體使用此組態，並且無論是否遇到問題，都可以充分利用超過 20,000 個 IOPS。
<a href="#">Distribute both preparation and completion</a>	工作負載驅動超過 20,000 個 IOPS，並且允許驅動程式選擇分發不改善效能，或是 0 之外的核心正在遇到高負載。	此組態會啟用 IO 準備和 IO 完成的分發。

#### Note

未同時分發 IO 完成 (設定 `DpcRedirection` 而不設定 `NotifierDistributed`) 的情況下，我們建議您不要分發 IO 準備，因為當準備階段平行執行時，完成階段會對準備階段過載較為敏感。

#### 登錄金鑰值

- `NotifierDistributed`

值 0 或不存在 – 完成階段將在核心 0 上執行。

值 1 – 驅動程式選擇執行完成階段或核心 0，或者每個連接磁碟有另一個核心。

值 2 – 驅動程式會在每個連接磁碟的另一個核心上執行完成階段。

- `DpcRedirection`

值 0 或不存在 – 準備階段將會在單一、任意核心上執行。

值 1 – 準備階段分發在多個核心之間。

## 預設組態

使用 8.4.0 之前的 AWS PV 驅動程式版本套用預設組態，或者如果在套用本節中的其他組態之一後發現效能或穩定性降低，則套用預設組態。

1. 使用 SSH 連線至您的執行個體。
2. 以系統管理員身分開啟新的 PowerShell 命令提示字元。
3. 執行下列命令以移除 NotifierDistributed 和 DpcRedirection 登錄金鑰。

```
Remove-ItemProperty -Path HKLM:\System\CurrentControlSet\Services\xenvbd\Parameters -Name NotifierDistributed
```

```
Remove-ItemProperty -Path HKLM:\System\CurrentControlSet\Services\xenvbd\Parameters -Name DpcRedirection
```

4. 將執行個體重新開機。

## 允許驅動程式選擇是否要分配完成

設定 NotiferDistributed 登錄金鑰，以允許 PV 儲存驅動程式選擇是否要分發 IO 完成。

1. 使用 SSH 連線至您的執行個體。
2. 以系統管理員身分開啟新的 PowerShell 命令提示字元。
3. 執行下列命令以設定 NotiferDistributed 登錄金鑰。

```
Set-ItemProperty -Type DWORD -Path HKLM:\System\CurrentControlSet\Services\xenvbd\Parameters -Value 0x00000001 -Name NotifierDistributed
```

4. 將執行個體重新開機。

## 分配準備和完成

設定 NotifierDistributed 和 DpcRedirection 登錄金鑰，以始終分發發準備和完成階段。

1. 使用 SSH 連線至您的執行個體。
2. 以系統管理員身分開啟新的 PowerShell 命令提示字元。
3. 執行下列命令以設定 NotifierDistributed 和 DpcRedirection 登錄金鑰。

```
Set-ItemProperty -Type DWORD -Path HKLM:\System\CurrentControlSet\Services\xenvbd\Parameters -Value 0x00000002 -Name NotifierDistributed
```

```
Set-ItemProperty -Type DWORD -Path HKLM:\System\CurrentControlSet\Services\xenvbd\Parameters -Value 0x00000001 -Name DpcRedirection
```

4. 將執行個體重新開機。

## AWS 適用於視窗執行個體的 NVMe

Amazon EBS 磁碟區和執行個體存放區磁碟區會以 NVMe 區塊裝置的形式公開在 [N AWS itro 系統上建置的執行個體上](#)。若要針對以 NVMe 區塊裝置形式公開的磁碟區充分利用 Amazon EBS 功能的效能和功能，執行個體必須安裝 AWS NVMe 驅動程式。根據預設，所有目前一代的 AWS 視窗 AMI 都會隨附安裝 AWS NVMe 驅動程式。

如需有關 EBS 和 NVMe 的詳細資訊，請參閱 [Amazon EBS 使用者指南中的 Amazon EBS 和 NVMe](#)。如需 SSD 執行個體存放區和 NVMe 的詳細資訊，請參閱 [SSD 執行個體存放區磁碟區](#)。

### 使用安裝或升級 AWS NVMe 驅動程式 PowerShell

如果您不是使用 Amazon 提供的最新 AWS 視窗 AMI，請使用下列程序安裝目前的 AWS NVMe 驅動程式。您應該在方便重新啟動執行個體時，執行這項更新。安裝指令碼會重新啟動您的執行個體，或是您必須重新啟動，以作為最後一個步驟。

#### 先決條件

PowerShell 3.0 或更新版本

#### 下載並安裝最新版 AWS NVMe 驅動程式

1. 我們建議您根據以下方式建立 AMI 作為備份，以防您需要復原變更。

- a. 停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。停止執行個體之前，請確認您已將所需的任何資料從執行個體存放區磁碟區複製到永久性儲存，例如 Amazon EBS 或 Amazon S3。
  - b. 在導覽窗格中，選擇 Instances (執行個體)。
  - c. 選取需要升級驅動程式的執行個體，並依序選取 Instance state (執行個體狀態)、Stop instance (停止執行個體)。
  - d. 停止執行個體後，請選取執行個體，依序選取 Actions (動作)、Image and templates (映像和範本)，然後選取 Create image (建立映像)。
  - e. 選擇 Instance state (執行個體狀態)、Start instance (啟動執行個體)。
2. 連線到您的執行個體，並以本機管理員登入。
  3. 使用下列其中一個選項下載驅動程式並解壓縮至您的執行個體：
    - 使用瀏覽器：
      - a. 將最新的驅動程式套件[下載](#)到執行個體。
      - b. 將 zip 封存檔解壓縮。
    - 使用 PowerShell：

```
Invoke-WebRequest https://s3.amazonaws.com/ec2-windows-drivers-downloads/NVMe/Latest/AWSNVMe.zip -outfile $env:USERPROFILE\nvme_driver.zip
Expand-Archive $env:userprofile\nvme_driver.zip -DestinationPath
$env:userprofile\nvme_driver
```

#### Note

如果您在下載檔案時收到錯誤訊息，而且您使用的是 Windows 伺服器 2016 或更早版本，則可能需要為您的 PowerShell 終端機啟用 TLS 1.2。您可以使用下列命令為目前 PowerShell 工作階段啟用 TLS 1.2，然後再試一次：

```
[Net.ServicePointManager]::SecurityProtocol =
[Net.SecurityProtocolType]::Tls12
```

4. 從nvme\_driver目錄(.install.ps1)執行install.ps1 PowerShell 指令碼，將驅動程式安裝到執行個體。如果出現錯誤，請確保您使用的是 PowerShell 3.0 或更高版本。

- a. (選擇性) 從 AWS NVMe 版本開始 1.5.0，小型電腦系統介面 (SCSI) 持續性保留支援 Windows 伺服器 2016 年及更新版本。此功能新增對具有共享 Amazon EBS 儲存體的 Windows Server 容錯移轉叢集的支援。依預設，安裝期間系統不會啟用此功能。

您可以在執行 `install.ps1` 指令碼以安裝驅動程式時啟用此功能，方法是將 `EnableSCSIPersistentReservations` 參數指定為 `$true` 值。

```
PS C:\> .\install.ps1 -EnableSCSIPersistentReservations $true
```

您可以在執行 `install.ps1` 指令碼以安裝驅動程式時停用此功能，方法是將 `EnableSCSIPersistentReservations` 參數指定為 `$false` 值。

```
PS C:\> .\install.ps1 -EnableSCSIPersistentReservations $false
```

- b. 從 AWS NVMe 開始 1.5.0，`install.ps1` 指令碼一律會與驅動程式一起安裝 `ebsnvme-id` 工具。

(選用) 對於版本 1.4.0、1.4.1 和 1.4.2，`install.ps1` 指令碼允許您指定 `ebsnvme-id` 工具是否應與驅動程式一起安裝。

- i. 若要安裝 `ebsnvme-id` 工具，請指定 `InstallEBSNVMeIdTool 'Yes'`。
- ii. 如果您不想安裝工具，請指定 `InstallEBSNVMeIdTool 'No'`。

如果您未指定 `InstallEBSNVMeIdTool`，並且工具已存在於 `C:\ProgramData\Amazon\Tools`，則套件預設會升級此工具。如果工具不存在，`install.ps1` 預設不會升級此工具。

如果您不想將工具安裝為套件的一部分，而且想要稍後安裝它，則可以在驅動程式套件中找到最新版本或工具。或者，您可以從 Amazon S3 下載版本 1.0.0：

[下載](#) `ebsnvme-id` 工具。

5. 如果安裝程式並未重新啟動您的執行個體，請重新啟動執行個體。

## 使用代理商安裝或升級 AWS NVMe 驅動程式

您可以使用「散發者」功能來安裝 NVMe 驅動程式套件一次或排程更新。AWS Systems Manager

1. 如需如何使用 Distributor 安裝 NVMe 驅動程式套件的指示，請參閱 Amazon EC2 Systems Manager 使用者指南中的[安裝或更新套件](#)。
2. 對於「名稱」，選擇AWSNVMe。
3. 對於安裝類型，請選擇卸載並重新安裝。
4. (選用) 透過指定 AdditionalArguments 的值來自訂安裝。
  - a. 從 AWS NVMe 開始1.5.0，此驅動程式支援視窗伺服器 2016 年及更新版本的 SCSI 持續性保留。依預設，安裝期間系統不會啟用此功能。若要啟用此功能，請將 AdditionalArguments 指定為 {"SSM\_EnableSCSIPersistentReservations": \$true}。如果您不想啟用此功能，則請將 AdditionalArguments 指定為 {"SSM\_EnableSCSIPersistentReservations": \$false}。
  - b. 從 AWS NVMe 開始1.5.0，install.ps1指令碼一律會安裝工ebsnvme-id具。

(選用) 對於版本 1.4.0、1.4.1 和 1.4.2，install.ps1 指令碼允許您指定 ebsnvme-id 工具是否應與驅動程式一起安裝。

- i. 若要安裝 ebsnvme-id 工具，請將 AdditionalArguments 指定為 {"SSM\_InstallEBSNVMeIdTool": "Yes"}。
- ii. 如果您不想安裝工具，則請將 AdditionalArguments 指定為 {"SSM\_InstallEBSNVMeIdTool": "No"}。

如果您未針對 AdditionalArguments 指定 SSM\_InstallEBSNVMeIdTool，並且工具已存在於 C:\ProgramData\Amazon\Tools，則套件預設會升級此工具。如果工具不存在，套件預設不會升級此工具。「其他引數」必須使用有效的 JSON 語法進行格式化。如需如何傳遞 aws configure 套件之其他引數的範例，請參閱 [Amazon EC2 Systems Manager 文件](#)。

如果您不想將工具安裝為套件的一部分，而且想要稍後安裝它，則可以在驅動程式套件中找到最新版本的工具。或者，您可以從 Amazon S3 下載版本 1.0.0：

[下載](#) ebsnvme-id 工具。

5. 如果安裝程式並未重新啟動您的執行個體，請重新啟動執行個體。

## 設定 SCSI 持續保留

安裝 AWS NVMe 驅動程式版本 1.5.0 或更新版本之後，您可以使用 Windows 伺服器 2016 及更新版本的 Windows 登錄來啟用或停用 SCSI 持續性保留。您必須重新啟動這些登錄檔的執行個體，變更才會生效。

您可以使用以下命令 (將 `EnableSCSIPersistentReservations` 設定為值 1)，啟用 SCSI 持續保留。

```
PS C:\> $registryPath = "HKLM:\SYSTEM\CurrentControlSet\Services\AWSNVMe\Parameters\Device"
Set-ItemProperty -Path $registryPath -Name EnableSCSIPersistentReservations -Value 1
```

您可以使用以下命令 (將 `EnableSCSIPersistentReservations` 設定為值 0)，停用 SCSI 持續保留。

```
PS C:\> $registryPath = "HKLM:\SYSTEM\CurrentControlSet\Services\AWSNVMe\Parameters\Device"
Set-ItemProperty -Path $registryPath -Name EnableSCSIPersistentReservations -Value 0
```

## AWS NVMe 驅動程式版本紀錄

下表說明 AWS NVMe 驅動程式的發行版本。

套件版本	驅動程式版本	詳細資訊	發行日期
<a href="#">1.5.1</a>	1.5.0	修正安裝指令碼以為 <code>ebsnvme-id</code> 工具建立資料夾 (如果不存在)。	2023 年 11 月 17 日
<a href="#">1.5.0</a>	1.5.0	已新增對執行 Windows Server 2016 及更新版本之執行個體的 Small Computer System Interface (SCSI) 持續保留的支援。 <code>ebsnvme-id</code> 工具 ( <code>ebsnvme-id.exe</code> ) 現在預設安裝。	2023 年 8 月 31 日
<a href="#">1.4.2</a>	1.4.2	修正 D3 執行個體上 AWS NVMe 驅動程式不支援執行個體儲存磁碟區的錯誤。	2023 年 3 月 16 日
<a href="#">1.4.1</a>	1.4.1	針對支援命名空間偏好寫入精細程度 (NPGW) 這項選用性 NVMe 功能的 EBS 磁碟區，報告命名空間偏好	2022 年 5 月 20 日



套件版本	驅動程式版本	詳細資訊	發行日期
		寫入精細程度。有關詳細資訊，請參閱 <a href="#">NVMe Base Specification, version 1.4</a> (NVMe 基本規格 1.4 版) 中的第 8.25 節 "Improving Performance through I/O Size and Alignment Adherence" (透過 I/O 大小和匹配依從性提高效能)。	
<a href="#">1.4.0</a>	1.4.0	<ul style="list-style-type: none"> <li>已新增對 iOCTLS 的支援，允許應用程式與 NVMe 設備互動。此支援可讓應用程式從 NVMe 設備中取得 IdentifyController、IdentifyNamespaces 和 NameSpace 清單。如需詳細資訊，請參閱 Microsoft 文件中的 <a href="#">通訊協定特定查詢</a>。</li> <li>AWSNVMe 在視窗伺服器 2008 R2 上安裝 1.4.0 將會失敗。AWSNVMe 視窗伺服器 2008 R2 支援版本 1.3.2 及更早版本。</li> <li>1.4.0 驅動程式版本和最新的 ebsnvme-id 工具 (ebsnvme-id.exe) 會組合在單一套件中。這種組合可讓您從單一套件同時安裝驅動程式和工具。如需詳細資訊，請參閱 <a href="#">使用安裝或升級 AWS NVMe 驅動程式 PowerShell</a>。</li> <li>錯誤修正與可靠性改進。</li> </ul>	2021 年 11 月 23 日
<a href="#">1.3.2</a>	1.3.2	修正主動處理 IO 之 EBS 磁碟區時可能導致資料毀損的問題。未修改線上 EBS 磁碟區 (例如，調整大小或變更類型) 的客戶不受影響。	2019 年 9 月 10 日
<a href="#">1.3.1</a>	1.3.1	可靠性改善。	2019 年 5 月 21 日
<a href="#">1.3.0</a>	1.3.0	裝置最佳化改善。	2018 年 8 月 31 日

套件版本	驅動程式版本	詳細資訊	發行日期
<a href="#">1.2.0</a>	1.2.0	針對所有支援執行個體 (包括裸機執行個體) 上的 AWS NVMe 裝置提升效能和可靠性。	2018 年 6 月 13 日
<a href="#">1.0.0</a>	1.0.0	AWS NVMe 驅動程式，支援執行 Windows 伺服器的執行個體類型	2018 年 2 月 12 日

## 訂閱通知

Amazon SNS 會在 EC2 Windows 驅動程式發佈新版本時通知您。使用下列程序訂閱這些通知。

從主控台訂閱 EC2 通知

1. 在 <https://console.aws.amazon.com/sns/v3/home> 開啟 Amazon SNS 主控台。
2. 如有必要，請在導覽列中將「區域」變更為美國東部 (維吉尼亞北部)。您必須選取此區域，因為您要訂閱的 SNS 通知位於本區域中。
3. 在導覽窗格中，選擇 Subscriptions (訂閱)。
4. 選擇 Create subscription (建立訂閱)。
5. 在 Create subscription (建立訂閱) 對話方塊中，執行下列動作：
  - a. 針對 TopicARN，複製下列 Amazon Resource Name (ARN)：  
`arn:aws:sns:us-east-1:801119661308:ec2-windows-drivers`
  - b. 針對 Protocol (通訊協定)，選擇 Email。
  - c. 針對 Endpoint (端點)，輸入可用於接收通知的電子郵件地址。
  - d. 選擇 Create subscription (建立訂閱)。
6. 您將會收到確認電子郵件。開啟電子郵件並遵循指示完成訂閱。

只要有新的 EC2 Windows 驅動程式發行，我們就會向訂閱者傳送通知。如果您不想再接收這些通知，請使用下列程序來取消訂閱。

取消訂閱 Amazon EC2 Windows 驅動程式通知

1. 在 <https://console.aws.amazon.com/sns/v3/home> 開啟 Amazon SNS 主控台。

2. 在導覽窗格中，選擇 Subscriptions (訂閱)。
3. 選取訂閱的核取方塊，然後選取 Actions (動作)，Delete subscriptions (刪除訂閱)。出現確認提示時，請選擇 Delete (刪除)。

若要使用訂閱 EC2 通知 AWS CLI

若要使用訂閱 EC2 通知 AWS CLI，請使用下列命令。

```
aws sns subscribe --topic-arn arn:aws:sns:us-east-1:801119661308:ec2-windows-drivers --
protocol email --notification-endpoint YourUserName@YourDomainName.ext
```

若要使用訂閱 EC2 通知 AWS Tools for Windows PowerShell

若要使用訂閱 EC2 通知 AWS Tools for Windows PowerShell，請使用下列命令。

```
Connect-SNSNotification -TopicArn 'arn:aws:sns:us-east-1:801119661308:ec2-windows-
drivers' -Protocol email -Region us-east-1 -Endpoint 'YourUserName@YourDomainName.ext'
```

## 設定您的 Windows 執行個體

啟動 Windows 執行個體之後，您可以以系統管理員身分登入，以針對啟動代理程式和 Windows 特定功能執行其他設定。下列主題著重於 Windows 執行個體組態。

### 目錄

- [設定 Amazon EC2 視窗執行個體的啟動設定](#)
- [為您的執行個體使用 EC2 快速啟動](#)
- [在 Windows 執行個體上使用 Amazon 彈性圖形加速器](#)
- [在 Windows 執行個體上安裝 WSL](#)

## 設定 Amazon EC2 視窗執行個體的啟動設定

Amazon EC2 啟動代理程式會在執行個體啟動期間執行任務，並在執行個體停止、稍後啟動或重新啟動時執行。如需有關特定代理程式的資訊，請參閱下列清單中的詳細資訊頁面。

- [使用 EC2Launch v2 設定 Windows 執行個體](#)

- [使用 EC2Launch 設定 Windows 執行個體](#)
- [使用 EC2Config 服務 \(舊版\) 設定 Windows 執行個體](#)

## 內容

- [比較 Amazon EC2 啟動代理](#)
- [設定 Windows 啟動代理程式的 DNS 尾碼](#)

## 比較 Amazon EC2 啟動代理

下表顯示了 EC2Config、EC2Launch v1 和 EC2Launch v2 之間的主要功能差異。

功能	EC2Config	EC2Launch v1	EC2Launch v2
Run as (執行身分)	Windows 服務	PowerShell 腳本	Windows 服務
支援	僅限舊版作業系統	Windows 2016 Windows 2019 (LTSC 和 SAC)	Windows 2016 Windows 2019 (LTSC 和 SAC) Windows 2022
組態檔案	XML	XML	YAML
設定管理員使用者名稱	否	否	是
使用者資料大小	16 KB	16 KB	60 KB (壓縮)
在 AMI 上預製的本機使用者資料	否	否	是，不可設定。
使用者資料中的任務組態	否	否	是
可設定桌布	否	否	是
自訂任務執行順序	否	否	是

功能	EC2Config	EC2Launch v1	EC2Launch v2
可設定的工作	15	9	啟動時為 20 個
支援 Windows 事件檢視器	是	否	是
事件檢視器事件類型的數目	2	0	30

### Note

EC2Config 文件僅供歷史參考之用。它執行的作業系統版本已不再受到 Microsoft 的支援。強烈建議您升級至最新的啟動服務。

## 設定 Windows 啟動代理程式的 DNS 尾碼

使用 Amazon EC2 啟動代理程式，您可以設定 Windows 執行個體用於網域名稱解析的 DNS 尾碼清單。啟動代理程式會覆寫 System\CurrentControlSet\Services\Tcpip\Parameters\SearchList 登錄機碼中的標準 Windows 設定，方法是將下列值新增至 DNS 尾碼搜尋清單：

- 執行個體的網域
- 執行個體網域下放所產生的尾碼
- NV 域名
- 每張網路介面卡所指定的網域

所有啟動代理程式都支援 DNS 尾碼組態。如需詳細資訊，請參閱您的特定啟動代理程式版本：

- 如需 setDnsSuffix 工作以及如何在 EC2Launch v2 中設定 DNS 尾碼的相關資訊，請參閱 [setDnsSuffix](#)
- 如需 DNS 尾碼清單設定，以及如何啟用或停用 EC2Launch v1 的遞增功能的相關資訊，請參閱 [設定 EC2Launch](#)
- 如需 DNS 尾碼清單設定以及如何啟用或停用 EC2Config 的遞增功能的相關資訊，請參閱 [EC2Config 設定檔](#)

## 域名下放

網域名稱遞減是一種 Active Directory 行為，允許子網域中的電腦存取父網域中的資源，而無需使用完整網域名稱。根據預設，網域名稱下放會繼續執行，直到網域名稱進度中只剩下兩個節點為止。

如果執行個體已連線至網域，Launch 代理程式會對網域名稱執行遞減，並將結果新增至 **System\CurrentControlSet\Services\Tcpip\Parameters\SearchList** 登錄機碼中維護的 DNS 尾碼搜尋清單。代理程式會使用下列登錄機碼中的設定來判斷遞增行為。

- **System\CurrentControlSet\Services\Tcpip\Parameters\UseDomainNameDevolution**
  - 如果沒有設置，禁用下放
  - 設定為時1，啟用下積分 (預設)
  - 設定為時0，停用下放
- **System\CurrentControlSet\Services\Dnscache\Parameters\DomainNameDevolutionLevel**
  - 未設定時，請使用圖層 2 (預設值)
  - 當設定為3或更大時，使用值設定等級

當您停用遞增功能或將遞增設定變更為較高層級時，System\CurrentControlSet\Services\Tcpip\Parameters\SearchList 登錄機碼樣式會包含先前新增的尾碼。它們不會自動移除。您可以手動更新清單，也可以清除清單，讓代理程式執行程序以設定新清單。

### Note

若要清除登錄中的 DNS 尾碼清單，您可以執行下列命令。

```
PS C:\> Invoke-CimMethod -ClassName Win32_NetworkAdapterConfiguration -
Methodname "SetDNSSuffixSearchOrder" -Arguments @{ DNSDomainSuffixSearchOrder =
$null } | Out-Null
```

## 下放示例

下列範例顯示網域名稱在下放過程中的進展情況。

corp.example.com

- 進展到 example.com

locale.region.corp.example.com

1. 進展到 region.corp.example.com
2. 進展到 corp.example.com
3. 進展到 example.com

locale.region.corp.example.com與設置 DomainNameDevolutionLevel=3

1. 進展到 region.corp.example.com
2. 進展到。corp.example.com由於等級設定，進度會在此停止。

## 使用 EC2Launch v2 設定 Windows 執行個體

預設情況下，執行 Windows Server 2022 的所有支援的 Amazon EC2 執行個體都包括 EC2Launch v2 啟動代理程式 (EC2Launch.exe)。我們亦提供將 EC2Launch v2 安裝為預設啟動代理程式的 Windows Server 2016 和 2019 AMI。除了包含 EC2Launch v1 的 Windows Server 2016 和 2019 AMI 之外，亦提供了這些 AMI。您可以從 Amazon EC2 主控台內的 AMI 頁面，在搜尋中輸入以下字首來搜尋預設包含 EC2Launch v2 的 Windows AMI：EC2LaunchV2-Windows\_Server-\*。

EC2Launch v2 在執行個體啟動期間執行任務，並在執行個體停止並稍後啟動或重新啟動時執行。EC2Launch v2 也可依據需求執行任務。其中部分作業為自動啟用，其他作業則必須手動啟用。EC2Launch v2 服務支援所有 EC2Config 和 EC2Launch 功能。

此服務會使用組態檔來控制其操作。您可以使用圖形化工具或直接將組態檔編輯為單一 .yml 檔案 (agent-config.yml) 來更新組態檔。服務二進位檔案位於 %ProgramFiles%\Amazon\EC2Launch 目錄中。

EC2Launch v2 會發佈 Windows 事件日誌，協助您疑難排解錯誤並設定觸發程序。如需詳細資訊，請參閱 [Windows 事件日誌](#)。

### 支援的作業系統

- Windows Server 2022
- Windows Server 2019 (長期服務通道和半年通道)

- [Windows Server 2016](#)

## EC2Launch v2 部分內容

- [EC2Launch v2 概觀](#)
- [安裝最新版本的 EC2Launch v2](#)
- [遷移到 EC2Launch v2](#)
- [停止、重新啟動、刪除或解除安裝 EC2Launch v2](#)
- [訂閱 EC2Launch v2 服務通知](#)
- [EC2Launch v2 設定](#)
- [EC2Launch v2 故障診斷](#)
- [EC2Launch v2 版本歷程記錄](#)

## EC2Launch v2 概觀

EC2Launch v2 是一種服務，可在執行個體啟動期間執行任務，並在執行個體停止並稍後啟動或重新啟動時執行。

### 概觀主題

- [EC2Launch v2 概念](#)
- [EC2Launch v2 任務](#)
- [遙測](#)

若要比較啟動代理程式版本功能，請參閱[比較 Amazon EC2 啟動代理](#)。

## EC2Launch v2 概念

考慮 EC2Launch v2 時，下列概念有助於理解。

### 任務

您可以調用任務以在執行個體上執行動作。您可以在 `agent-config.yml` 檔案中或透過使用者資料來設定任務。如需 EC2Launch v2 可用任務的清單，請參閱 [EC2Launch v2 任務](#)。如需了解任務組態結構描述和詳細資訊，請參閱 [EC2Launch v2 任務組態](#)。

### 階段



階段是 EC2Launch v2 代理程式所執行任務的邏輯群組。某些工作只能在特定階段中執行。其他人可以在多個階段運行。使用 `agent-config.yml` 時，必須指定階段清單，以及要在每個階段中執行的任務清單。

此服務會以下列順序執行各階段：

階段 1：開機

階段 2：網路

第三階段：PreReady

Windows 已準備就緒

PreReady 階段完成後，服務會將 `Windows is ready` 訊息傳送到 Amazon EC2 主控台。

第四階段：PostReady

使用者資料會在 PostReady 階段中執行。有些指令碼版本會在 `agent-config.yml` 檔案 PostReady 階段之前執行，有些版本會在之後執行，如下所示：

`agent-config.yml` 之前

- YAML 使用者資料版本 1.1
- XML 使用者資料

`agent-config.yml` 之後

- YAML 使用者資料 1.0 版 (舊版供回溯相容性之用)

如需階段和任務的範例，請參閱 [範例：agent-config.yml](#)。

使用使用者資料時，您必須指定代理程式要執行的任務清單。此階段是隱含的。如需任務範例，請參閱 [範例：使用者資料](#)。

EC2Launch v2 會依照您在 `agent-config.yml` 和使用者資料中指定的順序來執行任務清單。階段按順序執行。下一個階段會在前一個階段完成之後開始。任務也會依序執行。

頻率

任務頻率確定何時執行任務，這取決於開機內容。大多數任務只有一個允許的頻率。您可以指定 `executeScript` 任務的頻率。

您將在 [EC2Launch v2 任務組態](#) 中看到以下頻率。

- 一次 – 當 AMI 已初次啟動 (完成 Sysprep)，執行任務一次。
- 始終 – 每次啟動代理程式執行時，任務都會執行。啟動代理程式會執行於下列時間：
  - 執行個體啟動或重新啟動
  - 執行 EC2Launch 服務
  - EC2Launch.exe run 被叫用

## agent-config

agent-config 是位於 EC2Launch v2 組態資料夾中的檔案。其中包括開機、PreReady網路和 PostReady 階段的組態。此檔案用於指定執行個體組態，設定應在 AMI 第一次啟動或後續啟動時執行的任務。

依預設，EC2Launch v2 安裝會安裝一個 agent-config 檔案，它包含標準 Amazon Windows AMI 中使用的推薦組態。您可以更新該組態檔案，以變更 EC2Launch v2 指定的 AMI 的預設開機經驗。

## 使用者資料

使用者資料是啟動執行個體時可設定的資料。您可以更新使用者資料，以動態變更自訂 AMI 或快速入門 AMI 的設定方式。EC2Launch v2 支援 60 kB 的使用者資料輸入長度。使用者資料僅包含 UserData 階段，因此會在 agent-config 檔案之後執行。您可以在使用啟動執行個體精靈啟動執行個體時輸入使用者資料，也可以從 EC2 主控台修改使用者資料。如需處理使用者資料的資訊，請參閱 [Amazon EC2 如何處理 Windows 執行個體的使用者資料](#)。

## EC2Launch v2 任務

EC2Launch v2 可以在每次開機時執行下列任務：

- 設定新的和選擇性的自訂桌布來呈現執行個體的相關資訊。
- 設定在本機電腦上建立的系統管理員帳戶的屬性。
- 將 DNS 尾碼新增至搜尋尾碼清單。只有不存在的尾碼才會加入到清單中。
- 為任何額外的磁碟區設定磁碟機代號，並將它們延伸至使用可用空間。
- 將檔案從配置寫入磁碟。
- 執行 EC2Launch v2 組態檔或從中指定的指令碼。user-data 腳本 user-data 可以是純文本或壓縮，並以 base64 格式提供。
- 執行具有指定參數的程式。

- 設定電腦名稱。
- 將執行個體資訊傳送到 Amazon EC2 主控台。
- 傳送 RDP 憑證指紋到 Amazon EC2 主控台。
- 動態擴充作業系統的分割區，以包含任何尚未分割的空間。
- 執行使用者資料。如需指定使用者資料的詳細資訊，請參閱[EC2Launch v2 任務組態](#)。
- 設定非持久性靜態路由以到達中繼資料服務和 AWS KMS 伺服器。
- 將非開機分割區設定為mbr或gpt。
- 在 Sysprep 後啟動 Systems Manager 服務。
- 最佳化 ENA 設定。
- 為更新的 Windows 版本啟用開啟 SSH。
- 啟用巨型訊框。
- 將 Sysprep 設定為在 EC2Launch v2 中執行。
- 發佈 Windows 事件日誌。

## 遙測

遙測是其他資訊，可協助您進一步瞭解您的需求、診斷問題並提供可改善使用體驗的功能 AWS 服務。

EC2Launch v2 版本 2.0.592 及更新版本會收集遙測，例如用量指標和錯誤。這些資料是從執行 EC2Launch v2 的 Amazon EC2 執行個體中收集的資料。這包括所有的視窗 AMI 擁有。AWS

EC2Launch v2 會收集下列類型的遙測：

- 用量資訊 – 代理程式命令、安裝方法和排程的執行頻率。
- 錯誤和診斷資訊 — 代理程式安裝錯誤代碼、執行錯誤代碼和錯誤呼叫堆疊。

收集資料的範例：

```
2021/07/15 21:44:12Z: EC2LaunchTelemetry: IsAgentScheduledPerBoot=true
2021/07/15 21:44:12Z: EC2LaunchTelemetry: IsUserDataScheduledPerBoot=true
2021/07/15 21:44:12Z: EC2LaunchTelemetry: AgentCommandCode=1
2021/07/15 21:44:12Z: EC2LaunchTelemetry: AgentCommandErrorCode=5
2021/07/15 21:44:12Z: EC2LaunchTelemetry: AgentInstallCode=2
```

```
2021/07/15 21:44:12Z: EC2LaunchTelemetry: AgentInstallErrorCode=0
```

預設為啟用遙測。您可隨時停用遙測收集。如果啟用遙測，EC2Launch v2 會傳送遙測資料，而不會傳送其他通知給客戶。

### 遙測可見性

啟用遙測時，會在 Amazon EC2 主控台輸出中顯示，如下所示。

```
2021/07/15 21:44:12Z: Telemetry: <Data>
```

### 停用執行個體的遙測

若要停用單一執行個體的遙測，您可以設定系統環境變數，或使用 MSI 來修改安裝。

若要藉由設定系統環境變數來停用遙測，請以管理員身分執行下列命令。

```
setx /M EC2LAUNCH_TELEMETRY 0
```

若要使用 MSI 來停用遙測，請在[下載 MSI](#) 後執行以下命令。

```
msiexec /i ".\AmazonEC2Launch.msi" Remove="Telemetry" /q
```

### 安裝最新版本的 EC2Launch v2

您可使用以下其中一種方法，在 EC2 執行個體上安裝 EC2Launch v2 代理程式：

- 從 Amazon S3 下載代理程式並使用視窗進行安裝 PowerShell。如需下載 URL，請參閱 [在 Amazon S3 上下載 EC2Launch v2](#)。
- 透過 SSM Distributor 進行安裝。
- 從 EC2 Image Builder 元件進行安裝。
- 從已預先安裝 EC2Launch v2 的 AMI 啟動執行個體。

#### Warning

AmazonEC2Launch.msi 會解除安裝先前版本的 EC2 啟動服務，例如 EC2Launch (v1) 和 EC2Config。

如需了解安裝步驟，請選取符合您偏好方法的索引標籤。

## Windows PowerShell

若要使用 Windows 安裝最新版本的 EC2Launch v2 代理程式 PowerShell，請依照下列步驟執行。

1. 建立您的本機目錄。

```
New-Item -Path "$env:USERPROFILE\Desktop\EC2Launchv2" -ItemType Directory
```

2. 為您的下載位置設定 URL。使用您將使用的 Amazon S3 URL 執行以下命令。如需下載 URL，請參閱 [在 Amazon S3 上下載 EC2Launch v2](#)

```
$Url = "Amazon S3 URL/AmazonEC2Launch.msi"
```

3. 使用以下複合命令來下載和安裝代理程式：

```
$DownloadFile = "$env:USERPROFILE\Desktop\EC2Launchv2\" + $(Split-Path -Path $Url -Leaf)
Invoke-WebRequest -Uri $Url -OutFile $DownloadFile
msiexec /i "$DownloadFile"
```

### Note

如果您在下載檔案時收到錯誤訊息，而且您使用的是 Windows 伺服器 2016 或更早版本，則可能需要為您的 PowerShell 終端機啟用 TLS 1.2。您可以使用下列命令為目前 PowerShell 工作階段啟用 TLS 1.2，然後再試一次：

```
[Net.ServicePointManager]::SecurityProtocol =
[Net.SecurityProtocolType]::Tls12
```

4. 若要驗證安裝情況，請檢查 msi 檔案是否存在於您執行個體的 EC2Launch v2 目錄中 (C:\ProgramData\Amazon\EC2Launch)。

## AWS Systems Manager Distributor

若要使用 AWS Systems Manager 快速設定來設定 EC2Launch v2 的自動更新，請參閱 [使用經銷商快速設定自動安裝和更新](#)

您也可以從「AWS Systems Manager 代理商」執行AWSEC2Launch-Agent套件的一次性安裝。如需有關如何從 Systems Manager Distributor 安裝套件的指示，請參閱《AWS Systems Manager 使用者指南》中的[安裝或更新套件](#)。

## EC2 Image Builder component

您可以在使用 EC2 Image Builder 建置自訂映像時，安裝 ec2launch-v2-windows 元件。如需如何使用 EC2 Image Builder 建置自訂映像的指示，請參閱 EC2 Image Builder 使用者指南中的[使用 EC2 Image Builder 主控台精靈建立映像管道](#)。

## AMI

EC2Launch v2 預設會預先安裝在以下 Windows Server 2022 和 UEFI AMI 上：

- Windows\_Server-2022-English-Full-Base
- Windows\_Server-2022-English-Core-Base
- 所有其他語言的 Windows Server 2022 AMI
- 安裝 SQL 的 Windows Server 2022 AMI
- Windows\_Server-2022-English-Core-EKS\_Optimized

EC2Launch v2 也已預先安裝在下列 Windows Server AMI 上。您可以在 Amazon EC2 主控台找到這些 AMI，或使用下列搜尋字首：AWS CLI 中的 EC2LaunchV2-。

- EC2LaunchV2-Windows\_Server-2019-English-Core-Base
- EC2LaunchV2-Windows\_Server-2019-English-Full-Base
- EC2LaunchV2-Windows\_Server-2016-English-Core-Base
- EC2LaunchV2-Windows\_Server-2016-English-Full-Base
- EC2LaunchV2-Windows\_Server-2012\_R2\_RTM-English-Full-Base
- EC2LaunchV2-Windows\_Server-2012\_RTM-English-Full-Base

使用「AWS Systems Manager 代理商快速設定」自動安裝和更新 EC2Launch v2

透過「AWS Systems Manager 代理商快速設定」，您可以設定 EC2Launch v2 的自動更新。下列程序會在您的執行個體上設定 Systems Manager 關聯，以您指定的頻率自動更新 EC2Launch v2 代理程式。「代理商快速設定」所建立的「關聯」可以包含 AWS 帳戶和區域內的執行環境，或是 AWS 組織內的執行環境。如需有關設定組織的詳細資訊，請參閱《AWS Organizations 使用者指南》中的[〈教學課程：建立和設定組織〉](#)。

開始之前，請確定您的執行個體符合所有先決條件。

## 必要條件

若要使用「代理商快速設定」設定自動更新，您的執行個體必須符合下列先決條件。

- 您至少有一個支援 EC2Launch v2 的執行中執行個體。請參閱支援的作業系統[EC2Launch v2](#)。
- 您已在執行個體上執行 Systems Manager 設定工作。如需詳細資訊，請參閱《AWS Systems Manager 使用指南》中的〈[設定 Systems Manager](#)〉。
- EC2Launch v2 必須是您執行個體上唯一安裝的啟動代理程式。如果您安裝了一個以上的啟動代理程式，您的「代理商快速設定」組態將會失敗。使用散發商快速設定來設定 EC2Launch v2 之前，請先解除安裝 EC2Config 或 EC2Launch v1 啟動代理程式 (如果有的話)。

## 設定 EC2 啟動 v2 的代理商快速設定

若要使用「代理商快速設定」建立 EC2Launch v2 的組態，請在完成代理[商套件](#)部署的步驟時使用下列設定：

- 軟體套件：Amazon EC2 啟動 v2 代理程式。
- 更新頻率：從清單中選取頻率。
- 目標：從可用的部署選項中選擇。

若要檢查組態的狀態，請瀏覽至中的「Systems Manager 快速設定組態」標籤 AWS Management Console。

1. 開啟主 AWS Systems Manager 控制台，[網址為 https://console.aws.amazon.com/systems-manager/](https://console.aws.amazon.com/systems-manager/)。
2. 在導覽窗格中，選擇快速設定。
3. 在組態索引標籤中，選擇與您建立的模型組態相關聯的列。「組態」索引標籤會列出您的組態，並包含重要詳細資料摘要，例如「區域」、「部署狀態」和「關聯狀態」。

### Note

每個 EC2Launch v2 代理商組態的關聯名稱都會以下列前置詞開頭：AWS-QuickSetup-Distributor-EC2Launch-Agent-

4. 若要檢視詳細資訊，請選取組態並選擇檢視詳細資料。

如需詳細資訊和疑難排解步驟，請參閱AWS Systems Manager 使用指南中的[疑難排解快速設定結果](#)。

在 Amazon S3 上下載 EC2Launch v2

若要安裝最新版本 EC2Launch v2，請從下列其中一個位置下載安裝程式：

#### Note

我們即將汰除 32 位元的安裝連結。建議您使用 64 位元的安裝連結來安裝 EC2Launch v2。如果您需要 32 位元的啟動代理程式，請使用 [EC2Config](#)。

- 64 位元 - <https://s3.amazonaws.com/amazon-ec2launch-v2/windows/amd64/latest/AmazonEC2Launch.msi>
- 32 位元— <https://s3.amazonaws.com/amazon-ec2launch-v2/windows/386/latest/AmazonEC2Launch.msi>

## 設定安裝選項

在安裝或升級 EC2Launch v2 時，您可以使用 EC2Launch v2 安裝對話方塊或命令列 shell 中的 `msiexec` 命令來設定安裝選項。

首次在執行個體上執行 EC2Launch v2 安裝程式時，該安裝程式會初始化執行個體上的啟動代理程式設定，如下所示：

- 安裝程式會建立本機路徑，並將啟動代理程式檔案寫入該路徑。這有時被稱為全新安裝。
- 安裝程式會建立 `EC2LAUNCH_TELEMETRY` 環境變數 (若尚無)，並根據您的組態進行設定。

如需有關設定的詳細資訊，請選取符合您將使用的設定方法的索引標籤。

## Amazon EC2Launch Setup dialog

在安裝或升級 EC2Launch v2 時，您可以透過 EC2Launch v2 安裝對話方塊來設定以下安裝選項。

### 基本安裝選項

#### 傳送遙測

在您將此功能納入安裝對話方塊中時，安裝程式會將 `EC2LAUNCH_TELEMETRY` 環境變數的值設定為 1。若您停用傳送遙測，安裝程式會將環境變數設定為 0。



在 EC2Launch v2 代理程式執行時，其會讀取 EC2LAUNCH\_TELEMETRY 環境變數，決定是否要上傳遙測資料。若值等於 1，則會上傳資料。若不是，則不會上傳。

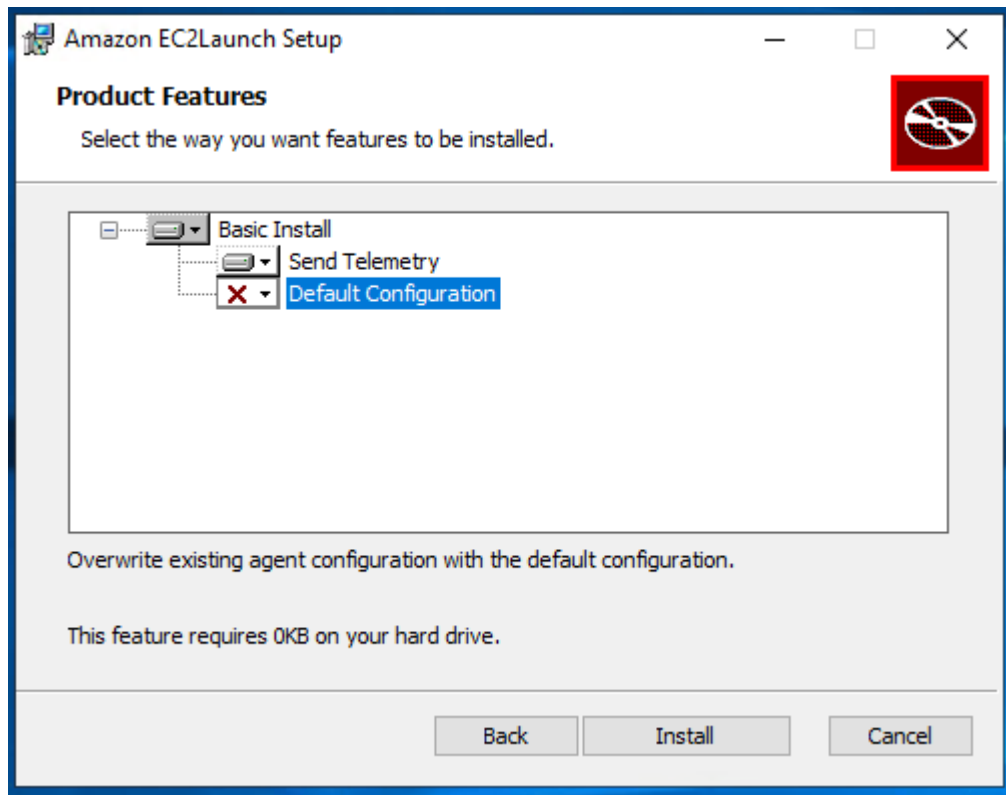
### 預設組態

EC2Launch v2 的預設組態為覆寫本機啟動代理程式 (若已具備)。首次在執行個體上執行安裝時，預設組態會執行全新安裝。若您在初始安裝時停用預設組態，則安裝會失敗。

若再次在執行個體上執行安裝，您可以停用預設組態，執行不會取代 %ProgramData%/Amazon/EC2Launch/config/agent-config.yml 檔案的升級。

### 範例：使用遙測來升級 EC2Launch v2

以下範例會顯示設定為升級目前安裝及啟用遙測的 EC2Launch v2 設定對話方塊。此組態會在不取代代理程式組態檔案的情況下執行安裝，並將 EC2LAUNCH\_TELEMETRY 環境變數設定為 1。



### Command line

在安裝或升級 EC2Launch v2 時，您可以使用命令列 shell 中的 msixexec 命令來設定以下安裝選項。

## ADDLOCAL 參數值

### 基本 (必填)

安裝啟動代理程式。如果 ADDLOCAL 參數中不存在此值，則安裝結束。

### 全新

當您在 ADDLOCAL 參數中納入 Clean 值時，安裝程式會將代理程式組態檔案寫入以下位置：`%ProgramData%/Amazon/EC2Launch/config/agent-config.yml`。如果代理程式組態檔案已存在，則會覆寫檔案。

當您將 Clean 值放在 ADDLOCAL 參數之外時，安裝程式會執行不會取代代理程式組態檔案的升級。

### 遙測

當您在 ADDLOCAL 參數中納入 Telemetry 值時，安裝程式會將 EC2LAUNCH\_TELEMETRY 環境變數設為 1。

當您將 Telemetry 值放在 ADDLOCAL 參數之外時，安裝程式會將環境變數設為 0。

在 EC2Launch v2 代理程式執行時，其會讀取 EC2LAUNCH\_TELEMETRY 環境變數，決定是否要上傳遙測資料。若值等於 1，則會上傳資料。若不是，則不會上傳。

### 範例：使用遙測安裝 EC2Launch v2

```
& msixexec /i "C:\Users\Administrator\Desktop\EC2Launchv2\AmazonEC2Launch.msi"
ADDLOCAL="Basic,Clean,Telemetry" /q
```

## 確認 EC2Launch v2 版本

請用下列程序之一確認執行個體上安裝的 EC2Launch v2 版本。

### Windows PowerShell

使用視窗驗證已安裝的 EC2 啟動 v2 版本 PowerShell，如下所示。

1. 從 AMI 啟動執行個體，並連線至執行個體。
2. 在中執行下列命令，PowerShell 以驗證 EC2Launch v2 的已安裝版本：

```
& "C:\Program Files\Amazon\EC2Launch\EC2Launch.exe" version
```

## Windows Control Panel

在 Windows 控制台中確認已安裝的 EC2Launch v2 版本，如下所示。

1. 從 AMI 啟動執行個體，並連線至執行個體。
2. 開啟 Windows 控制台，然後選擇程式和功能。
3. 從已安裝程式清單中，找出 Amazon EC2Launch。其版本編號將出現在 Version (版本) 欄中。

若要檢視視窗 AMI 的最新更新，請參閱視 AWS 窗 AMI 參考資料中的視 AWS 窗 AMI 版本歷史記錄。

如需 EC2Launch v2 的最新版本，請參閱 [EC2Launch v2 版本歷史記錄](#)。

如需 EC2Launch v2 遷移工具的最新版本，請參閱 [EC2Launch v2 遷移工具版本歷史記錄](#)。

您會在 EC2Launch v2 服務有新版本發佈時收到通知。如需詳細資訊，請參閱 [訂閱 EC2Launch v2 服務通知](#)。

## 遷移到 EC2Launch v2

EC2Launch 遷移工具會先解除安裝已安裝的啟動代理程式 (EC2Config 和 EC2Launch v1)，再安裝 EC2Launch v2，來進行升級。先前啟動服務的適用組態會自動移轉至新服務。遷移工具不會偵測任何連結至 EC2Launch v1 指令碼的排程任務；因此，不會自動在 EC2Launch v2 中設定這些任務。若要設定這些任務，請編輯 [agent-config.yml](#) 檔案，或使用 [EC2Launch v2 設定對話方塊](#)。例如，如果執行個體有執行 InitializeDisks.ps1 的排程任務，則執行遷移工具後，您必須在 EC2Launch v2 設定對話方塊中指定要初始化的磁碟區。請參閱 [使用 EC2Launch v2 設定對話方塊變更設定](#) 程序的步驟 6。

您可以下載移轉工具或使用 SSM RunCommand 文件進行安裝。

您可以從下列位置下載此工具：

### Note

我們即將汰除 32 位元的遷移工具連結。建議您使用 64 位元的連結來遷移至 EC2Launch v2。如果您需要 32 位元的啟動代理程式，請使用 [EC2Config](#)。

- 64 位 — <https://s3.amazonaws.com/amazon-ec2launch-v2-utils/ MigrationTool /視窗/安排4/最新/ EC2 Tool.zip LaunchMigration>
- 32 位 — <https://s3.amazonaws.com/amazon-ec2launch-v2-utils/ MigrationTool /視窗/386/最新/EC2 Tool.zip LaunchMigration>

### Note

您必須以管理員身分執行 EC2Launch v2 遷移工具。在您執行遷移工具之後，EC2Launch v2 會安裝為服務。此工具不會立即執行。依預設，工具會在執行個體啟動期間執行，並在執行個體停止後啟動或重新啟動時執行。

使用 [AWSEC2Launch-RunMigration](#) SSM 文件可以透過 SSM 執行命令遷移至最新 EC2Launch v2 版本。文件不需要任何參數。如需有關使用 SSM 執行命令的詳細資訊，請參閱 [AWS Systems Manager 執行命令](#)。

遷移工具會將 EC2Config 中的下列設定套用至 EC2Launch v2。

- 如果 `Ec2DynamicBootVolumeSize` 設定為 `false`，請移除 EC2Launch v2 boot 階段
- 如果 `Ec2SetPassword` 設定為 `Enabled`，請將 EC2Launch v2 密碼類型設定為 `random`
- 如果 `Ec2SetPassword` 設定為 `Disabled`，請將 EC2Launch v2 密碼類型設定為 `doNothing`
- 如果 `SetDnsSuffixList` 設定為 `false`，請移除 EC2Launch v2 `setDnsSuffix` 任務
- 如果 `EC2SetComputerName` 設定為 `true`，將 EC2Launch v2 `setHostName` 任務新增至 `yaml` 組態

遷移工具會將 EC2Launch v1 中的下列設定套用至 EC2Launch v2。

- 如果 `ExtendBootVolumeSize` 設定為 `false`，請移除 EC2Launch v2 boot 階段
- 如果 `AdminPasswordType` 設定為 `Random`，請將 EC2Launch v2 密碼類型設定為 `random`
- 如果將 `AdminPasswordType` 設定為 `Specify`，則將 EC2Launch v2 密碼類型設定為 `static`，並將密碼資料設定為 `AdminPassword` 中指定的密碼
- 如果 `SetWallpaper` 設定為 `false`，請移除 EC2Launch v2 `setWallpaper` 任務
- 如果 `AddDnsSuffixList` 設定為 `false`，請移除 EC2Launch v2 `setDnsSuffix` 任務
- 如果 `SetComputerName` 設定為 `true`，請新增 EC2Launch v2 `setHostName` 任務

## 停止、重新啟動、刪除或解除安裝 EC2Launch v2

您可以像管理任何其他 Windows 服務一樣管理 EC2Launch v2 服務。

EC2Launch v2 在開機時執行一次，並執行所有設定的任務。執行任務後，服務會進入停止狀態。當您重新啟動服務時，服務會再次執行所有已設定的任務，並回到已停止的狀態。

若要將更新設定套用至執行個體，您可停止並重新啟動服務。如果您手動安裝 EC2Launch v2，則必須先停止服務。

### 若要停 EC2Launch v2 服務

1. 啟動和連線至 Windows 執行個體。
2. 在開始功能表上，選擇系統管理工具，然後開啟服務。
3. 在服務清單中，用滑鼠右鍵按一下 Amazon EC2Launch，然後選取停止。

### 重新啟動 EC2Launch v2 服務

1. 啟動和連線至 Windows 執行個體。
2. 在開始功能表上，選擇系統管理工具，然後開啟服務。
3. 在服務清單中，用滑鼠右鍵按一下 Amazon EC2Launch，然後選取重新啟動。

如果不需要更新設定、建立您自己的 AMI 或使用 AWS Systems Manager，您可刪除並解除安裝服務。移除服務將移除其登錄子機碼。解除安裝服務將移除檔案、登錄子金鑰和任何服務捷徑。

### 刪除 EC2Launch v2 服務

1. 開啟命令提示字元視窗。
2. 執行以下命令：

```
sc delete EC2Launch
```

### 解除安裝 EC2Launch v2

1. 啟動和連線至 Windows 執行個體。
2. 在 Start (開始) 功能表上，選擇 Control Panel (控制台)。
3. 依次開啟 Programs (程式) 和 Programs and Features (程式和功能)。

4. 在程式清單中，選擇 Amazon EC2Launch。若要確認您選擇的是 v2，請檢查 Version (版本) 資料行。
5. 選擇解除安裝。

### 訂閱 EC2Launch v2 服務通知

Amazon SNS 會在 EC2Launch v2 服務發佈新版本時通知您。使用下列程序訂閱這些通知。

### 訂閱 EC2Launch v2 通知

1. 登入 AWS Management Console 並開啟 Amazon SNS 主控台，網址為 <https://console.aws.amazon.com/sns/v3/home>。
2. 如有必要，請在導覽列中將「區域」變更為美國東部 (維吉尼亞北部)。您必須選取此區域，因為您要訂閱的 SNS 通知已在本區域中建立完成。
3. 在導覽窗格中，選擇 Subscriptions (訂閱)。
4. 選擇 Create subscription (建立訂閱)。
5. 在 Create subscription (建立訂閱) 對話方塊中，執行下列動作：
  - a. 對於主題 ARN，請使用以下 Amazon Resource Name (ARN) `arn:aws:sns:us-east-1:309726204594:amazon-ec2launch-v2`。
  - b. 對於 通訊協定，選擇 電子郵件。
  - c. 針對 Endpoint (端點)，請輸入可用於接收通知的電子郵件地址。
  - d. 選擇 Create subscription (建立訂閱)。
6. 您會收到請求確認訂閱的電子郵件。開啟電子郵件並遵循指示完成訂閱。

每當 EC2Launch v2 服務有新版本發佈時，我們會將通知傳送給訂閱者。如果您不想再接收這些通知，請使用下列程序來取消訂閱。

1. 開啟 Amazon SNS 主控台。
2. 在導覽窗格中，選擇 Subscriptions (訂閱)。
3. 選取訂閱，然後選取 Actions (動作)、Delete subscriptions (刪除訂閱)。出現確認提示時，請選擇 Delete (刪除)。

### EC2Launch v2 設定


本節包含有關如何進行 EC2Launch v2 設定的資訊。

主題包括：

- [使用 EC2Launch v2 設定對話方塊變更設定](#)
- [EC2Launch v2 目錄結構](#)
- [使用 CLI 設定 EC2Launch v2](#)
- [EC2Launch v2 任務組態](#)
- [EC2Launch v2 結束程式碼和重新啟動](#)
- [EC2Launch v2 和 Sysprep](#)

使用 EC2Launch v2 設定對話方塊變更設定

下列程序說明如何使用 EC2Launch v2 對話方塊啟用或停用設定。

 Note

如果您在 `agent-config.yml` 檔案中不當設定自訂任務，並嘗試開啟 Amazon EC2Launch 設定對話方塊，您將收到錯誤。有關架構範例，請參閱 [範例：agent-config.yml](#)。

1. 啟動和連線至 Windows 執行個體。
2. 從「開始」功能表選擇所有程式，然後瀏覽至 EC2Launch 設定。

### Amazon EC2Launch settings ✕

General | DNS suffix | Wallpaper | Volumes

**Set computer name**

- Set the computer name of the instance
- Set to "ip-<hex private IPv4 address>"
- Use custom name
- Reboot after setting computer name

**Extend boot volume**

- Extend OS partition to use free space for boot volume

**Set administrator account**

- Set administrator account
- Administrator username (leave blank for default)
- Administrator password settings
  - Random (retrieve from console)
  - Specify (temporarily stored in configuration file)
  - Do not set

**Start SSM service**

- Re-enable and start SSM service after Sysprep

**Optimize ENA**

- Optimize receive side scaling and receive queue depth

**Enable SSH**

- Enable OpenSSH for later Windows versions

**Enable Jumbo Frames**

- Enable Jumbo Frames
- Important: Do not enable Jumbo Frames if you are not familiar with them

**Prepare for imaging**



### 3. 在 Ec2 服務屬性對話方塊的一般索引標籤中，啟用或停用下列設定。

#### a. Set Computer Name (設定電腦名稱)

如果啟用此設定 (預設為停用)，則會在每次開機時將目前主機名稱與所需主機名稱進行比較。如果主機名稱不相符，則會重設主機名稱，然後系統會選擇性地重新開機以取得新的主機名稱。如果未指定自訂主機名稱，則會使用十六進位格式的私有 IPv4 地址產生名稱，例如：ip-AC1F4E6。若要避免現有的主機名稱遭到修改，請勿啟用此設定。

#### b. 延伸開機磁碟區

此設定將動態擴充 Disk 0/Volume 0，以包含任何尚未分割的空間。此設定在執行個體是從自訂大小的根裝置磁碟區啟動時很有用。

#### c. 設定管理員帳戶

啟用後，您可以為本機電腦上建立的管理員帳戶設定使用者名稱和密碼屬性。如果未啟用此功能，遵循 Sysprep 的系統不會建立系統管理員帳戶。只有在 adminPassword 是 adminPasswordtype 時，才要在 Specify 中提供密碼。

密碼類型的定義如下：

##### i. Random

EC2Launch 會產生密碼並以使用者金鑰將其加密。系統會在執行個體啟動後停用此設定，以便執行個體重新開機或停止再啟動時，能持續使用這個密碼。

##### ii. Specify

EC2Launch 會使用您在 adminPassword 中指定的密碼。如果該密碼不符合系統要求，EC2Launch 會改為產生隨機密碼。系統會將該密碼以純文字存放於 agent-config.yml，並在 Sysprep 設定管理員密碼後予以刪除。EC2Launch 會以使用者金鑰加密該密碼。

##### iii. Do not set

EC2Launch 會使用您在未使用的 .xml 檔案中指定的密碼。如果您未在 unattend.xml 中指定密碼，會停用管理員帳戶。

#### d. 啟動 SSM 服務

選取此選項時，Systems Manager 服務會啟用，以啟動以下 Sysprep。EC2Launch v2 將執行[先前](#)所述的所有任務，而且 SSM Agent 將處理 Systems Manager 功能的請求，例如執行命令和狀態管理員。

您可使用執行命令升級現有執行個體，以使用 EC2Launch v2 服務和 SSM Agent 的最新版本。如需詳細資訊，請參閱《AWS Systems Manager 使用者指南》中的[使用執行命令更新 SSM Agent](#)。

e. 最佳化 ENA

選取時，ENA 設定會設定為確保 ENA 接收端調整和接收佇列深度設定已針對最佳化 AWS。如需詳細資訊，請參閱[設定 RSS CPU 親和性](#)。

f. 啟用 SSH

此設定會啟用適用於較新版本 Windows 的 OpenSSH，以允許遠端系統管理。

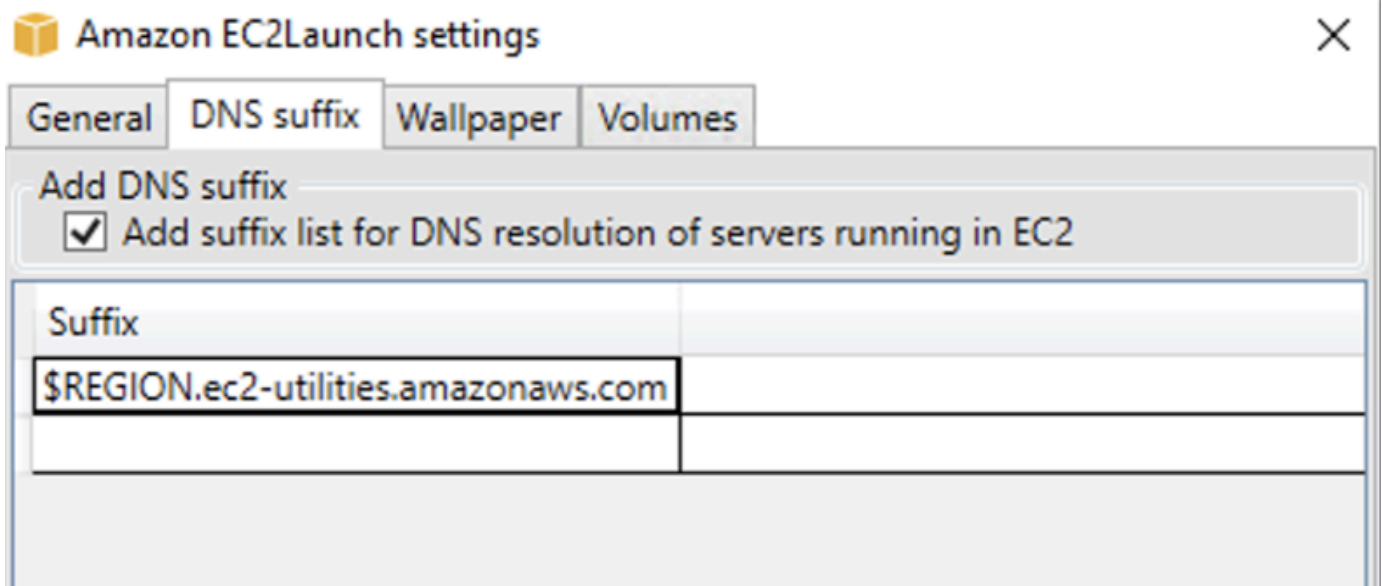
g. 啟用巨型訊框

選取以啟用「巨型訊框」。「巨型訊框」可能會對您的網路通訊產生意想不到的影響，因此在啟用之前，請務必了解「巨型訊框」對您的系統的影響。如需「巨型訊框」的詳細資訊，請參閱[巨型訊框 \(9001 MTU\)](#)。

h. 準備建立映像

選取是否希望 EC2 執行個體在使用 Sysprep 的情況下關閉。當您想要使用 EC2Launch v2 執行 Sysprep 時，請選擇 Shutdown with Sysprep (使用 Sysprep 關機)。


4. 在 DNS 尾碼標籤上，您可以選取是否要為 EC2 中執行之伺服器的 DNS 解析新增 DNS 尾碼清單，而不需提供完整網域名稱。DNS 尾碼可以包含變數 \$REGION 和 \$AZ。只有不存在的尾碼才會新增到清單中。



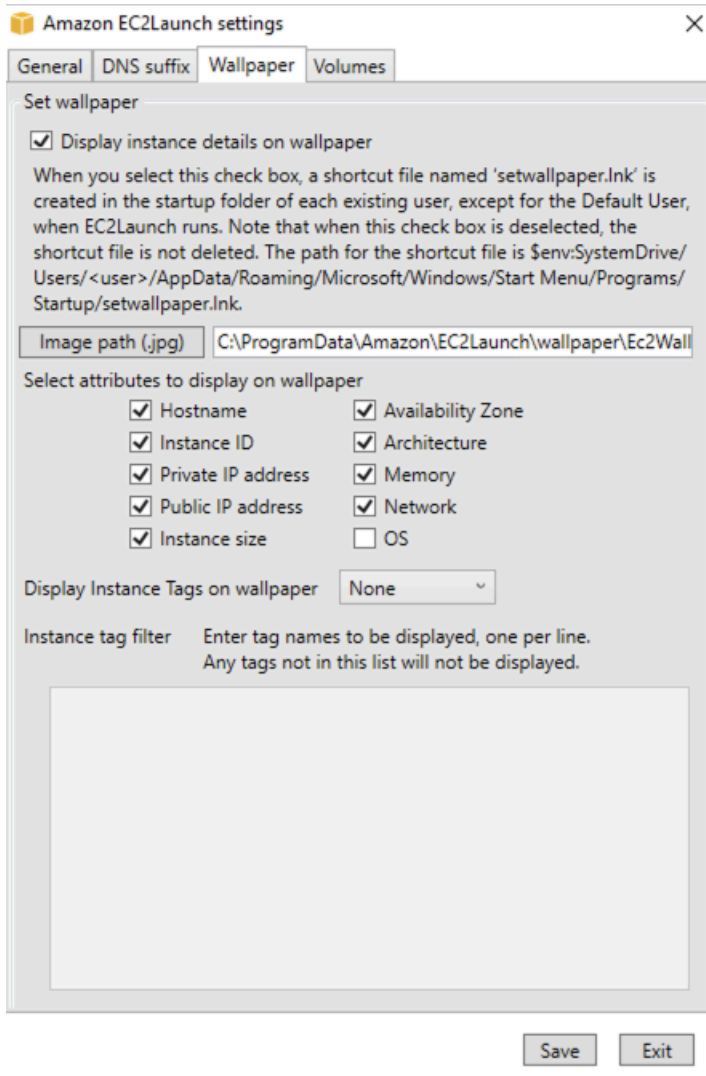
5. 在桌布索引標籤中，您可以使用背景圖片來設定執行個體桌布，並指定要顯示之桌布的執行個體詳細資訊。Amazon EC2 會在您每次登入時產生詳細資訊。

您可以使用下列控制項來設定桌布。

- 在桌布上顯示執行個體詳細資訊 - 此核取方塊會啟用或停用桌布上的執行個體詳細資訊顯示。
- 圖片路徑 (.jpg) - 指定要用作桌布背景的图片路徑。
- 選取要在桌布上顯示的屬性 - 選取要顯示在桌布上之執行個體詳細資訊的核取方塊。清除要從桌布中移除的先前選取的執行個體詳細資訊的核取方塊。
- 在桌布上顯示執行個體標籤 - 選取下列其中一個設定以在桌布上顯示執行個體標籤：
  - 無 - 不在桌布上顯示任何執行個體標籤。
  - 顯示全部 - 在桌布上顯示所有執行個體標籤。
  - 顯示篩選結果 - 在桌布上顯示指定的執行個體標籤。當您選取此設定時，可以在執行個體標籤篩選方塊中新增您想要顯示在桌布上的執行個體標籤。

 Note

您必須在中繼資料中啟用標籤，才能在桌布上顯示標籤。如需有關執行個體標籤和中繼資料的詳細資訊，請參閱 [使用執行個體中繼資料中的執行個體標籤](#)。



- 在磁碟區標籤上，選取是否要初始化連線至執行個體的磁碟區。啟用會設定任何額外磁碟區的磁碟機代號，並將它們延伸到使用可用空間。如果您選取全部，則會初始化所有儲存磁碟區。如果您選取裝置，則只會初始化清單中指定的裝置。您必須為每個要初始化的裝置輸入裝置。使用 EC2 主控台上列出的裝置，例如，`xvdb` 或 `/dev/nvme0n1`。下拉式清單會顯示連接到執行個體的儲存磁碟區。若要輸入未連接到執行個體的裝置，請在文字欄位中輸入該裝置。

名稱、字母和分割區是選填欄位。如果未指定分割區的值，則會以磁碟分割類型初始化大於 2 TB 的儲存磁碟 gpt 區，而小於 2 TB 的儲存磁碟區會以磁碟 mbr 分割類型初始化。如果已設定裝置，且非 NTFS 裝置包含磁碟分割區表格，或磁碟的前 4 KB 會包含資料，接著系統會略過磁碟並記錄動作。

## Amazon EC2Launch settings



- General
- DNS suffix
- Wallpaper
- Volumes

### Initialize volumes

Initialize     All     Devices

### Devices

If you choose Devices, only the devices listed below are initialized. You must enter the Device for each device to be initialized. Use the devices listed on the EC2 console, for example, xvdb or /dev/nvme0n1. Name, Letter, and Partition are optional.

Device	Name	Letter	Partition
--------	------	--------	-----------

以下是從 EC2Launch 對話方塊中輸入的設定建立的範例組態 YAML 檔案。

```
version: 1.0
config:
 - stage: boot
tasks:
 - task: extendRootPartition
 - stage: preReady
 tasks:
 - task: activateWindows
 inputs:
 activation:
 type: amazon
 - task: setDnsSuffix
 inputs:
 suffixes:
 - $REGION.ec2-utilities.amazonaws.com
 - task: setAdminAccount
 inputs:
 password:
 type: random
 - task: setWallpaper
 inputs:
 path: C:\ProgramData\Amazon\EC2Launch\wallpaper\Ec2Wallpaper.jpg
 attributes:
 - hostName
 - instanceId
 - privateIpAddress
 - publicIpAddress
 - instanceSize
 - availabilityZone
 - architecture
 - memory
 - network
 - stage: postReady
 tasks:
 - task: startSsm
```

## EC2Launch v2 目錄結構

EC2Launch v2 應該安裝在以下目錄中：

- 服務二進位檔案： %ProgramFiles%\Amazon\EC2Launch

- 服務資料 (設定、日誌檔和狀態檔) : %ProgramData%\Amazon\EC2Launch

#### Note

根據預設，Windows 會將檔案和資料夾隱藏在 C:\ProgramData 下。若要檢視 EC2Launch v2 目錄和檔案，您必須在 Windows 檔案總管中輸入路徑或變更資料夾屬性，才可顯示隱藏的檔案和資料夾。

%ProgramFiles%\Amazon\EC2Launch 目錄包含二進位檔案和支援的程式庫。它包括以下子目錄：

- settings
  - EC2LaunchSettingsUI.exe – 用於修改 agent-config.yml 檔案的使用者介面
  - YamldotNet.dll – 用於支援使用者介面中的某些操作的 DLL
- tools
  - ebsnvme-id.exe – 用於檢查執行個體上 EBS 磁碟區工具
  - AWSAcpiSpcrReader.exe – 用於確定要使用的正確 COM 連接埠工具
  - EC2LaunchEventMessage.dll – DLL，用於支援 EC2 啟動的 Windows 事件記錄。
- service
  - EC2LaunchService.exe – 啟動代理程式以服務方式執行時，所啟動的 Windows 服務可執行檔。
- EC2Launch.exe – 主 EC2Launch 可執行檔
- EC2LaunchAgentAttribution.txt – EC2 Launch 中使用的代碼屬性

%ProgramData%\Amazon\EC2Launch 目錄包含下列子目錄。服務產生的所有資料，包括日誌、組態和狀態，都會儲存在此目錄中。

- config – 組態

服務組態檔案儲存在此目錄中為 agent-config.yml。此檔案可以更新，以修改、新增或移除服務所執行的預設任務。在此目錄中建立檔案的許可限制為系統管理員帳戶，以防止權限提升。

- log – 執行個體日誌

服務 (agent.log)、主控台 (console.log)、效能 (bench.log) 和錯誤 (error.log) 的日誌會儲存在此目錄中。日誌會在後續執行服務時附加。

- [state](#) – 服務狀態資料

服務用來判斷應該執行哪些任務的狀態會儲存在此處。有一個 `.run-once` 檔案會指出服務是否已在 Sysprep 之後執行 (因此，下次執行時會略過頻率為一次的任務)。此子目錄包含 `state.json` 和 `previous-state.json` 以追蹤每個任務的狀態。

- [sysprep](#) – Sysprep

此目錄包含的檔案，可用來判斷 Sysprep 在建立可重複使用的自訂 Windows AMI 時要執行哪些作業。

## 使用 CLI 設定 EC2Launch v2

您可以使用命令列介面 (CLI) 來配置 EC2Launch 設定並管理服務。下節包含您可用來管理 EC2Launch v2 之 CLI 命令的說明和使用資訊。

### 命令

- [collect-logs](#)
- [get-agent-config](#)
- [list-volumes](#)
- [重設](#)
- [run](#)
- [status](#)
- [sysprep](#)
- [validate](#)
- [version](#)
- [桌布](#)

### collect-logs

收集 EC2Launch 的日誌、壓縮檔案，並將它們置於指定的目錄中。

### 範例

```
ec2launch collect-logs -o C:\Mylogs.zip
```

### 用途



```
ec2launch collect-logs [flags]
```

## Flags

```
-h, --help
```

collect-logs 的說明

```
-o, --output string
```

壓縮輸出日誌的路徑

```
get-agent-config
```

以指定的格式 (JSON 或 YAML) 列印 agent-config.yml。如果未指定任何格式，則會以先前指定的格式列印 agent-config.yml。

## 範例

```
ec2launch get-agent-config -f json
```

## 範例 2

下面的 PowerShell 命令顯示如何編輯和保存 JSON 格式的 agent-config 文件。

```
$config = & "$env:ProgramFiles/Amazon/EC2Launch/EC2Launch.exe" --format json |
 ConvertFrom-Json
$jumboFrame =@"
{
 "task": "enableJumboFrames"
}
"@
$config.config | %{if($_.stage -eq 'postReady'){$_tasks += (ConvertFrom-Json -
 InputObject $jumboFrame)}}
$config | ConvertTo-Json -Depth 6 | Out-File -encoding UTF8
$env:ProgramData/Amazon/EC2Launch/config/agent-config.yml
```

## 用途

```
ec2launch get-agent-config [flags]
```

## Flags

```
-h, --help
```

## get-agent-config 的說明

-f, --format string

agent-config 檔案的輸出格式：json、yaml

## list-volumes

列出連接到執行個體的所有儲存磁碟區，包括暫時磁碟區和 EBS 磁碟區。

## 範例

```
ec2launch list-volumes
```

## 用途

```
ec2launch list-volumes
```

## Flags

-h, --help

## list-volumes 的說明

## 重設

此任務的主要目標是在下次代理程式執行時重設該代理程式。為此，reset 命令會從本機 EC2Launch 目錄刪除 EC2Launch v2 的所有代理程式狀態資料 (請參閱：[EC2Launch v2 目錄結構](#))。選擇性重設會刪除服務和 Sysprep 日誌。

指令碼行為取決於代理程式執行指令碼的模式 (內嵌或分離)。

## 內嵌 (預設)

EC2Launch v2 代理程式會一次執行一個指令碼 (detach: false)。這是預設設定。

### Note

當內嵌指令碼發出 reset 或 sysprep 命令時，它會立即執行並重設代理程式。目前任務完成後，代理程式就會關閉，而不執行任何後續任務。

例如，如果發出命令的任務後緊接著一個 startSsm 任務 (預設情況下會在使用者資料執行之後包含)，則任務不會執行，而且 Systems Manager 服務永遠不會啟動。

## Detached

EC2Launch v2 代理程式會與其他任務同時執行指令碼 (`detach: true`)。

### Note

當分離的指令碼發出 `reset` 或 `sysprep` 命令時，這些命令會等待代理程式完成再執行。`executeScript` 之後的任務仍會執行。

## 範例

```
ec2launch reset -c
```

## 用途

```
ec2launch reset [flags]
```

## Flags

`-c, --clean`

在 `reset` 前清除執行個體日誌

`-h, --help`

`reset` 的說明

`run`

執行 EC2Launch v2。

## 範例

```
ec2launch run
```

## 用途

```
ec2launch run [flags]
```

## Flags

`-h, --help`

## run 的說明

### status

取得 EC2Launch v2 代理程式的狀態。選擇性地封鎖處理程序，直到代理程式完成為止。處理程序結束程式碼會決定代理程式狀態：

- 0 – 代理程式已執行且成功。
- 1 – 代理程式已執行且失敗。
- 2 – 代理程式仍在執行中。
- 3 – 代理程式處於未知狀態。代理程式狀態為未執行或已停止。
- 4 – 嘗試擷取代理程式狀態時發生錯誤。
- 5 – 代理程式未執行，且上次已知執行的狀態未知。這可能表示以下其中一項：
  - `state.json` 和 `previous-state.json` 均被刪除。
  - `previous-state.json` 已毀損。

這是執行 [reset](#) 命令後的代理程式狀態。

範例：

```
ec2launch status -b
```

### 用途

```
ec2launch status [flags]
```

### Flags

`-b, --block`

封鎖處理程序，直到代理程式完成執行

`-h, --help`

### status 的說明

### sysprep

此任務的主要目標是在下次代理程式執行時重設該代理程式。為此，`sysprep` 命令會重設代理程式狀態、更新 `unattend.xml` 檔案、停用 RDP，然後執行 Sysprep。

指令碼行為取決於代理程式執行指令碼的模式 (內嵌或分離)。

### 內嵌 (預設)

EC2Launch v2 代理程式會一次執行一個指令碼 (`detach: false`)。這是預設設定。

#### Note

當內嵌指令碼發出 `reset` 或 `sysprep` 命令時，它會立即執行並重設代理程式。目前任務完成後，代理程式就會關閉，而不執行任何後續任務。

例如，如果發出命令的任務後緊接著一個 `startSsm` 任務 (預設情況下會在使用者資料執行之後包含)，則任務不會執行，而且 Systems Manager 服務永遠不會啟動。

### Detached

EC2Launch v2 代理程式會與其他任務同時執行指令碼 (`detach: true`)。

#### Note

當分離的指令碼發出 `reset` 或 `sysprep` 命令時，這些命令會等待代理程式完成再執行。`executeScript` 之後的任務仍會執行。

### 範例：

```
ec2launch sysprep
```

### 用途

```
ec2launch sysprep [flags]
```

### Flags

```
-c,--clean
```

在 `sysprep` 前清除執行個體日誌

```
-h,--help
```

### Sysprep 說明

`-s,--shutdown`

在 `sysprep` 後關閉執行個體

`validate`

驗證 `agent-config` 檔案 `C:\ProgramData\Amazon\EC2Launch\config\agent-config.yml`。

範例

```
ec2launch validate
```

用途

```
ec2launch validate [flags]
```

Flags

`-h, --help`

`validate` 的說明

`version`

取得可執行版本。

範例

```
ec2launch version
```

用途

```
ec2launch version [flags]
```

Flags

`-h, --help`

`version` 的說明

## 桌布

將新桌布設定為提供的桌布路徑 (.jpg 檔案) , 並顯示選取的執行個體詳細資料。

## 語法

```
ec2launch wallpaper ^
--path="C:\ProgramData\Amazon\EC2Launch\wallpaper\Ec2Wallpaper.jpg" ^
--all-tags ^
--
attributes=hostName,instanceId,privateIpAddress,publicIpAddress,instanceSize,availabilityZone,a
```

## 輸入

### 參數

`--allowed-tags` [***tag-name-1, tag-name-n***]

(選用) 要顯示在桌布上的 Base64 編碼 JSON 執行個體標籤名稱陣列。您可以使用此標籤或 `--all-tags` , 但不能同時使用兩者。

`--attributes` ***attribute-string-1, attribute-string-n***

(選用) 要將設定套用至桌布的 wallpaper 屬性字串清單 (以逗號分隔)。

`[--path | -p]` ***path-string***

(必要) 指定 wallpaper 背景圖片檔案路徑。

## 旗標

`--all-tags`

(選用) 在桌布上顯示所有執行個體標籤。您可以使用此標籤或 `--allowed-tags` , 但不能同時使用兩者。

`[--help | -h]`

顯示 wallpaper 命令的說明。

## EC2Launch v2 任務組態

本節包含 `agent-config.yml` 和使用者資料的組態結構描述任務、詳細資訊及範例。

## 任務與範例

- [結構描述：agent-config.yml](#)
- [結構描述：使用者資料](#)
- [任務定義](#)

### 結構描述：agent-config.yml

agent-config.yml 檔案的結構如下所示。請注意，不能在同一階段重複任務。有關任務屬性，請參閱後面的任務說明。

文件結構：agent-config.yml

### JSON

```
{
 "version": "1.0",
 "config": [
 {
 "stage": "string",
 "tasks": [
 {
 "task": "string",
 "inputs": {
 ...
 }
 },
 ...
]
 },
 ...
]
}
```

### YAML

```
version: 1.0
config:
- stage: string
 tasks:
 - task: string
 inputs:
```



```
...
...
...
```

## 範例：agent-config.yml

下列範例顯示 agent-config.yml 組態檔的設定。

```
version: 1.0
config:
- stage: boot
 tasks:
 - task: extendRootPartition
- stage: preReady
 tasks:
 - task: activateWindows
 inputs:
 activation:
 type: amazon
 - task: setDnsSuffix
 inputs:
 suffixes:
 - $REGION.ec2-utilities.amazonaws.com
 - task: setAdminAccount
 inputs:
 password:
 type: random
 - task: setWallpaper
 inputs:
 path: C:\ProgramData\Amazon\EC2Launch\wallpaper\Ec2Wallpaper.jpg
 attributes:
 - hostName
 - instanceId
 - privateIpAddress
 - publicIpAddress
 - instanceSize
 - availabilityZone
 - architecture
 - memory
 - network
- stage: postReady
 tasks:
 - task: startSsm
```

## 結構描述：使用者資料

下列 JSON 和 YAML 範例顯示使用者資料的文件結構。Amazon EC2 會剖析您在文件中指定的 tasks 陣列中命名的每個任務。每個任務都有自己的屬性集和要求集。如需詳細資訊，請參閱 [任務定義](#)。

### Note

任務只能在使用者資料任務陣列中出現一次。

## 文件結構：使用者資料

### JSON

```
{
 "version": "1.1",
 "tasks": [
 {
 "task": "string",
 "inputs": {
 ...
 },
 },
 ...
]
}
```

### YAML

```
version: 1.1
tasks:
- task: string
 inputs:
 ...
...
```

## 範例：使用者資料

如需使用者資料的詳細資訊，請參閱 [Amazon EC2 如何處理 Windows 執行個體的使用者資料](#)。

下列 YAML 文件範例會顯示 EC2Launch v2 以使用者資料的身分執行的 PowerShell 指令碼，以建立檔案。

```

version: 1.1
tasks:
- task: executeScript
 inputs:
 - frequency: always
 type: powershell
 runAs: localSystem
 content: |-
 New-Item -Path 'C:\PowerShellTest.txt' -ItemType File

```

針對與舊版啟動代理程式相容的使用者資料，您可以使用 XML 格式。EC2Launch v2 會將指令碼作為 UserData 階段中的 executeScript 任務執行。為了符合 EC2Launch v1 和 EC2Config 行為，預設情況下使用者資料指令碼會作為連接/內嵌的程序執行。

您可以新增選用標籤來自訂指令碼的執行方式。例如，除了在執行個體啟動時執行一次使用者資料指令碼之外，若還想在執行個體重新啟動時執行，您可以使用下列標記：

```
<persist>true</persist>
```

範例：

```

<powershell>
 $file = $env:SystemRoot + "\Temp" + (Get-Date).ToString("MM-dd-yy-hh-mm")
 New-Item $file -ItemType file
</powershell>
<persist>true</persist>

```

您可以使用 <powershellArguments> 標籤指定一個或多個 PowerShell 引數。如果未傳遞任何引數，EC2Launch v2 預設會新增下列引數：。-ExecutionPolicy Unrestricted

範例：

```

<powershell>
 $file = $env:SystemRoot + "\Temp" + (Get-Date).ToString("MM-dd-yy-hh-mm")
 New-Item $file -ItemType file
</powershell>
<powershellArguments>-ExecutionPolicy Unrestricted -NoProfile -NonInteractive</
powershellArguments>

```

若要將 XML 使用者資料指令碼作為分離的程序執行，請將下列標籤新增至您的使用者資料。

```
<detach>>true</detach>
```

範例：

```
<powershell>
 $file = $env:SystemRoot + "\Temp" + (Get-Date).ToString("MM-dd-yy-hh-mm")
 New-Item $file -ItemType file
</powershell>
<detach>>true</detach>
```

### Note

舊版啟動代理程式不支援分離標籤。

變更日誌：使用者資料

下表列出使用者資料的變更，並將其交叉參考至適用的 EC2Launch v2 代理程式版本。

使用者資料版本	詳細資訊	引入於
1.1	<ul style="list-style-type: none"> <li>使用者資料任務會在代理程式組態檔案中的 PostReady 階段之前執行。</li> <li>在啟動 Systems Manager Agent 之前執行使用者資料 (與 EC2Launch v1 和 EC2Config 的行為相同)。<sup>*</sup></li> </ul>	EC2Launch v2 2.0.1245 版
1.0	<ul style="list-style-type: none"> <li>將被棄用。</li> <li>使用者資料任務會在代理程式組態檔案中的 PostReady 階段之後執行。這不能與 EC2Launch v1 回溯相容。</li> <li>受 Systems Manager Agent 啟動與使用者資料任務之間的競爭條件影響。</li> </ul>	EC2Launch v2 2.0.0 版

<sup>\*</sup> 與預設 agent-config.yml 檔案搭配使用時。

## 任務定義

每個任務都有自己的屬性集和要求集。如需詳細資訊，請參閱您要在文件中包含的個別任務。

### 任務

- [activateWindows](#)
- [enableJumboFrames](#)
- [enableOpenSsh](#)
- [executeProgram](#)
- [executeScript](#)
- [extendRootPartition](#)
- [initializeVolume](#)
- [optimizeEna](#)
- [集 AdminAccount](#)
- [setDnsSuffix](#)
- [setHostName](#)
- [setWallpaper](#)
- [startSsm](#)
- [sysprep](#)
- [writeFile](#)

### activateWindows

針對一組 AWS KMS 伺服器啟動視窗。如果偵測到執行個體為自帶授權 (BYOL)，則會略過啟動。

Frequency — 一次

AllowedStages — [PreReady]

Inputs —

activation : (映射)

type : (字串) 要使用的啟動類型，設定為 amazon

### 範例

```
task: activateWindows
inputs:
 activation:
 type: amazon
```

## enableJumboFrames

啟用「巨型訊框」，以增加網路介面卡的最大傳輸單位 (MTU)。如需詳細資訊，請參閱 [巨型訊框 \(9001 MTU\)](#)。

Frequency – 始終

AllowedStages — [PostReady, UserData]

Inputs — 無

### 範例

```
task: enableJumboFrames
```

## enableOpenSsh

啟用 Windows OpenSSH，並將執行個體的公開金鑰新增至授權的金鑰資料夾。

Frequency — 一次

AllowedStages — [PreReady, UserData]

Inputs — 無

### 範例

以下範例示範如何在執行個體上啟用 OpenSSH，以及如何將執行個體的公有金鑰新增至授權的金鑰資料夾。此組態僅適用於執行 Windows Server 2019 及更新版本的執行個體。

```
task: enableOpenSsh
```

## executeProgram

執行具有可選引數和指定頻率的程式。

階段：您可以在 PreReady、PostReady 和 UserData 階段執行 executeProgram 任務。

頻率：可設定，請參閱輸入。

## 輸入

您可以設定執行期參數，如下所示：

頻率 (字串)

(必要) 僅指定下列其中一個值：

- once
- always

路徑 (字串)

(必要) 要執行的可執行檔的檔案路徑。

引數 (字串清單)

(選用) 以逗號分隔的引數清單，作為輸入提供給程式。

runAs (字串)

(必要) 必須設定為 `localSystem`

## 輸出

所有任務都會將日誌檔案條目寫入 `agent.log` 檔案。executeProgram 任務的其他輸出會分別儲存在動態命名的資料夾中，如下所示：

```
%LocalAppData%\Temp\EC2Launch#####\outputfilename.tmp
```

輸出檔案的確切路徑包含在 `agent.log` 檔案中，例如：

```
Program file is created at: C:\Windows\system32\config\systemprofile\AppData\Local
\Temp\EC2Launch123456789\ExecuteProgramInputs.tmp
Output file is created at: C:\Windows\system32\config\systemprofile\AppData\Local
\Temp\EC2Launch123456789\Output.tmp
Error file is created at: C:\Windows\system32\config\systemprofile\AppData\Local
\Temp\EC2Launch123456789\Err.tmp
```

**executeProgram** 任務的輸出檔案

**ExecuteProgramInputs.tmp**

包含可執行檔的路徑，以及 executeProgram 任務在執行時傳送給其所有輸入參數。

## Output.tmp

包含 executeProgram 任務執行之程式的執行期輸出。

## Err.tmp

包含 executeProgram 任務執行之程式的執行期錯誤訊息。

## 範例

下列範例說明如何透過 executeProgram 任務，從執行個體上的本機目錄執行可執行檔。

### 範例 1：具有一個引數的設定可執行檔

此範例顯示在靜默模式下執行設定可執行檔的 executeProgram 任務。

```
task: executeProgram
inputs:
- frequency: always
 path: C:\Users\Administrator\Desktop\setup.exe
 arguments: ['-quiet']
```

### 範例 2：具有兩個引數的 VLC 可執行檔

此範例顯示執行 VLC 可執行檔的 executeProgram 任務，其中兩個引數作為輸入參數傳遞。

```
task: executeProgram
inputs:
- frequency: always
 path: C:\vlc-3.0.11-win64.exe
 arguments: ['/L=1033', '/S']
runAs: localSystem
```

## executeScript

執行具有可選引數和指定頻率的指令碼。指令碼行為取決於代理程式執行指令碼的模式 (內嵌或分離)。

### 內嵌 (預設)

EC2Launch v2 代理程式會一次執行一個指令碼 (detach: false)。這是預設設定。



**Note**

當內嵌指令碼發出 `reset` 或 `sysprep` 命令時，它會立即執行並重設代理程式。目前任務完成後，代理程式就會關閉，而不執行任何後續任務。

例如，如果發出命令的任務後緊接著一個 `startSsm` 任務 (預設情況下會在使用者資料執行之後包含)，則任務不會執行，而且 Systems Manager 服務永遠不會啟動。

**Detached**

EC2Launch v2 代理程式會與其他任務同時執行指令碼 (`detach: true`)。

**Note**

當分離的指令碼發出 `reset` 或 `sysprep` 命令時，這些命令會等待代理程式完成再執行。`executeScript` 之後的任務仍會執行。

**階段：**您可以在 `PreReady`、`PostReady` 和 `UserData` 階段執行 `executeScript` 任務。

**頻率：**可設定，請參閱輸入。

**輸入**

您可以設定執行期參數，如下所示：

**頻率 (字串)**

(必要) 僅指定下列其中一個值：

- `once`
- `always`

**類型 (字串)**

(必要) 僅指定下列其中一個值：

- `batch`
- `powershell`

**引數 (字串清單)**

(選用) 傳遞給 shell 的字串引數清單。`type: batch` 不支援此參數。如果未傳遞任何引數，EC2Launch v2 預設會新增下列引數：`-ExecutionPolicy Unrestricted`

## 內容 (字串)

(必要) 指令碼內容。

## runAs (字串)

(必要) 僅指定下列其中一個值：

- admin
- localSystem

## 分離 (布林值)

(選用) EC2Launch v2 代理程式預設為一次執行一個指令碼 (`detach: false`)。若要與其他任務同時執行指令碼，請將值設定為 `true` (`detach: true`)。

### Note

當 `detach` 設定為 `true` 時，指令碼結束代碼 (包括 3010) 沒有任何作用。

## 輸出

所有任務都會將日誌檔案條目寫入 `agent.log` 檔案。executeScript 任務執行之指令碼的其他輸出會分別儲存在動態命名的資料夾中，如下所示：

```
%LocalAppData%\Temp\EC2Launch#####\outputfilename.ext
```

輸出檔案的確切路徑包含在 `agent.log` 檔案中，例如：

```
Program file is created at: C:\Windows\system32\config\systemprofile\AppData\Local
\Temp\EC2Launch123456789\UserScript.ps1
Output file is created at: C:\Windows\system32\config\systemprofile\AppData\Local
\Temp\EC2Launch123456789\Output.tmp
Error file is created at: C:\Windows\system32\config\systemprofile\AppData\Local
\Temp\EC2Launch123456789\Err.tmp
```

## executeScript 任務的輸出檔案

### UserScript.ext

包含 executeScript 任務執行的指令碼。檔案的副檔名取決於您在 executeScript 任務的 `type` 參數中指定的指令碼類型，如下所示：

- 如果類型為 batch，則檔案副檔名為 .bat。
- 如果類型為 powershell，則檔案副檔名為 .ps1。

### Output.tmp

包含 executeScript 任務執行之指令碼的執行期輸出。

### Err.tmp

包含 executeScript 任務執行之指令碼的執行期錯誤訊息。

## 範例

下列範例示範如何使用 executeScript 任務執行內嵌指令碼。

### 範例 1：Hello World 輸出文字檔案

此範例顯示 executeScript 執行 PowerShell 指令碼以在 C: 磁碟機上建立「Hello world」文字檔的工作。

```
task: executeScript
inputs:
- frequency: always
 type: powershell
 runAs: admin
 content: |-
 New-Item -Path 'C:\PowerShellTest.txt' -ItemType File
 Set-Content 'C:\PowerShellTest.txt' "Hello world"
```

### 範例 2：執行兩個指令碼

此範例顯示 executeScript 任務可以執行多個指令碼，而指令碼類型不一定需要相符。

第一個指令碼 (type: powershell) 會將執行個體上目前執行的程序摘要寫入位於 C: 磁碟機上的文字檔案。

第二個指令碼 (batch) 會將系統資訊寫入 Output.tmp 檔案。

```
task: executeScript
inputs:
- frequency: always
 type: powershell
```

```
content: |
 Get-Process | Out-File -FilePath C:\Process.txt
runAs: localSystem
- frequency: always
type: batch
content: |
 systeminfo
```

### 範例 3：具有重新啟動功能的等冪系統組態

此範例顯示 `executeScript` 任務，其執行等冪指令碼以執行下列系統組態，並在每個步驟之間重新啟動：

- 重新命名電腦。
- 將電腦加入網域。
- 啟用 Telnet。

該指令碼確保每個操作只執行一次。這可以防止重新啟動循環，並讓指令碼具有等冪性。

```
task: executeScript
inputs:
- frequency: always
type: powershell
runAs: localSystem
content: |-
 $name = $env:ComputerName
 if ($name -ne $desiredName) {
 Rename-Computer -NewName $desiredName
 exit 3010
 }
 $domain = Get-ADDomain
 if ($domain -ne $desiredDomain)
 {
 Add-Computer -DomainName $desiredDomain
 exit 3010
 }
 $telnet = Get-WindowsFeature -Name Telnet-Client
 if (-not $telnet.Installed)
 {
 Install-WindowsFeature -Name "Telnet-Client"
 exit 3010
 }
```

## extendRootPartition

延伸根磁碟區以使用磁碟上的所有可用空間。

Frequency — 一次

AllowedStages — [Boot]

Inputs — 無

### 範例

```
task: extendRootPartition
```

## initializeVolume

初始化連接到執行個體的空白磁碟區，以便啟動和分割這些磁碟區。如果啟動代理程式偵測到磁碟區並非空白，就會略過初始化。如果磁碟區的前 4 KiB 空白，或磁碟區沒有 [Windows 可辨識的磁碟機配置](#)，磁碟區會視為空白。

不論磁碟機是否已經初始化，在此任務執行時永遠都會套用 letter 輸入參數。

initializeVolume 任務會執行下列動作。

- 將磁碟屬性 offline 和 readonly 設為 false。
- 建立分割區。如果未在 partition 輸入參數中指定分割區類型，會套用下列預設值：
  - 如果磁碟大小小於 2 TB，請將分割區類型設為 mbr。
  - 如果磁碟大小大於等於 2 TB，請將分割區類型設為 gpt。
- 將磁碟區格式化為 NTFS。
- 將磁碟區標籤設定如下：
  - 如果有指定，請使用 name 輸入參數的值。
  - 如果磁碟區為暫時性，且未指定名稱，請將磁碟區標籤設為 Temporary Storage Z。
- 如果磁碟區為暫時性 (SSD 或 HDD，而非 Amazon EBS)，請在磁碟區的根目錄建立 Important.txt 檔案，其中包含下列內容：

```
This is an 'Instance Store' disk and is provided at no additional charge.
```

```
*This disk offers increased performance since it is local to the host
```

```
*The number of Instance Store disks available to an instance vary by instance type
```

\*DATA ON THIS DRIVE WILL BE LOST IN CASES OF IMPAIRMENT OR STOPPING THE INSTANCE.  
PLEASE ENSURE THAT ANY IMPORTANT DATA IS BACKED UP FREQUENTLY

For more information, please refer to: [Amazon EC2 #####](#).

- 將磁碟機代號設定為 letter 輸入參數中指定的值。

階段：您可以在 PostReady 和 UserData 階段執行 initializeVolume 任務。

頻率：始終。

## 輸入

您可以設定執行期參數，如下所示：

devices (映射清單)

(視情況而定) 啟動代理程式初始化之每個裝置的組態。initialize 輸入參數設為 devices 時，此為必要項目。

- device (字串，必要) – 在建立執行個體期間識別裝置。例如，xvdb、xvdf 或 \dev \nvme0n1。
- letter (字串，選用) – 一個字元。要指派的磁碟機代號。
- name (字串，選用) – 要指派的磁碟區名稱。
- partition (字串，選用) – 為要建立的分割區類型指定下列任一個值，或根據磁碟區大小讓啟動代理程式設定預設值：
  - MBR
  - gpt

initialize (字串)

(必要) 僅指定下列其中一個值：

- all
- devices

## 範例

下列範例說明 initializeVolume 任務的輸入組態範例。

範例 1：在執行個體上初始化兩個磁碟區

此範例說明在執行個體上初始化兩個次要磁碟區的 `initializeVolume` 任務。範例中的裝置 `DataVolume2` 為暫時性。

```
task: initializeVolume
inputs:
 initialize: devices
 devices:
 - device: xvdb
 name: DataVolume1
 letter: D
 partition: mbr
 - device: /dev/nvme0n1
 name: DataVolume2
 letter: E
 partition: gpt
```

## 範例 2：初始化連接至執行個體的 EBS 磁碟區

此範例說明 `initializeVolume` 任務，這會初始化所有連接至執行個體的空白 EBS 磁碟區。

```
task: initializeVolume
inputs:
 initialize: all
```

## optimizeEna

根據目前的執行個體類型最佳化 ENA 設定；可能會將執行個體重新開機。

Frequency — 永遠

AllowedStages — [PostReady, UserData]

Inputs — 無

## 範例

```
task: optimizeEna
```

## 集 AdminAccount

為在本機電腦上建立的預設管理員帳戶設定屬性。

## Frequency — 一次

AllowedStages — [PreReady]

### Inputs —

name : (字串) 管理員帳戶的名稱

password : (映射)

type : (字符) 策略來設置密碼，無論是 static、random 或 doNothing

data : (字串) 儲存資料，如果 type 欄位是靜態的話

### 範例

```
task: setAdminAccount
inputs:
 name: Administrator
 password:
 type: random
```

## setDnsSuffix

將 DNS 尾碼新增至搜尋尾碼清單。只有不存在的尾碼才會加入到清單中。如需啟動代理程式如何設定 DNS 尾碼的詳細資訊，請參閱。[設定 Windows 啟動代理程式的 DNS 尾碼](#)

## Frequency – 始終

AllowedStages — [PreReady]

### Inputs —

suffixes : (字串清單) 一或多個有效 DNS 尾碼清單；有效的替代變數為 \$REGION 和 \$AZ

### 範例

```
task: setDnsSuffix
inputs:
 suffixes:
 - $REGION.ec2-utilities.amazonaws.com
```

## setHostName

將電腦的主機名稱設定為自訂字串，或者如果 hostName 未指定，則設定私有 IPv4 地址。



## Frequency — 永遠

AllowedStages — [PostReady, UserData]

Inputs —

hostName : (字串) 選擇性的主機名稱，必須格式化如下。

- 必須等於或少於 15 個字元
- 必須只包含英數字元 (a-z、A-Z、0-9) 和連字號 (-) 字元。
- 不得完全由數字字元組成。

reboot : (布林值) 表示當主機名被更改時是否允許重新啟動

### 範例

```
task: setHostName
inputs:
 reboot: true
```

## setWallpaper

除了 Default User 之外，在每個現有使用者的啟動資料夾中建立 setwallpaper.lnk 捷徑檔案。使用者在執行個體啟動後第一次登入時，此捷徑檔案會執行。這會使用顯示執行個體屬性的自訂桌布，設定執行個體。

捷徑檔案路徑為：

```
$env:SystemDrive/Users/<user>/AppData/Roaming/Microsoft/Windows/Start Menu/Programs/Startup/setwallpaper.lnk
```

### Note

移除 setWallpaper 任務時，不會刪除此捷徑檔案。如需詳細資訊，請參閱 [setWallpaper 任務未啟用，但桌布會在重新啟動時重設。](#)

階段：您可以在 PreReady 和 UserData 階段設定桌布。

頻率：always

## 桌布組態

您可以使用下列設定來配置桌布。

### 輸入

您提供的輸入參數，以及可以設定桌布的屬性：

#### 屬性 (字串清單)

(選用) 您可以將以下一個或多個屬性新增至桌布：

- architecture
- availabilityZone
- hostName
- instanceId
- instanceSize
- memory
- network
- privateIpAddress
- publicIpAddress

#### instanceTags

(選用) 您可以使用下列其中一個選項來進行此設定。

- AllTags(字串) — 將所有實體標籤新增至桌布。

```
instanceTags: AllTags
```

- instanceTags (字串清單) – 指定要新增至桌布的執行個體標籤名稱清單。例如：

```
instanceTags:
 - Tag 1
 - Tag 2
```

#### 路徑 (字串)

(必要) 要用於桌布圖片的本機 .jpg 格式圖片檔案的檔案名稱路徑。

## 範例

下列範例顯示設定桌布背景圖片檔案路徑的桌布組態輸入，以及名為 Tag 1 和 Tag 2 的執行個體標籤，以及包含執行個體的主機名稱、執行個體 ID 及私有和公有 IP 地址的屬性。

```
task: setWallpaper
inputs:
 path: C:\ProgramData\Amazon\EC2Launch\wallpaper\Ec2Wallpaper.jpg
 attributes:
 - hostName
 - instanceId
 - privateIpAddress
 - publicIpAddress
 instanceTags:
 - Tag 1
 - Tag 2
```

### Note

您必須在中繼資料中啟用標籤，才能在桌布上顯示標籤。如需有關執行個體標籤和中繼資料的詳細資訊，請參閱 [使用執行個體中繼資料中的執行個體標籤](#)。

## startSsm

依照 Sysprep 開始 Systems Manager (SSM) 服務。

Frequency — 永遠

AllowedStages — [PostReady, UserData]

Inputs — 無

### 範例

```
task: startSsm
```

## sysprep

重設服務狀態、更新 unattend.xml、停用 RDP，以及執行 Sysprep。只有在所有其他任務完成後才會執行此任務。

Frequency — 一次

## AllowedStages — [UserData]

### Inputs —

`clean` : (布林值) 在執行 Sysprep 之前清除執行個體日誌

`shutdown` : (布林值) 在執行 Sysprep 後關閉執行個體

### 範例

```
task: sysprep
inputs:
 clean: true
 shutdown: true
```

## writeFile

將檔案寫入目的地。

Frequency — 請參閱 Inputs

## AllowedStages — [PostReady, UserData]

### Inputs —

`frequency` : (字符) 其中一個 `once` 或 `always`

`destination` : (字符) 要寫入內容的路徑

`content` : (字符) 要寫入文本的目的地

### 範例

```
task: writeFile
inputs:
- frequency: once
 destination: C:\Users\Administrator\Desktop\booted.txt
 content: Windows Has Booted
```

## EC2Launch v2 結束程式碼和重新啟動

您可以使用 EC2Launch v2 定義指令碼處理結束程式碼的方式。根據預設，在指令碼中執行的最後一個命令的結束程式碼會報告為整個指令碼的結束程式碼。例如，如果指令碼包含三個命令，而第一個命令失敗，但後續命令成功，則會因最終命令成功而將執行狀態報告為 `success`。

如果您希望指令碼重新啟動執行個體，則必須在指令碼中指定 `exit 3010`，即使重新啟動是指令碼中的最後一個步驟也是如此。`exit 3010` 會指示 EC2Launch v2 重新啟動執行個體並再次呼叫指令碼，直到傳回不是 3010 的結束程式碼，或直到達到最大重新啟動計數為止。EC2Launch v2 允許每個任務最多 5 次重新啟動。如果您嘗試使用不同的機制 (例如 `Restart-Computer`) 從指令碼重新啟動執行個體，則指令碼執行狀態將不一致。例如，狀態可能會卡在重新啟動迴圈或不執行重新啟動。

如果您使用與舊版代理程式相容的 XML 使用者資料格式，則使用者資料執行的次數可能會超出您的預期。如需詳細資訊，請參閱故障診斷一節中的 [服務會多次執行使用者資料](#)。

## EC2Launch v2 和 Sysprep

EC2Launch v2 服務將執行 Sysprep，此為 Microsoft 的工具，可讓您建立自訂且可重複使用的 Windows AMI。EC2Launch v2 呼叫 Sysprep 時，會使用 `%ProgramData%\Amazon\EC2Launch` 中的檔案來決定要執行的操作。您可以使用 EC2Launch 設定對話方塊間接編輯這些檔案，或直接使用 YAML 編輯器或文字編輯器來編輯這些檔案。但有些進階設定無法在 EC2Launch 設定對話方塊中使用，因此您必須直接編輯這些項目。

如果您在更新執行個體設定後從執行個體建立 AMI，新設定將套用到從新 AMI 啟用的任何執行個體。如需有關建立 AMI 的詳細資訊，請參閱 [創建一個 Amazon EBS-backed AMI](#)。

## EC2Launch v2 故障診斷

本節說明 EC2Launch v2 的常見故障診斷案例、有關檢視 Windows 事件日誌的資訊，以及主控台日誌輸出和訊息。

### 故障診斷主題

- [常見的疑難排解案例：](#)
- [Windows 事件日誌](#)
- [EC2Launch v2 主控台日誌輸出](#)

常見的疑難排解案例：

本節說明常見的疑難排解案例和解決步驟。

### 案例

- [服務無法設定桌布](#)
- [服務無法執行使用者資料](#)

- [服務只執行一次任務](#)
- [服務無法執行任務](#)
- [服務會多次執行使用者資料](#)
- [在移轉至 EC2Launch v2 後，來自 EC2Launch v1 的排程任務無法執行](#)
- [服務會初始化不為空的 EBS 磁碟區](#)
- [setWallpaper 任務未啟用，但桌布會在重新啟動時重設](#)
- [服務卡在執行狀態](#)
- [無效的 agent-config.yml 導致無法開啟 EC2Launch v2 設定對話方塊](#)
- [task:executeScript should be unique and only invoked once](#)

### 服務無法設定桌布

#### 解析度

1. 檢查 %AppData%\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\setwallpaper.lnk 是否存在。
2. 檢查 %ProgramData%\Amazon\EC2Launch\log\agent.log 是否發生任何錯誤。

### 服務無法執行使用者資料

可能原因：服務可能在執行使用者資料之前失敗。

#### 解析度

1. 檢查 %ProgramData%\Amazon\EC2Launch\state\previous-state.json。
2. 查看 boot、network、preReady 和 postReadyLocalData 是否已全部標記為成功。
3. 如果其中一個階段失敗，請檢查 %ProgramData%\Amazon\EC2Launch\log\agent.log 是否有特定的錯誤。

### 服務只執行一次任務

#### 解析度

1. 檢查任務的頻率。
2. 如果服務已經在 Sysprep 之後執行，且任務頻率設定為 once，則任務將不會再次執行。

3. 如果您希望 EC2Launch v2 每次執行時都能執行任務，請將任務的頻率設定為 `always`。

### 服務無法執行任務

#### 解析度

1. 檢查 `%ProgramData%\Amazon\EC2Launch\log\agent.log` 中的最新項目。
2. 如果沒有發生錯誤，請嘗試從 `"%ProgramFiles%\Amazon\EC2Launch\EC2Launch.exe"` `run` 手動執行服務，以查看任務是否成功。

### 服務會多次執行使用者資料

#### 解析度

EC2Launch v1 和 EC2Launch v2 處理使用者資料的方式不同。當 `persist` 設定為 `true` 時，EC2Launch v1 會將使用者資料視為執行個體上的排程任務來執行。如果將 `persist` 設定為 `false`，即使任務由於重新啟動而退出或在執行時中斷，也不會進行排程。

EC2Launch v2 會將使用者資料視為代理程式任務執行，並追蹤其執行狀態。如果使用者資料導致電腦重新啟動，或是使用者資料在執行時中斷，執行狀態將一直保持為 `pending`，且使用者資料將會在下次執行個體啟動時再次執行。如果您想防止使用者資料指令碼多次執行，請使指令碼等冪。

下列範例等冪指令碼會設定電腦名稱並加入網域。

```
<powershell>
$name = $env:computername
if ($name -ne $desiredName) {
Rename-Computer -NewName $desiredName
}
$domain = Get-ADDomain
if ($domain -ne $desiredDomain)
{
Add-Computer -DomainName $desiredDomain
}
$telnet = Get-WindowsFeature -Name Telnet-Client
if (-not $telnet.Installed)
{
Install-WindowsFeature -Name "Telnet-Client"
}
}</powershell>
```

```
<persist>>false</persist>
```

在移轉至 EC2Launch v2 後，來自 EC2Launch v1 的排程任務無法執行

### 解析度

遷移工具不會偵測任何連結至 EC2Launch v1 指令碼的排程任務；因此，不會自動在 EC2Launch v2 中設定這些任務。若要設定這些任務，請編輯 [agent-config.yml](#) 檔案，或使用 [EC2Launch v2 設定對話方塊](#)。例如，如果執行個體有執行 InitializeDisks.ps1 的排程任務，則執行遷移工具後，您必須在 EC2Launch v2 設定對話方塊中指定要初始化的磁碟區。請參閱 [使用 EC2Launch v2 設定對話方塊變更設定](#) 程序的步驟 6。

服務會初始化不為空的 EBS 磁碟區

### 解析度

在初始化磁碟區之前，EC2Launch v2 會嘗試偵測磁碟區是否為空。如果磁碟區不為空，則會略過初始化程序。偵測為非空白的任何磁碟區都不會初始化。如果磁碟區的前 4 KiB 為空，或磁碟區沒有 [Windows 可辨識的磁碟機配置](#)，磁碟區會被視為空。在 Linux 系統上初始化並格式化的磁碟區沒有 Windows 可辨識的磁碟機配置，例如 MBR 或 GPT。因此，該磁碟區將視為空磁碟區且已初始化。如果您想要保留此資料，請勿依賴 EC2Launch v2 空白磁碟機偵測。而是在 [EC2Launch v2 設定對話方塊](#) (請參閱步驟 6) 或 [agent-config.yml](#) 中指定要初始化的磁碟區。

**setWallpaper** 任務未啟用，但桌布會在重新啟動時重設

setWallpaper 任務會在 Default User 以外的每個現有使用者啟動資料夾中，建立 setwallpaper.lnk 捷徑檔案。使用者在執行個體啟動後第一次登入時，此捷徑檔案會執行。這會使用顯示執行個體屬性的自訂桌布，設定執行個體。移除 setWallpaper 任務不會刪除此捷徑檔案。您必須手動刪除此檔案，或使用指令碼將其刪除。

捷徑路徑為：

```
$env:SystemDrive/Users/<user>/AppData/Roaming/Microsoft/Windows/Start Menu/
Programs/Startup/setwallpaper.lnk
```

### 解析度

手動刪除此檔案，或使用指令碼將其刪除。

刪除捷徑檔案的範例 PowerShell 指令碼

```
foreach ($userDir in (Get-ChildItem "C:\Users" -Force -Directory).FullName)
```



```
{
 $startupPath = Join-Path $userDir -ChildPath "AppData\Roaming\Microsoft\Windows\Start
Menu\Programs\Startup"
 if (Test-Path $startupPath)
 {
 $wallpaperSetupPath = Join-Path $startupPath -ChildPath "setwallpaper.lnk"
 if (Test-Path $wallpaperSetupPath)
 {
 Remove-Item $wallpaperSetupPath -Force -Confirm:$false
 }
 }
}
```

## 服務卡在執行狀態

### Description

EC2Launch v2 會被封鎖，其日誌訊息 (agent.log) 內容如下：

```
2022-02-24 08:08:58 Info:

2022-02-24 08:08:58 Info: EC2Launch Service starting
2022-02-24 08:08:58 Info: Windows event custom log exists: Amazon EC2Launch
2022-02-24 08:08:58 Info: ACPI SPCR table not supported. Bailing Out
2022-02-24 08:08:58 Info: Serial port is in use. Waiting for Serial Port...
2022-02-24 08:09:00 Info: ACPI SPCR table not supported. Use default console port.
2022-02-24 08:09:02 Info: ACPI SPCR table not supported. Use default console port.
2022-02-24 08:09:04 Info: ACPI SPCR table not supported. Use default console port.
2022-02-24 08:09:06 Info: ACPI SPCR table not supported. Use default console port.
```

### 可能的原因

SAC 處於啟用狀態，並使用序列連接埠。如需詳細資訊，請參閱[使用 SAC 疑難排解您的 Windows 執行個體](#)。

### 解析度

請嘗試下列步驟來解決此問題：

- 停用正在使用序列連接埠的服務。
- 如果希望服務繼續使用序列連接埠，請編寫自訂指令碼來執行啟動代理程式任務，並將其做為排程任務呼叫。

## 無效的 `agent-config.yml` 導致無法開啟 EC2Launch v2 設定對話方塊

### 描述

EC2Launch v2 設定將嘗試剖析 `agent-config.yml` 檔案，然後開啟對話方塊。如果 YAML 組態檔案不遵循支援的結構描述，對話方塊將顯示以下錯誤：

```
Unable to parse configuration file agent-config.yml. Review configuration file. Exiting application.
```

### 解析度

1. 驗證組態檔案是否遵循[支援的結構描述](#)。
2. 如果要從頭開始，請將預設組態檔案複製到 `agent-config.yml`。您可以使用「任務組態」區段中提供的[範例 `agent-config.yml`](#)。
3. 您也可以透過刪除 `agent-config.yml` 重新開始。EC2Launch v2 設定產生空的組態檔案。

## `task:executeScript` should be unique and only invoked once

### 描述

不能在同一階段重複任務。

### 解析度

某些任務必須作為陣列輸入，如 [executeScript](#) 和 [executeProgram](#)。如需將指令碼編寫為陣列的範例，請參閱 [executeScript](#)。

### Windows 事件日誌

EC2Launch v2 會針對重要事件發佈 Windows 事件日誌，例如服務啟動、Windows 已就緒，以及任務成功和失敗。事件 ID 會唯一識別特定事件。每個事件都包含階段、任務和層級資訊，以及說明。您可以使用事件 ID 來設定特定事件的觸發程序。

事件 ID 提供事件的相關資訊，並唯一識別某些事件。事件 ID 的最低有效位元顯示事件的嚴重性。

事件	最低有效位元
Success	. . .0
Informational	. . .1

事件	最低有效位元
Warning	. . .2
Error	. . .3

服務啟動或停止時產生的服務相關事件包含單一數字事件識別符。

事件	單一數字識別符
Success	0
Informational	1
Warning	2
Error	3

事件的 EC2LaunchService.exe 事件訊息以 Service: 為開頭。事件的 EC2Launch.exe 事件訊息不是以 Service: 為開頭。

四位數的事件 ID 包括關於階段、任務和事件嚴重性的資訊。

### 主題

- [事件 ID 格式](#)
- [事件 ID 範例](#)
- [Windows 事件日誌結構描述](#)

### 事件 ID 格式

下表顯示 EC2Launch v2 事件識別符的格式。

3	2 1	0
S	T	L

3	2 1	0
---	-----	---

表格中的字母和數字代表下列事件類型和定義。

Event type	定義
S (階段)	0 - 服務層級訊息 1 - 開機 2 - 網路 3- PreReady 5 - Windows 已準備就緒 6- PostReady 7 - 使用者資料
T (任務)	由對應的兩個值所代表的任務對每個階段都不同。若要檢視完整的事件清單，請參閱 <a href="#">Windows 事件日誌結構描述</a> 。
L (活動等級)	0 - 成功 1 - 資訊 2 - 警告 3 - 錯誤

### 事件 ID 範例

以下是範例事件 ID。

- 5000 - Windows 已做好使用準備

- 3010-在 PreReady 階段激活窗口任務成功
- 6013-在 PostReady 本機資料階段設定桌布工作發生錯誤

## Windows 事件日誌結構描述

MessageId/事件識別碼	事件訊息
. . .0	Success
. . .1	Informational
. . .2	Warning
. . .3	Error
x	EC2Launch service-level logs
0	EC2Launch service exited successfully
1	EC2Launch service informational logs
2	EC2Launch service warning logs
3	EC2Launch service error logs
10	Replace state.json with previous-state.json
100	Serial Port
200	Sysprep
300	PrimaryNic
400	Metadata
x000	Stage (1 digit), Task (2 digits), Status (1 digit)

MessageId/事件識別碼	事件訊息
1000	Boot
1010	Boot - extend_root_partition
2000	Network
2010	Network - add_routes
3000	PreReady
3010	PreReady - activate_windows
3020	PreReady - install_egpu_manager
3030	PreReady - set_monitor_on
3040	PreReady - set_hibernation
3050	PreReady - set_admin_account
3060	PreReady - set_dns_suffix
3070	PreReady - set_wallpaper
3080	PreReady - set_update_schedule
3090	PreReady - output_log
3100	PreReady - enable_open_ssh
5000	Windows is Ready to use
6000	PostReadyLocalData
7000	PostReadyUserData
6010/7010	PostReadyLocal/UserData - set_wallpaper

MessageId/事件識別碼	事件訊息
6020/7020	PostReadyLocal/UserData - set_update_schedule
6030/7030	PostReadyLocal/UserData - set_hostname
6040/7040	PostReadyLocal/UserData - execute_program
6050/7050	PostReadyLocal/UserData - execute_script
6060/7060	PostReadyLocal/UserData - manage_package
6070/7070	PostReadyLocal/UserData - initialize_volume
6080/7080	PostReadyLocal/UserData - write_file
6090/7090	PostReadyLocal/UserData - start_ssm
7100	PostReadyUserData - enable_op en_ssh
6110/7110	PostReadyLocal/UserData - enable_jumbo_frames

## EC2Launch v2 主控台日誌輸出

本節包含 EC2Launch v2 的範例主控台日誌輸出，並列出所有 EC2Launch v2 主控台日誌錯誤訊息，以協助您針對問題進行故障診斷。如需執行個體主控台輸出及其存取方式的詳細資訊，請參閱[the section called “執行個體主控台輸出”](#)。

## 輸出

- [EC2Launch v2 主控台日誌輸出](#)
- [EC2Launch v2 主控台日誌訊息](#)

## EC2Launch v2 主控台日誌輸出

以下是 EC2Launch v2 的範例主控台日誌輸出。

```
2023/11/30 20:18:53Z: Windows sysprep configuration complete.
2023/11/30 20:18:57Z: Message: Waiting for access to metadata...
2023/11/30 20:18:57Z: Message: Meta-data is now available.
2023/11/30 20:18:57Z: AMI Origin Version: 2023.11.15
2023/11/30 20:18:57Z: AMI Origin Name: Windows_Server-2022-English-Full-Base
2023/11/30 20:18:58Z: OS: Microsoft Windows NT 10.0.20348
2023/11/30 20:18:58Z: OsVersion: 10.0
2023/11/30 20:18:58Z: OsProductName: Windows Server 2022 Datacenter
2023/11/30 20:18:58Z: OsBuildLabEx: 20348.1.amd64fre.fe_release.210507-1500
2023/11/30 20:18:58Z: OsCurrentBuild: 20348
2023/11/30 20:18:58Z: OsReleaseId: 2009
2023/11/30 20:18:58Z: Language: en-US
2023/11/30 20:18:58Z: TimeZone: UTC
2023/11/30 20:18:58Z: Offset: UTC +0000
2023/11/30 20:18:58Z: Launch: EC2 Launch v2.0.1643
2023/11/30 20:18:58Z: AMI-ID: ami-1234567890abcdef1
2023/11/30 20:18:58Z: Instance-ID: i-1234567890abcdef0
2023/11/30 20:18:58Z: Instance Type: c5.large
2023/11/30 20:19:00Z: Driver: AWS NVMe Driver v1.5.0.33
2023/11/30 20:19:00Z: SubComponent: AWS NVMe Driver v1.5.0.33;
 EnableSCSIPersistentReservations: 0
2023/11/30 20:19:00Z: Driver: AWS PV Driver Package v8.4.3
2023/11/30 20:19:01Z: Driver: Amazon Elastic Network Adapter v2.6.0.0
2023/11/30 20:19:01Z: RDPCERTIFICATE-SUBJECTNAME: EC2AMAZ-S01T009
2023/11/30 20:19:01Z: RDPCERTIFICATE-THUMBPRINT:
 1234567890ABCDEF1234567890ABCDEF1234567890
2023/11/30 20:19:09Z: SSM: Amazon SSM Agent v3.2.1705.0
2023/11/30 20:19:13Z: Username: Administrator
2023/11/30 20:19:13Z: Password: <Password>
1234567890abcdef1EXAMPLEPASSWORD
</Password>
2023/11/30 20:19:14Z: User data format: no_user_data
2023/11/30 20:19:14Z: EC2LaunchTelemetry: IsTelemetryEnabled=true
2023/11/30 20:19:14Z: EC2LaunchTelemetry: AgentOsArch=windows_amd64
2023/11/30 20:19:14Z: EC2LaunchTelemetry: IsAgentScheduledPerBoot=true
```



```
2023/11/30 20:19:14Z: EC2LaunchTelemetry: AgentCommandErrorCode=0
2023/11/30 20:19:14Z: Message: Windows is Ready to use
```

## EC2Launch v2 主控台日誌訊息

以下是所有 EC2Launch v2 主控台日誌訊息的清單。

```
Message: Error EC2Launch service is stopping. {error message}
 Error setting up EC2Launch agent folders
 See instance logs for detail
 Error stopping service
 Error initializing service
Message: Windows sysprep configuration complete
Message: Invalid administrator username: {invalid username}
Message: Invalid administrator password
Username: {username}
Password: <Password>{encrypted password}</Password>
AMI Origin Version: {amiVersion}
AMI Origin Name: {amiName}
Microsoft Windows NT {currentVersion}.{currentBuildNumber}
OsVersion: {currentVersion}
OsProductName: {productName}
OsBuildLabEx: {buildLabEx}
OsCurrentBuild: {currentBuild}
OsReleaseId: {releaseId}
Language: {language}
TimeZone: {timeZone}
Offset: UTC {offset}
Launch agent: EC2Launch {BuildVersion}
AMI-ID: {amiId}
Instance-ID: {instanceId}
Instance Type: {instanceType}
RDPCERTIFICATE-SUBJECTNAME: {certificate subject name}
RDPCERTIFICATE-THUMBPRINT: {thumbprint hash}
SqlServerBilling: {sql billing}
SqlServerInstall: {sql patch leve, edition type}
Driver: AWS NVMe Driver {version}
Driver: Inbox NVMe Driver {version}
Driver: AWS PV Driver Package {version}
Microsoft-Hyper-V is installed.
Unable to get service status for vmms
Microsoft-Hyper-V is {status}
SSM: Amazon SSM Agent {version}
AWS VSS Version: {version}
```

```

Message: Windows sysprep configuration complete
Message: Windows is being configured. SysprepState is {state}
Windows is still being configured. SysprepState is {state}
Message: Windows is Ready to use
Message: Waiting for meta-data accessibility...
Message: Meta-data is now available.
Message: Still waiting for meta-data accessibility...
Message: Failed to find primary network interface...retrying...
User data format: {format}

```

## EC2Launch v2 版本歷程記錄

### 版本歷程記錄

- [EC2Launch v2 版本歷史記錄](#)
- [EC2Launch v2 遷移工具版本歷史記錄](#)

### EC2Launch v2 版本歷史記錄

下表說明 EC2Launch v2 的發行版本。

版本	詳細資訊	發行日期
2.0.1924	<ul style="list-style-type: none"> <li>• 更新 EC2 啟動設定使用者介面。</li> <li>• 更新桌布 CLI 命令。</li> <li>• 更新了 EC2 啟動安裝程式。</li> </ul>	2024年6月10日
2.0.1914	<ul style="list-style-type: none"> <li>• 新增具有未指定閘道位址的路由 (0.0.0.0適用::於 IPv4 或 IPv6)。</li> <li>• 一律新增 IPv4 和 IPv6 路由。</li> <li>• 修正未指定Administrator 使用者名稱時新增至agent-config.yml 檔案的問題。</li> <li>• 已修改 EC2 啟動 v2 權限。</li> </ul>	2024年6月5日

版本	詳細資訊	發行日期
2.0.1881	<ul style="list-style-type: none"> <li>在任務中添加了一個加密密碼選 <code>setAdminAccount</code> 項。</li> <li>添加了 CLI 命令來加密代理程序配置 <code>.yaml</code> 中的靜態密碼。</li> <li>修正 XML 使用者資料在以系統管理員權限執行時不會新增 PowerShell 引數的問題。如需詳細資訊，請參閱 <a href="#">Amazon EC2 如何處理 Windows 執行個體的使用者資料</a>。</li> <li>調整 <code>executeScript</code> 工作和使用者資料指令碼使用 <code>LocalSystem</code> 權限執行時的 PowerShell 引數。當引數為空時，代理程式會使用下列預設值：<code>-ExecutionPolicy Unrestricted</code></li> <li>禁止將重複的驅動程序版本打印到控制台日誌。</li> </ul>	2024年5月8日
2.0.1815	<ul style="list-style-type: none"> <li>在 <code>sysprep</code> 之前調整了關鍵設置問題的錯誤處理失敗。</li> <li>修正桌布和主機名稱工作在指派給主要網路介面的多個 IP 位址的執行個體上，可能會使用不正確 IP 位址的問題。</li> <li>桌布和主機名稱工作已變更為先從 IMDS 取得私有 IP，如果 IMDS 停用，則會失敗回到 WMI。</li> <li>修正 <code>sc1</code> 磁碟區因暫時性錯誤而無法初始化的 <code>initializeVolume</code> 工作問題。</li> </ul>	2024年3月6日
2.0.1739	<ul style="list-style-type: none"> <li>修正以 Windows 管理員使用者身分執行的 <code>executeScript</code> 工作無法擷取結束代碼的問題。</li> </ul>	2024年1月17日

版本	詳細資訊	發行日期
2.0.1702	<ul style="list-style-type: none"> <li>• 將 Telemetry.log 許可限制為 read-execute，僅適用於標準使用者。</li> <li>• 將 EC2Launch Windows 服務設定為在啟動失敗時重新啟動。</li> <li>• 透過記錄 route.exe stderr 輸出，使 add-routes 故障可操作。</li> <li>• 修正當路由指標超出 [1, 9999] 範圍時發生的問題。</li> <li>• 為數種新的執行個體類型新增了桌布支援。</li> <li>• 修正以 Windows 管理員使用者身分執行並將輸出傳送至 stderr 的使用者資料指令碼所造成的問題。</li> </ul>	2024 年 1 月 4 日
2.0.1643	<ul style="list-style-type: none"> <li>• 將 ebsnvme-id.exe 工具更新至 1.1.0.7 版。</li> <li>• 修正以 'metal-*' 開頭 (例如 metal-48x1) 的金屬執行個體類型的接收端調整 (RSS) 和接收佇列深度設定的問題。</li> <li>• 已移除對封鎖代理程式之 XML 使用者資料命令進行報告的遙測事件。</li> <li>• 已更新 setDnsSuffix 任務，以根據登錄項目限制網域名稱轉移：HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Dnscache\Parameters\DomainNameDevolutionLevel。</li> <li>• 新增一個公有任務以及可新增網路路由的 CLI。</li> <li>• 注意 – 這是正式支援 Windows Server 2012 的最新版本。</li> <li>• 注意 – 這是正式支援 32 位元作業系統的最新版本。</li> </ul>	2023 年 10 月 4 日

版本	詳細資訊	發行日期
2.0.1580	<ul style="list-style-type: none"> <li>修改日誌檔案許可時，已變更啟動代理程式處理錯誤的方式。</li> <li>已新增連線至序列埠資源的逾時時間。如果序列埠正在使用中，則逾時允許啟動代理程式繼續執行。</li> </ul>	2023 年 9 月 5 日
2.0.1521	<ul style="list-style-type: none"> <li>已棄用 EC2Launch.exe、reset 和 sysprep 命令的 <code>-block</code> 旗標。</li> <li>EC2Launch.exe 已更新，可偵測並處理內嵌 <code>executeScript</code> 任務中使用的 <code>reset</code> 和 <code>sysprep</code> 命令。這些命令會導致代理程式在 <code>executeScript</code> 任務執行後停止執行。</li> <li>已更新 XML 使用者資料指令碼，預設情況下會內嵌執行。</li> <li>啟用 XML 使用者資料指令碼，以使用新的 <code>detach</code> 標籤在分離狀態下執行。如需詳細資訊，請參閱<a href="#">使用者資料指令碼</a>。</li> <li>對代理程式日誌進行了下列變更。 <ul style="list-style-type: none"> <li>更新代理程式日誌訊息。</li> <li>從代理程式日誌中移除 <code>executeScript</code> 內容和輸出。</li> <li>從代理程式日誌中移除 <code>executeProgram</code> 引數和輸出。</li> </ul> </li> <li>對主控台日誌進行了下列變更。 <ul style="list-style-type: none"> <li>為主控台日誌新增 <code>EnableSCSIPersistentReservations</code> 值。</li> </ul> </li> </ul>	2023 年 7 月 3 日

版本	詳細資訊	發行日期
2.0.1303	<ul style="list-style-type: none"> <li>• 新增網絡路由時新增了額外的錯誤處理和日誌行。</li> <li>• 允許executeScript 和 PreReady 階段中的executeProgram 任務。</li> <li>• 已更新 executeProgram 任務，以產生類似於 executeScript 任務輸出的輸出檔案。如需詳細資訊，請參閱 <a href="#">executeProgram</a>。</li> <li>• 已新增遙測，以監控 XML 使用者資料中封鎖代理程式命令的使用情況。</li> </ul>	2023 年 5 月 3 日
2.0.1245	<ul style="list-style-type: none"> <li>• 以純文字形式記錄當機呼叫堆疊，改善當機的可見性。</li> <li>• 已新增 EventLog 服務做為啟動相依性，以修正 Amazon EC2Launch 服務啟動速度比服務更快時的當機問題。EventLog</li> <li>• 讓 XML 使用者資料從代理程式設定檔 (例如 EC2Launch v1 和 EC2Config) 在 PostReady 階段之前執行。</li> <li>• 新增 YAML 使用者資料 1.1 版，讓使用者資料在代理程式設定檔的 PostReady 階段前執行 (YAML 使用者資料版本 1.0 在代理程式設定檔的 PostReady 階段之後執行)。</li> </ul>	2023 年 3 月 8 日
2.0.1173	<ul style="list-style-type: none"> <li>• 新增一個可選功能以在桌布上顯示執行個體標籤。如需詳細資訊，請參閱 <a href="#">setWallpaper</a>。</li> <li>• 未正確設定 Elastic Graphics 的安全群組時，會新增錯誤處理。</li> <li>• 修復未啟用執行個體中繼資料服務時的逾時。</li> </ul>	2023 年 2 月 6 日

版本	詳細資訊	發行日期
2.0.1121	<ul style="list-style-type: none"> <li>修正未指派公有 IPv4 地址時會將 404 錯誤列印至桌布的問題。</li> <li>修復當裝置的磁碟機代號設為 D 時磁碟區檔案系統格式為 RAW 而非 NTFS 的問題。</li> <li>修復 NVMe SSD 磁碟區錯誤識別為 EBS 磁碟區的問題。</li> <li>修復在停用 IMDS 後啟動 Windows 時的錯誤。</li> </ul>	2023 年 1 月 4 日
2.0.1082	<ul style="list-style-type: none"> <li>修正停用 IMDS 時 <code>setWallpaper :privateIpAddress</code> 欄位為空白的問題。</li> <li>修正停用 IMDS 時將主機名稱設定為私有 IPv4 地址的問題。</li> <li>修正在 Windows Server 2012 上初始化磁碟區的問題。</li> <li>修正設定巨型訊框的問題。</li> <li>修正執行個體啟動時未指定 SSH 金鑰時的錯誤。</li> <li>修復 Windows 沒有 'Releaseld' 註冊表項時 Windows 服務器 2012 上的錯誤。</li> </ul>	2022 年 12 月 7 日
2.0.1011	<ul style="list-style-type: none"> <li>修正 PnPDeviceID 為空白時尋找網路轉接器的邏輯。</li> </ul>	2022 年 11 月 11 日
2.0.1009	<ul style="list-style-type: none"> <li>使用 PCI 區段資訊來選取主控台連接埠。</li> </ul>	2022 年 11 月 8 日

版本	詳細資訊	發行日期
2.0.982	<ul style="list-style-type: none"> <li>• 新增重試邏輯以獲取 RDP 資訊。</li> <li>• 修正 d2.8xlarge 執行個體的磁碟區初始化期間的錯誤。</li> <li>• 修正在重新啟動後可能會選取錯誤網路轉接器的問題。</li> <li>• 當 ACPI SPCR 無法使用時，移除誤報錯誤訊息。</li> </ul>	2022 年 10 月 31 日
2.0.863	<ul style="list-style-type: none"> <li>• 更新 IMDS 等待邏輯，以僅傳出 IMDSv2 請求。</li> <li>• 新增邏輯，將磁碟機代號指派給已初始化但未掛載的磁碟區。</li> <li>• 當 key pair type 不受支援時，印出更詳細的錯誤訊息。</li> <li>• 修復 3010 重新啟動的程式碼錯誤。</li> <li>• 新增檢查無效 base64 編碼的使用者資料功能。</li> </ul>	2022 年 7 月 6 日
2.0.698	<ul style="list-style-type: none"> <li>• 修復了執行指令碼時日誌輸出中的錯字。</li> </ul>	2022 年 1 月 30 日
2.0.674	<ul style="list-style-type: none"> <li>• 遙測會上傳已啟用/已停用的隱私權控制。</li> <li>• 修正 index out of bounds 錯誤。</li> <li>• 在 sysprep 期間移除桌布捷徑。</li> </ul>	2021 年 11 月 15 日
2.0.651	<ul style="list-style-type: none"> <li>• 新增邏輯以在安裝 EC2Launch v2 期間，解除安裝舊式代理程式。</li> <li>• 修正根磁碟區未列為磁碟區 0 時的 list-volume CLI 問題。</li> </ul>	2021 年 10 月 7 日



版本	詳細資訊	發行日期
2.0.592	<ul style="list-style-type: none"> <li>修正錯誤以正確報告階段狀態。</li> <li>當日誌檔關閉時，移除錯誤的警示錯誤訊息。</li> <li>新增遙測。</li> </ul>	2021 年 8 月 31 日
2.0.548	<ul style="list-style-type: none"> <li>為十六進位 IP 主機名稱新增開頭的零。</li> <li>修正 <code>enableOpenSsh</code> 任務的檔案許可。</li> <li>修正 <code>sysprep</code> 命令當機。</li> </ul>	2021 年 8 月 4 日
2.0.470	<ul style="list-style-type: none"> <li>修正在網路階段中等待 DHCP 指派 IP 給執行個體的錯誤。</li> <li>修正 <code>SearchList</code> 登錄機碼不存在時的 <code>setDnsSuffix</code> 錯誤。</li> <li>修正 <code>setDnsSuffix</code> 中 DNS 轉移邏輯中的錯誤。</li> <li>在中繼重新開機後新增網路路由。</li> <li>允許 <code>initializeVolume</code> 重新用字母標示現有磁碟區。</li> <li>從版本子命令中移除額外的資訊。</li> </ul>	2021 年 7 月 20 日
2.0.285	<ul style="list-style-type: none"> <li>在分離的程序中新增了執行使用者指令碼的選項。</li> <li>舊式使用者資料 (XML 使用者資料) 現在會在分離的程序中執行，這與之前啟動代理程式類似的行為。</li> <li>將 CLI 標記新增至 <code>sysprep</code> 和 <code>reset</code> 命令，這可讓其在服務停止之前進行封鎖。</li> <li>限制組態資料夾許可。</li> </ul>	2021 年 3 月 8 日

版本	詳細資訊	發行日期
2.0.207	<ul style="list-style-type: none"><li>將選擇性 <code>hostName</code> 欄位新增至 <code>setHostName</code> 工作。</li><li>修正重新開機錯誤。重新啟動任務 <code>executeScript</code> 並將 <code>executeProgram</code> 標記為執行中。</li><li>在狀態命令中新增更多傳回碼。</li><li>新增引導服務，以修正在執行 <code>t2.nano</code> 執行個體類型時的啟動問題。</li><li>修正全新安裝模式，以移除安裝程式未追蹤的檔案。</li></ul>	2021 年 2 月 2 日
2.0.160	<ul style="list-style-type: none"><li>修正 <code>validate</code> 命令，以偵測無效的階段名稱。</li><li>在 <code>addroutes</code> 任務中新增 <code>w32tm resync</code> 命令。</li><li>修正變更 DNS 尾碼搜尋順序的問題。</li><li>新增檢查條件，以更詳細地報告無效的使用者資料。</li></ul>	2020 年 12 月 4 日
2.0.153	在中增加系統表示功能。 <code>UserData</code>	2020 年 11 月 3 日

版本	詳細資訊	發行日期
2.0.146	<ul style="list-style-type: none"> <li>修復了非英語 AMI 的 RootExtend 問題。</li> <li>授與使用者群組寫入日誌檔案的權限。</li> <li>為 GPT 磁碟區建立 MS 預留的磁碟分割。</li> <li>在 Amazon EC2Launch 設定中，新增清單磁碟區命令和磁碟區下拉式清單。</li> <li>添加用於以 YAML 或 JSON 格式打印代理配置 .yml 文件的 get-agent-config 命令。</li> <li>如果沒有偵測到公開金鑰，則清除靜態密碼。</li> </ul>	2020 年 10 月 6 日
2.0.124	<ul style="list-style-type: none"> <li>新增在桌布上顯示作業系統版本的選項。</li> <li>初始化加密的 EBS 磁碟區。</li> <li>為沒有本機 DNS 名稱的 VPC 新增路由。</li> </ul>	2020 年 9 月 10 日
2.0.104	<ul style="list-style-type: none"> <li>如果不存在，則建立 DNS 尾碼搜尋清單。</li> <li>如果沒有要求，則可略過休眠。</li> </ul>	2020 年 8 月 12 日
2.0.0	初始版本。	2020 年 6 月 30 日

## EC2Launch v2 遷移工具版本歷史記錄

下表說明 EC2Launch v2 遷移工具的發行版本。

版本	詳細資訊	發行日期
1.0.396	<ul style="list-style-type: none"> <li>使用最新版本的 EC2Launch v2 代理程式更新移轉工具：2.0.1924。</li> </ul>	2024年6月11日
1.0.394	<ul style="list-style-type: none"> <li>使用最新版本的 EC2Launch v2 代理程式更新移轉工具：2.0.1914。</li> </ul>	2024年6月6日
1.0.384	<ul style="list-style-type: none"> <li>使用最新版本的 EC2Launch 代理程式更新移轉工具：2.0.1881。</li> </ul>	2024年5月8日
1.0.358	<ul style="list-style-type: none"> <li>使用最新版本的 EC2Launch 代理程式更新移轉工具：2.0.1815。</li> </ul>	2024年3月8日
1.0.345	<ul style="list-style-type: none"> <li>使用最新版本的 EC2Launch v2 代理程式更新移轉工具：2.0.1739。</li> </ul>	2024年1月18日
1.0.342	<ul style="list-style-type: none"> <li>使用最新版本的 EC2Launch 代理程式更新移轉工具：2.0.1702。</li> </ul>	2024年1月5日
1.0.331	<ul style="list-style-type: none"> <li>使用最新版的 EC2Launch v2 代理程式更新遷移工具：2.0.1643。</li> <li>修復執行 <code>.Install.ps1 -DryRun</code> 時發生的錯誤。</li> <li>修正從 EC2Config 遷移期間，密碼組態未正確設定為 <code>random</code> 的問題。</li> <li>修正從 EC2Launch 遷移期間，<code>setWallpaper</code> 設定為 <code>False</code> 時會發生的錯誤。</li> </ul>	2023年11月3日
1.0.303	<p>使用最新版的 EC2Launch v2 代理程式更新遷移工具：2.0.1580。</p>	2023年9月14日

版本	詳細資訊	發行日期
1.0.286	使用最新版的 EC2Launch v2 代理程式更新遷移工具：2.0.152 1。	2023 年 7 月 14 日
1.0.272	使用最新版的 EC2Launch v2 代理程式更新遷移工具：2.0.130 3。	2023 年 5 月 3 日
1.0.262	使用最新版的 EC2Launch v2 代理程式更新遷移工具：2.0.124 5。	2023 年 3 月 9 日
1.0.241	將 EC2Launch v2 代理程式的版本編號更新至 2.0.1011。	2022 年 12 月 7 日
1.0.218	<ul style="list-style-type: none"> <li>驗證從執行個體中繼資料中擷取的區域值。</li> <li>修復了語言套件中的遷移失敗錯誤。</li> <li>將 EC2Launch v2 代理程式的版本編號更新至 2.0.863。</li> </ul>	2022 年 9 月 3 日
1.0.162	<ul style="list-style-type: none"> <li>將移除舊式代理程式的邏輯移動至 EC2Launch v2 MSI。</li> <li>將 EC2Launch v2 代理程式的版本編號更新至 2.0.698。</li> </ul>	2022 年 3 月 18 日
1.0.136	將 EC2Launch v2 代理程式的版本編號更新至 2.0.651。	2021 年 10 月 13 日
1.0.130	將 EC2Launch v2 代理程式的版本編號更新至 2.0.548。	2021 年 8 月 5 日
1.0.113	使用 IMDSv2 取代 IMDSv1。	2021 年 6 月 4 日
1.0.101	將 EC2Launch v2 代理程式的版本編號更新至 2.0.285。	2021 年 3 月 12 日
1.0.86	將 EC2Launch v2 代理程式的版本編號更新至 2.0.207。	2021 年 2 月 3 日
1.0.76	將 EC2Launch v2 代理程式的版本編號更新至 2.0.160。	2020 年 12 月 4 日

版本	詳細資訊	發行日期
1.0.69	將 EC2Launch v2 代理程式的版本編號更新至 2.0.153。	2020 年 11 月 5 日
1.0.65	將 EC2Launch v2 代理程式的版本編號更新至 2.0.146。	2020 年 10 月 9 日
1.0.60	將 EC2Launch v2 代理程式的版本編號更新至 2.0.124。	2020 年 9 月 10 日
1.0.54	<ul style="list-style-type: none"><li>• 如果未安裝代理程式，則安裝 EC2Launch v2。</li><li>• 將 EC2Launch v2 代理程式的版本編號更新至 2.0.104。</li><li>• 解偶 SSM Agent。</li></ul>	2020 年 8 月 12 日
1.0.50	刪除 NuGet 依賴關係。	2020 年 8 月 10 日
1.0.0	初始版本。	2020 年 6 月 30 日

## 使用 EC2Launch 設定 Windows 執行個體

EC2 啟動是一組視窗 PowerShell 腳本，取代了 EC2 配置服務在視窗服務器 2016 年和 2019 AMI 上。其中有許多 AMI 仍然可用。所有受支援 Windows 版本的最新啟動代理程式是 EC2Launch v2，它會取代 EC2Config 和 EC2Launch。如需詳細資訊，請參閱 [使用 EC2Launch v2 設定 Windows 執行個體](#)。

### Note

為了搭配使用 EC2Launch 與 IMDSv2，版本必須為 1.3.2002730 或以上版本。

## 目錄

- [EC2Launch 任務](#)
- [遙測](#)

- [安裝最新版本的 EC2Launch](#)
- [確認 EC2Launch 版本](#)
- [EC2Launch 目錄結構](#)
- [設定 EC2Launch](#)
- [EC2Launch 版本歷史記錄](#)

## EC2Launch 任務

根據預設，EC2Launch 會在初始執行個體開機期間執行下列任務：

- 設定呈現執行個體資訊的新桌面。
- 將電腦名稱設定為執行個體的私人 IPv4 位址。
- 傳送執行個體資訊到 Amazon EC2 主控台。
- 傳送 RDP 憑證指紋到 EC2 主控台。
- 為管理員帳戶設定隨機密碼。
- 新增 DNS 尾碼。
- 動態擴充作業系統分割區，以包含任何未分割的空間。
- 執行使用者資料 (若指定)。如需指定使用者資料的詳細資訊，請參閱[使用執行個體使用者資料](#)。
- 設置持久靜態路由以訪問元數據服務和服務 AWS KMS 器。

### Important

如果從這個執行個體建立自訂 AMI，則會擷取這些路由做為作業系統組態的一部分，任何從該 AMI 啟動的新執行個體會保留相同路由，而不論子網放置何處。若要更新路由，請參閱[啟動自訂 AMI 時更新 Server 2016 和更新版本的中繼資料/KMS 路由](#)。

下列任務可協助維護與 EC2Config 服務回溯相容的功能。您也可以設定 EC2Launch 在啟動期間執行這些任務：

- 初始化輔助 EBS 磁碟區。
- 傳送 Windows 事件日誌到 EC2 主控台日誌。
- 傳送「Windows 已可使用」訊息到 EC2 主控台。

如需 Windows Server 2019 的詳細資訊，請參閱 Microsoft.com 上的[比較 Windows Server 版本功能](#)。

## 遙測

遙測是其他資訊，可協 AWS 助您進一步瞭解您的需求、診斷問題，並提供可改善 AWS 服務體驗的功能。

EC2Launch 版本 1.3.2003498 及更新版本會收集遙測，例如用量指標和錯誤。這些資料是從執行 EC2Launch 的 Amazon EC2 執行個體中收集的資料。這包括所有的視窗 AMI 擁有。AWS

EC2Launch 會收集下列類型的遙測：

- 用量資訊 – 代理程式命令、安裝方法和排程的執行頻率。
- 錯誤和診斷資訊 – 代理程式安裝和執行錯誤代碼。

收集資料的範例：

```
2021/07/15 21:44:12Z: EC2LaunchTelemetry: IsAgentScheduledPerBoot=true
2021/07/15 21:44:12Z: EC2LaunchTelemetry: IsUserDataScheduledPerBoot=true
2021/07/15 21:44:12Z: EC2LaunchTelemetry: AgentCommandCode=1
2021/07/15 21:44:12Z: EC2LaunchTelemetry: AgentCommandErrorCode=5
2021/07/15 21:44:12Z: EC2LaunchTelemetry: AgentInstallCode=2
2021/07/15 21:44:12Z: EC2LaunchTelemetry: AgentInstallErrorCode=0
```

預設為啟用遙測。您可隨時停用遙測收集。如果啟用遙測，EC2Launch 會傳送遙測資料，而不會傳送其他通知給客戶。

您選擇啟用或停用遙測收集。

您可以選擇加入或退出遙測收集。您選取加入或退出遙測收集，以確保我們遵守您的遙測選項。

## 遙測可見性

啟用遙測時，它會在 Amazon EC2 主控台輸出中顯示，如下所示：

```
2021/07/15 21:44:12Z: Telemetry: <Data>
```

## 停用執行個體的遙測

若要藉由設定系統環境變數來停用遙測，請以管理員身分執行下列命令：



```
setx /M EC2LAUNCH_TELEMETRY 0
```

若要在安裝期間停用遙測，請執行 `install.ps1`，如下所示：

```
.\install.ps1 -EnableTelemetry:$false
```

## 安裝最新版本的 EC2Launch

使用下列程序可在執行個體上下載和安裝 EC2Launch 最新版本。

### 下載和安裝 EC2Launch 最新版本

1. 如果您已在執行個體上安裝和設定 EC2Launch，請備份 EC2Launch 組態檔案。安裝程序不會保留此檔案的變更內容。根據預設，檔案位於 `C:\ProgramData\Amazon\EC2-Windows\Launch\Config` 目錄。
2. 下載 [EC2-Windows-Launch.zip](#) 到執行個體上的目錄。
3. 下載 [install.ps1](#) 到您下載 `EC2-Windows-Launch.zip` 的同一個目錄。
4. 執行 `install.ps1`
5. 如果您備份了 EC2Launch 組態檔案，請將其複製到 `C:\ProgramData\Amazon\EC2-Windows\Launch\Config` 目錄。

### 若要使用下載並安裝最新版本的 EC2Launch PowerShell

如果您已在執行個體上安裝和設定 EC2Launch，請備份 EC2Launch 組態檔案。安裝程序不會保留此檔案的變更內容。根據預設，檔案位於 `C:\ProgramData\Amazon\EC2-Windows\Launch\Config` 目錄。

若要使用安裝最新版本的 EC2Launch PowerShell，請從視窗執行下列指令 PowerShell

```
mkdir $env:USERPROFILE\Desktop\EC2Launch
$url = "https://s3.amazonaws.com/ec2-downloads-windows/EC2Launch/latest/EC2-Windows-Launch.zip"
$DownloadZipFile = "$env:USERPROFILE\Desktop\EC2Launch\" + $(Split-Path -Path $url - Leaf)
Invoke-WebRequest -Uri $url -OutFile $DownloadZipFile
$url = "https://s3.amazonaws.com/ec2-downloads-windows/EC2Launch/latest/install.ps1"
$DownloadZipFile = "$env:USERPROFILE\Desktop\EC2Launch\" + $(Split-Path -Path $url - Leaf)
Invoke-WebRequest -Uri $url -OutFile $DownloadZipFile
```

```
& $env:USERPROFILE\Desktop\EC2Launch\install.ps1
```

### Note

如果您在下載檔案時收到錯誤訊息，而且您正在使用視窗伺服器 2016，則可能需要為您的 PowerShell 終端機啟用 TLS 1.2。您可以使用下列命令為目前 PowerShell 工作階段啟用 TLS 1.2，然後再試一次：

```
[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
```

查看 C:\ProgramData\Amazon\EC2-Windows\Launch 以驗證安裝。

確認 EC2Launch 版本

使用下列視窗 PowerShell 指令來驗證 EC2Launch 的已安裝版本。

```
PS C:\> Test-ModuleManifest -Path "C:\ProgramData\Amazon\EC2-Windows\Launch\Module\Ec2Launch.psd1" | Select Version
```

### EC2Launch 目錄結構

根據預設，EC2Launch 會安裝於根目錄 C:\ProgramData\Amazon\EC2-Windows\Launch 中的 Windows Server 2016 和更新版本 AMI。

### Note

根據預設，Windows 會將檔案和資料夾隱藏在 C:\ProgramData 下。若要檢視 EC2Launch 目錄和檔案，您必須在 Windows 檔案總管中輸入路徑或變更資料夾屬性，才可顯示隱藏的檔案和資料夾。

Launch 目錄包含下列子目錄。

- Scripts— 包含組成 EC2Launch 的 PowerShell 指令碼。
- Module – 含有建置 Amazon EC2 相關指令碼的模組。
- Config – 含有您可以自訂的指令碼組態檔案。
- Sysprep – 含有 Sysprep 資源。

- Settings – 含有 Sysprep 圖形使用者介面的應用程式。
- Library - 包含 EC2 啟動代理程式的共用程式庫。
- Logs – 含有指令碼產生的日誌檔案。

## EC2 啟動版本及更新版本 **1.3.2004592**

Administrators 群組的使用者擁有所有 EC2Launch 目錄的 Full control 權限。不在「管理員」群組中的使用者擁有除外的所有 EC2Launch 目錄的 Read & execute 權限。C:\ProgramData\Amazon\EC2-Windows\Launch\Module\Config 目錄僅限於屬於 Administrators 群組成員的使用者。

## EC2 啟動版本和更早版本 **1.3.2004491**

除了以外 C:\ProgramData，所有 EC2Launch 目錄都會繼承其權限。C:\ProgramData\Amazon\EC2-Windows\Launch\Module\Scripts 此資料夾會繼承建立 C:\ProgramData 時的所有初始權限，但會移除一般使用者在目錄 CreateFiles 中存取的權限。

## 設定 EC2Launch

在您的執行個體第一次初始化後，您可以設定 EC2Launch 再次執行並執行其他啟動任務。

### 工作

- [設定初始化任務](#)
- [排程 EC2Launch 在每次開機時都執行](#)
- [初始化磁碟機和映射磁碟機代號](#)
- [傳送 Windows 事件日誌到 EC2 主控台。](#)
- [在成功開機後傳送「Windows 已就緒」訊息](#)

## 設定初始化任務

在 LaunchConfig.json 檔案中指定設定以啟用或停用下列初始化任務：

- 將電腦名稱設定為執行個體私有 IPv4 地址。
- 將監視器設定為永遠保持開啟狀態。
- 設定新的桌面。
- 新增 DNS 尾碼清單。

**Note**

這會新增下列網域的 DNS 尾碼查詢，並設定其他標準尾碼。如需啟動代理程式如何設定 DNS 尾碼的詳細資訊，請參閱。[設定 Windows 啟動代理程式的 DNS 尾碼](#)

```
region.ec2-utilities.amazonaws.com
```

- 擴充開機磁碟區大小。
- 設定管理員密碼。

### 進行初始化設定

1. 在要設定的執行個體上，於文字編輯器中開啟下列檔案：C:\ProgramData\Amazon\EC2-Windows\Launch\Config\LaunchConfig.json。
2. 視需要更新下列設定並儲存變更。只有在 adminPassword 是 adminPasswordtype 時，才要在 Specify 中提供密碼。

```
{
 "setComputerName": false,
 "setMonitorAlwaysOn": true,
 "setWallpaper": true,
 "addDnsSuffixList": true,
 "extendBootVolumeSize": true,
 "handleUserData": true,
 "adminPasswordType": "Random | Specify | DoNothing",
 "adminPassword": "password that adheres to your security policy (optional)"
}
```

密碼類型的定義如下：

#### Random

EC2Launch 會產生密碼並以使用者金鑰將其加密。系統會在執行個體啟動後停用此設定，以便執行個體重新開機或停止再啟動時，能持續使用這個密碼。

## Specify

EC2Launch 會使用您在 `adminPassword` 中指定的密碼。如果該密碼不符合系統要求，EC2Launch 會改為產生隨機密碼。系統會將該密碼以純文字存放於 `LaunchConfig.json`，並在 Sysprep 設定管理員密碼後予以刪除。EC2Launch 會以使用者金鑰加密該密碼。

## DoNothing

EC2Launch 會使用您在 `unattend.xml` 檔案中指定的密碼。如果您未在 `unattend.xml` 中指定密碼，會停用管理員帳戶。

3. 在 Windows 中 PowerShell，執行下列命令來排程指令碼以 Windows 排程工作的形式執行。該指令碼會在下一次開機期間執行一次，然後停用這些任務，以避免再次執行。

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 - Schedule
```

## 排程 EC2Launch 在每次開機時都執行

您可排程 EC2Launch 在每次開機時都執行，不是只在初次開機時執行。

## 啟用 EC2Launch 在每次開機時都執行

1. 開啟視窗 PowerShell 並執行下列命令：

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 - SchedulePerBoot
```

2. 或使用下列命令執行可執行檔：

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Settings\Ec2LaunchSettings.exe
```

然後選取 Run EC2Launch on every boot。您可指定 EC2 執行個體 Shutdown without Sysprep 或 Shutdown with Sysprep。

### Note

當您啟用 EC2Launch 以便在每次開機時執行時，下次 EC2Launch 執行時會發生下列情況：

- 如果 AdminPasswordType 仍然設定為 Random，EC2Launch 將在下次開機時產生新密碼。開機之後，AdminPasswordType 會自動設定為 DoNothing，以防止 EC2Launch 在後續開機時產生新密碼。若要防止 EC2Launch 在第一次開機時產生新密碼，請手動將 AdminPasswordType 設定為 DoNothing，然後再重新開機。
- HandleUserData 會設回 false，除非使用者資料將 persist 設成 true。如需詳細資訊，請參閱 [the section called “使用者資料指令碼”](#)。

## 初始化磁碟機和映射磁碟機代號

在 DriveLetterMappingConfig.json 檔案中指定設定，以將磁碟機代號映射到您的 EC2 執行個體上的磁碟區。指令碼會初始化尚未初始化和分割的磁碟機。如需有關在 Windows 中取得磁碟區詳細資訊的更多詳細資訊，請參閱 Microsoft 文件中的 [Get-Volume](#)。

### 將磁碟機代號映射到磁碟區

1. 在文字編輯器中開啟 C:\ProgramData\Amazon\EC2-Windows\Launch\Config\DriveLetterMappingConfig.json 檔案。
2. 指定下列磁碟區設定並儲存您的變更：

```
{
 "driveLetterMapping": [
 {
 "volumeName": "sample volume",
 "driveLetter": "H"
 }
]
}
```

3. 開啟視窗，PowerShell 然後使用下列命令來執行 EC2Launch 指令碼，以初始化磁碟：

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeDisks.ps1
```

若要在每次執行個體開機時初始化磁碟，請新增 -Schedule 標記，如下所示：

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeDisks.ps1 -Schedule
```

傳送 Windows 事件日誌到 EC2 主控台。

在 EventLogConfig.json 檔案中指定設定，以傳送 Windows 事件日誌到 EC2 主控台日誌。

進行設定以傳送 Windows 事件日誌

1. 在執行個體上，於文字編輯器中開啟 C:\ProgramData\Amazon\EC2-Windows\Launch\Config\EventLogConfig.json 檔案。
2. 進行下列日誌設定並儲存您的變更：

```
{
 "events": [
 {
 "logName": "System",
 "source": "An event source (optional)",
 "level": "Error | Warning | Information",
 "numEntries": 3
 }
]
}
```

3. 在 Windows 中 PowerShell，執行下列命令，以便系統將指令碼排定為在每次執行個體開機時以 Windows 排程工作的形式執行。

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\SendEventLogs.ps1 -
Schedule
```

日誌可能要花費三分鐘 (含) 以上才會顯示在 EC2 主控台日誌中。

在成功開機後傳送「Windows 已就緒」訊息

EC2Config 服務會在每次開機後，傳送「Windows 已就緒」訊息到 EC2 主控台。EC2Launch 只會在初次開機後傳送這個訊息。若要與 EC2Config 服務回溯相容，您可以排程 EC2Launch 在每次開機後傳送這個訊息。在執行個體上，開啟 Windows PowerShell 並執行下列命令。系統會排程指令碼做為 Windows 排程任務執行。

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\SendWindowsIsReady.ps1 -
Schedule
```

## EC2Launch 版本歷史記錄

自 Windows Server 2016 起，Windows AMI 包含一組 Windows PowerShell 指令碼，稱為 EC2Launch。EC2Launch 會在初始執行個體開機期間執行任務。如需 Windows AMI 中包含的 EC2 啟動版本的相關資訊，請參閱 [AWS 視窗 AMI 版本歷程記錄](#)。

若要下載和安裝 EC2Launch 的最新版本，請參閱 [安裝最新版本的 EC2Launch](#)。

下表說明 EC2Launch 發行版本。請注意，版本格式在 1.3.610 版後已變更。

版本	詳細資訊	發行日期
1.3.2004891	<ul style="list-style-type: none"> <li>修正未按預期設定的 false 問題。HandleUserData</li> <li>已將 Encrypted 密碼選項新增至 LaunchConfig.json。</li> <li>已變更預設 Settings UI 為加密使用者指定密碼的行為。</li> <li>添 SetAdminPasswordConfig.ps1 加了將 Specify 密碼選項轉換為代理程序配置文件中的 Encrypted 密碼選項。</li> </ul>	2024 年五月三十一日
1.3.2004617	<ul style="list-style-type: none"> <li>修正了設置壁紙時的錯誤。</li> </ul>	2024 年一月十五日
1.3.2004592	<ul style="list-style-type: none"> <li>已更新 install.ps1 為 %ProgramData%\Amazon\EC2-Windows\Launch 設定的存取許可。</li> <li>已將 EC2Launch 資料夾/檔案存取限制為 read-execute，僅適用於標準使用者帳戶。</li> <li>已變更代理程式以停止等待執行個體中繼資料服務 (IMDS) 來初始化 (若沒有為執行個體啟用 IMDS)。</li> <li>在等待 IMDS 初始化時新增了五分鐘的逾時時間。</li> <li>將代理程式變更為在 Windows is Ready 訊息之前 (而非之後) 將遙測寫入執行個體主控台日誌。</li> </ul>	2024 年 1 月 2 日



版本	詳細資訊	發行日期
	<p>為數種新的執行個體類型新增了桌布支援。</p> <p>如需 EC2Launch 目錄的存取權限和使用者帳戶權限的詳細資訊，請參閱。<a href="#">the section called “EC2Launch 目錄結構”</a></p>	
1.3.2004491	<ul style="list-style-type: none"> <li>新增遙測以監控指定管理員密碼選項的使用情況。</li> </ul>	2023 年 11 月 9 日
1.3.2004462	<ul style="list-style-type: none"> <li>在每次寫入序列主控台後新增排清。</li> </ul>	2023 年 10 月 18 日
1.3.2004438	<ul style="list-style-type: none"> <li>根據登錄項目限制網域名稱轉移：HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Dnscache\Parameters\DomainNameDevolutionLevel。</li> <li>僅將 UserdataExecution.log 許可限制給 Administrators。</li> <li>當日誌初始化失敗時，在 Windows Event Log 中新增錯誤訊息。</li> </ul>	2023 年 10 月 4 日
1.3.2004256	<ul style="list-style-type: none"> <li>為主控台日誌新增 EnableSCSIPersistentReservations 值。</li> <li>為 Get-ConsolePort 新增重試功能。</li> </ul>	2023 年 7 月 7 日
1.3.2004052	<ul style="list-style-type: none"> <li>修正在執行個體啟動時未指定 SSH 金鑰時出現的錯誤。</li> <li>已更新為在失敗時重試啟動 AmazonSSMAgent Windows 服務。</li> <li>如果 SysprepInstance.BeforeSysprep cmd 因非零結束代碼而失敗，則更新為失敗 .ps1。</li> </ul>	2023 年 3 月 8 日

版本	詳細資訊	發行日期
1.3.2003975	<ul style="list-style-type: none"> <li>修正影響 Packer AMI 建置 (SysprepInstance.ps1 傳回 1 的 \$LastErrorCode ) 的問題。</li> </ul>	2022 年 12 月 24 日
1.3.2003961	<ul style="list-style-type: none"> <li>已修正快速啟動執行個體上明確指定的管理員密碼會被隨機密碼覆寫的問題。</li> <li>已修正 SSM Agent 無法在較小的執行個體類型上啟動的問題。</li> <li>已修正執行個體主控台日誌包含 RDPCERTIFICATE-THUMBPRINT: 000000000000000000000000 而非有效 RDP 憑證指紋值的問題。</li> </ul>	2022 年 12 月 6 日
1.3.2003923	<ul style="list-style-type: none"> <li>修正 PnPDeviceID 為空白時尋找網路轉接器的邏輯。</li> </ul>	2022 年 11 月 9 日
1.3.2003919	<ul style="list-style-type: none"> <li>已更新取得ConsolePort 以使用 PCI 區段資訊。</li> <li>已修正在重新啟動後可能會選取錯誤網路轉接器的問題。</li> <li>已修正 start-SSM-Agent 逾時邏輯。</li> <li>修正了 Send-AdminCredentials 函數別名的向後兼容性。</li> </ul>	2022 年 11 月 8 日
1.3.2003857	<ul style="list-style-type: none"> <li>選取主要網路轉接器時，應優先考慮具有預設閘道的轉接器。</li> <li>擴充了記憶體內密碼加密。</li> </ul>	2022 年 10 月 3 日
1.3.2003824	<ul style="list-style-type: none"> <li>已修正 setComputerName 期間的錯誤。</li> <li>新增在偵測到 BYOL 帳單代碼時略過 Windows 啟用的邏輯。</li> <li>新增記憶體內密碼加密。</li> <li>已修正 m6id.4xlarge 磁碟區初始化期間的錯誤。</li> </ul>	2022 年 8 月 30 日

版本	詳細資訊	發行日期
1.3.2003691	<ul style="list-style-type: none"> <li>更新了 IMDS 等待邏輯，以僅發出 IMDSv2 請求。</li> <li>修正了影響 eGPU 安裝的錯誤。</li> </ul>	2022 年 6 月 21 日
1.3.2003639	<ul style="list-style-type: none"> <li>新增了網路轉接器等待邏輯，以防止在初始化之前使用。</li> <li>已修正輕微的問題。</li> </ul>	2022 年 5 月 10 日
1.3.2003498	<ul style="list-style-type: none"> <li>新增的遙測。</li> <li>新增了設定 UI 的捷徑。</li> <li>格式化 PowerShell 腳本。</li> <li>修復了 BeforeSysprep .cmd 完成之前發生關機的問題。</li> </ul>	2022 年 1 月 31 日
1.3.2003411	已變更密碼產生邏輯，以排除複雜性低的密碼。	2021 年 8 月 4 日
1.3.2003364	更新了安裝-EgpuManager 具有 IMDSv2 支持。	2021 年 6 月 7 日
1.3.2003312	<ul style="list-style-type: none"> <li>在 setMonitorAlwaysOn 設定前後新增日誌行。</li> <li>將 AWS 硝基飛地包版本添加到控制台日誌中。</li> </ul>	2021 年 5 月 4 日
1.3.2003284	透過更新將使用者資料存放至 LocalAppData 的位置，來改善許可模型。	2021 年 3 月 23 日
1.3.2003236	<ul style="list-style-type: none"> <li>已更新設定使用者密碼的方法於 Set-AdminAccount 和 Randomize-LocalAdminPassword 中。</li> <li>修正 InitializeDisks，以在將磁碟設定為可寫入之前檢查是否已設定為唯讀。</li> </ul>	2021 年 2 月 11 日
1.3.2003210	install.ps1 的本地化修正程式。	2021 年 1 月 7 日

版本	詳細資訊	發行日期
1.3.2003205	更新 %ProgramData%AmazonEC2-WindowsLaunch ModuleScripts 目錄權限的 install.ps1 安全性修正程式。	2020 年 12 月 28 日
1.3.2003189	新增路由後新增 w32tm resync。	2020 年 12 月 4 日
1.3.2003155	已更新執行個體類型資訊。	2020 年 8 月 25 日
1.3.2003150	新增 OsCurrentBuild 和 OsReleaseId 至主控台輸出。	2020 年 4 月 22 日
1.3.2003040	修正 IMDS 第 1 版備用邏輯。	2020 年 4 月 7 日
1.3.2002730	已新增對 IMDS V2 的支援。	2020 年 3 月 3 日
1.3.2002240	已修正輕微的問題。	2019 年 10 月 31 日
1.3.2001660	修正在第一次執行 Sysprep 後未使用密碼的使用者自動登入的問題。	2019 年 7 月 2 日
1.3.2001360	已修正輕微的問題。	2019 年 3 月 27 日
1.3.2001220	所有 PowerShell 指令碼已簽署。	2019 年 2 月 28 日
1.3.2001200	已修正 InitializeDisks .ps1 的問題：在 Windows Server 容錯移轉叢集中的節點上執行指令碼，會將磁碟機代號與本機磁碟機代號相符的遠端節點上的磁碟機格式化。	2019 年 2 月 27 日
1.3.2001160	已修正 Windows 2019 中遺漏桌布的問題。	2019 年 2 月 22 日

版本	詳細資訊	發行日期
1.3.2001040	<ul style="list-style-type: none"> <li>新增將顯示器設定為永遠不關閉的外掛程式，以修復 ACPI 問題。</li> <li>寫入主控制台的 SQL Server 版本。</li> </ul>	2019 年 1 月 21 日
1.3.2000930	修正在已啟用 ipv6 的 ENI 上新增路由到中繼資料的問題。	2019 年 1 月 2 日
1.3.2000760	<ul style="list-style-type: none"> <li>RSS 的新增預設組態和 ENA 裝置的接收佇列設定。</li> <li>Sysprep 時停用休眠。</li> </ul>	2018 年 12 月 5 日
1.3.2000630	<ul style="list-style-type: none"> <li>DNS 伺服器新增路由 169.254.169.253/32。</li> <li>設定管理使用者的新增篩選條件</li> <li>執行個體休眠改進。</li> <li>排程 EC2Launch 在每次開機時都執行的新增選項。</li> </ul>	2018 年 11 月 9 日
1.3.2000430.0	<ul style="list-style-type: none"> <li>已新增路由 169.254.169.123/32 至 AMZN 時間服務。</li> <li>已新增路由 169.254.169.249/32 至 GRID 授權服務。</li> <li>嘗試啟動 Systems Manager 時新增了 25 秒的逾時。</li> </ul>	2018 年 9 月 19 日
1.3.200039.0	<ul style="list-style-type: none"> <li>已修正 EBS NVME 磁碟區不正確的磁碟機代號。</li> <li>已新增 NVME 驅動程式版本的其他記錄。</li> </ul>	2018 年 8 月 15 日
1.3.2000080	已修正輕微的問題。	
1.3.610	已修正將輸出和錯誤重新導向到使用者資料之檔案的問題。	
1.3.590	<ul style="list-style-type: none"> <li>在桌面中新增了缺少的執行個體類型。</li> <li>已修正磁碟機代號映射和磁碟安裝的問題。</li> </ul>	
1.3.580	<ul style="list-style-type: none"> <li>已修正 Get-Metadata ，以對 Web 請求使用預設系統代理設定。</li> <li>在磁碟初始化中新增了 NVMe 的特殊案例。</li> <li>已修正輕微的問題。</li> </ul>	

版本	詳細資訊	發行日期
1.3.550	新增了 <code>-NoShutdown</code> 選項以啟用 Sysprep 而不關機。	
1.3.540	已修正輕微的問題。	
1.3.530	已修正輕微的問題。	
1.3.521	已修正輕微的問題。	
1.3.0	<ul style="list-style-type: none"><li>已修正電腦名稱變更的十六進位長度問題。</li><li>已修正電腦名稱變更可能發生重新開機迴圈的問題。</li><li>已修正桌面設定中的問題。</li></ul>	
1.2.0	<ul style="list-style-type: none"><li>更新以顯示有關 EC2 系統日誌中安裝之作業系統 (OS) 的資訊。</li><li>更新以顯示 EC2 系統日誌中的 EC2Launch 和 SSM Agent 版本。</li><li>已修正輕微的問題。</li></ul>	

版本	詳細資訊	發行日期
1.1.2	<ul style="list-style-type: none"> <li>更新以顯示 EC2 系統日誌中的 ENA 驅動程式資訊。</li> <li>更新以從主要 NIC 篩選邏輯排除 Hyper-V。</li> <li>新增 AWS KMS 伺服器端和連接埠到登錄機碼以進行 KMS 啟用。</li> <li>已改善多位使用者的桌面設定。</li> <li>更新以清除來自持久性存放區的路由。</li> <li>更新以從 DNS 尾碼清單中的可用區域移除 z。</li> <li>更新以解決使用者資料中 &lt; 執行AsLocal系統 &gt; 標籤的問題。</li> </ul>	
1.1.1	初始版本。	

## 使用 EC2Config 服務 (舊版) 設定 Windows 執行個體

### Note

EC2Config 文件僅供歷史參考之用。它執行的作業系統版本已不再受到 Microsoft 的支援。強烈建議您升級至最新的啟動服務。

所有 Windows Server 2022 的最新啟動服務是 [EC2Launch v2](#)，它會取代 EC2Config 和 EC2Launch。

適用於視窗伺服器 2016 年之前版本的視窗 AMI 包含選用的服務，即 EC2Config 服務 (`EC2Config.exe`)。EC2Config 會在啟動期間啟動執行個體及執行作業時及每次您停止或啟動執行個體時啟動。EC2Config 也可依據需求執行作業。其中部分作業為自動啟用，其他作業則必須手動啟用。雖然此項服務為選用，其中可存取專屬的進階功能。此服務會在 LocalSystem 帳戶中執行。

**Note**

EC2Launch 已取代 Windows Server 2016 和 2019 的 Windows AMI 上的 EC2Config。如需詳細資訊，請參閱 [使用 EC2Launch 設定 Windows 執行個體](#)。所有受支援 Windows Server 版本的最新啟動服務是 [EC2Launch v2](#)，它會取代 EC2Config 和 EC2Launch。

EC2Config 使用設定檔案控制其運作。您可使用圖形工具或直接編輯 XML 檔案，以更新這些設定檔案。服務的二進位檔案和其他檔案包含在 %ProgramFiles%\Amazon\EC2ConfigService 目錄中。

**內容**

- [EC2Config 作業](#)
- [安裝最新版本的 EC2Config](#)
- [停止、重新啟動、刪除或解除安裝 EC2Config](#)
- [設定和 AWS Systems Manager](#)
- [EC2Config 和 Sysprep](#)
- [EC2 服務屬性](#)
- [EC2Config 設定檔](#)
- [設定 EC2Config 服務的代理伺服器設定](#)
- [EC2Config 版本記錄](#)
- [故障診斷使用 EC2Config 服務的問題](#)

**EC2Config 作業**

EC2Config 會在執行個體初次啟動時執行初始啟動作業，接著停用執行個體。若要重新執行這些作業，您必須先明確啟用執行個體，然後再關閉執行個體，或手動執行 Sysprep。這些作業如下：

- 設定管理員帳戶的隨機加密密碼。
- 產生並安裝遠端桌面連線所用的主機憑證。
- 動態擴充作業系統的分割區，以包含任何尚未分割的空間。
- 執行指定的使用者資料 (和 Cloud-Init，如果尚未安裝)。如需指定使用者資料的詳細資訊，請參閱 [使用執行個體使用者資料](#)。



EC2Config 會在執行個體每次啟動時執行下列作業：

- 變更主機名稱，以符合 Hex 表示法中的私有 IP 地址 (預設停用此作業，但必須啟用才能在執行個體啟動時執行)。
- 設定金鑰管理伺服器 (AWS KMS)，檢查 Windows 啟用狀態，並視需要啟動 Windows。
- 掛載所有的 Amazon EBS 磁碟區和執行個體存放區磁碟區，然後將磁碟區名稱對應至磁碟機代號。
- 將事件日誌項目寫入主控台，以協助故障診斷 (預設停用此作業，但必須啟用才能在執行個體啟動時執行)。
- 將 Windows 已就緒的狀態寫入主控台。
- 新增自訂路由表至主要網路轉接器，以在連接單個 NIC 或多個 NIC 時啟用下列的 IP 地址：169.254.169.250、169.254.169.251 和 169.254.169.254。這些地址供 Windows 啟用和當您要存取執行個體中繼資料時使用。

#### Note

如果 Windows 作業系統設定為使用 IPv4，則可以使用這些 IPv4 link-local 地址。  
如果 Windows 作業系統已停用 IPv4 網路通訊協定堆疊，並改用 IPv6，請新增 [fd00:ec2::240] 取代 169.254.169.250 和 169.254.169.251。然後新增 [fd00:ec2::254] 取代 169.254.169.254。

EC2Config 會在每次有使用者登入時執行下列作業：

- 在桌面背景顯示桌布資訊。

執行個體執行時，您可要求 EC2Config 隨需執行下列作業：

- 執行 Sysprep 並關閉執行個體，如此才能從中建立 AMI。如需詳細資訊，請參閱 [使用視窗系統建立 AMI](#)。

## 安裝最新版本的 EC2Config

根據預設，EC2Config 包含在 Windows Server 2016 之前版本的 AMI 內。當 EC2Config 服務更新時，新的視窗 AMI 會 AWS 包含最新版本的服務。不過，您需要更新自己的 Windows AMI 和執行個體為最新版的 EC2Config。

**Note**

EC2Launch 將取代 Windows Server 2016 和 2019 AMI 上的 EC2Config。如需詳細資訊，請參閱 [使用 EC2Launch 設定 Windows 執行個體](#)。所有受支援 Windows Server 版本的最新啟動服務是 [EC2Launch v2](#)，它會取代 EC2Config 和 EC2Launch。

如需如何接收 EC2Config 更新通知的詳細資訊，請參閱 [訂閱 EC2Config 服務通知](#)。如需各版本變更的詳細資訊，請參閱 [EC2Config 版本記錄](#)。

**開始之前**

- 確認您擁有 .NET Framework 3.5 SP1 或更新版本。
- 根據預設，安裝程式會在安裝期間將您的設定檔換成預設設定，並在安裝完成後重新啟動 EC2Config 服務。如果變更 EC2Config 服務設定，請從 config.xml 目錄複製 %Program Files%\Amazon\Ec2ConfigService\Settings 檔案。更新 EC2Config 服務後，您可還原此檔案，以保留設定變更。
- 如果您的 EC2Config 版本早於 2.1.19 版，而您要安裝 2.2.12 版或更早版本，您必須先安裝 2.1.19 版。若要安裝 2.1.19 版，請下載 [EC2Install\\_2.1.19.zip](#)，解壓縮檔案，然後執行 EC2Install.exe。

**Note**

如果您的 EC2Config 版本早於 2.1.19 版，而您要安裝 2.3.313 版或更新版本，則您可直接安裝，無需先安裝 2.1.19 版。

**確認 EC2Config 版本**

請用下列程序確認執行個體上安裝的 EC2Config 版本。

**確認安裝的 EC2Config 版本**

1. 從 AMI 啟動執行個體，並連線至執行個體。
2. 從 Control Panel (控制台) 中，選取 Programs and Features (程式和功能)。
3. 從已安裝程式清單中，找出 Ec2ConfigService。其版本編號將出現在 Version (版本) 欄中。

## 更新 EC2Config

請用下列程序在執行個體上下載並安裝最新版的 EC2Config。

### 下載並安裝最新版的 EC2Config

1. 下載並解壓縮 [EC2Config 安裝程式](#)。
2. 執行 EC2Install.exe。如需完整的選項清單，請執行 EC2Install 並加入 /? 選項。根據預設，設定程式會顯示提示。若要執行命令且不加入提示，請使用 /quiet 選項。

#### Important

若要保留所儲存之 config.xml 檔案中的自訂設定，請執行 EC2Install 並加入 /norestart 選項，還原設定，然後手動重新啟動 EC2Config 服務。

3. 如果您執行的是 EC2Config 4.0 版或更新版本，您必須從 Microsoft Services 內嵌在執行個體上重新開始 SSM Agent。

#### Note

您重新啟動或者停止並啟動執行個體後，執行個體系統日誌或 Trusted Advisor 檢查中才會顯示更新的 EC2Config 版本資訊。

若要使用下載並安裝最新版本的 EC2Config PowerShell

若要使用下載、解壓縮和安裝最新版本的 EC2Config PowerShell，請從視窗執行下列命令：  
PowerShell

```
$Url = "https://s3.amazonaws.com/ec2-downloads-windows/EC2Config/EC2Install.zip"
$DownloadZipFile = "$env:USERPROFILE\Desktop\" + $(Split-Path -Path $Url -Leaf)
$ExtractPath = "$env:USERPROFILE\Desktop\"
Invoke-WebRequest -Uri $Url -OutFile $DownloadZipFile
$ExtractShell = New-Object -ComObject Shell.Application
$ExtractFiles = $ExtractShell.Namespace($DownloadZipFile).Items()
$ExtractShell.NameSpace($ExtractPath).CopyHere($ExtractFiles)
Start-Process $ExtractPath
Start-Process `
 -FilePath $env:USERPROFILE\Desktop\EC2Install.exe `
 -ArgumentList "/S"
```

**Note**

如果您在下載檔案時收到錯誤訊息，而且您使用的是 Windows 伺服器 2016 或更早版本，則可能需要為您的 PowerShell 終端機啟用 TLS 1.2。您可以使用下列命令為目前 PowerShell 工作階段啟用 TLS 1.2，然後再試一次：

```
[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
```

查看 C:\Program Files\Amazon\ 是否有 Ec2ConfigService 目錄以驗證安裝。

### 停止、重新啟動、刪除或解除安裝 EC2Config

您可用跟任何其他服務一樣的方式來管理 EC2Config 服務。

若要將更新設定套用至執行個體，您可停止並重新啟動服務。如果正在手動安裝 EC2Config，必須先停止服務。

#### 停止 EC2Config 服務

1. 啟動和連線至 Windows 執行個體。
2. 在 Start (開始) 選單上，指向 Administrative Tools (管理工具)，然後按一下 Services (服務)。
3. 從服務清單中，在 EC2Config 上按一下滑鼠右鍵，然後選取 Stop (停止)。

#### 重新啟動 EC2Config 服務

1. 啟動和連線至 Windows 執行個體。
2. 在 Start (開始) 選單上，指向 Administrative Tools (管理工具)，然後按一下 Services (服務)。
3. 從服務清單中，在 EC2Config 上按一下滑鼠右鍵，然後選取 Restart (重新啟動)。

如果不需要更新設定、建立您自己的 AMI 或使用 AWS Systems Manager，您可刪除並解除安裝服務。移除服務將移除其登錄子機碼。解除安裝服務將移除檔案、登錄子機碼和任何服務捷徑。

#### 刪除 EC2Config 服務

1. 開啟命令提示字元視窗。
2. 執行以下命令：

```
sc delete ec2config
```

## 解除安裝 EC2Config

1. 啟動和連線至 Windows 執行個體。
2. 在 Start (開始) 選單上，按一下 Control Panel (控制台)。
3. 按兩下 Programs and Features (程式和功能)。
4. 在程式清單上選取 EC2 ConfigService，然後按一下 [解除安裝]。

## 設定和 AWS Systems Manager

EC2Config 服務會處理 Windows Server 2016 先前 Windows Server 版本 (2016 年 11 月之前發佈) 之 AMI 所建立執行個體的 Systems Manager 請求。

2016 年 11 月之後發佈的 Windows Server 2016 先前 Windows Server 版本的 AMI 所建立的執行個體，包含 EC2Config 服務和 SSM Agent。EC2Config 將執行先前所述的所有作業，而且 SSM Agent 將處理 Systems Manager 功能的請求，例如執行命令和狀態管理員。

您可使用執行命令，將使用的現有執行個體升級到 EC2Config 服務和 SSM Agent 的最新版本。如需詳細資訊，請參閱《AWS Systems Manager 使用者指南》中的 [使用執行命令更新 SSM Agent](#)。

## EC2Config 和 Sysprep

EC2Config 服務將執行 Sysprep，此為 Microsoft 的工具，可讓您建立自訂且可重複使用的 Windows AMI。EC2Config 呼叫 Sysprep 時，會使用 %ProgramFiles%\Amazon\EC2ConfigService\Settings 中的檔案來決定要執行的操作。您可間接用 EC2 Service Properties (EC2 服務屬性) 對話方塊編輯這些檔案，或直接用 XML 編輯器或文字編輯器編輯這些檔案。但有些進階設定無法在 Ec2 Service Properties (Ec2 服務屬性) 對話方塊中使用，因此您必須直接編輯這些項目。

如果您在更新執行個體設定後從執行個體建立 AMI，新設定將套用到從新 AMI 啟用的任何執行個體。如需有關建立 AMI 的詳細資訊，請參閱 [創建一個 Amazon EBS-backed AMI](#)。

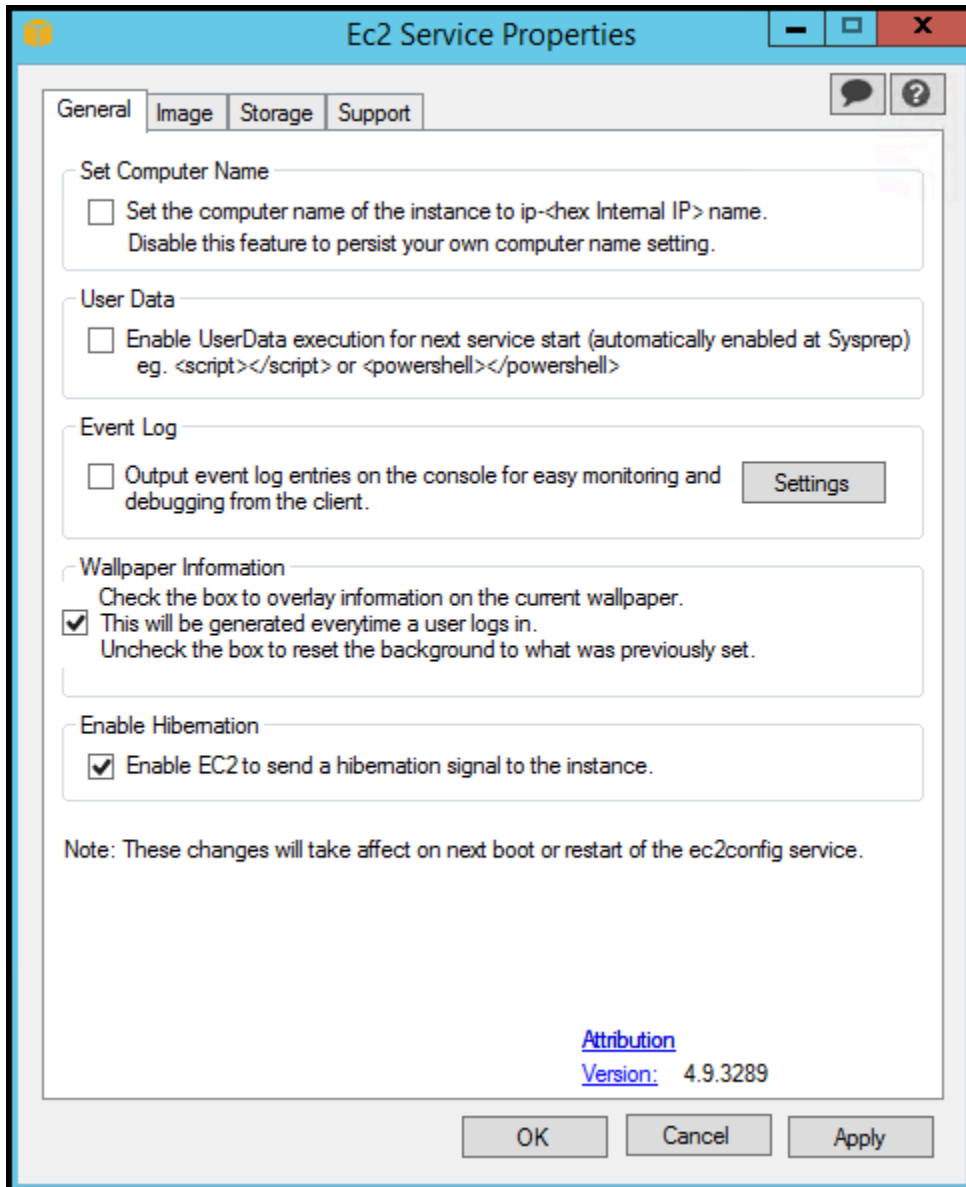
## EC2 服務屬性

下列程序說明如何使用 Ec2 Service Properties (Ec2 服務屬性) 對話方塊啟用或停用設定。

### 使用 Ec2 Service Properties (Ec2 服務屬性) 對話方塊變更設定

1. 啟動和連線至 Windows 執行個體。

- 從 [開始] 功能表中，按一下 [所有程式]，然後按一下 [EC2 ConfigService 設定]。



- 在 EC2 Service Properties (EC2 服務屬性) 對話方塊的 General (一般) 索引標籤中，啟用或停用下列設定。

#### Set Computer Name (設定電腦名稱)

如果啟用此設定 (預設為停用)，每次開機時會將主機名稱與目前的內部 IP 地址比對；如果主機名稱與內部 IP 地址不符，將重設主機名稱以包含內部 IP 地址，接著系統會重新啟動，以選用新的主機名稱。若要自行設定主機名稱，或避免現有的主機名稱遭到修改，請勿啟用此設定。

## 使用者資料

使用者資料執行可讓您指定執行個體中繼資料中的指令碼。根據預設，這些指令碼會在初始啟動時執行。您也可將其設定為在下次重新啟動或啟動執行個體時，或每次重新啟動或啟動執行個體時執行。

如果指令碼較大，建議透過使用者資料下載指令碼，然後執行指令碼。

如需詳細資訊，請參閱 [使用者資料執行](#)。

## 事件日誌

使用此設定可於開機時在主控台上顯示事件日誌項目，以方便進行監控和除錯。

按一下 Settings (設定)，指定要傳送至主控台之日誌項目的篩選條件。預設篩選條件會傳送系統事件日誌中最近的三個錯誤項目至主控台。

## Wallpaper Information (桌布資訊)

使用此設定可在桌面背景顯示系統資訊。以下是顯示在桌面背景上的資訊範例。

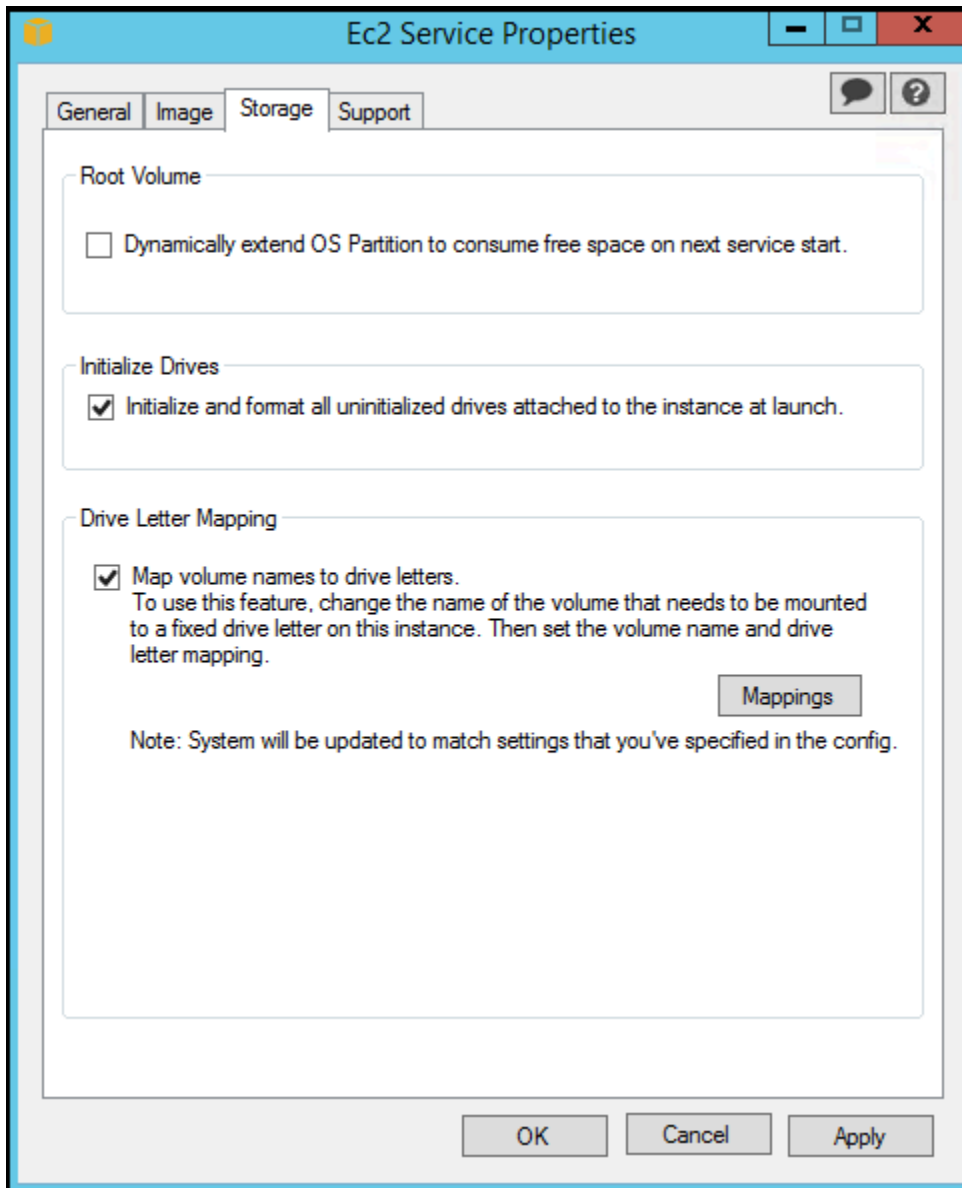
```
Hostname : WIN-U0RFOJCTPUU
Instance ID : i-d583f76a
Public IP Address : 54.208.43.227
Private IP Address : 172.31.42.195
Availability Zone : us-east-1b
Instance Size : t2.micro
Architecture : AMD64
```

顯示在桌面背景上的資訊由設定檔 EC2ConfigService\Settings\WallpaperSettings.xml 控制。

## 啟用休眠

使用此設定，允許 EC2 向作業系統發出執行休眠的信號。

- 按一下 Storage (儲存) 索引標籤。您可以啟用或停用以下設定。



### Root Volume (根磁碟區)

此設定將動態擴充 Disk 0/Volume 0，以包含任何尚未分割的空間。此設定在執行個體是從自訂大小的根裝置磁碟區啟動時很有用。

### Initialize Drives (初始化磁碟機)

此設定會在啟動時格式化及掛載連接至執行個體的所有磁碟區。

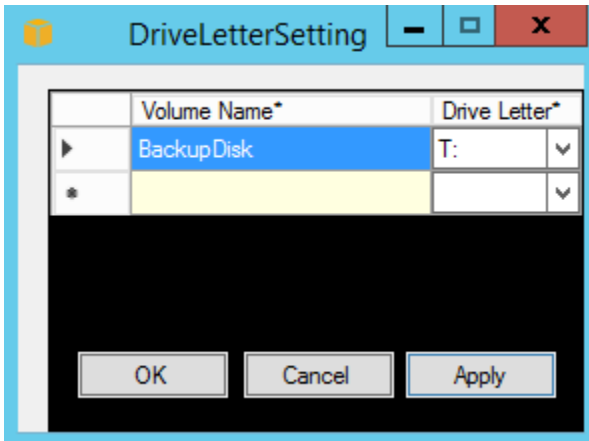
### Drive Letter Mapping (磁碟機代號映射)

系統會對應連接至執行個體的磁碟區與磁碟機代號。如為 Amazon EBS 磁碟區，預設為指派 D: 到 Z: 的磁碟機代號。對於實例存儲卷，默認值取決於驅動程序。AWS PV 驅動程序和



Citrix PV 驅動程式會指派執行個體儲存磁碟區磁碟機代號，從 Z: 到 A:。Red Hat 驅動程式會為執行個體存放區磁碟區指派 D: 到 Z: 的磁碟機代號。

若要為磁碟區選擇磁碟機代號，請按一下 Mappings (映射)。在 [DriveLetter設定] 對話方塊中，指定每個磁碟區的磁碟區名稱和磁碟機代號值，按一下 [套用]，然後按一下 [確定]。建議選取不會與可能正在使用中的磁碟機代號衝突的代號，例如選取較中間的英文字母做為磁碟機代號。



指定磁碟機代號映射，並連接含有與您指定的其中一個磁碟區名稱相同標籤的磁碟區後，EC2Config 便會自動指派指定的磁碟機代號給該磁碟區。但假如磁碟機代號已在使用中，磁碟機代號映射便會失敗。請注意，EC2Config 不會變更在您指定磁碟機代號映射時已連接之磁碟區的磁碟機代號。

5. 若要儲存設定，等日後再繼續設定，請按一下 OK (確定) 關閉 EC2 Service Properties (EC2 服務屬性) 對話方塊。如果已完成執行個體的自訂且想從該執行個體建立 AMI，請參閱[使用視窗系統建立 AMI](#)。

## EC2Config 設定檔

設定檔可控制 EC2Config 服務的操作。這些檔案位在 C:\Program Files\Amazon\Ec2ConfigService\Settings 目錄：

- ActivationSettings.xml - 使用金鑰管理伺服器 (AWS KMS) 控制產品啟用。
- AWS.EC2.Windows.CloudWatch.json— 控制要傳送到哪些效能計數器，以 CloudWatch 及要傳送至記錄的 CloudWatch 記錄檔。
- BundleConfig.xml—控制 EC2Config 如何準備執行個體後端執行個體以用於建立 AMI。
- Config.xml—控制主要設定。
- DriveLetterConfig.xml—控制磁碟機代號映射。

- EventLogConfig.xml—控制執行個體開機時顯示在主控台上的事件日誌資訊。
- WallpaperSettings.xml—控制顯示在桌面背景上的資訊。

## ActivationSettings.xml

此檔案包含控制產品啟用的設定。Windows 開機時，EC2Config 服務會檢查 Windows 是否已啟動。如果 Windows 尚未啟動，服務會搜尋指定的 AWS KMS 伺服器，嘗試啟動 Windows。

- SetAutodiscover - 表示是否自動偵測 AWS KMS 。
- TargetKMServer 儲存的私人 IP 位址。AWS KMS 和您的執行個體必須在相同的區域。
- DiscoverFromZone 從指定的 DNS 區域探索 AWS KMS 伺服器。
- ReadFromUserData 從 UserData 取得 AWS KMS 伺服器。
- LegacySearchZones 從指定的 DNS 區域探索 AWS KMS 伺服器。
- DoActivate—使用區段中的指定設定嘗試啟用。此值可以為 true 或 false。
- LogResultToConsole—將結果顯示在主控台。

## BundleConfig.xml

此檔案包含控制 EC2Config 如何準備執行個體以用於建立 AMI 的設定。

- AutoSysprep—指出是否自動使用 Sysprep。變更值為 Yes 將使用 Sysprep。
- SetRDPCertificate—設定遠端桌面伺服器的自我簽署憑證。這可安全地傳送 RDP 至執行個體。如果新執行個體應取得該憑證，請變更值為 Yes。

此設定不適用於 Windows Server 2016 以前版本的作業系統執行個體，因為它們可以產生自己的憑證。

- SetPasswordAfterSysprep—在新啟動的執行個體上設定隨機密碼，以使用者啟動金鑰加密，然後將加密的密碼輸出到主控台。如果新執行個體不應設定為隨機加密密碼，請變更此設定的值為 No。

## Config.xml

### 外掛程式

- **Ec2SetPassword**—每次啟動執行個體時產生隨機加密密碼。此功能自初次啟動後將預設停用，因此重新啟動本執行個體不會變更使用者設定的密碼。變更此設定為 `Enabled`，將繼續在每次啟動執行個體時產生密碼。

如果您計劃要從執行個體建立 AMI，此設定極為重要。

- **Ec2SetComputerName**—依執行個體的 IP 地址將執行個體的主機名稱設為唯一名稱，並重新啟動執行個體。若要自行設定主機名稱，或避免現有的主機名稱遭到修改，您必須停用此設定。
- **Ec2InitializeDrives**—在啟動期間初始化及格式化所有磁碟區。此功能預設為啟用。
- **Ec2EventLog**—在主控制台顯示事件日誌項目。根據預設，會顯示系統事件日誌中最近的三個錯誤項目。若要指定要顯示的事件日誌項目，請編輯 `EventLogConfig.xml` 目錄內的 `EC2ConfigService\Settings` 檔案。如需此檔案中設定的詳細資訊，請參閱 MSDN 程式庫中的 [Eventlog 金鑰](#)。
- **Ec2ConfigureRDP**—在執行個體上設定自我簽署憑證，讓使用者可使用遠端桌面安全存取執行個體。此設定不適用於 Windows Server 2016 以前版本的作業系統執行個體，因為它們可以產生自己的憑證。
- **Ec2OutputRDPcert**—顯示遠端桌面憑證資訊至主控台，讓使用者可對照指紋進行確認。
- **Ec2SetDriveLetter**—依使用者定義的設定來設定掛載磁碟區的磁碟機代號。根據預設，Amazon EBS 磁碟區掛載至執行個體時，可用執行個體上的磁碟機代號掛載。若要指定磁碟機代號映射，請編輯 `DriveLetterConfig.xml` 目錄內的 `EC2ConfigService\Settings` 檔案。
- **Ec2WindowsActivate**—外掛程式將處理 Windows 啟用。外掛程式會檢查 Windows 是否啟動。如果沒有，它會更新用 AWS KMS 戶端設定，然後啟動 Windows。

若要修改 AWS KMS 設定，請編輯位於 `EC2ConfigService\Settings` 目錄中的 `ActivationSettings.xml` 檔案。

- **Ec2DynamicBootVolumeSize**—擴充 `Disk 0/Volume 0`，以包含任何尚未分割的空間。
- **Ec2HandleUserData**—在 `Sysprep` 執行後建立並執行使用者於初次啟動執行個體時建立的指令碼。包裝在腳本標籤中的命令將保存到批處理文件中，並將包裝在 PowerShell 標籤中的命令保存到 `.ps1` 文件中（與「EC2 服務屬性」對話方塊中的「用戶數據」核取方塊相對應）。
- **Ec2ElasticGpuSetup**—若執行個體與彈性 GPU 相關聯，則安裝彈性 GPU 軟體套件。
- **Ec2FeatureLogging**—將 Windows 功能安裝和對應的服務狀態傳送至主控台。僅支援 Microsoft Hyper-V 功能和對應的 `vmms` 服務。

## 全域設定

- **ManageShutdown**—確定執行個體後端 AMI 啟動的執行個體不會在執行 `Sysprep` 時終止。

- `SetDnsSuffixList`—設定 Amazon EC2 網路轉接器的 DNS 尾碼。如此可對執行於 Amazon EC2 的伺服器進行 DNS 解析，而不用提供完全合格的網域名稱。

**Note**

這會新增下列網域的 DNS 尾碼查詢，並設定其他標準尾碼。如需啟動代理程式如何設定 DNS 尾碼的詳細資訊，請參閱。[設定 Windows 啟動代理程式的 DNS 尾碼](#)

```
region.ec2-utilities.amazonaws.com
```

- `WaitForMetaDataAvailable`—確定 EC2Config 服務將等待直到中繼資料變成可存取且網路變成可用，然後再繼續開機。此檢查會確定 EC2Config 可從中繼資料取得用於啟用和其他外掛程式的資訊。
- `ShouldAddRoutes`—新增自訂路由表至主要網路轉接器，以在連接多個 NIC 時啟用下列的 IP 地址：169.254.169.250、169.254.169.251 和 169.254.169.254。這些地址供 Windows 啟用和當您要存取執行個體中繼資料時使用。
- `RemoveCredentialsfromSyspreponStartup`—下次服務啟動時從 `Sysprep.xml` 移除管理員密碼。若要確保此密碼保持一致，請編輯此設定。

### DriveLetterConfig.xml

此檔案包含控制磁碟機代號映射的設定。根據預設，磁碟區可映射至任何可用的磁碟機代號。您可將磁碟區掛載至特定的磁碟機代號，如下所示。

```
<?xml version="1.0" standalone="yes"?>
<DriveLetterMapping>
 <Mapping>
 <VolumeName></VolumeName>
 <DriveLetter></DriveLetter>
 </Mapping>
 . . .
 <Mapping>
 <VolumeName></VolumeName>
 <DriveLetter></DriveLetter>
 </Mapping>
</DriveLetterMapping>
```

- VolumeName—磁碟區標籤。例如 *My Volume*。若要指定執行個體儲存體磁碟區的映射，請使用 Temporary Storage X 標籤，其中的 X 為 0 至 25 的數字。
- DriveLetter—磁碟機代號。例如 *M:*。假如磁碟機代號已在使用中，映射便會失敗。

### EventLogConfig.xml

此檔案包含控制執行個體開機時顯示在主控台上之事件日誌資訊的設定。根據預設，會顯示系統事件日誌中最近的三個錯誤項目。

- Category—傳送至主控台的事件日誌金鑰。
- ErrorType—事件類型 (例如 Error、Warning、Information。)
- NumEntries—此類別存放的事件數量。
- LastMessageTime—為避免重複推送相同的訊息，服務會在每次推送訊息時更新此值。
- AppName—事件來源或記錄事件的應用程式。

### WallpaperSettings.xml

此檔案包含控制顯示在桌面背景上之資訊的設定。預設顯示下列資訊。

- Hostname—顯示電腦名稱。
- Instance ID—顯示執行個體的 ID。
- Public IP Address—顯示執行個體的公有 IP 地址。
- Private IP Address—顯示執行個體的私有 IP 地址。
- Availability Zone—顯示執行個體正在執行的可用區域。
- Instance Size—顯示執行個體的類型。
- Architecture—顯示 PROCESSOR\_ARCHITECTURE 環境變數的設定。

移除此項目，將移除預設顯示的任何資訊。您可新增額外要顯示的執行個體中繼資料，如下所示。

```
<WallpaperInformation>
 <name>display_name</name>
 <source>metadata</source>
 <identifier>meta-data/path</identifier>
</WallpaperInformation>
```

您可新增額外要顯示的系統環境變數，如下所示。

```
<WallpaperInformation>
 <name>display_name</name>
 <source>EnvironmentVariable</source>
 <identifier>variable-name</identifier>
</WallpaperInformation>
```

## InitializeDrivesSettings.xml

此檔案包含控制 EC2Config 如何初始化磁碟機的設定。

根據預設，EC2Config 會初始化作業系統未將其設為上線的磁碟機。您可自訂外掛程式，如下所示。

```
<InitializeDrivesSettings>
 <SettingsGroup>setting</SettingsGroup>
</InitializeDrivesSettings>
```

使用設定群組指定初始化磁碟機的方式：

### FormatWith 修剪

在格式化磁碟機時啟用 TRIM 命令。磁碟機完成格式化和初始化後，系統將還原 TRIM 組態。

自 EC2Config 3.18 版起，TRIM 命令在磁碟格式化操作期間預設為停用，如此將可改善格式化的時間。針對 EC2Config 3.18 版和更新版本，請使用此設定啟用磁碟格式化操作期間的 TRIM 命令。

### FormatWithout 修剪

格式化磁碟機時停用 TRIM 命令，以改善 Windows 的格式化時間。磁碟機完成格式化和初始化後，系統將還原 TRIM 組態。

### DisableInitialize 磁碟機

停用新磁碟機的格式化。請用此設定手動初始化磁碟機。

## 設定 EC2Config 服務的代理伺服器設定

您可以將 EC2Config 服務設定為透過代理使用下列其中一種方法進行通訊：AWS SDK for .NET、system.net 元素或 Microsoft 群組原則和網際網路資源管理器。使用 AWS SDK for .NET 是首選方法，因為您可以指定登入認證。

### 方法

- [使用 AWS SDK for .NET \(偏好\) 設定代理伺服器設定](#)
- [使用 system.net 元素設定代理伺服器設定](#)
- [使用 Microsoft 群組政策和 Microsoft Internet Explorer 設定代理伺服器設定](#)

## 使用 AWS SDK for .NET (偏好) 設定代理伺服器設定

您可指定 proxy 檔案中的 Ec2Config.exe.config 元素，以設定 EC2Config 服務的 proxy 設定。如需詳細資訊，請參閱 [AWS SDK for .NET 的組態檔參考](#)。

### 指定 Ec2Config.exe.config 中的代理伺服器元素

1. 在要讓 EC2Config 服務透過 proxy 通訊的執行個體上，編輯 Ec2Config.exe.config 檔案。根據預設，檔案位在下列目錄：`%ProgramFiles%\Amazon\Ec2ConfigService`。
2. 將以下 aws 元素新增到 configSections。請勿將此元素新增到任何現有的 sectionGroups。

#### 針對 EC2Config 3.17 版或更早版本

```
<configSections>
 <section name="aws" type="Amazon.AWSSection, AWSSDK"/>
</configSections>
```

#### 針對 EC2Config 3.18 版或更新版本

```
<configSections>
 <section name="aws" type="Amazon.AWSSection, AWSSDK.Core"/>
</configSections>
```

3. 將以下 aws 元素新增到 Ec2Config.exe.config 檔案。

```
<aws>
 <proxy
 host="string value"
 port="string value"
 username="string value"
 password="string value" />
</aws>
```

4. 儲存您的變更。

## 使用 system.net 元素設定代理伺服器設定

您可在 system.net 檔案的 Ec2Config.exe.config 元素中指定 proxy 的設定。如需詳細資訊，請參閱 MSDN 中的 [defaultProxy 元素 \(網路設定\)](#)。

指定 Ec2Config.exe.config 中的 system.net 元素

1. 在要讓 EC2Config 服務透過 proxy 通訊的執行個體上，編輯 Ec2Config.exe.config 檔案。根據預設，檔案位在下列目錄：`%ProgramFiles%\Amazon\Ec2ConfigService`。
2. 新增 defaultProxy 項目至 system.net。如需詳細資訊，請參閱 MSDN 中的 [defaultProxy 元素 \(網路設定\)](#)。

例如，下列設定將路由所有流量，以使用 Internet Explorer 目前設定的 proxy，但不包含中繼資料和授權流量，這些流量將繞過代理伺服器。

```
<defaultProxy>
 <proxy usesystemdefault="true" />
 <bypasslist>
 <add address="169.254.169.250" />
 <add address="169.254.169.251" />
 <add address="169.254.169.254" />
 <add address="[fd00:ec2::250]" />
 <add address="[fd00:ec2::254]" />
 </bypasslist>
</defaultProxy>
```

3. 儲存您的變更。

## 使用 Microsoft 群組政策和 Microsoft Internet Explorer 設定代理伺服器設定

EC2Config 服務在 Local System 使用者帳戶下執行。在執行個體上變更群組政策設定後，您可在 Internet Explorer 中為此帳戶指定執行個體範圍的代理伺服器設定。

## 使用群組政策和 Internet Explorer 設定代理伺服器設定

1. 在要讓 EC2Config 服務透過 proxy 通訊的執行個體上，以管理員身分開啟命令提示，輸入 **gpedit.msc** 並按 Enter 鍵。
2. 在本機群組政策編輯器中的 Local Computer Policy (本機電腦政策) 底下，選擇 Computer Configuration (電腦設定)、Administrative Templates (系統管理範本)、Windows Components (Windows 元件)、Internet Explorer。



3. 在右側窗格中，選擇 Make proxy settings per-machine (rather than per-user) (依每一部電腦設定 proxy 設定 (而不是依每一位使用者))，然後選擇 Edit policy setting (編輯政策設定)。
4. 選擇 Enabled (已啟用)，然後選擇 Apply (套用)。
5. 開啟 Internet Explorer，然後選擇 Tools (工具) 按鈕。
6. 選擇 Internet Option (網際網路選項)，然後選擇 Connections (連線) 索引標籤。
7. 選擇 LAN settings (LAN 設定)。
8. 在 Proxy server (代理伺服器) 底下，選擇 Use a proxy server for your LAN (為您的 LAN 使用代理伺服器) 選項。
9. 指定地址和連接埠資訊，然後選擇 OK (確定)。

### EC2Config 版本記錄

Windows Server 2016 以上版本的 Windows AMI 包含稱為 EC2Config 服務 (EC2Config.exe) 的選用服務。EC2Config 會在啟動期間啟動執行個體及執行作業時及每次您停止或啟動執行個體時啟動。

您會在 EC2Config 服務有新版本發佈時收到通知。如需詳細資訊，請參閱 [訂閱 EC2Config 服務通知](#)。

下表說明 EC2Config 的發佈版本。如需 SSM Agent 更新的詳細資訊，請參閱 [Systems Manager SSM Agent 版本備註](#)。

版本	詳細資訊	發行日期
4.9.5554	<ul style="list-style-type: none"> <li>• 根據登錄項目限制網域名稱轉移：HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Dnscache\Parameters\DomainNameDevolutionLevel。</li> <li>• SSM Agent 新版本 3.2.1630.0。</li> </ul>	2023 年 10 月 4 日
4.9.5467	<ul style="list-style-type: none"> <li>• 已新增重試功能，可探索主控台連接埠。</li> <li>• SSM Agent 新版本 3.1.2282.0。</li> </ul>	2023 年 8 月 1 日
4.9.5288	<ul style="list-style-type: none"> <li>• 已將 AWS Core SDK 更新為 3.7.103.23 版。</li> </ul>	2023 年 3 月 8 日

版本	詳細資訊	發行日期
	<ul style="list-style-type: none"> <li>修正 AWS-UpdateEC2Config SSM 文件無法在僅啟用 IMDSv2 的執行個體上更新 EC2Config 的問題。</li> <li>SSM Agent 新版本 3.1.2144.0。</li> </ul>	
4.9.5231	<ul style="list-style-type: none"> <li>SSM Agent 新版本 3.1.1927.0。</li> </ul>	2023 年 2 月 14 日
4.9.5103	<ul style="list-style-type: none"> <li>已修正 r5d 和 i4i 執行個體系列上無法正確識別暫時性磁碟區的問題。</li> <li>SSM Agent 新版本 3.1.1856.0。</li> </ul>	2022 年 12 月 5 日
4.9.5064	<ul style="list-style-type: none"> <li>已更新為使用 PCI 區段資訊來選取主控台連接埠。</li> <li>已簽署的 PowerShell 指令碼並新增版權標頭。</li> <li>已修正主要網路轉接器選取邏輯。</li> <li>SSM Agent 新版本 3.1.1732.0。</li> </ul>	2022 年 11 月 16 日
4.9.4588	<ul style="list-style-type: none"> <li>更新了 IMDS 等待邏輯，以僅發出 IMDSv2 請求。</li> <li>新增 libec2launch.dll 啟動代理程式共用庫。</li> <li>SSM Agent 3.1.1188.0 新版本。</li> </ul>	2022 年 5 月 31 日
4.9.4556	<ul style="list-style-type: none"> <li>新增等待邏輯以確保 NIC 在使用前完全初始化。</li> <li>新版本的 Log4Net 2.0.14.0 收取了安全修補程式。</li> <li>新版本的 SSM Agent 3.1.1045.0 收取了安全修補程式。</li> </ul>	2022 年 3 月 1 日
4.9.4536	<ul style="list-style-type: none"> <li>修復了 Temp 資料夾遺失時使用者資料損毀的問題。</li> <li>SSM Agent 新版本 3.1.804.0。</li> </ul>	2022 年 1 月 31 日

版本	詳細資訊	發行日期
4.9.4508	<ul style="list-style-type: none"> <li>修正正確運算 diskpart 指令碼路徑的問題。</li> <li>SSM Agent 新版本 3.1.338.0。</li> </ul>	2021 年 10 月 6 日
4.9.4500	<ul style="list-style-type: none"> <li>已更新具有 IMDS v2 支援的 Install-EgpuManagerConfig 。</li> <li>已更新 Web 連結以使用 https。</li> <li>SSM Agent 新版本 3.1.282.0</li> </ul>	2021 年 9 月 7 日
4.9.4419	<ul style="list-style-type: none"> <li>修正 IMDS 第 1 版後援邏輯</li> <li>已將 Windows 臨時目錄的所有使用更新為 EC2config 臨時目錄</li> <li>SSM Agent 新版本 3.0.1124.0</li> </ul>	2021 年 6 月 2 日
4.9.4381	<ul style="list-style-type: none"> <li>在 EC2 中增加了對 SSM 文檔架構 2.2 版本的支持 ConfigUpdater</li> <li>將 AWS 硝基飛地包版本添加到控制台日誌</li> <li>SSM Agent 新版本 3.0.529.0</li> </ul>	2021 年 5 月 4 日
4.9.4326	<ul style="list-style-type: none"> <li>移除了設定 UI 中的所有連結</li> <li>這是支援 Windows Server 2008 的最後一個 EC2Config 版本。</li> </ul>	2021 年 3 月 3 日
4.9.4279	<ul style="list-style-type: none"> <li>已修正與 Ec2ConfigMonitor 排程任務相關的安全性問題</li> <li>修正磁碟機代號對應問題和不正確的暫時磁碟計數</li> <li>新增 OsCurrentBuild 和 OsReleaseId 至主控台輸出</li> <li>SSM Agent 新版本 2.3.871.0</li> </ul>	2020 年 12 月 11 日

版本	詳細資訊	發行日期
4.9.4222	<ul style="list-style-type: none"> <li>修正 IMDS 第 1 版後援邏輯</li> <li>SSM Agent 新版本 2.3.842.0</li> </ul>	2020 年 4 月 7 日
4.9.4122	<ul style="list-style-type: none"> <li>已新增對 IMDS v2 的支援。</li> <li>SSM Agent 新版本 2.3.814.0</li> </ul>	2020 年 3 月 4 日
4.9.3865	<ul style="list-style-type: none"> <li>修正 Windows Server 2008 R2 在裸機執行個體上偵測 COM 連接埠的問題</li> <li>SSM Agent 新版本 2.3.722.0</li> </ul>	2019 年 10 月 31 日
4.9.3519	<ul style="list-style-type: none"> <li>SSM Agent 新版本 2.3.634.0</li> </ul>	2019 年 6 月 18 日
4.9.3429	<ul style="list-style-type: none"> <li>SSM Agent 新版本 2.3.542.0</li> </ul>	2019 年 4 月 25 日
4.9.3289	<ul style="list-style-type: none"> <li>SSM Agent 新版本 2.3.444.0</li> </ul>	2019 年 2 月 11 日
4.9.3270	<ul style="list-style-type: none"> <li>新增了將顯示器設定為永遠不關閉的外掛程式，以修復 ACPI 問題。</li> <li>寫入主控制台的 SQL Server 版本。</li> <li>SSM Agent 新版本 2.3.415.0</li> </ul>	2019 年 1 月 22 日
4.9.3230	<ul style="list-style-type: none"> <li>已更新磁碟機代號對應說明，以更切合功能。</li> <li>SSM Agent 新版本 2.3.372.0</li> </ul>	2019 年 1 月 10 日
4.9.3160	<ul style="list-style-type: none"> <li>主要 NIC 的等待時間增加</li> <li>RSS 的新增預設組態和 ENA 裝置的接收佇列設定</li> <li>Sysprep 時停用休眠</li> <li>SSM Agent 新版本 2.3.344.0</li> <li>AWS 開發套件升級至 3.3.29.13</li> </ul>	2018 年 12 月 15 日
4.9.3067	<ul style="list-style-type: none"> <li>執行個體休眠改進</li> <li>SSM Agent 新版本 2.3.235.0</li> </ul>	2018 年 11 月 8 日

版本	詳細資訊	發行日期
4.9.3034	<ul style="list-style-type: none"><li>DNS 伺服器新增路由 169.254.169.253/32</li><li>SSM Agent 新版本 2.3.193.0</li></ul>	2018 年 10 月 24 日
4.9.2986	<ul style="list-style-type: none"><li>所有 EC2Config 相關二進位檔的新增簽署</li><li>SSM Agent 新版本 2.3.136.0</li></ul>	2018 年 10 月 11 日
4.9.2953	SSM Agent 新版本 (2.3.117.0)	2018 年 10 月 2 日
4.9.2926	SSM Agent 新版本 (2.3.68.0)	2018 年 9 月 18 日
4.9.2905	<ul style="list-style-type: none"><li>SSM Agent 新版本 (2.3.50.0)</li><li>已新增路由 169.254.169.123/32 至 AMZN 時間服務</li><li>已新增路由 169.254.169.249/32 至 GRID 授權服務</li><li>已修正造成 EBS NVMe 磁碟區被標示為暫時性的問題</li></ul>	2018 年 9 月 17 日
4.9.2854	SSM Agent 新版本 (2.3.13.0)	2018 年 8 月 17 日
4.9.2831	SSM Agent 新版本 (2.2.916.0)	2018 年 8 月 7 日
4.9.2818	SSM Agent 新版本 (2.2.902.0)	2018 年 7 月 31 日
4.9.2756	SSM Agent 新版本 (2.2.800.0)	2018 年 6 月 27 日
4.9.2688	SSM Agent 新版本 (2.2.607.0)	2018 年 5 月 25 日
4.9.2660	SSM Agent 新版本 (2.2.546.0)	2018 年 5 月 11 日

版本	詳細資訊	發行日期
4.9.2644	SSM Agent 新版本 (2.2.493.0)	2018 年 4 月 26 日
4.9.2586	SSM Agent 新版本 (2.2.392.0)	2018 年 3 月 28 日
4.9.2565	<ul style="list-style-type: none"> <li>SSM Agent 新版本 (2.2.355.0)</li> <li>修正 M5 和 C5 執行個體的問題 (找不到 PV 驅動程式)</li> <li>新增執行個體類型、最新 PV 驅動程式和 NVMe 驅動程式的主控台記錄</li> </ul>	2018 年 3 月 13 日
4.9.2549	SSM Agent 新版本 (2.2.325.0)	2018 年 3 月 8 日
4.9.2461	SSM Agent 新版本 (2.2.257.0)	2018 年 2 月 15 日
4.9.2439	SSM Agent 新版本 (2.2.191.0)	2018 年 2 月 6 日
4.9.2400	SSM Agent 新版本 (2.2.160.0)	2018 年 1 月 16 日
4.9.2327	<ul style="list-style-type: none"> <li>SSM Agent 新版本 (2.2.120.0)</li> <li>新增 Amazon EC2 裸機執行個體上的 COM 連接埠探索</li> <li>新增 Amazon EC2 裸機執行個體上的 Hyper-V 狀態記錄</li> </ul>	2018 年 1 月 2 日
4.9.2294	SSM Agent 新版本 (2.2.103.0)	2017 年 12 月 4 日
4.9.2262	SSM Agent 新版本 (2.2.93.0)	2017 年 11 月 15 日

版本	詳細資訊	發行日期
4.9.2246	SSM Agent 新版本 (2.2.82.0)	2017 年 11 月 11 日
4.9.2218	SSM Agent 新版本 (2.2.64.0)	2017 年 10 月 29 日
4.9.2212	SSM Agent 新版本 (2.2.58.0)	2017 年 10 月 23 日
4.9.2203	SSM Agent 新版本 (2.2.45.0)	2017 年 10 月 19 日
4.9.2188	SSM Agent 新版本 (2.2.30.0)	2017 年 10 月 10 日
4.9.2180	<ul style="list-style-type: none"> <li>SSM Agent 新版本 (2.2.24.0)</li> <li>新增 GPU 執行個體的 Elastic GPU 外掛程式</li> </ul>	2017 年 10 月 5 日
4.9.2143	SSM Agent 新版本 (2.2.16.0)	2017 年 10 月 1 日
4.9.2140	SSM Agent 新版本 (2.1.10.0)	
4.9.2130	SSM Agent 新版本 (2.1.4.0)	
4.9.2106	SSM Agent 新版本 (2.0.952.0)	
4.9.2061	SSM Agent 新版本 (2.0.922.0)	
4.9.2047	SSM Agent 新版本 (2.0.913.0)	
4.9.2031	SSM Agent 新版本 (2.0.902.0)	
4.9.2016	<ul style="list-style-type: none"> <li>SSM Agent 新版本 (2.0.879.0)</li> <li>修正了視窗服務器 2003 的 CloudWatch 日誌目錄路徑</li> </ul>	

版本	詳細資訊	發行日期
4.9.1981	<ul style="list-style-type: none"> <li>SSM Agent 新版本 (2.0.847.0)</li> <li>修正 EBS 磁碟區內所產生之 <code>important.txt</code> 的問題。</li> </ul>	
4.9.1964	SSM Agent 新版本 (2.0.842.0)	
4.9.1951	<ul style="list-style-type: none"> <li>SSM Agent 新版本 (2.0.834.0)</li> <li>修正暫時性硬碟機之磁碟機代號 Z: 無法對應的問題。</li> </ul>	
4.9.1925	<ul style="list-style-type: none"> <li>SSM Agent 新版本 (2.0.822.0)</li> <li>(錯誤) 此版本不是 SSM Agent v4.9.1775 的有效更新目標。</li> </ul>	
4.9.1900	SSM Agent 新版本 (2.0.805.0)	
4.9.1876	<ul style="list-style-type: none"> <li>SSM Agent 新版本 (2.0.796.0)</li> <li>修正管理使用者資料執行的輸出/錯誤重新導向問題。</li> </ul>	
4.9.1863	<ul style="list-style-type: none"> <li>SSM Agent 新版本 (2.0.790.0)</li> <li>修正連接多個 EBS 磁碟區至 Amazon EC2 執行個體時發生的問題。</li> <li>改進 CloudWatch 了採取配置路徑，保持向後兼容性。</li> </ul>	
4.9.1791	SSM Agent 新版本 (2.0.767.0)	
4.9.1775	SSM Agent 新版本 (2.0.761.0)	
4.9.1752	SSM Agent 新版本 (2.0.755.0)	
4.9.1711	SSM Agent 新版本 (2.0.730.0)	



版本	詳細資訊	發行日期
4.8.1676	SSM Agent 新版本 (2.0.716.0)	
4.7.1631	SSM Agent 新版本 (2.0.682.0)	
4.6.1579	<ul style="list-style-type: none"> <li>SSM Agent 新版本 (2.0.672.0)</li> <li>修正 v4.3、v4.4 和 v4.5 的代理程式更新問題</li> </ul>	
4.5.1534	SSM Agent 新版本 (2.0.645.1)	
4.4.1503	SSM Agent 新版本 (2.0.633.0)	
4.3.1472	SSM Agent 新版本 (2.0.617.1)	
4.2.1442	SSM Agent 新版本 (2.0.599.0)	
4.1.1378	SSM Agent 新版本 (2.0.558.0)	
4.0.1343	<ul style="list-style-type: none"> <li>執行命令、狀態管理員、CloudWatch 代理程式和網域加入支援已移至另一個稱為 SSM 代理程式的代理程式。SSM Agent 將在 EC2Config 升級時一併安裝。如需詳細資訊，請參閱 <a href="#">設定和 AWS Systems Manager</a>。</li> <li>如果您已在 EC2Config 中設定代理伺服器，則您需要先更新 SSM Agent 的代理伺服器，然後再升級。如果未更新代理伺服器設定，將無法使用執行命令管理執行個體。為避免此問題，請在更新至新版本前先參閱下列資訊：《AWS Systems Manager 使用者指南》中的 <a href="#">在 Windows 執行個體上安裝和設定 SSM Agent</a>。</li> <li>如果您先前使用本機組態檔案 (AWS.EC2.Windows.CloudWatch.json ) 在執行個體上啟用 CloudWatch 整合，則需要設定檔案以與 SSM Agent 搭配使用。</li> </ul>	

版本	詳細資訊	發行日期
3.19.1153	<ul style="list-style-type: none"> <li>對於具有舊 AWS KMS 配置的實例重新啟用激活插件。跳過 BYOL 使用者的啟動。</li> <li>更改默認 TRIM 行為在磁盤格式化操作期間被禁用，並添加 FormatWith TRIM 用於覆蓋帶有用戶數據的 InitializeDisks 插件。</li> </ul>	
3.18.1118	<ul style="list-style-type: none"> <li>修正以可靠地新增路由表至主要網路轉接器。</li> <li>更新以改善對 AWS 服務的支援。</li> </ul>	
3.17.1032	<ul style="list-style-type: none"> <li>修正篩選條件設為相同類別時出現的重複系統日誌。</li> <li>修正以避免磁碟初始化時發生停滯。</li> </ul>	
3.16.930	新增在啟動時將「Window is Ready to use」(Window 已可開始使用) 事件記錄至 Windows 事件日誌的支援。	
3.15.880	修正以允許將 Systems Manager 執行命令輸出上傳至名稱含「.」字元的 S3 儲存貯體。	
3.14.786	<p>增加了覆蓋 InitializeDisks 插件設置的支持。例如：若要加速 SSD 磁碟初始化，您可在使用者資料中指定下列設定以暫時停用 TRIM：</p> <pre>&lt; InitializeDrives 設定 &gt; &lt; SettingsGroup &gt; FormatWithout 修剪 SettingsGroup &lt;/ 設定值 InitializeDrives</pre>	
3.13.727	Systems Manager RunCommand - 修正以在 Windows 重新開機後可靠地處理命令。	

版本	詳細資訊	發行日期
3.12.649	<ul style="list-style-type: none"> <li>• 修正以在執行命令/指令碼時從容處理重新開機。</li> <li>• 修正以從容取消執行命令。</li> <li>• 新增透過 Systems Manager 執行命令安裝應用程式時 (選擇性) 上傳 MSI 日誌至 S3 的支援。</li> </ul>	
3.11.521	<ul style="list-style-type: none"> <li>• 修正以啟用 Windows Server 2003 的 RDP 指紋產生。</li> <li>• 修正以將時區和 UTC 位移包含在 EC2Config 日誌行內。</li> <li>• Systems Manager 支援平行執行執行命令的命令。</li> <li>• 還原先前的變更，將已分割的磁碟設為上線。</li> </ul>	
3.10.442	<ul style="list-style-type: none"> <li>• 修正安裝 MSI 應用程式時的 Systems Manager 設定錯誤。</li> <li>• 修正以可靠地將儲存磁碟設為上線。</li> <li>• 更新以改善對 AWS 服務的支援。</li> </ul>	
3.9.359	<ul style="list-style-type: none"> <li>• 修正後期 Sysprep 指令碼，讓 Windows 更新設定保持為預設狀態。</li> <li>• 修正密碼產生外掛程式，以改善取得 GPO 密碼政策設定的可靠性。</li> <li>• 限制僅本機管理員群組擁有 EC2Config/SSM 日誌資料夾權限。</li> <li>• 更新以改善對 AWS 服務的支援。</li> </ul>	

版本	詳細資訊	發行日期
3.8.294	<ul style="list-style-type: none"> <li>• 修正當不在主要磁碟機上時，無法上傳記錄檔的問題。 CloudWatch</li> <li>• 加入重試邏輯，改善磁碟初始化流程。</li> <li>• 增加了改進了 SetPassword 插件在 AMI 創建過程中偶爾失敗時的錯誤處理。</li> <li>• 更新以改善對 AWS 服務的支援。</li> </ul>	
3.7.308	<ul style="list-style-type: none"> <li>• 改善 ec2config-cli 公用程式，以在執行個體內進行 config 測試與故障診斷。</li> <li>• 避免在 OpenVPN 適配器上添加靜態路由 AWS KMS 和元數據服務。</li> <li>• 修正使用者資料執行未執行「persist」標籤的問題。</li> <li>• 改善無法登入 EC2 主控台時的錯誤處理能力。</li> <li>• 更新以改善對 AWS 服務的支援。</li> </ul>	
3.6.269	<ul style="list-style-type: none"> <li>• 修正 Windows 啟用可靠性，在透過 AWS KMS 啟動 Windows 時先使用連結本機地址 169.254.0.250/251</li> <li>• 改善 Systems Manager、Windows 啟動和網域加入等狀況下的代理伺服器處理能力</li> <li>• 修正重複的使用者帳戶行新增到 Sysprep 回答檔案的問題</li> </ul>	

版本	詳細資訊	發行日期
3.5.228	<ul style="list-style-type: none"><li>• 解決 CloudWatch 外掛程式可能會耗用過多 CPU 和記憶體讀取 Windows 事件記錄檔的案例</li><li>• 在 EC2Config 設定使用者介面中新增 CloudWatch 組態文件的連結</li></ul>	
3.4.212	<ul style="list-style-type: none"><li>• 修正 EC2Config 搭配 VM-Import 使用時的問題。</li><li>• 修正 WiX 安裝程式內的服務命名問題。</li></ul>	
3.3.174	<ul style="list-style-type: none"><li>• 改善 Systems Manager 和網域加入問題的例外處理能力。</li><li>• 支援 Systems Manager SSM 結構描述版本控制的變更。</li><li>• 修正 Win2K3 上的暫時性磁碟格式化問題。</li><li>• 支援設定超過 2TB 磁碟大小的變更。</li><li>• 設定 GC 模式為預設值，減少虛擬記憶體用量。</li><li>• 支援從 <code>aws:psModule</code> 和 <code>aws:application</code> 外掛程式的 UNC 路徑下載成品。</li><li>• 改善 Windows 啟用外掛程式的記錄功能。</li></ul>	

版本	詳細資訊	發行日期
3.2.97	<ul style="list-style-type: none"><li>• 延遲載入 Systems Manager SSM 組件，以改善效能。</li><li>• 改善不正確之 sysprep2008.xml 的例外處理能力。</li><li>• Systems Manager 「套用」設定的命令列支援。</li><li>• 支援有電腦重新命名等待時加入網域的變更。</li><li>• 支援 aws:applications 外掛程式中的選擇性參數。</li><li>• 支援 aws:psModule 外掛程式中的命令陣列。</li></ul>	
3.0.54	<ul style="list-style-type: none"><li>• 啟用 Systems Manager 的支援。</li><li>• 透過 Systems Manager，將 EC2 Windows 執行個體自動網域加入至 AWS 目錄。</li><li>• 透過 Systems Manager 設定及上傳 CloudWatch 記錄/指標。</li><li>• 透過 Systems Manager 安裝 PowerShell 模塊。</li><li>• 透過 Systems Manager 安裝 MSI 應用程式。</li></ul>	
2.4.233	<ul style="list-style-type: none"><li>• 新增排程任務，從服務啟動故障中復原 EC2Config。</li><li>• 改善主控台日誌錯誤訊息。</li><li>• 更新以改善對 AWS 服務的支援。</li></ul>	

版本	詳細資訊	發行日期
2.3.313	<ul style="list-style-type: none"> <li>修正啟用記 CloudWatch 錄功能時，某些情況下會耗用大量記憶體的問題。</li> <li>修正升級錯誤，允許 ec2config 2.1.19 以下版本現在可升級至最新版。</li> <li>更新 COM 連接埠開啟例外，使其在日誌中更方便好用。</li> <li>EC2 configServiceSettings UI 禁用了調整大小，並修復了 UI 中的歸因和版本顯示位置。</li> </ul>	
2.2.12	<ul style="list-style-type: none"> <li>NullPointerException 在查詢用於確定偶爾返回空值的 Windows Sysprep 狀態的註冊表項時進行處理。</li> <li>發佈 finally 區塊中的非受管資源。</li> </ul>	
2.2.11	修正 CloudWatch 外掛程式中處理空白記錄行的問題。	
2.2.10	<ul style="list-style-type: none"> <li>移除透過使用者介面設定 CloudWatch 記錄設定</li> <li>讓使用者在%ProgramFiles%\Amazon\Ec2ConfigService\Settings\AWS.EC2.Windows.CloudWatch.json 檔案中定義 CloudWatch 防護記錄設定，以允許 future 的增強。</li> </ul>	
2.2.9	修正未處理的例外狀況並新增記錄功能。	
2.2.8	<ul style="list-style-type: none"> <li>修正 EC2Config 安裝程式中的 Windows OS 版本檢查，以支援 Windows Server 2003 SP1 和更新版本。</li> <li>修正為更新 Sysprep 組態檔而讀取登錄機碼時的 null 值處理能力。</li> </ul>	

版本	詳細資訊	發行日期
2.2.7	<ul style="list-style-type: none"><li>• 新增 Windows 2008 和更新版本上 EC2Config 在 Sysprep 執行期間執行的支援。</li><li>• 改善例外處理和記錄功能，以提升診斷效率</li></ul>	
2.2.6	<ul style="list-style-type: none"><li>• 減少上傳記錄事件時執行個體和 CloudWatch 記錄檔上的負載。</li><li>• 解決 CloudWatch 記錄外掛程式未始終保持啟用狀態的升級問題</li></ul>	
2.2.5	<ul style="list-style-type: none"><li>• 增加了將日誌上傳到日 CloudWatch 誌服務的支持。</li><li>• 修正 Ec2OutputRDPcert 外掛程式中的競爭條件問題</li><li>• 將 EC2Config 服務恢復選項更改為從操作重新啟動 TakeNo</li><li>• 新增更多 EC2Config 當機時的例外狀況資訊</li></ul>	
2.2.4	<ul style="list-style-type: none"><li>• 修正了 .cmd 中的一個錯字 PostSysprep</li><li>• 修正 EC2Config 在 OS2012+ 中無法釘選至開始選單的錯誤</li></ul>	



版本	詳細資訊	發行日期
2.2.3	<ul style="list-style-type: none"> <li>• 新增在安裝時未立即啟動服務之情況下安裝 EC2Config 的選項。若要使用，請從命令提示執行「Ec2Install.exe start=false」</li> <li>• 新增桌布外掛程式中用於控制新增/移除桌布的參數。要使用，請從命令提示符下運行「EC2 WallpaperInfo .exe 集」或「EC2 WallpaperInfo .exe 還原」</li> <li>• 添加了對 RealTimeIsUniversal 鍵的檢查，將 RealTimeIsUniversal 註冊表項的不正確設置輸出到控制台</li> <li>• 移除 EC2Config 對 Windows 暫存資料夾的相依性</li> <li>• 移除了對 .Net 3.5 的 UserData 執行相依性</li> </ul>	
2.2.2	<ul style="list-style-type: none"> <li>• 新增對服務停止行為的檢查，以檢查資源是否發佈</li> <li>• 修正加入網域時執行時間過長的問題</li> </ul>	
2.2.1	<ul style="list-style-type: none"> <li>• 更新安裝程式，以允許從舊版升級</li> <li>• 修復了 .Net4.5 唯一環境中的 EC2 WallpaperInfo 錯誤</li> <li>• 修正間歇性的驅動程式偵測錯誤</li> <li>• 新增靜音安裝選項。執行 Ec2Install.exe 並加入「-q」選項。例如：「Ec2Install.exe -q」</li> </ul>	
2.2.0	<ul style="list-style-type: none"> <li>• 新增對僅限於 .Net4 和 .Net4.5 環境的支援</li> <li>• 更新安裝程式</li> </ul>	

版本	詳細資訊	發行日期
2.1.19	<ul style="list-style-type: none"> <li>• 新增使用 Intel 網路驅動程式時的暫時性磁碟標籤支援 (例如 C3 執行個體類型)。如需詳細資訊，請參閱 <a href="#">Amazon EC2 上的增強型聯網</a>。</li> <li>• 新增 AMI Origin Version 和 AMI Origin Name 支援至主控台輸出</li> <li>• 變更主控台輸出，以執行一致的格式化/剖析</li> <li>• 更新說明檔案</li> </ul>	
2.1.18	<ul style="list-style-type: none"> <li>• 為完成通知添加 EC2 配置 WMI 對象 ( -命名空間根\ Amazon 類 EC2_ ) ConfigService</li> <li>• 改善含大量事件日誌之啟動 WMI 查詢的效能；避免其在初始執行時長時間大量消耗 CPU</li> </ul>	
2.1.17	<ul style="list-style-type: none"> <li>• 修復了標準輸出和標準錯誤緩衝填充的 UserData 執行問題</li> <li>• 修正 w2k8 OS 以上版本主控台輸出有時出現不正確 RDP 指紋的問題</li> <li>• 控制台輸出現在包含「RDP 證書-SubjectName:」，適用於 Windows 2008 +，其中包含機器名稱值</li> <li>• 新增 D:\ 至磁碟機代號映射下拉式清單</li> <li>• 移動說明按鈕至右上角並變更外觀/使用體驗</li> <li>• 新增意見回饋問卷連結至右上角</li> </ul>	

版本	詳細資訊	發行日期
2.1.16	<ul style="list-style-type: none"><li>• 一般索引標籤包含新版本的 EC2Config 下載頁面連結</li><li>• 桌面牆紙覆蓋現在存儲在用戶本地 Appdata 文件夾中，而不是我的文檔中，以支持 MyDoc 重定向</li><li>• MSSQLServer 名稱與 Post-Sysprep 指令碼中的系統同步 (2008+)</li><li>• 重新排序應用程式資料夾 (將檔案移至 Plugin 目錄並移除重複的檔案)</li><li>• 變更系統日誌輸出 (主控台) :<ul style="list-style-type: none"><li>* 改為日期、名稱、值格式，以方便剖析 (請開始將相依性遷移至新格式)</li><li>* 增加了「EC2SetPassword」插件狀態</li><li>* 新增 Sysprep 開始和結束時間</li></ul></li><li>• 修正非英文版作業系統上暫時性磁碟未標籤為「Temporary Storage」的問題</li><li>• 修正執行 Sysprep 後的 EC2Config 解除安裝問題</li></ul>	

版本	詳細資訊	發行日期
2.1.15	<ul style="list-style-type: none"> <li>• 將中繼資料服務請求最佳化</li> <li>• 中繼資料現在將繞過代理伺服器設定</li> <li>• 暫時性磁碟已標籤為「Temporary Storage」，且找到的 Important.txt 將放入磁碟區 (僅限 Citrix PV 驅動程式)。如需詳細資訊，請參閱 <a href="#">升級 Windows 執行個體上的 PV 驅動程式</a>。</li> <li>• 暫時性磁碟將指派 Z 到 A 的代號 (僅限 Citrix PV 驅動程式) - 指派可用磁碟機代號映射外掛程式覆寫，覆寫為磁碟區標籤「Temporary Storage X」(其中 X 為 0-25 的數字)</li> <li>• UserData 現在立即在「窗口已準備就緒」之後運行</li> </ul>	
2.1.14	桌面桌布修正	
2.1.13	<ul style="list-style-type: none"> <li>• 桌面桌布預設將顯示主機名稱</li> <li>• 移除對 Windows 時間服務的相依性</li> <li>• 針對單一介面獲指派多重 IP 的狀況新增路由表</li> </ul>	
2.1.11	<ul style="list-style-type: none"> <li>• 變更 Ec2Activation 外掛程式</li> <li>• -每隔 30 天檢查一次啟用狀態</li> <li>• -如果寬限期仍有 90 天 (共 180 天)，則重新嘗試啟用</li> </ul>	

版本	詳細資訊	發行日期
2.1.10	<ul style="list-style-type: none"><li>• 桌面桌布疊加層不再與 Sysprep 或不使用 Sysprep 關機一致</li><li>• 在開頭為 <code>&lt;persist&gt;&gt;true&lt;/persist&gt;</code> 的每個服務上執行的使用者資料選項</li><li>• 更改的位置和名稱/ DisableWin 更新.cmd 為 /腳本/ .cmd PostSysprep</li><li>• 在 /Script PostSysprep /.cmd 中，系統管理員密碼預設為不過期</li><li>• 解除安裝將會從 c:\windows\setup\script\ .cmd 移除 EC2 組態 PostSysprep 指令碼 CommandComplete</li><li>• 新增路由表對自訂介面指標的支援</li></ul>	
2.1.9	UserData 執行不再限於 3851 個字元	

版本	詳細資訊	發行日期
2.1.7	<ul style="list-style-type: none"><li>• OS 版本和語言識別符寫入至主控台</li><li>• EC2Config 版本寫入至主控台</li><li>• PV 驅動程式版本寫入至主控台</li><li>• 偵測錯誤檢查，並在找到錯誤後於下次開機時輸出至主控台</li><li>• 新增選項至 config.xml，以維持 Sysprep 登入資料</li><li>• 新增啟動時無法使用 ENI 情況下的路由重試邏輯</li><li>• 使用者資料執行 PID 寫入至主控台</li><li>• 從 GPO 擷取的最小產生密碼長度</li><li>• 設定服務啟動重試 3 次</li><li>• 增加了 S3_DownloadFile.ps1 和 S3_上傳文件.ps1的例子到/腳本文件夾</li></ul>	

版本	詳細資訊	發行日期
2.1.6	<ul style="list-style-type: none"> <li>• 新增版本資訊至 General (一般) 索引標籤</li> <li>• Bundle 索引標籤重新命名為 Image (影像)</li> <li>• 簡化指定密碼的流程，並將密碼相關 UI 從 General (一般) 索引標籤移至 Image (影像) 索引標籤</li> <li>• Disk Settings (磁碟設定) 索引標籤重新命名為 Storage (儲存)</li> <li>• 新增 Support (支援) 索引標籤，並提供可執行故障排除的常用工具</li> <li>• Windows Server 2003 sysprep.ini 設為預設擴充 OS 分割區</li> <li>• 新增私有 IP 地址至桌布</li> <li>• 私有 IP 地址將在桌布上顯示</li> <li>• 新增主控台輸出的重試邏輯</li> <li>• 修正中繼資料存取功能的通訊連接埠例外狀況 -- 其會導致 EC2Config 在顯示主控台輸出之前終止</li> <li>• 每次開機時檢查啟用狀態 -- 視需要啟動</li> <li>• 修正相關路徑的問題 -- 發生在手動從啟動資料夾執行桌布捷徑時；指向管理員/日誌</li> <li>• 修正 Windows Server 2003 使用者 (管理員除外) 的預設背景色彩</li> </ul>	

版本	詳細資訊	發行日期
2.1.2	<ul style="list-style-type: none"><li>• UTC (Zulu) 的主控台時間戳記</li><li>• 移除 Sysprep 索引標籤上的超連結顯示</li><li>• 新增在 Windows 2008+ 第一次開機時動態擴充根磁碟區的功能</li><li>• 啟用設定密碼時，現在允取 EC2Config 設定密碼</li><li>• EC2Config 會在執行 Sysprep 之前檢查啟用狀態 (若未啟動將出現警告)</li><li>• Windows Server 2003 Sysprep.xml 現在預設為 UTC 時區，而非亞太地區</li><li>• 隨機啟動伺服器</li><li>• Drive Mapping (磁碟機映射) 索引標籤重新命名為 Disk Settings (磁碟設定)</li><li>• Initialize Drives UI (初始化磁碟機 UI) 項目從 General (一般) 索引標籤移至 Disk Settings (磁碟設定) 索引標籤</li><li>• 說明按鈕現在指向 HTML 說明檔案</li><li>• 更新 HTML 說明檔案的變更</li><li>• 更新磁碟機代號映射的「備註」文字</li><li>• 在 Sysprep 之前將 InstallUpdates.ps1 添加到 /Script 文件夾中，用於自動執行修補程序和清理</li></ul>	



版本	詳細資訊	發行日期
2.1.0	<ul style="list-style-type: none"> <li>桌面桌布預設在第一次登入 (未中斷/重新連接) 後顯示執行個體資訊</li> <li>PowerShell 可以通過圍繞代碼從用戶數據運行&lt;powershell&gt;&lt;/powershell&gt;</li> </ul>	

## 訂閱 EC2Config 服務通知

Amazon SNS 會在 EC2Config 服務發佈新版本時通知您。使用下列程序訂閱這些通知。

### 訂閱 EC2Config 通知

1. 在 <https://console.aws.amazon.com/sns/v3/home> 開啟 Amazon SNS 主控台。
2. 如有必要，請在導覽列中將「區域」變更為美國東部 (維吉尼亞北部)。您必須選取此區域，因為您要訂閱的 SNS 通知已在本區域中建立完成。
3. 在導覽窗格中，選擇 Subscriptions (訂閱)。
4. 選擇 Create subscription (建立訂閱)。
5. 在 Create subscription (建立訂閱) 對話方塊中，執行下列動作：
  - a. 對於 Topic ARN (主題 ARN)，請使用下列 Amazon Resource Name (ARN)：
 

`arn:aws:sns:us-east-1:801119661308:ec2-windows-ec2config`
  - b. 針對 Protocol (通訊協定)，選擇 Email。
  - c. 針對 Endpoint (端點)，輸入可用於接收通知的電子郵件地址。
  - d. 選擇 Create subscription (建立訂閱)。
6. 您會收到要求確認訂閱的電子郵件。開啟電子郵件並遵循指示完成訂閱。

每當 EC2Config 服務有新版本發佈時，我們將傳送通知給訂閱者。如果您不想再接收這些通知，請使用下列程序來取消訂閱。

### 取消訂閱 EC2Config 通知

1. 開啟 Amazon SNS 主控台。

2. 在導覽窗格中，選擇 Subscriptions (訂閱)。
3. 選取訂閱，接著選取 Actions (動作)、Delete subscriptions (刪除訂閱)，要求確認時請選取 Delete (刪除)。

### 故障診斷使用 EC2Config 服務的問題

以下資訊有助於排解 EC2Config 服務的疑難問題。

#### 在無法使用的執行個體上更新 EC2Config

請用下列程序更新使用遠端桌面無法存取之 Windows Server 執行個體上的 EC2Config 服務。

#### 更新無法連線之 Amazon EBS 後端 Windows 執行個體上的 EC2Config

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 尋找受影響的執行個體。選取執行個體，並依序選取 Instance state (執行個體狀態)、Stop instance (停止執行個體)。

#### Warning

停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區的資料，請將資料備份至持久性儲存。

4. 選擇 Launch instances (啟動執行個體)，並在與受影響執行個體相同的可用區域中，建立暫時性的 t2.micro 執行個體。使用與用來啟動受影響之執行個體不同的 AMI。

#### Important

若您沒有在與受影響的執行個體相同的可用區域內建立執行個體，您將無法將受影響執行個體的根磁碟區連接到新的執行個體。

5. 在 EC2 主控台中，選擇 Volumes (磁碟區)。
6. 尋找受影響執行個體的根磁碟區。[分開磁碟區](#)，再[將磁碟區連接](#)至先前建立的暫時執行個體。使用預設裝置名稱 (xvdf) 連接它。
7. 使用遠端桌面連接至暫時性執行個體，然後使用 Disk Management 公用程式將[磁碟區變成可用](#)。
8. [下載](#)最新版的 EC2Config 服務。將 .zip 檔案中的檔案解壓縮到所連接之磁碟機上的 Temp 目錄中。

9. 在暫時性執行個體上，開啟 Run (執行) 對話方塊，輸入 **regedit**，然後按 Enter 鍵。
10. 選擇 HKEY\_LOCAL\_MACHINE。從 File (檔案) 功能表中，選擇 Load Hive (載入 Hive)。選擇磁碟機，然後導覽並開啟下列檔案：Windows\System32\config\SOFTWARE。出現提示時，指定機碼名稱。
11. 選取剛載入的機碼並導覽至 Microsoft\Windows\CurrentVersion。選擇 RunOnce 機碼。如果該機碼不存在，從內容 (按一下右鍵) 選單中選擇 CurrentVersion，選擇 New (新增)，然後選擇 Key (機碼)。將機碼命名為 RunOnce。
12. 從內容 (按一下右鍵) 選單中選擇 RunOnce 機碼選擇 New (新增)，然後選擇 String Value (字串值)。在名稱輸入 Ec2Install，在資料輸入 C:\Temp\Ec2Install.exe /quiet。
13. 選擇 HKEY\_LOCAL\_MACHINE\*specified key name*\Microsoft\Windows NT\CurrentVersion\Winlogon 機碼。從內容 (按一下右鍵) 選單中選擇 New (新增)，然後選擇 String Value (字串值)。在名稱輸入 **AutoAdminLogon**，在數值資料輸入 **1**。
14. 選擇 HKEY\_LOCAL\_MACHINE\*specified key name*\Microsoft\Windows NT\CurrentVersion\Winlogon> 機碼。從內容 (按一下右鍵) 選單中選擇 New (新增)，然後選擇 String Value (字串值)。在名稱輸入 **DefaultUserName**，在數值資料輸入 **Administrator**。
15. 選擇 HKEY\_LOCAL\_MACHINE\*specified key name*\Microsoft\Windows NT\CurrentVersion\Winlogon 機碼。從內容 (按一下右鍵) 選單中選擇 New (新增)，然後選擇 String Value (字串值)。在名稱輸入 **DefaultPassword**，在數值資料輸入密碼。
16. 在 Registry Editor (登錄編輯程式) 導覽窗格中，選擇一開始開啟登錄編輯程式時建立的暫時性機碼。
17. 從 File (檔案) 選單中，選擇 Unload Hive (解除載入 Hive)。
18. 在 Disk Management 公用程式中，選擇先前連接的磁碟機，開啟內容 (按一下滑鼠右鍵) 選單，然後選擇 Offline (離線)。
19. 在 Amazon EC2 主控台中，將受影響的磁碟區與暫時性執行個體分離，然後將其重新連接至裝置名稱為 /dev/sda1 的執行個體。您必須指定此裝置名稱，將磁碟區指定為根磁碟區。
20. [停止並啟動 Amazon EC2 執行個體](#) 執行個體。
21. 執行個體啟動後，檢查系統日誌並確認出現 Windows is ready to use (Windows 已可開始使用) 的訊息。
22. 開啟登錄編輯程式，然後選擇 HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon。刪除您先前建立的「字串值」鍵：「AutoAdmin登入」、「DefaultUser名稱」和DefaultPassword。
23. 刪除或停止您在此程序中建立的暫時性執行個體。

## 為您的執行個體使用 EC2 快速啟動

每個 Amazon EC2 Windows 執行個體都必須經過標準的 Windows 作業系統 (OS) 啟動步驟，其中包括多次重新啟動，通常需要 15 分鐘或更長時間才能完成。已啟用 EC2 快速啟動功能的 Amazon EC2 Windows 伺服器 AMI 會完成其中一些步驟並預先重新開機，以減少啟動執行個體所需的時間。

當您設定適用於 EC2 快速啟動的 Windows 伺服器 AMI 時，Amazon EC2 會建立一組預先佈建的快照，以便加快啟動速度，如下所示。

1. Amazon EC2 會依據您的設定啟動一組暫時 t3 執行個體。
2. 當每個臨時執行個體完成標準啟動步驟時，Amazon EC2 會建立預先佈建的執行個體快照。它會將快照存放在您的 Amazon S3 儲存貯體中。
3. 快照準備就緒時，Amazon EC2 會終止相關聯的 t3 執行個體，以盡可能降低資源成本。
4. 下次 Amazon EC2 從啟用 EC2 快速啟動的 AMI 啟動執行個體時，會使用其中一個快照來顯著縮短啟動所需的時間。

Amazon EC2 會自動補充您現有的快照，因為它們會使用這些快照從啟用 EC2 快速啟動的 AMI 啟動執行個體。

任何在啟用 EC2 快速啟動的情況下存取 AMI 的帳戶都可以受益於縮短的啟動時間。當 AMI 擁有者授予您啟動執行個體的權限時，預先佈建的快照來自 AMI 擁有者的帳戶。

如果與您共用支援 EC2 快速啟動的 AMI，您可以自行在共用 AMI 上啟用或停用更快的啟動。如果您為 EC2 快速啟動啟用共用 AMI，Amazon EC2 會直接在您的帳戶中建立預先佈建的快照。如果您耗盡帳戶中的快照，仍可使用 AMI 擁有者帳戶中的快照。

### Note

EC2 Fast Launch 會在啟動時立即刪除預先佈建的快照，以將儲存成本降至最低並防止重複使用。不過，如果刪除的快照符合保留規則，資源回收筒會自動保留它們。建議您檢閱資源回收筒保留規則的範圍，以免發生此類情況。如需詳細資訊，請參閱 [考量事項](#)。

此功能與 [EBS 快速快照還原](#) 不同。您必須針對每個快照確實啟用 EBS 快速快照還原，並有其相關費用。

下列影片示範如何透過相關關鍵術語及其定義的快速概觀來設定 Windows AMI 以加快啟動速度：[啟動 EC2 Windows 執行個體的速度提升 65% AWS](#)。

## 資源成本

設定適用於 EC2 快速啟動的視窗 AMI 無須支付任何服務費用。不過，標準定價適用於 Amazon EC2 使用的任何基礎 AWS 資源。若要進一步了解相關資源成本以及如何管理這些成本，請參閱 [使用 EC2 快速啟動管理資源成本](#)。

## 目錄

- [重要用語](#)
- [EC2 快速啟動先決條件](#)
- [為您的亞馬遜 EC2 視窗伺服器 AMI 設定 Amazon EC2 快速啟動設定](#)
- [在啟用 EC2 快速啟動的情況下檢視 AMI](#)
- [使用 EC2 快速啟動管理資源成本](#)
- [監控 EC2 快速啟動](#)
- [EC2 快速啟動的服務連結角色](#)

## 重要用語

EC2 快速啟動功能使用下列關鍵術語：

### 已預先佈建的快照

從 Windows AMI 啟動並啟用 EC2 快速啟動的執行個體快照，並且已完成下列 Windows 啟動步驟，視需要重新開機。

- Sysprep 特製化
- Windows 開箱即用體驗 (OOBE)

完成這些步驟後，EC2 Fast Launch 會停止執行個體，並建立快照，以便稍後根據您的組態從 AMI 快速啟動。

### 啟動頻率

控制 Amazon EC2 可在指定時間範圍內啟動的預先佈建快照的數量。當您為 AMI 啟用 EC2 快速啟動時，Amazon EC2 會在背景中建立一組預先佈建的一組初始快照。例如，如果啟動頻率設定為每小時五次啟動 (預設值)，則 EC2 Fast Launch 會建立一組初始設定的五個預先佈建快照。

當 Amazon EC2 從已啟用 EC2 快速啟動的 AMI 啟動執行個體時，它會使用其中一個預先佈建的快照來縮短啟動時間。使用快照時，它們會自動補充，直到達到啟動頻率指定的數量。

如果您預計從 AMI 啟動的執行個體數量會出現峰值 (例如，在特殊事件期間)，則可以預先增加啟動頻率，以涵蓋所需的其他執行個體。當您的啟動速率恢復正常時，則可以將頻率降低回原始值。

當您遇到啟動次數高於預期時，則可能會用到所有可用的預先佈建快照。這不會引發任何啟動失敗。但是，它可能會使某些執行個體經歷標準啟動過程，直到快照得到補充為止。

## 目標資源計數

已啟用 EC2 快速啟動的 Amazon EC2 視窗伺服器 AMI 可保留的預先佈建快照數量。

## 最大平行啟動數

控制 Amazon EC2 可同時啟動多少個執行個體，以便為 EC2 快速啟動建立預先佈建的快照。如果您的目標資源計數高於設定的最大平行啟動數，Amazon EC2 會啟動開始建立快照之最大平行啟動數設定所指定的執行個體數。當這些執行個體完成此程序時，Amazon EC2 會擷取快照並停止執行個體。然後，它會繼續啟動更多執行個體，直到可用的快照總數達到目標資源計數為止。最大平行啟動數的值必須為 6 或更大。

## EC2 快速啟動先決條件

在設定 EC2 快速啟動之前，請確認您已符合為中的 AMI 建立快照所需的下列先決條件：AWS 帳戶

- 如果您不使用啟動範本來設定設定，請確定已針對您使用 EC2 快速啟動的區域設定預設 VPC。

### Note

如果您在計劃設定 EC2 快速啟動的區域中不小心刪除了預設 VPC，您可以在該區域中建立新的預設 VPC。如需進一步了解，請參閱《Amazon VPC 使用者指南》中的[建立預設的 VPC](#)。

- 若要指定非預設 VPC，您必須在設定 Windows Fast Launch 時使用啟動範本。如需詳細資訊，請參閱 [設定 EC2 快速啟動時使用啟動範本](#)。
- 如果您的帳戶包含對 Amazon EC2 執行個體強制實施 IMDSv2 的政策，您必須建立指定中繼資料組態強制執行 IMDSv2 的啟動範本。
- 私有 EC2 快速啟動 AMI 必須支援使用者資料指令碼執行。
- 若要為 AMI 設定 EC2 快速啟動，您必須使用 Sysprep 關閉選項來建立 AMI。EC2 快速啟動功能目前不支援從執行中執行個體建立的 AMI。

若要使用 Sysprep 建立 AMI，請參閱 [使用視窗系統建立 AMI](#)。

- AWS 帳戶 中所有 AMI 的最大平行啟動數預設配額為每個區域 40。您可為自己的帳戶請求增加 Service Quotas，如下所示。
  1. 登入 AWS Management Console 並開啟 Service Quotas 主控台，網址為 <https://console.aws.amazon.com/servicequotas/>。
  2. 在導覽窗格中，選擇 AWS 服務。
  3. 在搜尋列中輸入 EC2 Fast Launch，然後選取結果。
  4. 選取 Parallel instance launches 的連結。這會將帶您到平行執行個體啟動數服務配額詳細資訊頁面。
  5. 選擇 Request quota increase (請求增加配額)。

如需詳細資訊，請參閱《Service Quotas 使用者指南》中的[請求提高配額](#)。

## 為您的亞馬遜 EC2 視窗伺服器 AMI 設定 Amazon EC2 快速啟動設定

您可以針對您擁有的 Windows AMI 設定 EC2 快速啟動，或從 API AWS Management Console、開發套件或 AWS Command Line Interface () 與您共用的 AMI。CloudFormation AWS CLI在設定 EC2 快速啟動之前，請確認您的 AMI 符合建立預先佈建快照所需的所有必要條件。如需詳細資訊，請參閱[EC2 快速啟動先決條件](#)。

以下各節涵蓋 Amazon EC2 主控台和的設定步驟 AWS CLI。

### 啟用 EC2 快速啟動

若要啟用 EC2 快速啟動，請選擇符合您環境的索引標籤，然後按照步驟操作。

#### Note

在變更這些設定之前，請確保您的 AMI 和在其中執行的區域滿足所有 [EC2 快速啟動先決條件](#)。

### Console

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的 Images (映像) 下，選擇 AMIs (AIM)。
3. 選取要更新的 AMI，方法選取相應 Name (名稱) 旁邊的核取方塊。

4. 從 AMI 清單上方的動作功能表中選擇設定快速啟動。這會開啟 [設定快速啟動] 頁面，您可以在其中設定 EC2 快速啟動的設定。
5. 若要開始使用預先佈建的快照以更快地從 Windows AMI 啟動執行個體，請選取啟用 Windows 快速啟動核取方塊。
6. 從 Set anticipated launch frequency (設定預期啟動頻率) 下拉式清單中，選擇一個值以指定為涵蓋預期執行個體啟動磁碟區而建立和維護的快照數量。
7. 修改完成後，請選擇 Save changes (儲存變更)。

**Note**

如果您需要用啟動範本來指定非預設 VPC 或 IMDSv2 的中繼資料設定，請參閱 [設定 EC2 快速啟動時使用啟動範本](#)。

## AWS CLI

該 `enable-fast-launch` 命令會呼叫 Amazon EC2 [EnableFast啟動](#) API 操作。

語法:

```
aws ec2 enable-fast-launch \
 --image-id <value> \
 --resource-type <value> \ (optional)
 --snapshot-configuration <value> \ (optional)
 --launch-template <value> \ (optional)
 --max-parallel-launches <value> \ (optional)
 --dry-run | --no-dry-run \ (optional)
 --cli-input-json <value> \ (optional)
 --generate-cli-skeleton <value> \ (optional)
```

範例：

下列 [啟用快速啟動](#) 範例可為指定 AMI 啟用 EC2 快速啟動，啟動六個 parallel 執行個體進行預先佈建。ResourceType 設定為 snapshot，這是預設值。

```
aws ec2 enable-fast-launch \
 --image-id ami-01234567890abcdef \
 --max-parallel-launches 6 \
 --resource-type snapshot
```



```
--resource-type snapshot
```

輸出：

```
{
 "ImageId": "ami-01234567890abcdef",
 "ResourceType": "snapshot",
 "SnapshotConfiguration": {
 "TargetResourceCount": 10
 },
 "LaunchTemplate": {},
 "MaxParallelLaunches": 6,
 "OwnerId": "0123456789123",
 "State": "enabling",
 "StateTransitionReason": "Client.UserInitiated",
 "StateTransitionTime": "2022-01-27T22:16:03.199000+00:00"
}
```

## Tools for PowerShell

此指 `Enable-EC2FastLaunch` 令程式會呼叫 Amazon EC2 [EnableFast啟動](#) API 作業，以在您的視窗 AMI 上啟用 EC2 快速啟動。

語法：

```
Enable-EC2FastLaunch
 -ImageId <String>
 -LaunchTemplate_LaunchTemplateId <String>
 -LaunchTemplate_LaunchTemplateName <String>
 -MaxParallelLaunch <Int32>
 -ResourceType <String>
 -SnapshotConfiguration_TargetResourceCount <Int32>
 -LaunchTemplate_Version <String>
 -Select <String>
 -PassThru <SwitchParameter>
 -Force <SwitchParameter>
```

範例：

下列 [Enable-EC2FastLaunch](#) 範例會針對指定的 AMI 啟用 EC2 快速啟動，啟動六個 parallel 執行個體進行預先佈建。ResourceType 設定為 snapshot，這是預設值。

```
Enable-EC2FastLaunch `
 -ImageId ami-01234567890abcdef `
 -MaxParallelLaunch 6 `
 -Region us-west-2 `
 -ResourceType snapshot
```

輸出：

```
ImageId : ami-01234567890abcdef
LaunchTemplate :
MaxParallelLaunches : 6
OwnerId : 0123456789123
ResourceType : snapshot
SnapshotConfiguration : Amazon.EC2.Model.FastLaunchSnapshotConfigurationResponse
State : enabling
StateTransitionReason : Client.UserInitiated
StateTransitionTime : 2/25/2022 12:24:11 PM
```

## 停用 EC2 快速啟動

若要停用 EC2 快速啟動，請選擇符合您環境的索引標籤，然後按照步驟操作。

### Note

在變更這些設定之前，請確保您的 AMI 和在其中執行的區域滿足所有 [EC2 快速啟動先決條件](#)。

## Console

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的 Images (映像) 下，選擇 AMIs (AIM)。
3. 選取要更新的 AMI，方法選取相應 Name (名稱) 旁邊的核取方塊。
4. 從 AMI 清單上方的動作功能表中選擇設定快速啟動。這會開啟 [設定快速啟動] 頁面，您可以在其中設定 EC2 快速啟動的設定。
5. 清除 [啟用 Windows 快速啟動] 核取方塊以停用 EC2 快速啟動並移除預先佈建的快照。這會導致 AMI 將繼續為每個執行個體使用標準啟動流程。

**Note**

停用 Windows 映像最佳化時，將自動刪除所有現存的預先佈建快照。必須完成此步驟，然後才能再次開始使用該功能。

6. 修改完成後，請選擇 Save changes (儲存變更)。

## AWS CLI

該 `disable-fast-launch` 命令會呼叫 Amazon EC2 [DisableFast](#) 啟動 API 操作。

語法:

```
aws ec2 disable-fast-launch \
 --image-id <value> \
 --force | --no-force \ (optional)
 --dry-run | --no-dry-run \ (optional)
 --cli-input-json <value> \ (optional)
 --generate-cli-skeleton <value> \ (optional)
```

範例：

下列 [停用快速啟動](#) 範例會停用指定 AMI 上的 EC2 快速啟動，並清除現有預先佈建的快照。

```
aws ec2 disable-fast-launch \
 --image-id ami-01234567890abcdef
```

輸出：

```
{
 "ImageId": "ami-01234567890abcdef",
 "ResourceType": "snapshot",
 "SnapshotConfiguration": {},
 "LaunchTemplate": {
 "LaunchTemplateId": "lt-01234567890abcdef",
 "LaunchTemplateName": "EC2FastLaunchDefaultResourceCreation-
a8c6215d-94e6-441b-9272-dbd1f87b07e2",
 "Version": "1"
 },
}
```

```
"MaxParallelLaunches": 6,
"OwnerId": "0123456789123",
"State": "disabling",
"StateTransitionReason": "Client.UserInitiated",
"StateTransitionTime": "2022-01-27T22:47:29.265000+00:00"
}
```

## Tools for PowerShell

該Disable-EC2FastLaunch指令程式會呼叫 Amazon EC2 [DisableFast啟動](#) API 作業。

語法:

```
Disable-EC2FastLaunch
 -ImageId <String>
 -ForceStop <Boolean>
 -Select <String>
 -PassThru <SwitchParameter>
 -Force <SwitchParameter>
```

範例 :

下列[Disable-EC2FastLaunch](#)範例會停用指定 AMI 上的 EC2 快速啟動，並清除現有預先佈建的快照。

```
Disable-EC2FastLaunch -ImageId ami-01234567890abcdef
```

輸出 :

```
ImageId : ami-01234567890abcdef
LaunchTemplate :
Amazon.EC2.Model.FastLaunchLaunchTemplateSpecificationResponse
MaxParallelLaunches : 6
OwnerId : 0123456789123
ResourceType : snapshot
SnapshotConfiguration :
State : disabling
StateTransitionReason : Client.UserInitiated
StateTransitionTime : 2/25/2022 1:10:08 PM
```

## 設定 EC2 快速啟動時使用啟動範本

透過啟動範本，您可以設定 Amazon EC2 每次從該範本啟動執行個體時使用的一組啟動參數。您可以指定 AMI 等項目，以用於基礎映像、執行個體類型、儲存體、網路設定等。

啟動範本是選擇性的，但下列特定情況除外；當您設定更快速啟動功能時，您必須為 Windows AMI 使用啟動範本：

- 您必須使用啟動範本為 Windows AMI 指定非預設 VPC。
- 如果您的帳戶包含對 Amazon EC2 執行個體強制實施 IMDSv2 的政策，您必須建立指定中繼資料組態強制執行 IMDSv2 的啟動範本。

使用包含 EC2 主控台中繼資料組態的啟動範本，或在中執行啟用[快速啟動](#)命令 AWS CLI，或呼叫啟[EnableFast動](#) API 動作時。

當您使用啟動範本時，Amazon EC2 EC2 快速啟動不支援下列組態。如果您使用 EC2 快速啟動的啟動範本，則不得指定下列任何項目：

- 使用者資料指令碼
- 終止保護
- 停用的詮釋
- 定點選項
- 終止執行處理的關閉行為
- 網路介面、彈性圖形或 Spot 執行個體請求的資源標籤

### 指定非預設 VPC

#### 步驟 1：建立啟動範本

建立指定 Windows 執行個體下列詳細資訊的啟動範本：

- VPC 子網路。
- t3.xlarge 的執行個體類型。

如需詳細資訊，請參閱 [建立啟動範本](#)。

#### 步驟 2：為您的 EC2 快速啟動 AMI 指定啟動範本

選擇符合您程序的索引標籤：

## Console

若要從指定 EC2 快速啟動的啟動範本 AWS Management Console，請依照下列步驟執行：

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格的 Images (映像) 下，選擇 AMIs (AIM)。
3. 選取要更新的 AMI，方法選取相應 Name (名稱) 旁邊的核取方塊。
4. 從 AMI 清單上方的動作功能表中選擇設定快速啟動。這會開啟 [設定快速啟動] 頁面，您可以在其中設定 EC2 快速啟動的設定。
5. 此啟動範本方框會執行篩選搜尋，在您帳戶目前區域中搜尋與您輸入文字相符的啟動範本。在方框中指定所有或部分的啟動範本名稱或 ID，以比對出相符的啟動範本清單。例如，如果您在方框中輸入 fast，Amazon EC2 會在您帳戶目前的區域當中，搜尋所有名稱中包含「快速」的啟動範本。

若要新建一個啟動範本，請選擇 Create launch template (建立啟動範本)。

6. 當您選取啟動範本時，Amazon EC2 會在 Source template version (來源範本版本) 方框中列出該範本的預設版本。若要指定其他版本，請反白要替換的預設版本，然後在方框中輸入所需的版本編號。
7. 修改完成後，請選擇 Save changes (儲存變更)。

## AWS CLI, API

若要從中指定 EC2 快速啟動的啟動範本 AWS CLI，請在中執行啟用 [快速啟動命令時](#)，在 `--launch-template` 參數中指定啟動範本名稱或 ID。AWS CLI

若要在 API 請求中指定 EC2 快速啟動的啟動範本，請在呼叫啟動 API 動作時，在 `LaunchTemplate` 參數中指定 [EnableFast](#) 啟動範本名稱或 ID。

如需 EC2 啟動範本的詳細資訊，請參閱 [從啟動範本啟動執行個體](#)。

在啟用 EC2 快速啟動的情況下建立自訂映像

Amazon EC2 EC2 快速啟動與 EC2 Image Builder 整合，協助您在啟用 EC2 快速啟動的情況下建立自訂映像。如需詳細資訊，請參閱 EC2 Image Builder 使用者指南中的 [為啟用 EC2 快速啟動的 Windows AMI \(AWS CLI\) 建立分佈設定](#)。

## 在啟用 EC2 快速啟動的情況下檢視 AMI

您可以使用中的[描述快速啟動映像](#)命令 AWS CLI，或使用 PowerShell Cmdlet 的[Get-EC2FastLaunchImage](#)工具來取得啟用 EC2 快速啟動之 AMI 的詳細資料。

Amazon EC2 為結果中傳回的每個 Windows AMI 提供以下詳細資訊：

- 啟用 EC2 快速啟動的 AMI 的映像 ID。
- 用於預先佈建相關 Windows AMI 的資源類型。支援的值為：snapshot。
- 快照組態是一組參數，使用快照預先佈建關聯的 Windows AMI。
- 啟動範本資訊，包括相關 AMI 從預先佈建快照啟動 Windows 執行個體時使用的啟動範本的 ID、名稱和版本。
- 為建立資源而同時啟動的最大執行個體數。
- 關聯 AMI 的擁有者 ID。系統不會為與您共享的 AMI 填入此項目。
- 相關 AMI 的 EC2 快速啟動的目前狀態。支援的值包括：enabling | enabling-failed | enabled | enabled-failed | disabling | disabling-failed。

### Note

您還可以在 EC2 主控台的 Manage image optimization (管理映像最佳化) 頁面中，看到目前狀態顯示為 Image optimization state (映像最佳化狀態)。

- 相關 AMI 的 EC2 快速啟動變更為目前狀態的原因。
- 相關 AMI 的 EC2 快速啟動變更為目前狀態的時間。

選擇與您的命令列環境相符的索引標籤：

### AWS CLI

該describe-fast-launch-images命令調用 Amazon EC2 [DescribeFastLaunchImages](#)API 操作。

語法:

```
aws ec2 describe-fast-launch-images \
 --image-ids <value> \ (optional) \
 --filters <value> \ (optional) \
 --dry-run | --no-dry-run \ (optional) \
 --cli-input-json <value> \ (optional)
```

```
--starting-token <value> \ (optional)
--page-size <value> \ (optional)
--max-items <value> \ (optional)
--generate-cli-skeleton <value> \ (optional)
```

範例：

下列 [描述快速啟動映像](#) 範例說明為 EC2 快速啟動設定的帳戶中每個 AMI 的詳細資料。在此範例中，帳戶中只有一個 AMI 設定為 EC2 快速啟動。

```
aws ec2 describe-fast-launch-images
```

輸出：

```
{
 "FastLaunchImages": [
 {
 "ImageId": "ami-01234567890abcdef",
 "ResourceType": "snapshot",
 "SnapshotConfiguration": {},
 "LaunchTemplate": {
 "LaunchTemplateId": "lt-01234567890abcdef",
 "LaunchTemplateName": "EC2FastLaunchDefaultResourceCreation-
a8c6215d-94e6-441b-9272-dbd1f87b07e2",
 "Version": "1"
 },
 "MaxParallelLaunches": 6,
 "OwnerId": "0123456789123",
 "State": "enabled",
 "StateTransitionReason": "Client.UserInitiated",
 "StateTransitionTime": "2022-01-27T22:20:06.552000+00:00"
 }
]
}
```

## Tools for PowerShell

指 `Get-EC2FastLaunchImage` 令程式會呼叫 Amazon EC2 [DescribeFastLaunchImages](#) API 作業。

語法：

```
Get-EC2FastLaunchImage
```



```
-Filter <Filter[]>
-ImageId <String[]>
-MaxResult <Int32>
-NextToken <String>
-Select <String>
-NoAutoIteration <SwitchParameter>
```

範例：

下列 [Get-EC2FastLaunchImage](#) 範例說明為 EC2 快速啟動設定的帳戶中每個 AMI 的詳細資料。在此範例中，帳戶中只有一個 AMI 設定為 EC2 快速啟動。

```
Get-EC2FastLaunchImage -ImageId ami-01234567890abcdef
```

輸出：

```
ImageId : ami-01234567890abcdef
LaunchTemplate :
 Amazon.EC2.Model.FastLaunchLaunchTemplateSpecificationResponse
MaxParallelLaunches : 6
OwnerId : 0123456789123
ResourceType : snapshot
SnapshotConfiguration :
State : enabled
StateTransitionReason : Client.UserInitiated
StateTransitionTime : 2/25/2022 12:54:43 PM
```

## 使用 EC2 快速啟動管理資源成本

設定適用於 EC2 快速啟動的視窗 AMI 無須支付任何服務費用。不過，當您為 Amazon EC2 Windows AMI 啟用 EC2 快速啟動時，標準定價適用於 Amazon EC2 用來準備和存放預先佈建的快照的基礎 AWS 資源。您可以設定成本分配標籤，以協助您追蹤和管理與 EC2 快速啟動資源相關聯的成本。如需如何設定成本分配標籤的詳細資訊，請參閱 [追蹤帳單上的 EC2 快速啟動成本](#)。

下列範例示範如何配置與 EC2 快照相關聯的成本。

範例情境：AtoZ Example 公司擁有一個具有 50 GiB EBS 根磁碟區的 Windows AMI。他們為 AMI 啟用 EC2 快速啟動，並將目標資源計數設置為五。在一個月的過程中，使用 EC2 快速啟動 AMI 的費用約為 5.00 美元，成本明細如下：

1. 當 AtoZ 範例啟用 EC2 快速啟動時，Amazon EC2 會啟動五個小型執行個體。每個執行個體都會執行 Sysprep 和 OOB Windows 啟動步驟，並根據需要重新啟動。每個執行個體都需要幾分鐘的時間 (取決於該區域或可用區域 (AZ) 的忙碌程度以及 AMI 的大小，時間可能會有所不同)。

#### 成本

- 執行個體執行時間成本 (或最低執行時間，如果適用)：五個執行個體
  - 磁碟區成本：五個 EBS 根磁碟區
2. 預先佈建過程完成後，Amazon EC2 將產生執行個體的快照，並將其存放在 Amazon S3 中。快照通常會存放 4 到 8 小時，然後才會啟動使用。在這種情況下，每個快照的成本大約為 0.02 到 0.05 USD。

#### 成本

- 快照儲存 (Amazon S3)：五個快照
3. Amazon EC2 拍攝快照後，即會停止執行個體。此時，執行個體將不再累計成本。然而，EBS 磁碟區成本繼續累計。

#### 成本

- EBS 磁碟區：相關 EBS 根磁碟區的成本繼續存在。

#### Note

此處顯示的費用僅供示範用途。您的成本會有所不同，具體取決於您的 AMI 組態和定價計劃。

### 追蹤帳單上的 EC2 快速啟動成本

成本分配標籤可協助您整理 AWS 帳單，以反映與 EC2 快速啟動相關的成本。您可以使用 Amazon EC2 在為 EC2 快速啟動準備和存放預先佈建的快照時，將其新增至其建立的資源中的下列標籤：

標籤索引鍵：CreatedBy，值：EC2 Fast Launch

在帳單與成本管理主控台中啟用標籤並設定詳細的帳單報告後，user:CreatedBy 欄出現在報告中。此欄包含來自所有服務的值。不過，如果您下載了 CSV 檔案，則可以將資料匯入試算表，並在值中篩選 EC2 Fast Launch。啟動標籤 AWS Cost and Usage Report 時，此資訊也會顯示在中。

### 步驟 1：啟用使用者定義的成本分配標籤

若要在成本報告中包含資源標籤，您必須先在帳單與成本管理主控台中啟用該標籤。如需詳細資訊，請參閱《AWS Billing and Cost Management 使用者指南》中的[啟用使用者定義的成本分配標籤](#)。

#### Note

啟用最多可能需要 24 小時。

## 步驟 2：設定成本報告

如果您已設定成本報告，則下次啟用完成後執行報告時，標籤的資料欄會出現。若是第一次設定成本報告，請選擇以下其中一項。

- 請參閱《AWS Billing and Cost Management 使用者指南》中的[設定每月成本分配報告](#)。
- 請參閱《AWS Cost and Usage Report 使用者指南》中的[建立成本和用量報告](#)。

#### Note

最多可能需要 24 小時才能開始 AWS 將報告傳遞到 S3 儲存貯體。

您可以為您擁有的 Windows AMI 設定 EC2 快速啟動，或是從中的 Amazon EC2 主控台、API、開發套件或 ec2 命令與您共用的 AMI。[CloudFormation](#) AWS CLI 以下各節涵蓋 Amazon EC2 主控台和的設定步驟 AWS CLI。

您也可以建立自訂視窗 AMI，這些 AMI 已設定為 EC2 使用 EC2 Image Builder 快速啟動。如需詳細資訊，請參閱[針對已啟用 EC2 快速啟動的 Windows AMI 建立發佈設定 \(AWS CLI\)](#)。

## 監控 EC2 快速啟動

本節介紹如何監控已啟用 EC2 快速啟動的帳戶中的 Amazon EC2 視窗伺服器 AMI。

使用以監控 EC2 快速啟動狀態變更 EventBridge

在啟用 EC2 快速啟動的情況下，當 Windows AMI 的狀態變更時，Amazon EC2 會產生一個 EC2 Fast Launch State-change Notification 事件。然後 Amazon EC2 將狀態更改事件發送到 Amazon EventBridge（以前稱為 Amazon CloudWatch 活動）。

您可以建立 EventBridge 規則來觸發一或多個動作以回應狀態變更事件。例如，您可以建立 EventBridge 規則來偵測 EC2 快速啟動何時啟用，並執行下列動作：

- 將訊息傳送至 Amazon SNS 主題，通知其訂閱用戶。
- 叫用執行某些動作的 Lambda 函數。
- 將狀態變更資料傳送至 Amazon 資料 Firehose 進行分析。

如需詳細資訊，請參閱 [Amazon EventBridge 使用者指南中的建立針對事件做出回應的 Amazon EventBridge 規則](#)。

## 狀態變更事件

EC2 快速啟動功能會盡力發出 JSON 格式的狀態變更事件。Amazon EC2 以近乎實時的方 EventBridge 式將事件發送到。本節說明事件欄位，並顯示事件格式範例。

### EC2 Fast Launch State-change Notification

imageId

透過 EC2 快速啟動狀態變更識別 AMI。

resourceType

用於預先佈建的資源類型。支援的值為：snapshot。預設值為 snapshot。

state

指定 AMI 的 EC2 快速啟動功能的目前狀態。有效值包括以下項目：

- 啟用 — 您已為 AMI 啟用 EC2 快速啟動功能，而 Amazon EC2 已開始為預先佈建程序建立快照。
- 啟用失敗 — 發生錯誤，導致您第一次為 AMI 啟用 EC2 快速啟動時，預先佈建程序失敗。這可能會在預先佈建程序期間隨時發生。
- 啟用 — 已啟用 EC2 快速啟動功能。一旦 Amazon EC2 enabled 為新啟用的 EC2 快速啟動 AMI 建立第一個預先佈建的快照，狀態就會變更為。如果 AMI 已啟用並再次進行預先佈建，狀態變更會立即發生。
- 啟用-失敗 — 僅當 EC2 快速啟動 AMI 不是第一次完成預先佈建程序時，才會套用此狀態。如果禁用 EC2 Fast Launch 功能，然後再次啟用，或者在第一次完成預先佈建後發生組態變更或其他錯誤，則可能會發生這種情況。
- 停用 — AMI 擁有者已關閉 AMI 的 EC2 快速啟動功能，而 Amazon EC2 已開始清理程序。
- 停用 — EC2 快速啟動功能已停用。Amazon EC2 完成清理程序後，狀態就會變更為 disabled。

- 禁用失敗 — 出現問題，導致清理程序失敗。這意味著某些預先佈建的快照可能仍會保留在帳戶中。

## 狀態 TransitionReason

EC2 快速啟動 AMI 狀態更改的原因。

### Note

此事件訊息中的所有欄位都是必填欄位。

下列範例顯示新啟用的 EC2 快速啟動 AMI，該 AMI 已啟動第一個執行個體以啟動預先佈建程序。此時，狀態為 `enabling`。Amazon EC2 建立第一個預先佈建的快照後，狀態會變更為 `enabled`。

```
{
 "version": "0",
 "id": "01234567-0123-0123-0123-012345678901",
 "detail-type": "EC2 Fast Launch State-change Notification",
 "source": "aws.ec2",
 "account": "123456789012",
 "time": "2022-08-31T20:30:12Z",
 "region": "us-east-1",
 "resources": [
 "arn:aws:ec2:us-east-1:123456789012:image/ami-123456789012"
],
 "detail": {
 "imageId": "ami-123456789012",
 "resourceType": "snapshot",
 "state": "enabling",
 "stateTransitionReason": "Client.UserInitiated"
 }
}
```

## 使用監控 EC2 快速啟動指標 CloudWatch

具有 Amazon EC2 快速啟動功能的 Amazon EC2 AMI 將指標發送到亞馬遜 CloudWatch。您可以使用 AWS Management Console AWS CLI、或 API 列出 EC2 快速啟動傳送到的指標 CloudWatch。命 `AWS/EC2` 名空間包括下列 EC2 快速啟動指標：

指標	描述
NumberOfAvailableFastLaunchSnapshots	每個已啟用 EC2 快速啟動的 AMI 可用的預先佈建快照數量。
NumberOfInstancesFast推出	從預先佈建的快照啟用每個已啟用 EC2 快速啟動之 AMI 的執行個體數量。
NumberOfInstancesNotFastLaunched	由於啟動時缺少可用的預先佈建快照，導致每個啟用 EC2 快速啟動的 AMI 執行個體數量導致冷開機。
FastLaunchSnapshotUsedToRefillStartTime	Amazon EC2 從 EC2 快速啟動啟動新映像時的時間戳記可讓 AMI 在使用現有快照後建立另一個快照。
FastLaunchSnapshotCreation時間	測量 Amazon EC2 啟動執行個體並為啟用 EC2 快速啟動的 AMI 建立快照所花費的時間。

## EC2 快速啟動的服務連結角色

Amazon EC2 使用許可的服務連結角色，它需要代表您呼叫其他 AWS 服務。服務連結角色是一種獨特的 IAM 角色類型，可直接連結至 AWS 服務服務連結角色提供委派權限的安全方式，AWS 服務因為只有連結的服務可以擔任服務連結的角色。如需 Amazon EC2 如何使用 IAM 角色的詳細資訊，請參閱 [Amazon EC2 的 IAM 的角色](#)。

Amazon EC2 使用名為 `AWSServiceRoleForEC2FastLaunch` 的服務連結角色建立和管理一組預先佈建的快照，從而縮短從 Windows AMI 啟動執行個體所需的時間。

您不需要手動建立此服務連結角色。當您開始為 AMI 使用 EC2 快速啟動時，Amazon EC2 會為您建立服務連結角色 (如果尚未存在)。

### Note

如果從您的帳戶中刪除服務連結角色，您可以為另一個 Windows AMI 啟用 EC2 快速啟動，以在您的帳戶中重新建立角色。或者，您可以針對目前的 AMI 停用 EC2 快速啟動，然後再次啟用。但是，停用該功能會導致 AMI 對所有新執行個體使用標準啟動過程，而 Amazon EC2

會移除您的所有預先佈建快照。在所有預先佈建的快照都消失之後，您可以再次為 AMI 啟用 EC2 快速啟動。

Amazon EC2 不允許您編輯 `AWSServiceRoleForEC2FastLaunch` 服務連結角色。因為有各種實體可能會參考服務連結角色，所以您無法在建立角色之後變更角色名稱。然而，您可使用 IAM 來編輯角色描述。如需詳細資訊，請參閱《IAM 使用者指南》中的[編輯服務連結角色](#)。

您必須先刪除所有相關的資源，才能刪除服務連結角色。這樣可以在啟用 EC2 快速啟動的情況下保護與您的 Amazon EC2 Windows 伺服器 AMI 相關聯的 Amazon EC2 資源，因為您不會意外移除存取資源的許可。

在提供 Amazon EC2 服務的所有區域中，Amazon EC2 都支援 EC2 快速啟動服務連結角色。如需詳細資訊，請參閱 [區域](#)。

### **AWSServiceRoleForEC2FastLaunch** 授予的許可

Amazon EC2 使用 `EC2FastLaunchServiceRolePolicy` 受管政策來完成下列動作：

- `cloudwatch:PutMetricData`— 將與 EC2 快速啟動相關聯的指標資料張貼至 Amazon EC2 命名空間。
- `ec2:CreateLaunchTemplate`— 在啟用 EC2 快速啟動的情況下為您的 Amazon EC2 視窗伺服器 AMI 建立啟動範本。
- `ec2:CreateSnapshot`— 在啟用 EC2 快速啟動的情況下，為您的 Amazon EC2 視窗伺服器 AMI 建立預先佈建的快照。
- `ec2:CreateTags`— 為與啟用 EC2 快速啟動的 Amazon EC2 Windows 伺服器 AMI 啟動和預先佈建 Windows 執行個體相關聯的資源建立標籤。
- `ec2:DeleteSnapshots`— 如果先前啟用的 AMI 已關閉 EC2 快速啟動，請刪除所有關聯的預先佈建快照。
- `ec2:DescribeImages` – 描述所有資源的映像。
- `ec2:DescribeInstanceAttribute` – 描述所有資源的執行個體屬性。
- `ec2:DescribeInstanceState` – 描述所有資源的執行個體狀態。
- `ec2:DescribeInstances` – 描述所有資源的執行個體。
- `ec2:DescribeInstanceTypeOfferings` – 描述適用於所有資源的執行個體類型方案。
- `ec2:DescribeLaunchTemplates` – 描述所有資源的啟動範本。

- `ec2:DescribeLaunchTemplateVersions` – 描述所有資源的啟動範本版本。
- `ec2:DescribeSnapshots` – 描述所有資源的快照資源。
- `ec2:DescribeSubnets` – 描述所有資源的子網。
- `ec2:RunInstances`— 在啟用 EC2 快速啟動的情況下，從 Amazon EC2 Windows 伺服器 AMI 啟動執行個體，以執行佈建步驟。
- `ec2:StopInstances`— 停止從已啟用 EC2 快速啟動的 Amazon EC2 Windows 伺服器 AMI 啟動的執行個體，以建立預先佈建的快照。
- `ec2:TerminateInstances`— 在從執行個體建立預先佈建的快照之後，終止從已啟用 EC2 快速啟動的 Amazon EC2 Windows 伺服器 AMI 啟動的執行個體。
- `iam:PassRole` – 允許 `AWSServiceRoleForEC2FastLaunch` 服務連結角色使用啟動範本中的執行個體設定檔代表您啟動執行個體。

如需使用 Amazon EC2 受管政策的詳細資訊，請參閱 [AWS Amazon EC2 的受管政策](#)。

存取客戶受管金鑰，以便與加密的 AMI 和 EBS 快照搭配使用

#### 先決條件

- 若要讓 Amazon EC2 能夠代表您存取加密 AMI，您必須在客戶受管金鑰中擁有 `createGrant` 動作的許可。

當您為加密 AMI 啟用 EC2 快速啟動時，Amazon EC2 會確保授與該 `AWSServiceRoleForEC2FastLaunch` 角色使用客戶受管金鑰存取 AMI 的許可。代表您啟動執行個體和建立預先佈建的快照需要此許可。

## 在 Windows 執行個體上使用 Amazon 彈性圖形加速器

### Important

Amazon 彈性圖形於 2024 年 1 月 8 日終止使用壽命。對於需要圖形加速的工作負載，我們建議您使用 Amazon EC2 G4ad、G4dn 或 G5 執行個體。

Amazon Elastic Graphics 為您的 Windows 執行個體提供靈活、低成本和高效能的圖形加速功能。彈性圖形加速器有多種尺寸，是使用 GPU 圖形執行個體類型 (例如 G3) 的低成本替代方案。您可靈活選



擇滿足您應用程式運算、記憶體和儲存需求的執行個體類型。然後，為您的執行個體選擇滿足您工作負載圖形要求的加速器。

Elastic Graphics 適用於需要少量或間歇性額外圖形加速，且使用 OpenGL 圖形支援的應用程式。如果您需要存取直接連接的完整 GPU 並使用 DirectX、CUDA 或開放運算語言 (OpenCL) 平行運算框架，請改為使用加速運算執行個體類型執行個體。

## 目錄

- [Elastic Graphics 基本概念](#)
- [Elastic Graphics 定價](#)
- [Elastic Graphics 限制](#)
- [使用 Elastic Graphics](#)
- [Elastic Graphics 維護](#)
- [使用 CloudWatch 指標監視彈性圖形](#)
- [疑難排解](#)

## Elastic Graphics 基本概念

若要使用彈性圖形，請啟動 Windows 例證並在啟動期間指定例證的加速器類型。AWS 尋找可用的彈性圖形容量，並在執行個體與彈性圖形加速器之間建立網路連線。

### Note

不支援裸機執行個體。

彈性圖形加速器可在下列 AWS 區域中使用：us-east-1、us-east-2、us-west-2、ap-northeast-1、ap-southeast-1、ap-southeast-2、eu-central-1、和 eu-west-1。

下列的執行個體類型支援 Elastic Graphics 加速器：

- 一般用途：M3、M4、M5、M5d、M5dn、M5n、T2、T3

### Note

僅支援 t2.medium 和以上以及 t3.medium 和以上。

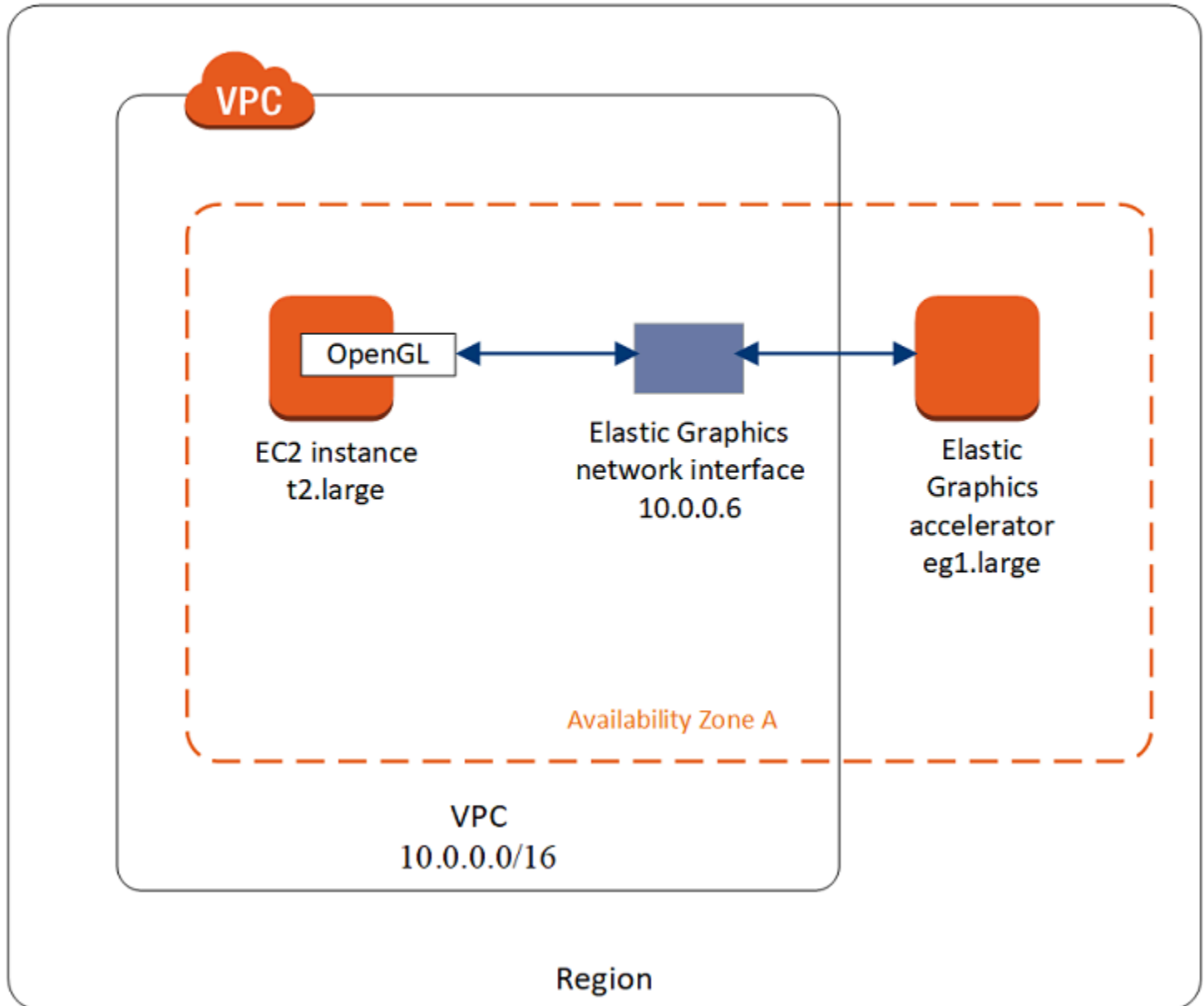
- 運算最佳化：C3、C4、C5、C5a、C5ad、C5d、C5n
- 記憶體最佳化：R3、R4、R5、R5d、R5dn、R5n、X1、X1e、z1d
- 儲存最佳化：D2、D3、D3en、H1、I3、I3en
- 加速運算：P2、P3、P3dn

提供下列 Elastic Graphics 加速器。您可以將任何 Elastic Graphics 加速器連接至任何支援的執行個體類型。

Elastic Graphics 加速器	圖形記憶體 (GB)
eg1.medium	1
eg1.large	2
eg1.xlarge	4
eg1.2xlarge	8

Elastic Graphics 加速器不構成您執行個體硬體的一部分。而是透過稱為「Elastic Graphics 網路介面」的網路介面連接到網路。當您使用圖形加速啟動或重新啟動執行個體時，即會為您在 VPC 中建立 Elastic Graphics 網路介面。

Elastic Graphics 網路介面會建立在與您執行個體相同的子網路和 VPC 中，並從該子網路獲指派私有的 IPv4 地址。連接到您 Amazon EC2 執行個體的加速器是從您執行個體所在之同一可用區域中的可用加速器集區中所配置。



Elastic Graphics 加速器支援 OpenGL 4.3 API 及舊版的 API 標準，可用於批次應用程式或 3D 圖形加速。執行個體上的 Amazon 最佳化 OpenGL 程式庫會偵測連接的加速器。它會將 OpenGL API 呼叫從您的執行個體導向至加速器，然後處理請求並傳回結果。執行個體和加速器之間的流量會使用與執行個體網路流量相同的頻寬，因此建議您準備足夠的可用網路頻寬。請向您的軟體供應商諮詢任何 OpenGL 合規和版本問題。

根據預設，VPC 的預設安全群組與 Elastic Graphics 網路介面相關聯。Elastic Graphics 網路流量使用 TCP 通訊協定和連接埠 2007。請確定您執行個體的安全群組允許這一點。如需詳細資訊，請參閱 [設定安全群組](#)。

## Elastic Graphics定價

當加速器為 `running` 狀態時，狀態為 `Ok` 並連接至執行個體的 Elastic Graphics 加速器，每秒都要付費。若加速器連接至 `pending`、`stopping`、`stopped`、`shutting-down` 或 `terminated` 狀態的執行個體，則不會向您收取費用。當加速器狀態為 `Unknown` 或 `Impaired` 時，也不收取費用。

加速器的定價僅適用於隨需費率。您可以將加速器連接到預留執行個體或 Spot 執行個體，但是需支付加速器的隨需價格。

如需詳細資訊，請參閱 [Amazon Elastic Graphics 定價](#)。

## Elastic Graphics 限制

開始使用 Elastic Graphics 加速器前，請注意下列限制：

- 您只可將加速器連接到使用 Microsoft Windows Server 2012 R2 或更新版本的 Windows 執行個體。目前不支援 Linux 執行個體。
- 您可以一次將一個加速器附加到執行個體上。
- 您只能在執行個體啟動期間附加加速器。您無法將加速器附加至現有執行個體。
- 您無法讓附加加速器的執行個體進入休眠。
- 執行個體之間無法共享一個加速器。
- 您無法分離加速器和執行個體，也無法將其轉移至另一個執行個體。如果您不再需要某個加速器，您必須終止您的執行個體。若要變更加速器類型，請從您的執行個體建立 AMI、終止該執行個體，然後使用不同的加速器規格啟動新執行個體。
- 只支援 4.3 版及舊版的 OpenGL API。DirectX、CUDA 和 OpenCL 並不支援。
- 無法透過您執行個體的裝置管理員看到或存取 Elastic Graphics 加速器。
- 您無法預留或排程加速器容量。

## 使用 Elastic Graphics

### Important

Amazon 彈性圖形於 2024 年 1 月 8 日終止使用壽命。對於需要圖形加速的工作負載，我們建議您使用 Amazon EC2 G4ad、G4dn 或 G5 執行個體。

您可以在啟動期間啟動執行個體，並建立其與 Elastic Graphics 加速器的關聯。您必須在您的執行個體上手動安裝必要的程式庫，以便與加速器進行通訊。如需限制的詳細資訊，請參閱[Elastic Graphics 限制](#)。

## 工作

- [設定安全群組](#)
- [啟動附有 Elastic Graphics 加速器的執行個體](#)
- [安裝 Elastic Graphics 所需的軟體](#)
- [確認執行個體的 Elastic Graphics 功能](#)
- [檢視 Elastic Graphics 資訊](#)
- [提交意見回饋](#)

## 設定安全群組

Elastic Graphics 需要有自我參考的安全群組來允許往返安全群組自身的所有傳入和傳出流量。安全群組必須包含以下傳入及傳出規則。

### 傳入

Type	通訊協定	連線埠	來源
Elastic Graphics	TCP	2007	安全群組 ID (其本身的資源 ID)

### 傳出

Type	通訊協定	連接埠範圍	目的地
Elastic Graphics	TCP	2007	安全群組 ID (其本身的資源 ID)

如果您使用 Amazon EC2 主控台啟動帶有 Elastic Graphics 加速器的執行個體，您可以允許啟動執行個體精靈來自動建立所需的安全群組規則，或選取先前建立的安全群組規則。

如果您要使用 AWS CLI 或 SDK 啟動執行個體，則必須指定先前建立的安全性群組。

## 為 Elastic Graphics 建立安全群組

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。

2. 在導覽窗格中，選擇 Security Groups (安全群組)，然後選擇 Create Security Group (建立安全群組)。
3. 在 Create Security Group (建立安全群組) 視窗中，執行下列動作：
  - a. 對於 Security group name (安全群組名稱)，輸入安全群組的描述性名稱，例如 Elastic Graphics security group。
  - b. (選用) 對於 Description (描述)，輸入安全群組的簡短描述。
  - c. 在 VPC 中，選取您打算讓 Elastic Graphics 在其中使用的 VPC。
  - d. 選擇建立安全群組。
4. 在導覽窗格中，選擇 Security Groups (安全群組)，選取您剛建立的安全群組，然後在 Details (詳細資訊) 索引標籤上，複製 Security group ID (安全群組 ID)。
5. 在 Inbound rules (傳入規則) 標籤上，選擇 Edit inbound rules (編輯傳入規則)，然後執行以下動作：
  - a. 選擇新增規則。
  - b. 針對 Type (類型)，選擇 Elastic Graphics。
  - c. 在 Source type (來源類型) 中，選擇 Custom (自訂)。
  - d. 在 Source (來源) 中，貼上先前複製的安全群組 ID。
  - e. 選擇儲存規則。
6. 在 Outbound rules (傳出規則) 上，選擇 Edit outbound rules (編輯傳出規則)，然後執行以下動作：
  - a. 選擇新增規則。
  - b. 針對 Type (類型)，選擇 Elastic Graphics。
  - c. 在 Destination type (目的地類型) 中，選擇 Custom (自訂)。
  - d. 在 Destination (目的地) 中，貼上先前複製的安全群組 ID。
  - e. 選擇儲存規則。

如需詳細資訊，請參閱 [適用於 Amazon EC2 執行個體的亞馬遜 EC2 安全群組](#)。

### 啟動附有 Elastic Graphics 加速器的執行個體

您可以在啟動期間建立 Elastic Graphics 加速器與執行個體的關聯。如果啟動失敗，可能原因如下：

- Elastic Graphics 加速器容量不足

- 已超過該區域的 Elastic Graphics 加速器限制
- VPC 中沒有足夠的私有 IPv4 地址，無法為加速器建立網路介面

如需詳細資訊，請參閱 [Elastic Graphics 限制](#)。

在執行個體啟動期間建立 Elastic Graphics 加速器的關聯 (主控台)

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在儀表板中，選擇 Launch Instance (啟動執行個體)。
3. 在「名稱和標籤」下，輸入「名稱」的值。您可以選擇性地選擇「新增其他標籤」，將更多標籤新增至與您要啟動的執行個體相關聯的資源。
4. 在應用程式和作業系統映像 (Amazon 機器映像) 下，選取 Windows AMI。
5. 在 Instance type (執行個體類型) 中，選取支援的執行個體類型。如需詳細資訊，請參閱 [Elastic Graphics 基本概念](#)。
6. 在 Key pair (login) (金鑰對 (登入)) 下，針對 Key pair name (金鑰對名稱)，選擇現有的金鑰對或建立新的金鑰對。
7. 在 [網路設定] 旁邊，選擇 [編輯]，然後指定要用於執行個體的網路設定。
  - a. 在 [網路] 中，為您的執行個體選取 VPC。
  - b. 在子網路中，選取要在其中啟動執行個體的字網路。
  - c. 對於防火牆 (安全群組)，您可以使用在中手動建立的安全性群組 [設定安全群組](#)，或讓主控台為您建立具有必要輸入和輸出規則的安全性群組。視需要新增其他安全群組。
8. (選擇性) 在設定儲存區下，設定根磁碟區大小，並視需要新增其他磁碟區。
9. 展開 [進階詳細資料] 區段。
10. 在 [進階詳細資料] 下方，針對 Elastic GPU，選取彈性圖形加速器類型。
11. 在 Summary (摘要) 面板中，選擇 Launch instance (啟動執行個體)。

在執行個體啟動期間建立 Elastic Graphics 加速器的關聯 (AWS CLI)

您可以搭配下列參數使用 [執行個體](#) AWS CLI 命令：

```
--elastic-gpu-specification Type=eg1.medium
```

若為 `--security-group-ids` 參數，您必須包含有所需傳入及傳出規則的安全性群組。如需詳細資訊，請參閱 [設定安全群組](#)。

欲在例證啟動期間關聯彈性圖形加速器 (Windows 專用工具 PowerShell)

使用「視窗 [New-EC2Instance](#) 工具」 PowerShell 指令。

安裝 Elastic Graphics 所需的軟體

如果您使用目前的 AWS Windows AMI 啟動執行個體，則會在第一次開機時自動安裝所需的軟體。如已使用未自動安裝所需軟體的 Windows AMI 啟動執行個體，您即必須在執行個體上手動安裝所需軟體。

安裝 Elastic Graphics 所需的軟體 (如有必要)

1. 連線到執行個體。
2. 下載 [Elastic Graphics 安裝程式](#)，並開啟之。安裝管理員會連線到 Elastic Graphics 端點並下載所需軟體的最新版本。

**Note**

如果下載連結無法運作，請嘗試不同的瀏覽器，或複製連結地址並將其貼到新的瀏覽器索引標籤。

3. 重新開機執行個體以確認。

確認執行個體的 Elastic Graphics 功能

執行個體上的 Elastic Graphics 套件包含可檢視加速器狀態的工具，並確認從執行個體到加速器的 OpenGL 命令是否可運作。

如果您執行個體使用未預先安裝 Elastic Graphics 套件的 AMI 來啟動，您可以自行下載並安裝該套件。如需詳細資訊，請參閱 [安裝 Elastic Graphics 所需的軟體](#)。

您可以使用下列其中一種方法來驗證執行個體上的 Elastic Graphics 功能。

**Note**

如果 Elastic Graphics 狀態監控或命令列工具傳回未預期的結果，請參閱 [解決狀況不良問題](#)。



## Elastic Graphics status monitor

您可以使用狀態監控工具來檢視連接的 Elastic Graphics 加速器狀態資訊。根據預設，此工具位於 Windows 執行個體任務列的通知區域中，會顯示圖形加速器的狀態。以下為可能值。

### 狀態良好

Elastic Graphics 加速器已啟用且狀態良好。

### 正在更新

目前正在更新 Elastic Graphics 加速器的狀態。可能需要幾分鐘才會顯示狀態。

### 停止服務

Elastic Graphics 加速器停止服務。請選擇 [Read More](#) (閱讀其他資訊) 以取得錯誤的詳細資訊。

## Elastic Graphics command line tool

您可使用 Elastic Graphics 命令列工具 `egcli.exe` 檢查加速器的狀態。如果加速器發生問題，此工具會將傳回錯誤訊息。

若要啟動該工具，請在您的執行個體中開啟命令提示並執行下列命令：

```
C:\Program Files\Amazon\EC2ElasticGPUs\manager\egcli.exe
```

此工具也支援下列參數：

`--json, -j`

指出是否顯示 JSON 訊息。可能的值為 `true` 和 `false`。預設值為 `true`。

`--imds, -i`

指示是否檢查執行個體中繼資料能否使用加速器。可能的值為 `true` 和 `false`。預設值為 `true`。

下列為範例輸出。OK 狀態指出加速器已啟用且運作正常。

```
EG Infrastructure is available.
Instance ID egpu-f6d94dfa66df4883b284e96db7397ee6
```

```
Instance Type eg1.large
EG Version 1.0.0.885 (Manager) / 1.0.0.95 (OpenGL Library) / 1.0.0.69 (OpenGL
Redirector)
EG Status: Healthy
JSON Message:
{
 "version": "2016-11-30",
 "status": "OK"
}
```

以下是 status 的可能值：

OK

Elastic Graphics 加速器已啟用且狀態良好。

UPDATING

正在更新 Elastic Graphics 驅動程式。

NEEDS\_REBOOT

Elastic Graphics 驅動程式已更新，需要重新開機 Amazon EC2 執行個體。

LOADING\_DRIVER

正在載入 Elastic Graphics 驅動程式。

CONNECTING\_EGPU

Elastic Graphics 驅動程式正在確認與 Elastic Graphics 加速器的連線。

ERROR\_UPDATE\_RETRY

更新 Elastic Graphics 驅動程式時發生錯誤，很快會重試更新。

ERROR\_UPDATE

更新 Elastic Graphics 驅動程式時發生無法復原的錯誤。

ERROR\_LOAD\_DRIVER

載入 Elastic Graphics 驅動程式時發生錯誤。

ERROR\_EGPU\_CONNECTIVITY

無法連線 Elastic Graphics 加速器。

## 檢視 Elastic Graphics 資訊

您可以檢視連接到執行個體的 Elastic Graphics 加速器資訊。

### 檢視 Elastic Graphics 加速器的資訊 (主控台)

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)，然後選取您的執行個體。
3. 在 Details (詳細資訊) 索引標籤中，尋找 Elastic Graphics ID。選擇 ID 以 Elastic Graphics 加速器的下列資訊：
  - 連接狀態
  - 類型
  - 運作狀態

### 檢視 Elastic Graphics 加速器的資訊 (AWS CLI)

您可以使用[描述彈性 gpus 命令](#) AWS CLI：

```
aws ec2 describe-elastic-gpus
```

您可以使用[描述網路介面](#) AWS CLI 指令，並依擁有者 ID 進行篩選，以檢視有關彈性圖形網路介面的資訊。

```
aws ec2 describe-network-interfaces --filters "Name=attachment.instance-owner-id,Values=amazon-elasticgpus"
```

### 檢視彈性圖形加速器的相關資訊 (Windows 專用工具 PowerShell) 的步驟

使用下列命令：

- [Get-EC2ElasticGpu](#)
- [Get-EC2NetworkInterface](#)

使用執行個體中繼資料來檢視 Elastic Graphics 加速器的相關資訊

1. 連線到使用 Elastic Graphics 加速器的 Windows 執行個體。

## 2. 執行以下任意一項：

- 從中 PowerShell，使用下列指令程式：

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/elastic-gpus/associations/egpu-f6d94dfa66df4883b284e96db7397ee6
```

- 從 Web 瀏覽器，將以下 URL 貼至網址欄位：

```
http://169.254.169.254/latest/meta-data/elastic-gpus/associations/egpu-f6d94dfa66df4883b284e96db7397ee6
```

## 提交意見回饋

您可以使用 Elastic Graphics 提交意見回饋，以便團隊精益求精。

使用 Elastic Graphics 狀態監控提交意見回饋

1. 在 Windows 執行個體任務列的通知區域中，開啟 Elastic Graphics 狀態監控。
2. 在左下角選擇 Feedback (意見回饋)。
3. 輸入您的意見回饋，然後選擇 Submit (提交)。

## Elastic Graphics 維護

### Important

Amazon 彈性圖形於 2024 年 1 月 8 日終止使用壽命。對於需要圖形加速的工作負載，我們建議您使用 Amazon EC2 G4ad、G4dn 或 G5 執行個體。

AWS 在下列情況下，可能會判斷彈性圖形加速器處於不健康狀態：

- 需要安全或基礎設施更新
- 需要軟體更新
- 基本主機發生問題

當 AWS 確定「彈性圖形」加速器處於不健康的狀態時，它會排程加速器進行淘汰。AWS 通知您加速器的暫緩處分，並提供您需要採取的補救步驟。

## 主題

- [我將如何收到通知？](#)
- [該怎麼辦？](#)
- [當加速器到達淘汰日期時會發生什麼情況？](#)

### 我將如何收到通知？

當安 AWS 排彈性圖形加速器進行退休時，它會將加速器退休通知傳送給您的 [AWS Health Dashboard](#)。AWS 也會傳送電子郵件至與您 AWS 帳戶相關聯的電子郵件地址。這與您用來登入 AWS Management Console 的電子郵件地址相同。

#### Note

如果您使用的電子郵件帳戶不會經常檢查，請使用 AWS Health Dashboard 來判斷是否有任何 Elastic Graphics 加速器已排程淘汰。您也可以在此 [「帳戶設定」頁面上變更 AWS 帳戶](#) 的聯絡資訊。

### 淘汰通知提供內容如下：

- 加速器所連接之執行個體的 ID
- 有關影響加速器之問題的資訊
- 加速器的淘汰日期
- 您應採取的補救步驟

### 該怎麼辦？

當您收到 Elastic Graphics 加速器排定淘汰的通知時，您必須 [停止並啟動加速器連接到的執行個體](#)，以便將狀況不良的舊加速器替換為狀況良好的新加速器。

我們建議您先關閉在執行個體上執行的圖形應用程式，然後再停止並重新啟動執行個體。

#### Important

如果您沒有在排定的淘汰日期之前停止並啟動執行個體，則與您的執行個體關聯的加速器將自動停止，這可能會導致您的應用程式停止工作。

您必須停止並啟動執行個體。重新啟動執行個體不會將狀況不良的加速器替換為狀況良好的加速器。

當加速器到達淘汰日期時會發生什麼情況？

當健康狀況不佳的「彈性圖形」加速器到達其排定的淘汰日期時，會將其 AWS 永久終止。若要讓狀況不良的加速器在淘汰日期前後得到替換，您必須停止並啟動加速器連接到的執行個體。

如果您沒有在排定的淘汰日期之前停止並啟動執行個體，則與您的執行個體關聯的加速器將自動停止，這可能會導致您的應用程式停止工作。

## 使用 CloudWatch 指標監視彈性圖形

### Important

Amazon 彈性圖形於 2024 年 1 月 8 日終止使用壽命。對於需要圖形加速的工作負載，我們建議您使用 Amazon EC2 G4ad、G4dn 或 G5 執行個體。

您可以使用 Amazon 監控彈性圖形加速器 CloudWatch，Amazon 會收集有關加速器效能的指標。這些統計資料會保存兩週的期間，以便您存取歷史資訊，並更清楚服務的執行方式。

根據預設，彈性圖形加速器會 CloudWatch 在 5 分鐘內將量度資料傳送至。

有關 Amazon 的更多信息 CloudWatch，請參閱 [Amazon CloudWatch 用戶指南](#)。

## Elastic Graphics 指標

AWS/ElasticGPUs 命名空間包含下列 Elastic Graphics 指標。

指標	描述
顯示卡ConnectivityCheck失敗	回報 Elastic Graphics 加速器的連線為作用中或失敗。0 值指出連線為作用中。1 值指出連線失敗。  單位：計數
顯示卡HealthCheck失敗	回報 Elastic Graphics 加速器是否在最後一分鐘通過運作狀態檢查。0 值指出已通過狀態檢查。值一 (1) 指出狀態檢查失敗。

指標	描述
	單位：計數
顯示卡 MemoryUtilization	已使用的 GPU 記憶體。 單位：MiB

## Elastic Graphics 維度

您可以利用下列維度來篩選 Elastic Graphics 加速器的指標資料。

維度	描述
EGPUId	依照 Elastic Graphics 加速器篩選資料。
InstanceId	依照 Elastic Graphics 加速器所連接之執行個體篩選資料。

## 檢視彈性圖形的 CloudWatch 指標

指標會先依服務命名空間分組，再依支援的維度分組。您可以使用下列程序檢視 Elastic Graphics 加速器的指標。

使用 CloudWatch 主控台檢視彈性圖形度量

1. 開啟主 CloudWatch 控制台，網址為 <https://console.aws.amazon.com/cloudwatch/>。
2. 如有必要請變更區域。從導覽列，選取您 Elastic Graphics 加速器所在的區域。如需詳細資訊，請參閱 [區域與端點](#)。
3. 在導覽窗格中，選擇 Metrics (指標)。
4. 針對 All metrics (所有指標)，選取 Elastic Graphics、Elastic Graphics Metrics (Elastic Graphics 指標)。

## 檢視 Elastic Graphics 指標 (AWS CLI)

使用下列 [list-metrics](#) 命令：

```
aws cloudwatch list-metrics --namespace "AWS/ElasticGPUs"
```

## 創建 CloudWatch 警報以監視彈性圖形

您可以建立 CloudWatch 警報，在警報狀態變更時傳送 Amazon SNS 訊息。警報會監看您指定期間內的單一指標，然後根據若干這樣的時段內相對於指定閾值的指標值，向 Amazon SNS 主題傳送通知。

例如，您可以建立警報來監控 Elastic Graphics 加速器的運作狀態，當圖形加速器在接連三個 5 分鐘的期間內未通過運作狀態檢查時傳送通知。

### 建立 Elastic Graphics 加速器運作狀態警報

1. 開啟主 CloudWatch 控制台，網址為 <https://console.aws.amazon.com/cloudwatch/>。
2. 在導覽窗格中，選擇 Alarms (警報)、Create Alarm (建立警報)。
3. 選擇 Select metric (選取指標)、Elastic Graphics、Elastic Graphics Metrics (Elastic Graphics 指標)。
4. 選取「GPU HealthCheck 失敗」量度，然後選擇「選取量度」。
5. 請如下所示設定警報：
  - a. 針對 Alarm details (警報詳細資訊)，輸入警報的名稱和描述。針對 Whenever (每當)，選擇  $\geq$  並輸入 1。
  - b. 針對 Actions (動作)，選取現有的通知清單，或選擇 New list (新增清單)。
  - c. 選擇 Create Alarm (建立警報)。

## 疑難排解

### Important

Amazon 彈性圖形於 2024 年 1 月 8 日終止使用壽命。對於需要圖形加速的工作負載，我們建議您使用 Amazon EC2 G4ad、G4dn 或 G5 執行個體。

下列是常見的錯誤和故障診斷步驟。

### 內容

- [調查應用程式效能問題](#)
- [OpenGL 轉譯效能問題](#)
- [遠端存取效能問題](#)
- [解決狀況不良問題](#)



- [檢查執行個體組態](#)
- [停止並啟動執行個體](#)
- [確認已安裝的元件](#)
- [檢查 Elastic Graphics 日誌](#)
- [為什麼我看到多個 ENI ?](#)

## 調查應用程式效能問題

Elastic Graphics 使用執行個體網路將 OpenGL 命令傳送到遠端連接的顯示卡。此外，執行使用 Elastic Graphics 加速器之 OpenGL 應用程式的桌面，通常會使用遠端存取技術存取。區分 OpenGL 轉譯與桌面遠端存取技術相關的效能問題非常重要。

## OpenGL 轉譯效能問題

OpenGL 轉譯效能取決於遠端執行個體上產生的 OpenGL 命令和影格數量。

轉譯效能可能因下列因素而異：

- Elastic Graphics 加速器效能
- 網路效能
- CPU 效能
- 轉譯模型與案例複雜性
- OpenGL 應用程式行為

評估效能的簡易方法是顯示遠端執行個體的轉譯影格數。Elastic Graphics 加速器在遠端執行個體上最多顯示 25 個 FPS，在降低網路用量的同時實現最佳感知品質。

## 顯示產生的影格數

1. 在文字編輯器中開啟下列檔案。如果檔案不存在，請先建立檔案。

```
C:\Program Files\Amazon\EC2ElasticGPUs\conf\eg.conf
```

2. 識別 [Application] 區段，或新增該區段 (如不存在)，然後新增下列組態參數：

```
[Application]
show_fps=1
```

### 3. 重新啟動應用程式並再次檢查 FPS。

如果 FPS 在更新轉譯場景時達到 15-25 個 FPS，則 Elastic Graphics 加速器執行達到峰值。您碰到的任何其他效能問題都可能與執行個體桌面的遠端存取有關。如果是這種情況，請參閱遠端存取效能問題一節。

如果 FPS 數目低於 15，您可以嘗試下列操作：

- 選取更強大的圖形加速器類型來提高 Elastic Graphics 加速器效能。
- 使用以下秘訣來提升整體網路效能：
  - 檢查往來 Elastic Graphics 加速器端點的傳入及傳出頻寬數量。您可以使用下列 PowerShell 指令擷取「彈性圖形」加速器端點：

```
PS C:\> (Invoke-WebRequest http://169.254.169.254/latest/meta-data/elastic-gpus/associations/[ELASTICGPU_ID]).content
```

- 從執行個體到 Elastic Graphics 加速器端點的網路流量，與 OpenGL 應用程式產生的命令數量有關。
- 從 Elastic Graphics 加速器端點到執行個體的網路流量，與圖形加速器產生的影格數有關。
- 如果網路使用率達到執行個體最大網路輸送量，請嘗試使用具有更高網路輸送量限額的執行個體。
- 改善 CPU 效能：
  - 除了 Elastic Graphics 加速器所需的功能之外，應用程式可能還需要大量的 CPU 資源。如果 Windows 任務管理員回報 CPU 資源使用率過高，請嘗試使用 CPU 能力更強的執行個體。

#### 遠端存取效能問題

連接 Elastic Graphics 加速器的執行個體可以使用不同的遠端存取技術來存取。效能和品質可能因下列因素而異：

- 遠端存取技術
- 執行個體效能
- 用戶端效能
- 網路延遲以及用戶端與執行個體之間的頻寬

遠端存取通訊協定的可能選擇包括：

- Microsoft 遠端桌面連線
- NICE DCV
- VNC

如需最佳化的詳細資訊，請參閱特定通訊協定。

### 解決狀況不良問題

如果 Elastic Graphics 加速器的狀態不良，請使用下列故障診斷步驟來解決問題。

### 檢查執行個體組態

如果 Elastic Graphics 命令列工具 `egcli.exe` 傳回類似以下的輸出，請確保[安全群組設定正確](#)，並且您啟動了已啟用執行個體中繼資料服務的執行個體。

```
EG Version 1.0.7.4240 (Manager) / N/A (OpenGL Library) / N/A (OpenGL Redirector)
EG Status: Out Of Service
Something prevented the EG Infrastructure to work properly.
```

### 停止並啟動執行個體

如果您的 Elastic Graphics 加速器狀態不良，停止執行個體並再次啟動是最簡單的選項。如需詳細資訊，請參閱[手動停止和啟動執行個體](#)。

#### Warning

停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區的資料，請將資料備份至持久性儲存。

### 確認已安裝的元件

開啟 Windows 控制台並確認已安裝以下元件：

- Amazon Elastic Graphics Manager
- Amazon Elastic Graphics OpenGL 程式庫
- Amazon EC2 Elastic GPUs OpenGL 重定向器

如果缺少這些項目的任何一個，您必須手動安裝它們。如需詳細資訊，請參閱 [安裝 Elastic Graphics 所需的軟體](#)。

## 檢查 Elastic Graphics 日誌

開啟 Windows 事件檢視器，展開應用程式及服務記錄檔區段，並在下列事件日誌中尋找錯誤：

- EC2ElasticGPUs
- EC2ElasticGPUs GUI

## 為什麼我看到多個 ENI？

使用彈性圖形加速器呼叫 EC2 執行個體時，會 [StartInstances](#) 在執行個體上建立新的彈性網路介面 (ENI)，以允許將 OpenGL 命令傳送至遠端連接的圖形卡。

如果您在相同 EC2 執行個體的短時間內 (幾秒鐘或更短) 內呼叫 [StartInstances](#) 多次，則每次呼叫都會建立新的網路介面。但是：

- Elastic Graphics 加速器只會使用一個網路介面。
- 額外的網路介面不會產生任何費用，並將在 24 小時內自動釋放。

## 在 Windows 執行個體上安裝 WSL

Windows Subsystem for Linux (WSL) 是免費下載軟體，可以安裝在 Windows 執行個體上。透過安裝 WSL，您可以直接在 Windows 執行個體上執行原生 Linux 命令列工具，並使用 Linux 工具與傳統 Windows 桌面來編寫指令碼。您可以在單一 Windows 執行個體上輕鬆地在 Linux 和 Windows 之間進行切換，這在開發環境中可能很有用。

如需 WSL 的詳細資訊，請參閱 Microsoft Build 網站上的 [Windows Subsystem for Linux 文件](#)。

## 限制

- WSL 有兩個版本：WSL 1 和 WSL 2。
  - 對於 .metal EC2 執行個體，可以安裝 WSL 1 或 WSL 2。
  - 對於虛擬化 EC2 執行個體，必須安裝 WSL 1。
- 對於 Windows Server 作業系統，WSL 只能安裝在執行下列項目的執行個體上：
  - Windows Server 2019

- Windows Server 2022

## 安裝 WSL

以下指示會在執行 Windows Server 2022 的 EC2 執行個體上安裝 WSL。如需在執行 Windows Server 2019 的 EC2 執行個體上安裝 WSL 的說明，請參閱 Microsoft 網站上的[在舊版 Windows Server 上安裝 WSL](#)。完成這些指示後，您可以使用下列指示中的步驟 3 來設定 WSL 以使用 WSL 1。

### 安裝水務證 1

1. 若要安裝 WSL，請在 EC2 執行個體上執行下列標準安裝命令，但請務必透過包含 `--enable-wsl1` 來啟用 WSL 1。依預設，會安裝 WSL 2。如果使用虛擬化執行個體類型啟動執行個體，則您必須完成此程序中的步驟 3，才能將版本設定為 WSL 1。

```
wsl --install --enable-wsl1 --no-launch
```

2. 重新啟動 EC2 執行個體。

```
shutdown -r -t 20
```

3. 若要設定 WSL 以使用 WSL 1，請在執行個體上執行下列命令。如需有關設定 WSL 版本的詳細資訊，請參閱 Microsoft Build 網站上[舊版 WSL 的手動安裝步驟](#)。

```
wsl --set-default-version 1
```

4. 安裝預設發行版。

```
wsl --install
```

### 安裝水務證 2

- 若要安裝 WSL，請在 EC2 執行個體上執行下列標準安裝命令。依預設，會安裝 WSL 2。如果要在 `.metal` 執行個體上安裝 WSL，則這是唯一要執行的步驟。

```
wsl --install
```

如需詳細資訊，請參閱 Microsoft Build 網站上的[使用 WSL 在 Windows 上安裝 Linux](#)。

# 將 Amazon EC2 Windows 執行個體升級至較新版本的 Windows Server

有兩種方法可以升級執行個體上執行的舊版 Windows Server：就地升級和移轉 (也稱為 side-by-side 升級)。就地升級會將作業系統檔案升級，通常保留您的個人設定及檔案不變。遷移會涉及擷取設定、組態和資料，並將這些移植到全新 Amazon EC2 執行個體上的較新作業系統。

Microsoft 傳統上都建議遷移到較新版本的 Windows Server，而非升級。遷移可能會減少升級錯誤或問題，但相較於就地升級可能需要更長的時間，因為需要佈建新的執行個體、計劃及移植應用程式，以及調整新執行個體上的組態設定。就地升級的速度較快，但軟體不相容可能會產生錯誤。

## 目錄

- [在 Windows 執行個體上執行就地升級](#)
- [在您的 Windows 執行個體上執行自動升級](#)
- [將 Windows 執行個體移轉至目前一代的執行個體類型](#)
- [Microsoft SQL 伺服器資料庫的視窗到 Linux 重新平台輔助程式](#)
- [對 Windows 執行個體進行升級的疑難排解](#)

## 在 Windows 執行個體上執行就地升級

在您執行就地升級前，您必須判斷執行個體執行的網路驅動程式為何。PV 網路驅動程式可讓您使用遠端桌面存取您的執行個體。執行個體使用 AWS PV、Intel 網路介面卡或增強型網路驅動程式。如需詳細資訊，請參閱 [Windows 執行個體的全虛擬驅動程式](#)。

## 在您開始就地升級之前

在您開始您的就地升級之前，請完成下列任務並記下以下重要詳細資訊。

- 閱讀 Microsoft 文件，了解升級需求、已知問題和限制。同時也請檢閱升級的正式說明。
  - [Upgrade Options for Windows Server 2012](#)
  - [Upgrade Options for Windows Server 2012 R2](#)
  - [Upgrade and conversion options for Windows Server 2016](#)
  - [Upgrade and conversion options for Windows Server 2019](#)
  - [Windows Server 2022 的升級或轉換選項](#)
  - [Windows Server Upgrade Center](#)

- 我們建議在至少配備 2 個 vCPU 或 4GB RAM 的執行個體上執行作業系統升級。必要時，您可以將執行個體變更為相同類型但更大的執行個體 (例如，將 t2.small 變更為 t2.large)、執行升級，然後再調整回原始大小。如果您需要保留執行個體大小，您可以使用[執行個體主控台螢幕擷取畫面](#)來監控進度。如需詳細資訊，請參閱 [變更執行個體類型](#)。
- 確認您 Windows 執行個體的根磁碟區具有足夠的可用磁碟空間。Windows 安裝程序可能不會警告您的磁碟空間不足。如需升級至特定作業系統所需磁碟空間的資訊，請參閱 Microsoft 文件。若磁碟區沒有足夠的空間，您可以擴展它。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的 Amazon EBS 彈性磁碟區](#)。
- 決定您的升級途徑。您必須將作業系統升級至相同的架構。例如，您必須將 32 位元系統升級至 32 位元系統。Windows Server 2008 R2 及更新版本為僅限 64 位元。
- 停用防毒軟體、反間諜軟體及防火牆。這些軟體類型可能會和升級程序衝突。請在升級完成之後再次啟用防毒軟體、反間諜軟體及防火牆。
- 如 [將 Windows 執行個體移轉至目前一代的執行個體類型](#) 所述更新至最新的驅動程式。
- 升級協助程式服務僅支援執行 Citrix PV 驅動程式的執行個體。若執行個體執行的是 Red Hat 驅動程式，您必須先手動[升級那些驅動程式](#)。

## 使用 AWS PV、Intel 網路介面卡或增強型網路驅動程式就地升級執行個體

透過下列程序，使用 AWS PV、Intel 網路介面卡或增強型網路驅動程式升級 Windows Server 執行個體。

### 執行就地升級

1. 為備份或測試用途，建立您計劃升級之系統的 AMI。您接著便能在複本上執行升級，模擬測試環境。若升級完成，您便可以將流量切換至此執行個體，而無須經歷太多停機時間。若升級失敗，您可以回復至備份。如需詳細資訊，請參閱 [創建一個 Amazon EBS-backed AMI](#)。
2. 確認您的 Windows Server 執行個體使用最新的網路驅動程式。
  - a. 若要更新您的 AWS PV 驅動程式，請參閱[升級 Windows 執行個體上的 PV 驅動程式](#)。
  - b. 若要更新 ENA 驅動程式，請參閱[安裝彈性網路介面卡 \(ENA\) 驅動程式](#)。
  - c. 若要更新您的 Intel 驅動程式，請參閱 [使用 EC2 執行個體上的英特爾 82599 VF 介面啟用增強型聯網](#)
3. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
4. 在導覽窗格中，選擇 Instances (執行個體)。尋找執行個體。記下執行個體的執行個體 ID 和可用區域。您在此程序的後續步驟中將需要此資訊。

5. 若您要從 Windows Server 2012 或 2012 R2 升級至 Windows Server 2016、2019 或 2022，請在您的執行個體上執行下列作業再繼續：
  - a. 解除安裝 EC2Config 服務。如需詳細資訊，請參閱 [停止、重新啟動、刪除或解除安裝 EC2Config](#)。
  - b. 安裝 EC2Launch v1 或 EC2Launch v2 代理程式。如需詳細資訊，請參閱 [使用 EC2Launch 設定 Windows 執行個體](#) 和 [使用 EC2Launch v2 設定 Windows 執行個體](#)。
  - c. 安裝 AWS Systems Manager SSM 代理程式。如需詳細資訊，請參閱《AWS Systems Manager 使用者指南》中的 [使用 SSM Agent](#)。
6. 從 Windows Server 安裝媒體快照建立新的磁碟區。
  - a. 在左側導覽窗格中，於 Elastic Block Store 下方，選擇 Snapshots (快照)。
  - b. 從篩選條件列中，選擇公有快照。
  - c. 在搜尋列中，指定下列篩選條件：
    - 選擇所有者別名，然後選擇 =，最後選擇 amazon。
    - 選擇說明，然後開始輸入 **Windows**。選取與您要升級到的系統架構及語言偏好設定相符的 Windows 篩選條件。例如，選擇 Windows 2019 英文版安裝媒體以升級至 Windows Server 2019。
  - d. 選取與您要升級到的系統架構和語言偏好設定相符的快照旁邊的核取方塊，然後選擇動作、從快照建立磁碟區。
  - e. 在建立磁碟區頁面中，選擇符合 Windows 執行個體的可用區域，然後選擇建立磁碟區。
7. 在頁面頂端的成功建立的磁碟區 `vol-1234567890example` 橫幅中，選擇您剛才建立的磁碟區 ID。
8. 選擇 Actions (動作)、Attach Volume (連接磁碟區)。
9. 在附接磁碟區頁面中，對於執行個體，選取 Windows 執行個體的執行個體 ID，然後選擇附接磁碟區。
10. 遵循使 [Amazon EBS 磁碟區可供使用中的步驟](#)，讓新磁碟區可供使用。

 Important

請勿初始化磁碟，因為這樣做會刪除現有資料。

11. 在 Windows 中 PowerShell，切換到新的磁碟區磁碟機。透過開啟連接到執行個體的安裝媒體磁碟區，開始升級。



- a. 若您要升級至 Windows Server 2016 或更新版本，請執行下列項目：

```
.\setup.exe /auto upgrade /dynamicupdate disable
```

**Note**

在將 /dynamicupdate 選項設定為停用的情況下執行 setup.exe 會阻止 Windows 在 Windows Server 升級程序期間中安裝更新，因為在升級期間中安裝更新可能會導致失敗。升級完成後，您可以使用 Windows Update 安裝更新。

- 若您要升級至較早版本的 Windows Server，請執行下列項目：

```
Sources\setup.exe
```

- b. 針對 Select the operating system you want to install (選取您要安裝的作業系統)，選取您 Windows Server 執行個體的完整安裝 SKU，然後選取 Next (下一步)。
- c. 針對 Which type of installation do you want? (您希望的安裝類型為何?)，選擇 Upgrade (升級)。
- d. 完成協助程式。

Windows Server 安裝程式會複製並處理檔案。在幾分鐘之後，您的遠端桌面工作階段便會關閉。升級所需要花費的時間取決於在您 Windows Server 執行個體上執行之應用程式和伺服器角色的數目。升級程序最少可能需要花費 40 分鐘至數小時。執行個體的兩項狀態檢查中的其中一項會在升級程序過程中失敗。當升級完成後，兩項狀態檢查便都會通過。您可以檢查主控台輸出的系統日誌，或使用 Amazon CloudWatch 指標處理磁碟和 CPU 活動，以判斷升級是否正在進行中。

**Note**

若升級至 Windows Server 2019，在升級完成之後，若需要的話，您可以手動變更桌面背景，移除先前作業系統的名稱。

若執行個體在數個小時之後仍無法通過兩項狀態檢查，請參閱[對 Windows 執行個體進行升級的疑難排解](#)。

## 升級後任務

1. 登入執行個體，初始化 .NET Framework 的升級，然後在提示後重新開機系統。
2. 如果您尚未在先前的步驟中這樣做，請安裝 EC2Launch v1 或 EC2Launch v2 代理程式。如需詳細資訊，請參閱 [使用 EC2Launch 設定 Windows 執行個體](#) 和 [使用 EC2Launch v2 設定 Windows 執行個體](#)。
3. 如果您升級到 Windows 伺服器 2012 R2，我們建議您將光伏驅動程式升級為 AWS 光伏驅動程式。若您升級以 Nitro 為基礎的執行個體，我們建議您安裝或升級 NVME 和 ENA 驅動程式。如需詳細資訊，請參閱 [Windows Server 2012 R2](#)、[使用安裝或升級 AWS NVMe 驅動程式 PowerShell](#) 或 [啟用 Windows 上的增強型網路](#)。
4. 重新啟用防毒軟體、反間諜軟體及防火牆。

## 在您的 Windows 執行個體上執行自動升級

您可以 AWS 使用自動 AWS Systems Manager 化手冊在上執行 Windows 和 SQL Server 執行個體的自動升級。

### 目錄

- [相關服務](#)
- [執行選項](#)
- [升級 Windows Server](#)
- [升級 SQL Server](#)

## 相關服務

自動升級程序會使用下列 AWS 服務：

- AWS Systems Manager。AWS Systems Manager 是一個功能強大的統一介面，可集中管理您的 AWS 資源。如需詳細資訊，請參閱 [AWS Systems Manager 使用者指南](#)。
- AWS Systems Manager 代理程式 (SSM 代理程式) 是可在 Amazon EC2 執行個體、現場部署伺服器或虛擬機器 (VM) 上安裝和設定的 Amazon 軟體。SSM Agent 讓 Systems Manager 能夠更新、管理和設定這些資源。此代理程式會處理 AWS Cloud 中來自 Systems Manager 服務的申請，然後依照要求中的指定執行。如需詳細資訊，請參閱《AWS Systems Manager 使用者指南》中的 [使用 SSM Agent](#)。

- AWS Systems Manager SSM 手冊。SSM Runbook 定義 Systems Manager 在受管執行個體上執行的動作。SSM 執行手冊使用 JavaScript 物件標記法 (JSON) 或 YAML，而且包含您指定的步驟和參數。本主題使用兩份 Systems Manager SSM Runbook 來進行自動化。如需詳細資訊，請參閱 AWS Systems Manager 使用者指南中的 [AWS Systems Manager Automation runbook 參考](#)。

## 執行選項

當您在 Systems Manager 主控台選取 Automation (自動化) 時，請選取 Execute (執行)。在您選取自動化文件後，系統會提示您選取自動化執行選項。請從以下選項中選擇。在本主題稍後所提供途徑的步驟中，我們使用 Simple execution (簡易執行) 選項。

### 簡易執行

如果您想要更新單一執行個體，但不想經歷每個自動化步驟來稽核結果，請選擇此選項。稍後的升級步驟中會進一步詳述此選項。

### Rate control (比率控制)

如果您要對多個執行個體套用升級，請選擇此選項。請定義以下設定。

- Parameter (參數)

此設定 (也在 Multi-Account and Region (多帳戶和區域) 設定中) 定義自動化如何展開。

- 目標

選取您要套用自動化的目標。此設定也在 Multi-Account and Region (多帳戶和區域) 設定中。

- Parameter Values (參數值)

使用自動化文件參數中定義的值。

- Resource Group (資源群組)

在中 AWS，資源是您可以使用的實體。範例包括 Amazon EC2 執行個體、AWS CloudFormation 堆疊或 Amazon S3 儲存貯體。如果您使用多個資源，則將它們作為一個組進行管理可能會很有用，而不是從一個服務移動到另一個 AWS 服務的每個任務。在某些情況下，您可能需要管理大量的相關資源，例如組成應用程式層的 EC2 執行個體。在此情況下，您在這些資源上可能需要一次執行大量動作。

- Tags (標籤)

標籤可協助您以不同的方式對 AWS 資源進行分類，例如，依目的、擁有者或環境。此分類在您擁有許多相同類型的資源時很有用。您可以使用指派的標籤快速識別特定資源。

- Rate Control (比率控制)

Rate Control (比率控制) 也在 Multi-Account and Region (多帳戶和區域) 設定中。當您設定比率控制參數時，請以目標計數或機群的百分比，定義機群中有多少要套用自動化。

### Multi-Account and Region (多帳戶和區域)

除了 Rate Control (比率控制) 下指定的參數 (也用在 Multi-Account and Region (多帳戶和區域) 設定中)，另外還有兩個設定：

- Accounts and organizational units (OUs) (帳戶和組織單位 (OU))

指定您要執行自動化的多個帳戶。

- AWS 區域

指定您要 AWS 區域 在其中執行自動化操作的多個位置。

### 手動執行

此選項類似於 Simple execution (簡易執行)，但可讓您逐步完成每個自動化步驟並稽核結果。

### 升級 Windows Server

[AWSEC2-CloneInstanceAndUpgradeWindows](#) Runbook 從您帳戶中的 Windows Server 執行個體建立 Amazon Machine Image (AMI)，並將此 AMI 升級到您選擇的支援版本。此多步驟程序需要最多兩個小時才會完成。

自動化升級程序中包含兩個 AMI：

- 目前執行中的執行個體。第一個 AMI 是目前執行中的執行個體 (尚未升級)。此 AMI 用來啟動另一個執行個體，以執行就地升級。當程序完成時，將從您的帳戶中刪除此 AMI，除非您明確要求保留原始執行個體。此設定由參數 `KeepPreUpgradeImageBackup` 處理 (預設值為 `false`，表示預設會刪除 AMI)。
- 已升級的 AMI。此 AMI 是自動化程序的成果。

最終成果是一個 AMI，即已升級的 AMI 執行個體。

當升級完成時，您可以在 Amazon VPC 中啟動新的 AMI，以測試應用程式功能。測試後，在執行另一次升級前，請先安排應用程式停機時間，再完全切換至已升級的執行個體。

## 必要條件

若要使用自動 AWS Systems Manager 化文件自動化您的 Windows 伺服器升級，您必須執行下列工作：

- 搭配指定的 IAM 政策建立 IAM 角色，以允許 Systems Manager 在您的 Amazon EC2 執行個體上執行自動化任務，並驗證您是否符合使用 Systems Manager 的先決條件。如需詳細資訊，請參閱《AWS Identity and Access Management 使用指南》中的[建立角色以將權限委派給 AWS 服務](#)。
- [選取您希望如何執行自動化的選項](#)。執行選項包括 Simple execution (簡易執行)、Rate control (比率控制)、Multi-account and Region (多帳戶和區域) 及 Manual execution (手動執行)。如需關於這些選項的詳細資訊，請參閱 [執行選項](#)。
- 確認 SSM Agent 安裝於您的執行個體上。如需詳細資訊，請參閱[在適用於 Windows Server 的 Amazon EC2 執行個體上安裝與設定 SSM Agent](#)。
- 視窗 PowerShell 3.0 或更新版本必須安裝在您的執行個體上。
- 對於加入 Microsoft Active Directory 網域的執行個體，建議您指定沒有連線到您的網域控制站的 SubnetId，以協助避免主機名稱衝突。
- 執行個體子網路必須具有連線至網際網路的輸出連線，以提供 Amazon S3 AWS 服務 等存取權，以及從 Microsoft 下載修補程式的存取權。如果子網路是公有子網路且執行個體具有公有 IP 地址，或子網路是私有子網路且具有將網際網路流量傳送至公有 NAT 裝置的路由，則符合此需求。
- 此自動化適用於執行 Windows Server 2008 R2、Windows Server 2012 R2、Windows Server 2016 和 Windows Server 2019 的執行個體。
- 確認執行個體在開機磁碟中有 20 GB 的可用磁碟空間。
- 如果執行個體未使用由提供的 Windows 授權 AWS，請指定包含 Windows 伺服器 2012 R2 安裝媒體的 Amazon EBS 快照識別碼。若要執行此作業：
  1. 確認 Amazon EC2 執行個體執行的是 Windows Server 2012 R2 或更新版本。
  2. 在執行個體執行的相同可用區域中建立 6 GB 的 Amazon EBS 磁碟區。將磁碟區連結到執行個體。例如，將其掛載為 D 磁碟機。
  3. 在 ISO 按一下滑鼠右鍵，並將其掛載至執行個體，例如做為 E 磁碟機。
  4. 從磁碟機 E:\ 將 ISO 的內容複製到磁碟機 D:\
  5. 從上述步驟 2 建立的 6 GB 磁碟區建立 Amazon EBS 快照。

## Windows Server 升級限制

此自動化不支援升級 Windows 網域控制站、叢集或 Windows 桌面作業系統。此自動化也不支援安裝了下列角色的適用於 Windows Server 的 Amazon EC2 執行個體：

- 遠端桌面工作階段主機 (RDSH)
- 遠端桌面連線代理人 (RDCB)
- 遠端桌面虛擬化主機 (RDVH)
- 遠端桌面 Web 存取 (RDWA)

## 執行 Windows Server 自動升級的步驟

請按照以下步驟使用 [AWSEC2-Windows 自動化執行手冊升級您的 CloneInstance AndUpgrade Windows](#) 伺服器執行個體。

1. 從 AWS Management Console (AWS 管理主控台) 開啟 Systems Manager。
2. 從左側導覽窗格中，在 Change Management (變更管理) 下選擇 Automation (自動化)。
3. 選擇 Execute automation (執行自動化)。
4. 搜尋稱為 AWSEC2-CloneInstanceAndUpgradeWindows 的自動化文件。
5. 當文件名稱出現時，請選取它。選取此文件時，將會出現文件詳細資訊。
6. 選擇 Execute automation (執行自動化) 以輸入此文件的參數。在頁面頂端，保持選取 Simple execution (簡易執行)。
7. 根據下列指導，輸入所要求的參數。

- InstanceID

類型：字串

(必要) 執行 Windows Server 2008 R2、2012 R2、2016 或 2019 的執行個體，並且已安裝 SSM Agent。

- InstanceProfile.

類型：字串

(必要) IAM 執行個體描述檔。這是用來針對 Amazon EC2 執行個體和 AWS AMI 執行 Systems Manager 自動化的 IAM 角色。如需詳細資訊，請參閱 AWS Systems Manager 使用者指南中的 [建立 Systems Manager 的 IAM 執行個體描述檔](#)。

- TargetWindowsVersion

類型：字串

(必要) 選取目標 Windows 版本。

- SubnetId

類型：字串

(必要) 這是升級程序的子網，也是您的 EC2 執行個體所在地方。確認子網路具有連線至 AWS 服務 (包括 Amazon S3) 以及 Microsoft 的輸出連線 (以便下載修補程式)。

- KeepPreUpgradedBackUp

類型：字串

(選用) 如果此參數設為 true，自動化會保留從執行個體建立的映像。預設設定為 false。

- RebootInstanceBeforeTakingImage

類型：字串

(選用) 預設為 false (不重新啟動)。如果此參數設為 true，Systems Manager 在為升級建立 AMI 之前會重新啟動執行個體。

8. 輸入參數後，選擇 Execute (執行)。自動化開始時，您可以監控執行進度。
9. 當自動化完成時，您會看到 AMI ID。您可以啟動 AMI，以驗證 Windows 作業系統已升級。

 Note

自動化沒有必要執行所有步驟。這些步驟取決於自動化和執行個體的行為。Systems Manager 可能略過有些非必要的步驟。

此外，某些步驟可能會逾時。Systems Manager 會嘗試升級和安裝所有最新的修補程式。不過，根據特定步驟可定義的逾時設定，修補程式有時會逾時。發生此情況時，Systems Manager 自動化會繼續下一個步驟，以確保內部作業系統升級到目標 Windows Server 版本。

10. 自動化完成之後，您可以使用 AMI ID 啟動 Amazon EC2 執行個體，以檢閱您的升級。如需如何從 AWS AMI 建立 Amazon EC2 執行個體的詳細資訊，請參閱[如何從自訂 AMI 啟動 EC2 執行個體？](#)

## 升級 SQL Server

[AWSEC2-CloneInstanceAndUpgradeSQLServer](#) 指令碼從您帳戶中執行 SQL Server 的 Amazon EC2 執行個體建立 AMI，然後將 AMI 升級到更新版本的 SQL Server。此多步驟程序需要最多兩個小時才會完成。

自動化會從執行個體建立 AMI，接著在您提供的子網中啟動新的 AMI。自動化接著會執行 SQL Server 的就地升級。升級完成之後，自動化在終止已升級的執行個體之前會建立新的 AMI。

自動化升級程序中包含兩個 AMI：

- 目前執行中的執行個體。第一個 AMI 是目前執行中的執行個體 (尚未升級)。此 AMI 用來啟動另一個執行個體，以執行就地升級。當程序完成時，將從您的帳戶中刪除此 AMI，除非您明確要求保留原始執行個體。此設定由參數 `KeepPreUpgradeImageBackUp` 處理 (預設值為 `false`，表示預設會刪除 AMI)。
- 已升級的 AMI。此 AMI 是自動化程序的成果。

最終成果是一個 AMI，即已升級的 AMI 執行個體。

當升級完成時，您可以在 Amazon VPC 中啟動新的 AMI，以測試應用程式功能。測試後，在執行另一次升級前，請先安排應用程式停機時間，再完全切換至已升級的執行個體。

## 必要條件

若要使用自動 AWS Systems Manager 化文件自動化您的 SQL Server 升級，您必須執行下列工作：

- 搭配指定的 IAM 政策建立 IAM 角色，以允許 Systems Manager 在您的 Amazon EC2 執行個體上執行自動化任務，並驗證您是否符合使用 Systems Manager 的先決條件。如需詳細資訊，請參閱《AWS Identity and Access Management 使用者指南》中的[建立角色以委派許可給 AWS 服務](#)。
- [選取您希望如何執行自動化的選項](#)。執行選項包括 Simple execution (簡易執行)、Rate control (比率控制)、Multi-account and Region (多帳戶和區域) 及 Manual execution (手動執行)。如需關於這些選項的詳細資訊，請參閱 [執行選項](#)。
- Amazon EC2 執行個體必須使用 Windows Server 2008 R2 (或更新版本) 和 SQL Server 2008 或更新版本。
- 確認 SSM Agent 安裝於您的執行個體上。如需詳細資訊，請參閱[在適用於 Windows Server 的 Amazon EC2 執行個體上使用 SSM Agent](#)。
- 確認執行個體具有足夠的可用磁碟空間：
  - 若您要從 Windows Server 2008 R2 升級至 2012 R2，或從 Windows Server 2012 R2 升級至更新版本的作業系統，請確認您的執行個體開機磁碟中有 20 GB 的可用磁碟空間。
  - 若您要從 Windows Server 2008 R2 升級至 2016 或更新版本，請確認執行個體開機磁碟中有 40 GB 的可用磁碟空間。
- 針對使用自有授權 (BYOL) 的 SQL Server 版本執行個體，適用下列額外的事前準備：



- 提供 Amazon EBS 快照 ID，其中包含目標 SQL Server 安裝媒體。若要執行此作業：
  1. 確認 Amazon EC2 執行個體執行的是 Windows Server 2008 R2 或更新版本。
  2. 在執行個體執行的相同可用區域中建立 6 GB 的 Amazon EBS 磁碟區。將磁碟區連結到執行個體。例如，將其掛載為 D 磁碟機。
  3. 在 ISO 按一下滑鼠右鍵，並將其掛載至執行個體，例如做為 E 磁碟機。
  4. 從磁碟機 E:\ 將 ISO 的內容複製到磁碟機 D:\
  5. 建立步驟 2 中所建立 6 GB 磁碟區的 Amazon EBS 快照。

## SQL Server 自動升級限制

使用 [AWSEC2-CloneInstance AndUpgrade SQLServer](#) 執行手冊執行自動升級時，會套用下列限制：

- 升級僅能在使用 Windows 身分驗證的 SQL Server 上執行。
- 確認執行個體上沒有待定的安全性修補程式更新。開啟 Control Panel (控制面板)，接著選擇 Check for updates (檢查更新)。
- 不支援在 HA 和鏡像模式中的 SQL Server 部署。

## 執行 SQL Server 自動升級的步驟

請依照下列步驟，使用 [AWSEC2-SQL 伺服器自動化執行手冊來升級您的 CloneInstance AndUpgrade SQL 伺服器](#)。

1. 下載 SQL Server 2016 .iso 檔案並掛載到來源伺服器 (如果您尚未這麼做)。
2. 掛載 .iso 檔案之後，將所有元件檔案複製到您選擇的任何磁碟區。
3. 建立磁碟區的 Amazon EBS 快照，並將快照 ID 複製到剪貼簿，供稍後使用。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的建立 Amazon EBS 快照](#)。
4. 將執行個體描述檔連接至 Amazon EC2 來源執行個體。這可讓 Systems Manager 與 EC2 執行個體通訊，並在新增至 AWS Systems Manager 服務後在其上執行命令。在此範例中，我們將角色命名為 SSM-EC2-Profile-Role，並將 AmazonSSMManagedInstanceCore 政策連接到角色。請參閱《AWS Systems Manager 使用者指南》中的 [建立 Systems Manager 的 IAM 執行個體設定檔](#)。
5. 在 AWS Systems Manager 主控台的左側導覽窗格中，選擇 [受管執行個體]。確認 EC2 執行個體在受管執行個體的清單中。如果幾分鐘後沒有看見您的執行個體，請參閱 AWS Systems Manager 使用者指南中的 [我的執行個體在哪裡？](#)。
6. 在左側導覽窗格中，在 Change Management (變更管理) 下選擇 Automation (自動化)。

7. 選擇 Execute automation (執行自動化)。
8. 搜尋稱為 AWSEC2-CloneInstanceAndUpgradeSQLServer 的自動化文件。
9. 選擇 AWSEC2-CloneInstanceAndUpgradeSQLServer SSM 文件，然後選擇 Next (下一步)。
10. 確保選取 Simple execution (簡易執行) 選項。
11. 根據下列指導，輸入所要求的參數。

- InstanceId

類型：字串

(必要) 執行 SQL Server 2008 R2 (或更新版本) 的執行個體。

- IamInstanceProfile

類型：字串

(必要) IAM 執行個體描述檔。

- SQLServerSnapshotId

類型：字串

(必要) 目標 SQL Server 安裝媒體的快照 ID。包含 SQL Server 授權的執行個體不需要此參數。

- SubnetId

類型：字串

(必要) 這是升級程序的子網，也是您的 EC2 執行個體所在地方。確認子網路具有連線至 AWS 服務 (包括 Amazon S3) 以及 Microsoft 的輸出連線 (以便下載修補程式)。

- KeepPreUpgradedBackup

類型：字串

(選用) 如果此參數設為 true，自動化會保留從執行個體建立的映像。預設設定為 false。

- RebootInstanceBeforeTakingImage

類型：字串

(選用) 預設為 false (不重新啟動)。如果此參數設為 true，Systems Manager 在為升級建立 AMI 之前會重新啟動執行個體。

- TargetSQLVersion

類型：字串

(選用) 目標 SQL Server 版本。預設值為 2016。

12. 輸入參數後，選擇 Execute (執行)。自動化開始時，您可以監控執行進度。
13. 當 Execution Status (執行狀態) 顯示 Success (成功) 時，請展開 Outputs (輸出) 來檢視 AMI 資訊。針對您選擇的 VPC，您可以使用 AMI ID 來啟動您的 SQL Server 執行個體。
14. 開啟 Amazon EC2 主控台。在左側導覽窗格中選擇 AMI。您應該會看到新的 AMI。
15. 若要確認已成功安裝 SQL Server 新版本，請選擇新的 AMI 並選擇 Launch (啟動)。
16. 選擇您要用於 AMI 的執行個體類型、您要部署到的 VPC 和子網，以及您要使用的儲存體。因為您是從 AMI 啟動新的執行個體，將會出現磁碟區，供您選擇加入您啟動的新 EC2 執行個體中。您可以移除任何這些磁碟區，或新增磁碟區。
17. 新增標籤以協助識別您的執行個體。
18. 將一或多個安全群組新增至執行個體。
19. 選擇 Launch Instance (啟動執行個體)。
20. 選取執行個體的標籤名稱，然後在 Actions (動作) 下拉式清單下，選取 Connect (連接)。
21. 驗證 SQL Server 新版本是新執行個體上的新資料庫引擎。

## 將 Windows 執行個體移轉至目前一代的執行個體類型

AWS Windows AMI 配置了 Microsoft 安裝媒體所使用的默認設置，並帶有一些自定義。這些自訂項目包括支援最新一代執行個體類型的驅動程式和設定，這些[執行個體是建立在 AWS Nitro System 上的](#)執行個體，例如 M5 或 C5。

在遷移至 Nitro 型執行個體 (包括裸機執行個體) 時，我們建議您在下列情況下遵循本主題中的步驟：

- 如果您是從自訂的 Windows AMI 啟動執行個體
- 如果您正從 Amazon 在 2018 年 8 月之前建立並提供的 Windows AMI 啟動執行個體

如需詳細資訊，請參閱 [Amazon EC2 Update – Additional Instance Types, Nitro System, and CPU Options](#)。

**Note**

下列遷移程序可以在 Windows Server 2008 R2 版和更新版本上執行。若要將 Linux 執行個體遷移至最新一代執行個體類型，請參閱 [the section called “變更執行個體類型”](#)。

## 內容

- [第 1 部分：安裝和升級 AWS PV 驅動程序](#)
- [第 2 部分：安裝和升級 ENA](#)
- [第 3 部分：升級 AWS NVMe 驅動程序](#)
- [第 4 部分：更新 EC2Config 與 EC2Launch](#)
- [第 5 部分：為 Bare Metal 執行個體安裝序列埠驅動程式](#)
- [第 6 部分：更新電源管理設定](#)
- [第 7 部分：更新新執行個體類型的 Intel 晶片組驅動程式](#)
- [\(替代方法\) 使用升級 AWS PV、ENA 和 NVMe 驅動程式 AWS Systems Manager](#)
- [將 Windows 執行個體從硝基遷移到 Xen 執行個體類型](#)

**Note**

或者，您可以使用 `AWSSupport-UpgradeWindowsAWSDrivers` 自動化文件，以自動執行第 1 部分、第 2 部分及第 3 部分中所述的程序。如果您選擇使用自動程序，請參閱 [\(替代方法\) 使用升級 AWS PV、ENA 和 NVMe 驅動程式 AWS Systems Manager](#)，然後繼續進行第 4 部分及第 5 部分。

## 開始之前

此程序假設您目前正在執行上一代 Xen 型執行個體類型 (例如 M4 或 C4)，而且您正在移轉至 Nitro System [上建置的執行個體](#)。AWS

您必須使用 3.0 或更新 PowerShell 版本才能成功執行升級。

**Note**

當遷移到最新一代執行個體時，現有 ENI 上的靜態 IP 或自訂 DNS 網路設定可能會遺失，因為執行個體預設使用新的增強型聯網轉接器裝置。

在遵循此程序中的步驟之前，我們建議您建立該執行個體的備份。從 [EC2 主控台](#) 中，選擇需要遷移的執行個體、開啟內容 (按一下右鍵) 功能表、選擇 Instance State (執行個體狀態)，然後選擇 Stop (停止)。

**Warning**

停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區上的任何資料，請確定您將資料備份至持久性儲存。

在 [EC2 console \(EC2 主控台\)](#) 中開啟執行個體的內容 (按一下右鍵) 功能表、選擇 Image (映像)，然後選擇 Create Image (建立映像)。

**Note**

將執行個體類型移轉或變更為最新一代後，即可完成這些指示的第 4 和第 5 部分。不過，如果您要專門移轉至裸機執行個體類型，建議您在移轉之前先完成這些項目。

## 第 1 部分：安裝和升級 AWS PV 驅動程式

雖然 Nitro 系統中未使用 AWS 光伏驅動程式，但如果您使用的是舊版 Citrix PV 或 AWS PV，您仍應該升級它們。最新的 AWS PV 驅動程式會解決之前版本驅動程式中的錯誤，這些錯誤可能會在您使用 Nitro 系統，或者需要遷移回 Xen 執行個體時出現。最佳作法是，我們建議您一律開啟 Windows 執行個體更新至最新的驅動程式 AWS。

請使用下列程序在視窗伺服器 2008 R2、視窗伺服器 2012 年 R2、視窗伺服器 2012 R2、視窗伺服器 2012 R2、視窗伺服器 2012 年 R2、視窗伺服器 2016 年或視窗伺服器 2019 上執行 AWS 光伏驅動程式的就地升級，或將光伏驅動程式升級至光伏驅動程式 AWS 如需詳細資訊，請參閱 [升級 Windows 執行個體上的 PV 驅動程式](#)。

若要升級網域控制站，請參閱 [升級網域控制站 \(AWS PV 升級\)](#)。

## 若要執行 AWS PV 驅動程式的升級或

1. 使用遠端桌面連線到執行個體並為執行個體準備升級。在執行升級前將所有非系統磁碟離線。如果您正在執行 AWS PV 驅動程式的就地更新，則不需要執行此步驟。在服務主控台中將非必要的服務設為手動啟動。
2. 將最新的驅動程式套件[下載](#)到執行個體。
3. 解壓縮資料夾的內容，然後執行 AWSPVDriverSetup.msi。

在執行 MSI 之後，執行個體會自動重新開機並升級驅動程式。執行個體最多可能會在 15 分鐘內無法使用。

在升級完成且執行個體在 Amazon EC2 主控台中通過兩項運作狀態檢查後，使用遠端桌面連線到執行個體，確認已安裝新的驅動程式。在裝置管理員中，於 Storage Controllers (儲存控制器) 下，尋找 AWS PV Storage Host Adapter (AWS PV 儲存體主機控制器)。確認驅動程式版本與驅動程式版本歷史記錄表格中列出的最新版本相同。如需詳細資訊，請參閱 [AWS 光伏驅動器封裝歷史](#)。

## 第 2 部分：安裝和升級 ENA

升級到最新的彈性網路轉接器驅動程式，以確保支援所有網路功能。如果您啟動執行個體，且其尚未啟用增強型聯網，您必須在執行個體上下載並安裝必要的網路轉接器驅動程式。然後，設定 enaSupport 執行個體屬性，以啟用增強型聯網。您只能在支援的執行個體類型上啟用此屬性，且需先安裝 ENA 驅動程式。如需詳細資訊，請參閱 [使用 EC2 執行個體上的彈性網路介面卡 \(ENA\) 啟用增強型聯網](#)。

1. 將最新的驅動程式[下載](#)到執行個體。
2. 將 zip 封存檔解壓縮。
3. 通過從解壓縮的文件夾中運行install.ps1 PowerShell腳本來安裝驅動程序。

### Note

請以管理員身分執行 install.ps1 指令碼，以避免發生安裝錯誤。

4. 請檢查您的 AMI 是否已啟動 enaSupport。若否，請遵循[使用 EC2 執行個體上的彈性網路介面卡 \(ENA\) 啟用增強型聯網](#)中的文件繼續進行。

## 第 3 部分：升級 AWS NVMe 驅動程序

AWS NVMe 驅動程式可用來與 Amazon EBS 和 SSD 執行個體存放區磁碟區互動，這些磁碟區在 Nitro 系統中以 NVMe 區塊裝置的形式公開，以獲得更好的效能。

**⚠ Important**

下列指示專門針對您在上一代執行個體上安裝或升級 AWS NVMe 時修改，目的是將執行個體遷移至最新一代的執行個體類型。

1. 將最新的驅動程式套件[下載](#)到執行個體。
2. 將 zip 封存檔解壓縮。
3. 執行 `dpinst.exe` 命令以安裝驅動程式。
4. 開啟 PowerShell 工作階段並執行下列命令：

```
PS C:\> start rundll32.exe sppnp.dll, Sysprep_Generalize_Pnp -wait
```

**i Note**

若要套用命令，您必須以系統管理員身分執行 PowerShell 工作階段。PowerShell (x86) 版本將導致錯誤。  
此命令僅在裝置驅動程式上執行 sysprep。它不會執行完整的 sysprep 準備。

5. Windows Server 2008 R2 和 Windows Server 2012 請關閉執行個體，將執行個體類型變更為最新一代的執行個體並啟動它，然後繼續進行第 4 部分。若您在遷移至最新的執行個體類型前，在上一代的執行個體類型上再次啟動了該執行個體，執行個體將不會啟動。對於其他支援的 Windows AMI，您可以在裝置 sysprep 之後隨時變更執行個體類型。

## 第 4 部分：更新 EC2Config 與 EC2Launch

對 Windows 執行個體而言，當最新的 EC2Config 與 EC2Launch 公用程式在 Nitro 系統 (包括 EC2 Bare Metal) 上執行時，其將提供其他功能與資訊。根據預設，EC2Config 包含在 Windows Server 2016 之前版本的 AMI 內。EC2Launch 將取代 Windows Server 2016 和更新版本 AMI 上的 EC2Config。

更新 EC2Config 與 EC2Launch 服務時，AWS 的新 Windows AMI 將包含最新版的服務。不過，您必須更新自己的 Windows AMI 和執行個體，獲得最新版的 EC2Config 與 EC2Launch。

### 安裝或更新 EC2Config

1. 下載並解壓縮 [EC2Config 安裝程式](#)。

2. 執行 EC2Install.exe。如需完整的選項清單，請執行 EC2Install 並加入 /? 選項。根據預設，設定程式會顯示提示。若要執行命令且不加入提示，請使用 /quiet 選項。

如需詳細資訊，請參閱 [安裝最新版本的 EC2Config](#)。

### 安裝或更新 EC2Launch

1. 如果您已在執行個體上安裝和設定 EC2Launch，請備份 EC2Launch 組態檔案。安裝程序不會保留此檔案的變更內容。根據預設，檔案位於 C:\ProgramData\Amazon\EC2-Windows\Launch\Config 目錄。
2. 下載 [EC2-Windows-Launch.zip](#) 到執行個體上的目錄。
3. 下載 [install.ps1](#) 到您下載 EC2-Windows-Launch.zip 的同一個目錄。
4. 執行 install.ps1。

#### Note

請以管理員身分執行 install.ps1 指令碼，以避免發生安裝錯誤。

5. 如果您備份了 EC2Launch 組態檔案，請將其複製到 C:\ProgramData\Amazon\EC2-Windows\Launch\Config 目錄。

如需詳細資訊，請參閱 [使用 EC2Launch 設定 Windows 執行個體](#)。

## 第 5 部分：為 Bare Metal 執行個體安裝序列埠驅動程式

i3.metal 執行個體類型會使用 PCI 型序列裝置，而非 I/O 連接埠型序列裝置。最新的 Windows AMI 會自動使用 PCI 型序列裝置並安裝好序列埠驅動程式。如果您不是使用 Amazon 於 2018 年 4 月 11 日或之後發布的 Windows AMI 所啟動的執行個體，您必須安裝序列埠驅動程式，為 EC2 功能 (例如密碼產生或主控台輸出) 啟用序列裝置。最新的 EC2Config 與 EC2Launch 公用程式也支援 i3.metal 並提供其他功能。若您尚未這樣做，請遵循第 4 部分中的步驟。

### 安裝序列埠驅動程式

1. 將序列埠驅動程式套件 [下載](#) 到執行個體。
2. 擷取資料夾內容、針對 aws\_ser.INF 開啟內容 (按一下右鍵) 功能表，然後選擇 install (安裝)。
3. 選擇 Okay (確定)。



## 第 6 部分：更新電源管理設定

電源管理設定的下列更新會設定永不關閉顯示器，讓 Nitro 系統上的作業系統能正常關機。Amazon 提供的所有 Windows AMI，自 2018 年 11 月 28 日起皆有此預設組態。

1. 開啟命令提示字元或 PowerShell 工作階段。
2. 執行下列命令：

```
powercfg /setacvalueindex 381b4222-f694-41f0-9685-ff5bb260df2e 7516b95f-f776-4464-8c53-06167f40cc99 3c0bc021-c8a8-4e07-a973-6b14cbcb2b7e 0
powercfg /setacvalueindex 8c5e7fda-e8bf-4a96-9a85-a6e23a8c635c 7516b95f-f776-4464-8c53-06167f40cc99 3c0bc021-c8a8-4e07-a973-6b14cbcb2b7e 0
powercfg /setacvalueindex a1841308-3541-4fab-bc81-f71556f20b4a 7516b95f-f776-4464-8c53-06167f40cc99 3c0bc021-c8a8-4e07-a973-6b14cbcb2b7e 0
```

## 第 7 部分：更新新執行個體類型的 Intel 晶片組驅動程式

u-6tb1.metal、u-9tb1.metal 和 u-12tb1.metal 執行個體類型必須使用晶片組驅動程式先前未安裝在 Windows AMI 上的硬體。如果您不是使用從 Amazon 於 2018 年 11 月 19 日或之後提供之 Windows AMI 啟動的執行個體，您必須使用 Intel 晶片組 INF 公用程式安裝驅動程式。

### 安裝晶片組驅動程式

1. [下載晶片組公用程式](#)到執行個體。
2. 將檔案解壓縮。
3. 執行 SetupChipset.exe。
4. 接受 Intel 軟體授權合約，並安裝晶片組驅動程式。
5. 重新啟動執行個體。

### (替代方法) 使用升級 AWS PV、ENA 和 NVMe 驅動程式 AWS Systems Manager

AWSsupport-UpgradeWindowsAWSDrivers 自動化文件可以自動執行第 1 部分、第 2 部分及第 3 部分中所述的步驟。此方法也可以修復驅動程式升級失敗的執行個體。

AWSsupport-UpgradeWindowsAWSDrivers 自動化文件升級或修復指定 EC2 執行個體上的儲存和網路 AWS 驅動程式。此文件會嘗試透過呼叫代理程 AWS Systems Manager 式 (SSM 代理程式) 來在線上安裝最新版本的 AWS 驅動程式。如果 SSM Agent 無法連絡，文件可以在明確要求時執行 AWS 驅動程式的離線安裝。

**Note**

此程序將無法在網域控制器上進行。若要更新網域控制器上的驅動程式，請參閱[升級網域控制站 \(AWS PV 升級\)](#)。

若要使用自動升級 AWS PV、ENA 和 NVMe 驅動程式 AWS Systems Manager

1. 開啟 <https://console.aws.amazon.com/systems-manager> 中的 Systems Manager 主控台。
2. 選擇 Automation (自動化)、Execute Automation (執行自動化)。
3. 搜尋並選取 AWSSupport-UpgradeWindows AWSDrivers 自動化文件，然後選擇 [執行自動化]。
4. 在「輸入參數」區段中，設定下列選項：

執行個體 ID

輸入要升級之執行個體的唯一 ID。

AllowOffline

(選用) 選擇下列其中一個選項：

- True – 選擇此選項來執行離線安裝。升級過程中執行個體會停止並重新啟動。

**Warning**

停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區上的任何資料，請確定您將資料備份至持久性儲存。

- False – (預設值) 若要執行線上安裝，請將此選項保持選取狀態。升級過程中執行個體會重新啟動。

**Important**

線上和離線升級會在嘗試升級作業之前建立 AMI。在自動化完成之後，AMI 會持續存在。保護您對 AMI 的存取，或刪除它 (若不再需要它的話)。

SubnetId

(選用) 輸入下列其中一個值：

- `SelectedInstanceSubnet` – (預設值)升級程序會將 helper 執行個體啟動至與要升級之執行個體相同的子網路。子網必須允許與 Systems Manager 端點 (`ssm.*`) 進行通訊。
  - `CreateNewVPC` – 升級程序會將 helper 執行個體啟動至新的 VPC。如果您不確定目標執行個體的字網是否允許與 `ssm.*` 端點進行通訊，請使用此選項。您的使用者必須擁有建立 VPC 的許可。
  - 特定子網路 ID – 指定要將 helper 執行個體啟動至其中之特定子網路的 ID。子網必須與要升級的執行個體位於相同的可用區域，而且它必須允許與 `ssm.*` 端點進行通訊。
5. 選擇 Execute (執行)。
  6. 允許升級以完成。完成線上升級最多可能需要 10 分鐘，而完成離線升級最需要多 25 分鐘。

## 將 Windows 執行個體從硝基遷移到 Xen 執行個體類型

下列程序假設您目前正在執行以 Nitro 為基礎的執行個體類型，而且您要移轉至以 Xen System 為基礎的執行個體，例如 M4 或 C4。如需執行個體類型規格，請參閱 [Amazon EC2 執行個體類型指南](#)。在遷移之前執行下列步驟，以避免在開機程序期間發生錯誤。

### 從硝基遷移到 Xen

1. 備份您的資料。
2. 確認您的 Windows [san 原則](#) 允許非根儲存磁碟區上線。
3. AWS 移轉到 Xen 執行個體之前，必須先在 Nitro 執行個體上安裝和升級 PV 驅動程式。如需安裝和升級 AWS PV 驅動程式的步驟，請參閱 [第 1 部分：安裝和升級 AWS PV 驅動程序](#)。
4. 更新至最新的 EC2Launch v2 版本。如需了解步驟，請參閱 [遷移到 EC2Launch v2](#)。
5. 開啟 PowerShell 工作階段並以系統管理員身分執行下列命令，以 sysprep 裝置驅動程式。執行 sysprep 可確保在 Xen 執行個體上開機所需的提早開機儲存驅動程式已在 Windows 中正確註冊。

#### Note

使用 PowerShell (x86) 版本執行指令會導致錯誤。這個命令只會將關鍵開機的裝置驅動程式新增至重要的裝置資料庫。它不會執行完整的 sysprep 準備。

```
Start-Process rundll32.exe sppnp.dll,Sysprep_Generalize_Pnp -wait
```

6. sysprep 程序完成後，執行 Xen 執行個體類型的遷移。

## Microsoft SQL 伺服器資料庫的視窗到 Linux 重新平台輔助程式

如需有關將 Microsoft SQL 伺服器資料庫從視窗重新平台化為 Linux 的詳細資訊，請參閱 Amazon EC2 上的 Microsoft SQL 伺服器使用指南中的 [Microsoft SQL 伺服器資料庫重新平台助理](#)。

### 對 Windows 執行個體進行升級的疑難排解

AWS 針對升級協助程式服務的問題或問題提供升級支援，此 AWS 公用程式可協助您執行涉及 Citrix PV 驅動程式的就地升級。

在升級之後，當 .NET 執行時間最佳化服務最佳化 .NET Framework 時，執行個體的平均 CPU 用量可能會暫時增加。這是預期的行為。

若執行個體在數個小時之後仍無法通過兩項狀態檢查，請檢查以下項目。

- 若您升級到 Windows Server 2008，兩項狀態檢查在數個小時之後仍然失敗，升級可能已失敗，並會顯示提示，讓您 Click OK (按一下確認) 以確認還原。因為無法在此狀態存取主控台，沒有任何方式能按一下按鈕。若要因應此問題，請透過 Amazon EC2 主控台或 API 重新開機。初始化重新開機可能需要十分鐘以上。執行個體可能會在 25 分鐘可用。
- 從伺服器移除應用程式或伺服器角色並再試一次。

若執行個體在從伺服器移除應用程式或伺服器角色後仍然無法通過兩項狀態檢查，請執行下列作業。

- 停止執行個體，並將根磁碟區連接到另一個執行個體。如需詳細資訊，請參閱 [等待中繼資料服務](#) 中如何停止並將根磁碟區連接到另一個執行個體的說明。
- 分析 [Windows 安裝日誌檔案及事件日誌](#)，查看故障。

針對其他作業系統升級或遷移的問題，我們建議檢閱 [在您開始就地升級之前](#) 中列出的文章。

# EC2 機群 和 Spot 機群

EC2 機群和 Spot 機群的設計是用於透過 AWS 啟動一個機群 (意即群組) 的執行個體。叢集中的每個執行個體都是以您在[啟動時手動設定的啟動範本](#)或一組啟動參數為基礎。

機群提供下列功能和優點：這些優點可讓您在多個 EC2 執行個體上執行應用程式時，最大限度地節省成本並最佳化可用性和效能。

## 使用多種執行個體類型與購買選項

在單一 API 呼叫中，機群可以啟動多種執行個體類型與購買選項 (Spot 執行個體和隨需執行個體)，讓您透過 Spot 執行個體用量來最佳化成本。您也可以將預留執行個體和 Savings Plan 折扣與機群中的隨需執行個體搭配使用，善加利用這些折扣。

## 跨可用區域分佈執行個體

機群會自動嘗試在多個可用區域內平均分佈執行個體以利達成高可用性。這會在可用區域變得無法使用時提供復原能力。

## 自動替換 Spot 執行個體

如果您的機群包含 Spot 執行個體，當您的 Spot 執行個體因執行個體健康狀態變化而中斷或受損時，它可以自動請求替換 Spot 容量。透過容量重新平衡，機群也可以監控和主動替換具有較高中斷風險的 Spot 執行個體。

如果您需要靈活管理執行個體生命週期或擴展機制的各個層面，EC2 叢集是一個不錯的選擇。您也可以使用 Spot 機群，但我們不建議您這麼做，因為它是沒有計劃投資的舊式 API。但是，如果您已經在使用 Spot 機群，則可以繼續使用它。Spot 機群和 EC2 機群提供相同的核心功能。

### Tip

一般而言，我們建議您不要使用 Amazon EC2 Auto Scaling 啟動 Spot 和隨需執行個體叢集，因為它提供了可用來管理叢集的其他功能。其他功能清單包括 Spot 和隨需執行個體的自動運作狀態檢查替換、以應用程式為基礎的運作狀態檢查，以及與 Elastic Load Balancing 的整合，可確保應用程式流量均勻分佈到運作良好的執行個體。當您使用 Amazon ECS、Amazon EKS (自我管理節點群組) 和 Amazon VPC 格子等 AWS 服務時，也可以使用自動擴展群組。如需詳細資訊，請參閱 [Amazon EC2 Auto Scaling 使用者指南](#)。

## 主題

- [EC2 Fleet](#)
- [Spot 叢集](#)
- [使用 Amazon 監控車隊事件 EventBridge](#)
- [EC2 Fleet 和 Spot Fleet 的教學課程](#)
- [EC2 Fleet 和 Spot Fleet 的範例組態](#)
- [機群配額](#)

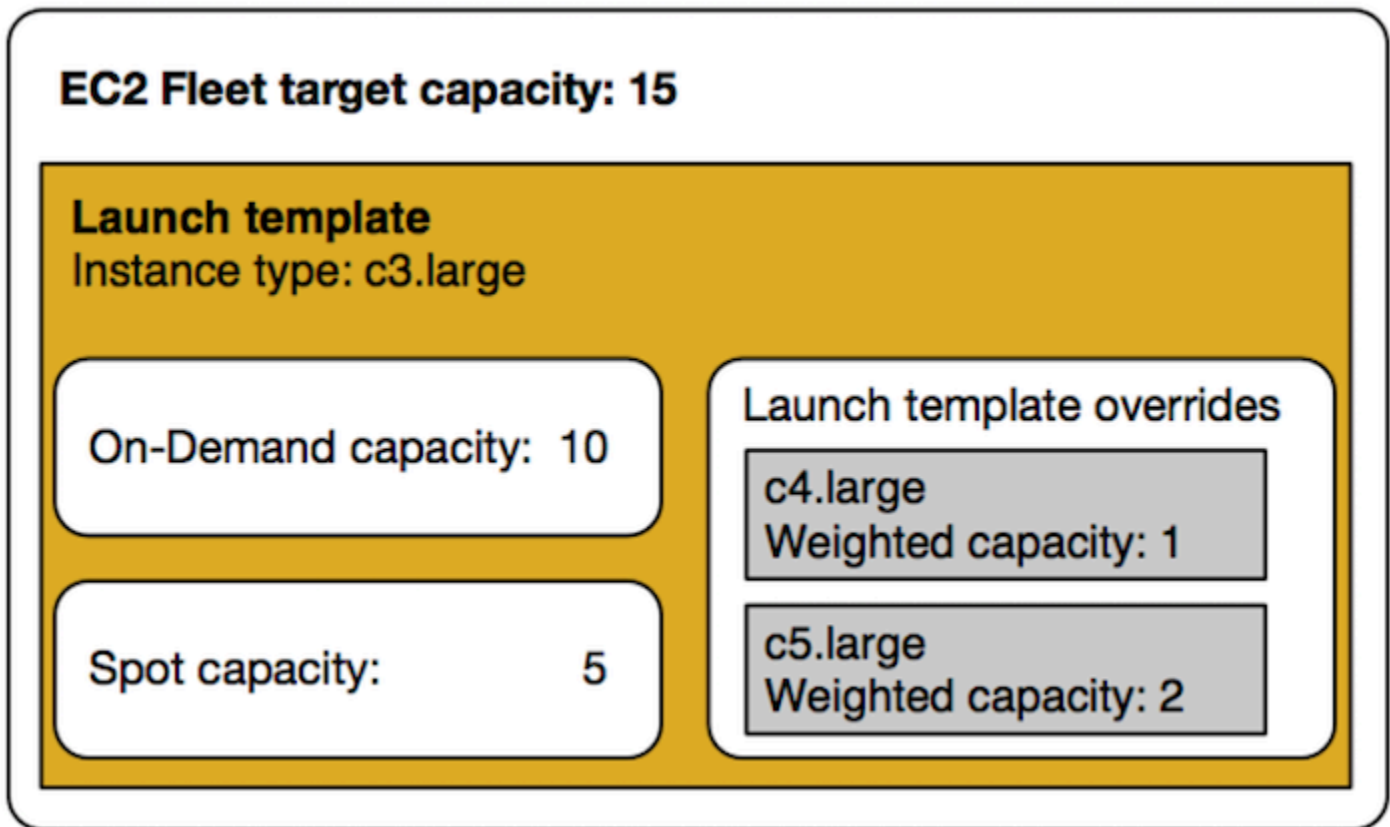
## EC2 Fleet

EC2 機群包含組態資訊用以啟動一個機群 (意即群組) 的執行個體。在單一 API 呼叫中，機群若同時使用隨需執行個體、預留執行個體和 Spot 執行個體購買選項，則可在多個可用區域上啟動多種執行個體類型。使用 EC2 Fleet，您可以：

- 定義個別的 Spot 和隨需容量目標，以及您願意支付的每小時金額上限
- 指定最適合您的應用程式的執行個體類型
- 指定 Amazon EC2 應如何在每個購買選項內分配您的機群容量

您也可以設定您願意為機群支付的每個小時金額上限，而 EC2 Fleet 會啟動執行個體，直到達到金額上限為止。達到您願意支付的金額上限時，機群會停止啟動執行個體，即使它未符合目標容量亦然。

EC2 Fleet 為滿足您在您請求中指定的目標容量請求，會嘗試啟動 Spot 執行個體，直到符合所需數目。如果您指定每個小時總價格上限，它會滿足容量，直到它達到您願意支付的金額上限為止。如果您的競價型執行個體中斷，叢集也會嘗試維持其目標 Spot 容量。如需詳細資訊，請參閱 [競價型執行個體的運作方式](#)。



您可以為每個 EC2 Fleet 指定無限數目的執行個體類型。可以使用 Spot 和隨需這兩種購買選項來佈建這些執行個體類型。您也可以指定多個可用區域，為每個執行個體指定不同的最高 Spot 價格，以及為每個機群選擇額外的 Spot 選項。Amazon EC2 會在機群啟動時使用指定的選項來佈建容量。

當機群正在執行時，如果 Amazon EC2 由於價格增加或執行個體故障而回收 Spot 執行個體，則 EC2 Fleet 可以嘗試用您指定的任何執行個體類型來替換執行個體。這能讓 Spot 定價高峰期間重新獲得容量變得更容易。您可以為每個機群制定靈活有彈性的資源分配策略。例如，在特定機群內，如果可用，您的主要容量可以是隨需補充和較低成本的 Spot 容量。

若您擁有預留執行個體且您在機群中指定隨需執行個體，則 EC2 Fleet 會使用預留執行個體。例如，如果您的機群將隨需執行個體指定為 c4.large，並且您擁有 c4.large 的預留執行個體，您會收到預留執行個體定價。如果您使用 Savings Plan，也同樣適用。

使用 EC2 Fleet 無需額外收費。您僅需支付機群為您啟動的 EC2 執行個體。

## 內容

- [EC2 Fleet 限制](#)
- [爆量效能執行個體](#)
- [EC2 Fleet 請求類型](#)

- [EC2 Fleet 組態策略](#)
- [使用 EC2 Fleet](#)

## EC2 Fleet 限制

下列限制適用於 EC2 Fleet：

- EC2 機群僅能透過 [Amazon EC2 API](#)、[AWS CLI](#)、[AWS SDK](#) 和 [AWS CloudFormation](#) 提供。
- EC2 叢集請求無法跨越 AWS 區域。您需要為每個區域建立個別的 EC2 Fleet。
- EC2 Fleet 請求不能橫跨同一個可用區域的不同子網。

## 爆量效能執行個體

如果您使用[爆量效能執行個體](#)啟動 Spot 執行個體，而且您計劃立即使用爆量效能 Spot 執行個體，且在短時間內沒有閒置時間來累積 CPU 額度，建議您以[標準模式](#)將其啟用，以避免支付更高的成本。如果您在[無限制模式](#)下啟動高載效能 競價型執行個體，而且 CPU 立即高載，則您將耗費高載的剩餘額度。如果您短暫使用執行個體，則您的執行個體沒有時間產生 CPU 額度來償還剩餘額度，而且當終止執行個體時，會針對剩餘額度向您收費。

只有在執行個體執行夠長時間，可為爆量產生 CPU 額度時，無限制模式才適合用於高載效能 競價型執行個體。否則，支付剩餘額度會使爆量效能 競價型執行個體比使用其他執行個體更為昂貴。如需詳細資訊，請參閱 [使用無限制模式與固定 CPU 的時機](#)。

啟動額度旨在透過提供足夠的運算資源來設定執行個體，創造 T2 執行個體具有生產力的初始啟動體驗。不允許重複啟動 T2 執行個體來獲得新的啟動額度。如果您需要持續的 CPU，可以賺取額度 (藉由在某些期間閒置)、為 T2 競價型執行個體使用[無限制模式](#)，或將執行個體類型與專屬 CPU 搭配使用。

## EC2 Fleet 請求類型

EC2 Fleet 請求有三種類型：

instant

如果您將請求類型設定為 instant，EC2 Fleet 會針對您所需容量送出同步的一次性請求。它會在 API 回應中，傳回已啟動的執行個體，以及無法啟動的執行個體錯誤。如需詳細資訊，請參閱 [使用「instant」類型的 EC2 Fleet](#)。



## request

如果您將請求類型設定為 `request`，EC2 Fleet 會針對您所需容量送出非同步的一次性請求。如果容量因為 Spot 中斷而減少，機群不會嘗試補充 競價型執行個體；如果無法使用容量，也不會在替代的 Spot 容量集區中提交請求。

## maintain

(預設) 如果您將請求類型設定為 `maintain`，EC2 Fleet 會針對您所需容量送出非同步請求，並自動補充所有中斷的 競價型執行個體以維持容量。

這三種請求類型都受益於分配策略。如需詳細資訊，請參閱 [競價型執行個體的分配策略](#)。

## 使用「instant」類型的 EC2 Fleet

`instant` 類型的 EC2 Fleet 是同步的一次性請求，只會嘗試啟動您所需的容量一次。API 回應列出已啟動的執行個體，以及無法啟動的執行個體錯誤。使用 `instant` 類型的 EC2 Fleet 具有本文描述的多種好處。本文末尾處會提供範例組態。

對於需要僅啟動 API 來啟動 EC2 執行個體的工作負載，您可以使用 API。RunInstances 但是，使用時 RunInstances，您只能啟動隨需執行個體或 Spot 執行個體，但不能在同一個請求中啟動兩者。此外，當您使用 RunInstances 啟動 Spot 執行個體時，您的競價型執行個體請求僅限於一種執行個體類型和一個可用區域。此操作針對單個 Spot 容量集區（一組具有相同執行個體類型和可用區域的未使用執行個體）。如果 Spot 容量集區沒有足夠的 Spot 執行個體容量供您的請求使用，則 RunInstances 呼叫會失敗。

建議您不 RunInstances 要使用啟動 Spot 執行個體，而是使用 `type` 參數設定 `instant` 為的 CreateFleet API，以獲得下列好處：

- 在一個請求中啟動隨需執行個體和 Spot 執行個體。EC2 Fleet 可以啟動隨需執行個體、Spot 執行個體或兩者皆有。如果有可用容量且您的請求中每個小時的價格上限超過 Spot 價格，則會滿足 Spot 執行個體的請求。
- 增加 Spot 執行個體的可用性。透過使用 `instant` 類型 EC2 Fleet，您可以按照 [Spot 最佳實務](#) 啟動 Spot 執行個體，獲得以下好處：
  - Spot 最佳實務：對執行個體類型和可用區域具有彈性。

好處：透過指定數個執行個體類型和可用區域，您可以增加 Spot 容量集區的數目。這可讓 Spot 服務更有機會尋找並配置您所需的 Spot 運算容量。一個很好的經驗法則是針對每個工作負載的至少 10 個執行個體類型上具有彈性，並確定所有可用區域均已設定為在 VPC 中使用。

- 發現最佳做法：使用price-capacity-optimized配置策略。

優點：price-capacity-optimized配置策略可識別最可用的 Spot 容量集區中的執行個體，然後從這些集區中最低價格自動佈建執行個體。由於您的 Spot 執行個體容量來自具有最佳容量的集區，因此可降低 Amazon EC2 需要回收容量時，中斷 Spot 執行個體的可能性。

- 取得更廣泛的功能組合。對於需要僅啟動 API 的工作負載，以及您希望管理執行個體生命週期而非讓 EC2 叢集為您管理的工作負載，請使用 EC2 叢集類型instant而非 API。[RunInstances](#) EC2 叢集提供比 RunInstances下列範例所示更廣泛的功能組合。對於所有其他工作負載，您應使用 Amazon EC2 Auto Scaling，因為它為各種工作負載提供更全面的功能組合，例如 ELB 後端應用程式、容器化工作負載和佇列處理任務。

您可以使用即時類型的 EC2 Fleet，在容量區塊中啟動執行個體。如需詳細資訊，請參閱 [教學課程：在容量區塊中啟動執行個體](#)。

AWS Amazon EC2 Auto Scaling 和 Amazon EMR 等服務使用類型即時的 EC2 叢集來啟動 EC2 執行個體。

立即類型 EC2 Fleet 的先決條件

如需建立 EC2 Fleet 的先決條件，請參閱 [EC2 Fleet 先決條件](#)。

instant 類型的 EC2 Fleet 如何運作

當使用 instant 類型的 EC2 Fleet 時，事件的順序如下所示：

1. 將[CreateFleet](#)要求類型設定為instant。如需詳細資訊，請參閱 [建立 EC2 Fleet](#)。請注意，在您進行 API 呼叫後，將無法修改它。
2. 在您進行 API 呼叫時，EC2 Fleet 會針對您所需容量送出同步的一次性請求。
3. API 回應列出已啟動的執行個體，以及無法啟動的執行個體錯誤。
4. 您可以描述您的 EC2 Fleet、列出與 EC2 Fleet 相關的執行個體，並檢視 EC2 Fleet 的歷史記錄。
5. 執行個體啟動後，您可以[刪除叢集請求](#)。刪除機群請求時，您也可以選擇終止相關聯的執行個體，或讓它們繼續執行。
6. 您隨時可終止執行個體。

範例

以下範例展示如何為不同的使用案例使用 instant 類型的 EC2 Fleet。如需有關使用 EC2 CreateFleet API 參數的詳細資訊，請參閱亞 Amazon EC2 API 參考[CreateFleet](#)中的。

## 範例

- [範例 1：使用容量最佳化分配策略啟動 Spot 執行個體](#)
- [範例 2：使用容量最佳化分配策略啟動單一 Spot 執行個體](#)
- [範例 3：使用執行個體權重啟動 Spot Fleet](#)
- [範例 4：在單一可用區域內啟動 Spot 執行個體](#)
- [範例 5：在單一可用區域內啟動單一執行個體類型的 Spot 執行個體](#)
- [範例 6：僅在可啟動最小目標容量時才啟動 Spot 執行個體](#)
- [範例 7：僅當可在單個可用區域中啟動相同執行個體類型的最低目標容量時，才能啟動 Spot 執行個體](#)
- [範例 8：使用多個啟動範本來啟動執行個體](#)
- [範例 9：啟動基於隨需執行個體的 Spot 執行個體](#)
- [範例 10：基於使用容量預留和優先順序分配策略的隨需執行個體，使用容量最佳化分配策略啟動 Spot 執行個體](#)
- [範例 11：使用 capacity-optimized-prioritized 配置策略啟動競價型執行個體](#)

### 範例 1：使用容量最佳化分配策略啟動 Spot 執行個體

下列範例指定 instant 類型 EC2 Fleet 中所需的參數：啟動範本、目標容量、預設購買選項，以及啟動範本覆寫。

- 啟動範本由其啟動範本名稱和版本編號識別。
- 12 個啟動範本覆寫會指定 4 個不同執行個體類型和 3 個不同的子網，每個都位於不同的可用區域。每個執行個體類型和子網組合都定義了一個 Spot 容量集區，從而產生 12 個 Spot 容量集區。
- 機群的目標容量為 20 個執行個體。
- 預設購買選項為 spot，這會導致機群嘗試將 20 個 Spot 執行個體啟動到 Spot 容量集區中，且擁有啟動中執行個體數目之最佳容量。

```
{
 "SpotOptions": {
 "AllocationStrategy": "capacity-optimized"
 },
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "ec2-fleet-1t1",
```

```
 "Version": "$Latest"
 },
 "Overrides": [
 {
 "InstanceType": "c5.large",
 "SubnetId": "subnet-fae8c380"
 },
 {
 "InstanceType": "c5.large",
 "SubnetId": "subnet-e7188bab"
 },
 {
 "InstanceType": "c5.large",
 "SubnetId": "subnet-49e41922"
 },
 {
 "InstanceType": "c5d.large",
 "SubnetId": "subnet-fae8c380"
 },
 {
 "InstanceType": "c5d.large",
 "SubnetId": "subnet-e7188bab"
 },
 {
 "InstanceType": "c5d.large",
 "SubnetId": "subnet-49e41922"
 },
 {
 "InstanceType": "m5.large",
 "SubnetId": "subnet-fae8c380"
 },
 {
 "InstanceType": "m5.large",
 "SubnetId": "subnet-e7188bab"
 },
 {
 "InstanceType": "m5.large",
 "SubnetId": "subnet-49e41922"
 },
 {
 "InstanceType": "m5d.large",
 "SubnetId": "subnet-fae8c380"
 },
 {
```

```

 "InstanceType": "m5d.large",
 "SubnetId": "subnet-e7188bab"
 },
 {
 "InstanceType": "m5d.large",
 "SubnetId": "subnet-49e41922"
 }
]
}
],
"TargetCapacitySpecification": {
 "TotalTargetCapacity": 20,
 "DefaultTargetCapacityType": "spot"
},
"Type": "instant"
}

```

## 範例 2：使用容量最佳化分配策略啟動單一 Spot 執行個體

您可以透過將多個類型的 EC2 叢集 API 呼叫設定 TotalTargetCapacity 為 1instant，一次以最佳方式啟動一個 Spot 執行個體。

下列範例指定 instant 類型 EC2 Fleet 中所需的參數：啟動範本、目標容量、預設購買選項，以及啟動範本覆寫。啟動範本由其啟動範本名稱和版本編號識別。12 個啟動範本覆寫會擁有 4 個不同執行個體類型和 3 個不同的子網，每個都位於不同的可用區域。機群的目標容量為 1 個執行個體，並且預設購買選項為 Spot，這會導致機群嘗試根據容量最佳化的分配策略從 12 個 Spot 容量集區之一啟動 Spot 執行個體，以從最可用的容量集區啟動 Spot 執行個體。

```

{
 "SpotOptions": {
 "AllocationStrategy": "capacity-optimized"
 },
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "ec2-fleet-lt1",
 "Version": "$Latest"
 },
 "Overrides": [
 {
 "InstanceType": "c5.large",
 "SubnetId": "subnet-fae8c380"
 }
]
 }
]
}

```

```
{
 "InstanceType": "c5.large",
 "SubnetId": "subnet-e7188bab"
},
{
 "InstanceType": "c5.large",
 "SubnetId": "subnet-49e41922"
},
{
 "InstanceType": "c5d.large",
 "SubnetId": "subnet-fae8c380"
},
{
 "InstanceType": "c5d.large",
 "SubnetId": "subnet-e7188bab"
},
{
 "InstanceType": "c5d.large",
 "SubnetId": "subnet-49e41922"
},
{
 "InstanceType": "m5.large",
 "SubnetId": "subnet-fae8c380"
},
{
 "InstanceType": "m5.large",
 "SubnetId": "subnet-e7188bab"
},
{
 "InstanceType": "m5.large",
 "SubnetId": "subnet-49e41922"
},
{
 "InstanceType": "m5d.large",
 "SubnetId": "subnet-fae8c380"
},
{
 "InstanceType": "m5d.large",
 "SubnetId": "subnet-e7188bab"
},
{
 "InstanceType": "m5d.large",
 "SubnetId": "subnet-49e41922"
}
}
```

```

]
 }
],
"TargetCapacitySpecification": {
 "TotalTargetCapacity": 1,
 "DefaultTargetCapacityType": "spot"
},
"Type": "instant"
}

```

### 範例 3：使用執行個體權重啟動 Spot Fleet

下列範例使用執行個體權重，這表示價格為每單位小時而非每執行個體小時的價格。假設一個工作負載單位需要 15 GB 的記憶體和 4 個 vCPU，每個啟動組態會列出不同的執行個體類型和不同的權重，具體取決於執行個體上可以執行的工作負載單位數。例如，m5.xlarge (4 個 vCPU 和 16 GB 記憶體) 可以執行一個單位，並且權重為 1，m5.2xlarge (8 個 vCPU 和 32 GB 記憶體) 可以執行 2 個單位，並且權重為 2，依此類推。目標總容量設定為 40 個單位。預設購買選項為 Spot，且分配策略容量最佳化，這會產生 40 個 m5.xlarge (40 除以 1)、20 個 m5.2xlarge (40 除以 2)、10 個 m5.4xlarge (40 除以 4)、5 個 m5.8xlarge (40 除以 8)，或基於容量最佳化分配策略，權重相加達到所需容量的執行個體類型的混合。

如需詳細資訊，請參閱 [EC2 Fleet 執行個體權重](#)。

```

{
 "SpotOptions":{
 "AllocationStrategy":"capacity-optimized"
 },
 "LaunchTemplateConfigs":[
 {
 "LaunchTemplateSpecification":{
 "LaunchTemplateName":"ec2-fleet-lt1",
 "Version":"$Latest"
 },
 "Overrides":[
 {
 "InstanceType":"m5.xlarge",
 "SubnetId":"subnet-fae8c380",
 "WeightedCapacity":1
 },
 {
 "InstanceType":"m5.xlarge",
 "SubnetId":"subnet-e7188bab",

```

```
 "WeightedCapacity":1
 },
 {
 "InstanceType":"m5.xlarge",
 "SubnetId":"subnet-49e41922",
 "WeightedCapacity":1
 },
 {
 "InstanceType":"m5.2xlarge",
 "SubnetId":"subnet-fae8c380",
 "WeightedCapacity":2
 },
 {
 "InstanceType":"m5.2xlarge",
 "SubnetId":"subnet-e7188bab",
 "WeightedCapacity":2
 },
 {
 "InstanceType":"m5.2xlarge",
 "SubnetId":"subnet-49e41922",
 "WeightedCapacity":2
 },
 {
 "InstanceType":"m5.4xlarge",
 "SubnetId":"subnet-fae8c380",
 "WeightedCapacity":4
 },
 {
 "InstanceType":"m5.4xlarge",
 "SubnetId":"subnet-e7188bab",
 "WeightedCapacity":4
 },
 {
 "InstanceType":"m5.4xlarge",
 "SubnetId":"subnet-49e41922",
 "WeightedCapacity":4
 },
 {
 "InstanceType":"m5.8xlarge",
 "SubnetId":"subnet-fae8c380",
 "WeightedCapacity":8
 },
 {
 "InstanceType":"m5.8xlarge",
```



```

 "SubnetId": "subnet-e7188bab",
 "WeightedCapacity": 8
 },
 {
 "InstanceType": "m5.8xlarge",
 "SubnetId": "subnet-49e41922",
 "WeightedCapacity": 8
 }
]
}
],
"TargetCapacitySpecification": {
 "TotalTargetCapacity": 40,
 "DefaultTargetCapacityType": "spot"
},
"Type": "instant"
}

```

#### 範例 4：在單一可用區域內啟動 Spot 執行個體

您可以將 Spot 選項 `SingleAvailabilityZone` 設定為 `true`，將叢集設定為在單一可用區域中啟動所有執行個體。

12 個啟動範本覆寫會擁有不同執行個體類型和子網 (每個都位於不同的可用區域)，但權重容量相同。總目標容量為 20 個執行個體，預設購買選項為 Spot，Spot 分配策略為容量最佳化。EC2 Fleet 使用啟動規格，從具有最佳容量的 Spot 容量集區啟動全部在單一可用區域中的 20 個 Spot 執行個體。

```

{
 "SpotOptions": {
 "AllocationStrategy": "capacity-optimized",
 "SingleAvailabilityZone": true
 },
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "ec2-fleet-lt1",
 "Version": "$Latest"
 },
 "Overrides": [
 {
 "InstanceType": "c5.4xlarge",
 "SubnetId": "subnet-fae8c380"
 }
]
 }
]
}

```

```
{
 "InstanceType": "c5.4xlarge",
 "SubnetId": "subnet-e7188bab"
},
{
 "InstanceType": "c5.4xlarge",
 "SubnetId": "subnet-49e41922"
},
{
 "InstanceType": "c5d.4xlarge",
 "SubnetId": "subnet-fae8c380"
},
{
 "InstanceType": "c5d.4xlarge",
 "SubnetId": "subnet-e7188bab"
},
{
 "InstanceType": "c5d.4xlarge",
 "SubnetId": "subnet-49e41922"
},
{
 "InstanceType": "m5.4xlarge",
 "SubnetId": "subnet-fae8c380"
},
{
 "InstanceType": "m5.4xlarge",
 "SubnetId": "subnet-e7188bab"
},
{
 "InstanceType": "m5.4xlarge",
 "SubnetId": "subnet-49e41922"
},
{
 "InstanceType": "m5d.4xlarge",
 "SubnetId": "subnet-fae8c380"
},
{
 "InstanceType": "m5d.4xlarge",
 "SubnetId": "subnet-e7188bab"
},
{
 "InstanceType": "m5d.4xlarge",
 "SubnetId": "subnet-49e41922"
}
}
```

```

]
 }
],
"TargetCapacitySpecification": {
 "TotalTargetCapacity": 20,
 "DefaultTargetCapacityType": "spot"
},
"Type": "instant"
}

```

### 範例 5：在單一可用區域內啟動單一執行個體類型的 Spot 執行個體

您可以設定叢集啟動相同執行個體類型的所有執行個體，並在單一可用區域中啟動所有執 SpotOptions SingleInstanceType 行個體，方 SingleAvailabilityZone 法是將。

12 個啟動範本覆寫會擁有不同執行個體類型和子網 (每個都位於不同的可用區域)，但權重容量相同。總目標容量為 20 個執行個體，預設購買選項為 Spot，Spot 分配策略為容量最佳化。EC2 Fleet 使用啟動規格，從具有最佳容量的 Spot 容量集區啟動全部在單一可用區域中，相同執行個體類型的 20 個 Spot 執行個體。

```

{
 "SpotOptions": {
 "AllocationStrategy": "capacity-optimized",
 "SingleInstanceType": true,
 "SingleAvailabilityZone": true
 },
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "ec2-fleet-lt1",
 "Version": "$Latest"
 },
 "Overrides": [
 {
 "InstanceType": "c5.4xlarge",
 "SubnetId": "subnet-fae8c380"
 },
 {
 "InstanceType": "c5.4xlarge",
 "SubnetId": "subnet-e7188bab"
 },
 {
 "InstanceType": "c5.4xlarge",

```

```
 "SubnetId": "subnet-49e41922"
 },
 {
 "InstanceType": "c5d.4xlarge",
 "SubnetId": "subnet-fae8c380"
 },
 {
 "InstanceType": "c5d.4xlarge",
 "SubnetId": "subnet-e7188bab"
 },
 {
 "InstanceType": "c5d.4xlarge",
 "SubnetId": "subnet-49e41922"
 },
 {
 "InstanceType": "m5.4xlarge",
 "SubnetId": "subnet-fae8c380"
 },
 {
 "InstanceType": "m5.4xlarge",
 "SubnetId": "subnet-e7188bab"
 },
 {
 "InstanceType": "m5.4xlarge",
 "SubnetId": "subnet-49e41922"
 },
 {
 "InstanceType": "m5d.4xlarge",
 "SubnetId": "subnet-fae8c380"
 },
 {
 "InstanceType": "m5d.4xlarge",
 "SubnetId": "subnet-e7188bab"
 },
 {
 "InstanceType": "m5d.4xlarge",
 "SubnetId": "subnet-49e41922"
 }
]
}
],
"TargetCapacitySpecification": {
 "TotalTargetCapacity": 20,
 "DefaultTargetCapacityType": "spot"
```

```
 },
 "Type": "instant"
}
```

### 範例 6：僅在可啟動最小目標容量時才啟動 Spot 執行個體

您可以透過將 Spot 選項設定為要一起啟動的最小目標容量，將叢集設定 `MinTargetCapacity` 為僅在可啟動最小目標容量時啟動執行個體。

12 個啟動範本覆寫會擁有不同執行個體類型和子網 (每個都位於不同的可用區域)，但權重容量相同。總目標容量和最小目標容量設定為 20 個執行個體，預設購買選項為 Spot，Spot 分配策略為容量最佳化。只有在能夠同時啟動全部 20 個執行個體時，EC2 Fleet 才能使用啟動範本覆寫，從具有最優容量的 Spot 容量集區啟動 20 個 Spot 執行個體。

```
{
 "SpotOptions": {
 "AllocationStrategy": "capacity-optimized",
 "MinTargetCapacity": 20
 },
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "ec2-fleet-lt1",
 "Version": "$Latest"
 },
 "Overrides": [
 {
 "InstanceType": "c5.4xlarge",
 "SubnetId": "subnet-fae8c380"
 },
 {
 "InstanceType": "c5.4xlarge",
 "SubnetId": "subnet-e7188bab"
 },
 {
 "InstanceType": "c5.4xlarge",
 "SubnetId": "subnet-49e41922"
 },
 {
 "InstanceType": "c5d.4xlarge",
 "SubnetId": "subnet-fae8c380"
 }
]
 }
]
}
```

```
 "InstanceType": "c5d.4xlarge",
 "SubnetId": "subnet-e7188bab"
 },
 {
 "InstanceType": "c5d.4xlarge",
 "SubnetId": "subnet-49e41922"
 },
 {
 "InstanceType": "m5.4xlarge",
 "SubnetId": "subnet-fae8c380"
 },
 {
 "InstanceType": "m5.4xlarge",
 "SubnetId": "subnet-e7188bab"
 },
 {
 "InstanceType": "m5.4xlarge",
 "SubnetId": "subnet-49e41922"
 },
 {
 "InstanceType": "m5d.4xlarge",
 "SubnetId": "subnet-fae8c380"
 },
 {
 "InstanceType": "m5d.4xlarge",
 "SubnetId": "subnet-e7188bab"
 },
 {
 "InstanceType": "m5d.4xlarge",
 "SubnetId": "subnet-49e41922"
 }
]
}
],
"TargetCapacitySpecification": {
 "TotalTargetCapacity": 20,
 "DefaultTargetCapacityType": "spot"
},
"Type": "instant"
}
```

範例 7：僅當可在單個可用區域中啟動相同執行個體類型的最低目標容量時，才能啟動 Spot 執行個體

只有當最小目標容量可以在單一可用區域中以單一執行個體類型啟動時，您才能設定叢集啟動執行個體，方法是將 Spot 選項 `MinTargetCapacity` 設定為要與 `SingleInstanceType` 和選 `SingleAvailabilityZone` 項一起啟動的最小目標容量。

覆寫啟動範本的 12 個啟動規格擁有不同執行個體類型和子網 (每個都位於不同的可用區域)，但權重容量相同。目標容量總計和最小目標容量都設定為 20 個執行個體，預設的購買選項為 Spot，Spot 配置策略為容量最佳化、真實且 `SingleInstanceType` `SingleAvailabilityZone` 為真。只有在能同時啟動全部 20 個執行個體時，EC2 Fleet 才能使用啟動規格，從具有最佳容量的 Spot 容量集區啟動全部在單一可用區域中，相同執行個體類型的 20 個 Spot 執行個體。

```
{
 "SpotOptions": {
 "AllocationStrategy": "capacity-optimized",
 "SingleInstanceType": true,
 "SingleAvailabilityZone": true,
 "MinTargetCapacity": 20
 },
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "ec2-fleet-lt1",
 "Version": "$Latest"
 },
 "Overrides": [
 {
 "InstanceType": "c5.4xlarge",
 "SubnetId": "subnet-fae8c380"
 },
 {
 "InstanceType": "c5.4xlarge",
 "SubnetId": "subnet-e7188bab"
 },
 {
 "InstanceType": "c5.4xlarge",
 "SubnetId": "subnet-49e41922"
 },
 {
 "InstanceType": "c5d.4xlarge",
 "SubnetId": "subnet-fae8c380"
 }
]
 }
]
}
```

```
 "InstanceType": "c5d.4xlarge",
 "SubnetId": "subnet-e7188bab"
 },
 {
 "InstanceType": "c5d.4xlarge",
 "SubnetId": "subnet-49e41922"
 },
 {
 "InstanceType": "m5.4xlarge",
 "SubnetId": "subnet-fae8c380"
 },
 {
 "InstanceType": "m5.4xlarge",
 "SubnetId": "subnet-e7188bab"
 },
 {
 "InstanceType": "m5.4xlarge",
 "SubnetId": "subnet-49e41922"
 },
 {
 "InstanceType": "m5d.4xlarge",
 "SubnetId": "subnet-fae8c380"
 },
 {
 "InstanceType": "m5d.4xlarge",
 "SubnetId": "subnet-e7188bab"
 },
 {
 "InstanceType": "m5d.4xlarge",
 "SubnetId": "subnet-49e41922"
 }
]
}
],
"TargetCapacitySpecification": {
 "TotalTargetCapacity": 20,
 "DefaultTargetCapacityType": "spot"
},
"Type": "instant"
}
```



## 範例 8：使用多個啟動範本來啟動執行個體

您可以透過指定多個啟動範本，將機群設定為針對不同執行個體類型或執行個體類型群組，啟動具有不同啟動規格的執行個體。在這個範例中，我們希望針對不同執行個體類型，擁有不同 EBS 磁碟區大小，並且我們在啟動範本 `ec2-fleet-lt-4xl`、`ec2-fleet-lt-9xl` 和 `ec2-fleet-lt-18xl` 中進行設定。

在此範例中，我們根據大小，針對 3 個執行個體類型使用 3 種不同的啟動範本。所有啟動範本上的啟動規格覆寫會根據執行個體類型上的 vCPU，使用執行個體權重。總目標容量為 144 個單位，預設購買選項為 Spot，Spot 分配策略為容量最佳化。EC2 Fleet 可以使用啟動範本 `ec2-fleet-4xl` 來啟動 9 個 `c5n.4xlarge` (144 除以 16)，或使用啟動範本 `ec2-fleet-9xl` 來啟動 4 個 `c5n.9xlarge` (144 除以 36)，或使用啟動範本 `ec2-fleet-18xl` 來啟動 2 個 `c5n.18xlarge` (144 除以 72)，或基於容量最佳化分配策略，權重相加達到所需容量的執行個體類型的混合。

```
{
 "SpotOptions": {
 "AllocationStrategy": "capacity-optimized"
 },
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "ec2-fleet-lt-18xl",
 "Version": "$Latest"
 },
 "Overrides": [
 {
 "InstanceType": "c5n.18xlarge",
 "SubnetId": "subnet-fae8c380",
 "WeightedCapacity": 72
 },
 {
 "InstanceType": "c5n.18xlarge",
 "SubnetId": "subnet-e7188bab",
 "WeightedCapacity": 72
 },
 {
 "InstanceType": "c5n.18xlarge",
 "SubnetId": "subnet-49e41922",
 "WeightedCapacity": 72
 }
]
 }
]
}
```

```
"LaunchTemplateSpecification":{
 "LaunchTemplateName":"ec2-fleet-lt-9x1",
 "Version":"$Latest"
},
"Overrides":[
 {
 "InstanceType":"c5n.9xlarge",
 "SubnetId":"subnet-fae8c380",
 "WeightedCapacity":36
 },
 {
 "InstanceType":"c5n.9xlarge",
 "SubnetId":"subnet-e7188bab",
 "WeightedCapacity":36
 },
 {
 "InstanceType":"c5n.9xlarge",
 "SubnetId":"subnet-49e41922",
 "WeightedCapacity":36
 }
]
},
{
 "LaunchTemplateSpecification":{
 "LaunchTemplateName":"ec2-fleet-lt-4x1",
 "Version":"$Latest"
 },
 "Overrides":[
 {
 "InstanceType":"c5n.4xlarge",
 "SubnetId":"subnet-fae8c380",
 "WeightedCapacity":16
 },
 {
 "InstanceType":"c5n.4xlarge",
 "SubnetId":"subnet-e7188bab",
 "WeightedCapacity":16
 },
 {
 "InstanceType":"c5n.4xlarge",
 "SubnetId":"subnet-49e41922",
 "WeightedCapacity":16
 }
]
}
```

```

 }
],
 "TargetCapacitySpecification": {
 "TotalTargetCapacity": 144,
 "DefaultTargetCapacityType": "spot"
 },
 "Type": "instant"
}

```

### 範例 9：啟動基於隨需執行個體的 Spot 執行個體

下列範例指定機群 20 個執行個體的總目標容量，以及 5 個隨需執行個體的目標容量。預設購買選項為 Spot。機群依照指定啟動 5 個隨需執行個體，但需要再啟動 15 個執行個體才能實現總目標容量。差異的購買選項計算方式為  $\text{TotalTargetCapacity} - \text{OnDemandTargetCapacity} = \text{DefaultTargetCapacityType}$ ，這會導致叢集啟動 15 個 Spot 執行個體，根據容量最佳化的配置策略，形成 12 個 Spot 容量集區之一。

```

{
 "SpotOptions": {
 "AllocationStrategy": "capacity-optimized"
 },
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "ec2-fleet-1t1",
 "Version": "$Latest"
 },
 "Overrides": [
 {
 "InstanceType": "c5.large",
 "SubnetId": "subnet-fae8c380"
 },
 {
 "InstanceType": "c5.large",
 "SubnetId": "subnet-e7188bab"
 },
 {
 "InstanceType": "c5.large",
 "SubnetId": "subnet-49e41922"
 },
 {
 "InstanceType": "c5d.large",
 "SubnetId": "subnet-fae8c380"
 }
]
 }
]
}

```

```
 },
 {
 "InstanceType": "c5d.large",
 "SubnetId": "subnet-e7188bab"
 },
 {
 "InstanceType": "c5d.large",
 "SubnetId": "subnet-49e41922"
 },
 {
 "InstanceType": "m5.large",
 "SubnetId": "subnet-fae8c380"
 },
 {
 "InstanceType": "m5.large",
 "SubnetId": "subnet-e7188bab"
 },
 {
 "InstanceType": "m5.large",
 "SubnetId": "subnet-49e41922"
 },
 {
 "InstanceType": "m5d.large",
 "SubnetId": "subnet-fae8c380"
 },
 {
 "InstanceType": "m5d.large",
 "SubnetId": "subnet-e7188bab"
 },
 {
 "InstanceType": "m5d.large",
 "SubnetId": "subnet-49e41922"
 }
]
}
],
"TargetCapacitySpecification": {
 "TotalTargetCapacity": 20,
 "OnDemandTargetCapacity": 5,
 "DefaultTargetCapacityType": "spot"
},
"Type": "instant"
}
```

## 範例 10：基於使用容量預留和優先順序分配策略的隨需執行個體，使用容量最佳化分配策略啟動 Spot 執行個體

您可以將容量保留的使用策略設定為，將叢集設定為在啟動具有預設目標容量類型為 Spot 的隨需執行個體基礎時，先使用隨需容量保留 `use-capacity-reservations-first`。因此，如果多個執行個體集區有未使用的容量預留，則會套用所選擇的隨需分配策略。在此範例中，隨需分配策略為優先順序。

在此範例中，有 6 個未使用的容量預留可供使用。這小於機群的目標隨需容量 10 個隨需執行個體。

帳戶在 2 個集區中有以下 6 個未使用的容量預留。每個集區中的容量保留數會以 `AvailableInstanceCount` 表示。

```
{
 "CapacityReservationId": "cr-111",
 "InstanceType": "m5.large",
 "InstancePlatform": "Linux/UNIX",
 "AvailabilityZone": "us-east-1a",
 "AvailableInstanceCount": 3,
 "InstanceMatchCriteria": "open",
 "State": "active"
}

{
 "CapacityReservationId": "cr-222",
 "InstanceType": "c5.large",
 "InstancePlatform": "Linux/UNIX",
 "AvailabilityZone": "us-east-1a",
 "AvailableInstanceCount": 3,
 "InstanceMatchCriteria": "open",
 "State": "active"
}
```

下列機群組態僅顯示此範例的相關組態。隨需配置策略的優先順序，容量保留的使用策略為 `use-capacity-reservations-firstSpot` 分配策略是容量最佳化。目標總容量為 20、隨需目標容量為 10、預設目標容量類型為 `spot`。

```
{
 "SpotOptions": {
 "AllocationStrategy": "capacity-optimized"
 },
 "OnDemandOptions": {
 "CapacityReservationOptions": {
```

```
 "UsageStrategy": "use-capacity-reservations-first"
 },
 "AllocationStrategy": "prioritized"
},
"LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "ec2-fleet-1t1",
 "Version": "$Latest"
 },
 "Overrides": [
 {
 "InstanceType": "c5.large",
 "SubnetId": "subnet-fae8c380",
 "Priority": 1.0
 },
 {
 "InstanceType": "c5.large",
 "SubnetId": "subnet-e7188bab",
 "Priority": 2.0
 },
 {
 "InstanceType": "c5.large",
 "SubnetId": "subnet-49e41922",
 "Priority": 3.0
 },
 {
 "InstanceType": "c5d.large",
 "SubnetId": "subnet-fae8c380",
 "Priority": 4.0
 },
 {
 "InstanceType": "c5d.large",
 "SubnetId": "subnet-e7188bab",
 "Priority": 5.0
 },
 {
 "InstanceType": "c5d.large",
 "SubnetId": "subnet-49e41922",
 "Priority": 6.0
 },
 {
 "InstanceType": "m5.large",
 "SubnetId": "subnet-fae8c380",
```

```

 "Priority": 7.0
 },
 {
 "InstanceType": "m5.large",
 "SubnetId": "subnet-e7188bab",
 "Priority": 8.0
 },
 {
 "InstanceType": "m5.large",
 "SubnetId": "subnet-49e41922",
 "Priority": 9.0
 },
 {
 "InstanceType": "m5d.large",
 "SubnetId": "subnet-fae8c380",
 "Priority": 10.0
 },
 {
 "InstanceType": "m5d.large",
 "SubnetId": "subnet-e7188bab",
 "Priority": 11.0
 },
 {
 "InstanceType": "m5d.large",
 "SubnetId": "subnet-49e41922",
 "Priority": 12.0
 }
]
}
],
"TargetCapacitySpecification": {
 "TotalTargetCapacity": 20,
 "OnDemandTargetCapacity": 10,
 "DefaultTargetCapacityType": "spot"
},
"Type": "instant"
}

```

使用上述組態建立 instant 類型機群之後，會啟動以下 20 個執行個體以符合目標容量：

- us-east-1a 中的 7 個 c5.large 隨需執行個體 – us-east-1a 中的 c5.large 是第一優先順序，並有 3 個未使用 c5.large 容量預留可供使用。容量預留會先用來啟動 3 個隨需執行個體，再根據隨需分配策略啟動額外的 4 個隨需執行個體，即此範例中的優先順序。

- us-east-1a 中的 3 個 m5.large 隨需執行個體 – us-east-1a 中的 m5.large 是第二優先順序，並有 3 個未使用 c3.large 容量預留可供使用。
- 來自 12 個 Spot 容量集區中的 10 個 Spot 執行個體，該集區根據容量最佳化的分配策略具有最佳容量。

啟動機群後，您可以執行 [describe-capacity-reservations](#)，以查看還剩餘多少個未使用的 容量預留。在此範例中，您應該會看到下列回應，其中顯示了已使用的所有 c5.large 和 m5.large 容量預留。

```
{
 "CapacityReservationId": "cr-111",
 "InstanceType": "m5.large",
 "AvailableInstanceCount": 0
}

{
 "CapacityReservationId": "cr-222",
 "InstanceType": "c5.large",
 "AvailableInstanceCount": 0
}
```

#### 範例 11：使用 capacity-optimized-prioritized 配置策略啟動競價型執行個體

下列範例指定 instant 類型 EC2 Fleet 中所需的參數：啟動範本、目標容量、預設購買選項，以及啟動範本覆寫。啟動範本由其啟動範本名稱和版本編號識別。覆寫啟動範本的 12 個啟動規格擁有已指派優先順序的 4 個不同執行個體類型和 3 個不同的子網，每個都位於不同的可用區域。叢集的目標容量為 20 個執行個體，而預設的購買選項為 Spot，這會導致叢集嘗試根據 capacity-optimized-prioritized 配置策略從 12 個 Spot 容量集區中的一個啟動 20 個 Spot 執行個體，這些執行個體會盡最大努力實作優先順序，但會先針對容量進行最佳化。

```
{
 "SpotOptions": {
 "AllocationStrategy": "capacity-optimized-prioritized"
 },
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "ec2-fleet-lt1",
 "Version": "$Latest"
 },
 "Overrides": [
```



```
{
 "InstanceType": "c5.large",
 "SubnetId": "subnet-fae8c380",
 "Priority": 1.0
},
{
 "InstanceType": "c5.large",
 "SubnetId": "subnet-e7188bab",
 "Priority": 1.0
},
{
 "InstanceType": "c5.large",
 "SubnetId": "subnet-49e41922",
 "Priority": 1.0
},
{
 "InstanceType": "c5d.large",
 "SubnetId": "subnet-fae8c380",
 "Priority": 2.0
},
{
 "InstanceType": "c5d.large",
 "SubnetId": "subnet-e7188bab",
 "Priority": 2.0
},
{
 "InstanceType": "c5d.large",
 "SubnetId": "subnet-49e41922",
 "Priority": 2.0
},
{
 "InstanceType": "m5.large",
 "SubnetId": "subnet-fae8c380",
 "Priority": 3.0
},
{
 "InstanceType": "m5.large",
 "SubnetId": "subnet-e7188bab",
 "Priority": 3.0
},
{
 "InstanceType": "m5.large",
 "SubnetId": "subnet-49e41922",
 "Priority": 3.0
}
```

```
 },
 {
 "InstanceType": "m5d.large",
 "SubnetId": "subnet-fae8c380",
 "Priority": 4.0
 },
 {
 "InstanceType": "m5d.large",
 "SubnetId": "subnet-e7188bab",
 "Priority": 4.0
 },
 {
 "InstanceType": "m5d.large",
 "SubnetId": "subnet-49e41922",
 "Priority": 4.0
 }
]
}
],
"TargetCapacitySpecification": {
 "TotalTargetCapacity": 20,
 "DefaultTargetCapacityType": "spot"
},
"Type": "instant"
}
```

## EC2 Fleet 組態策略

一個 EC2 Fleet 即為一個群組的 隨需執行個體 和 競價型執行個體。EC2 Fleet 也可以是一組容量區塊執行個體。

### 隨需執行個體和 Spot 執行個體

EC2 Fleet 會嘗試啟動所需的執行個體數目，以符合您在機群請求中指定的目標容量。機群的組成只能是僅 隨需執行個體、僅 競價型執行個體，或 隨需執行個體 和 競價型執行個體的組合。如果有可用容量且您的請求中每個小時的價格上限超過 Spot 價格，則會實現 競價型執行個體的請求。如果您的 競價型執行個體 中斷，叢集也會嘗試維持其目標容量。

您也可以設定您願意為機群支付的每個小時金額上限，而 EC2 Fleet 會啟動執行個體，直到達到金額上限為止。達到您願意支付的金額上限時，機群會停止啟動執行個體，即使它未符合目標容量亦然。

Spot 容量集區為一組具有相同執行個體類型和可用區域的未使用 EC2 執行個體。當您建立 EC2 Fleet 時可以包含多個啟動規格，這些規格因執行個體類型、可用區域、子網和最高價格而異。根據請求中包

含的啟動規格以及請求的組態，機群會選取用於實現請求的 Spot 容量集區。競價型執行個體來自選取的集區。

EC2 Fleet 使您能夠根據核心或執行個體數量或記憶體量來佈建適用於您應用程式的大量 EC2 容量。例如，您可以指定 EC2 Fleet 啟動 200 個執行個體的目標容量，其中 130 個是隨需執行個體，剩餘的則是競價型執行個體。

### 容量區塊執行個體

ML 的容量區塊可讓您為日後預留 GPU 執行個體，以支援短時間的機器學習 (ML) 工作負載。在容量區塊中執行的執行個體會自動放置在附近的 [Amazon EC2](#) 中 UltraClusters。如需容量區塊的詳細資訊，請參閱 [ML 的容量區塊](#)。

使用適當的組態策略來建立滿足您需求的 EC2 Fleet。

### 目錄

- [規劃 EC2 Fleet](#)
- [競價型執行個體的分配策略](#)
- [EC2 Fleet 的屬性型執行個體類型選擇](#)
- [針對隨需備份設定 EC2 Fleet](#)
- [容量重新平衡](#)
- [最高價格覆寫](#)
- [控制花費](#)
- [EC2 Fleet 執行個體權重](#)

## 規劃 EC2 Fleet

當您規劃 EC2 Fleet 時，建議您執行下列動作：

- 確定是否要建立提交所需目標容量之同步或非同步一次性請求的 EC2 Fleet，或者隨時間維持目標容量。如需詳細資訊，請參閱 [EC2 Fleet 請求類型](#)。
- 決定滿足您應用程式需求的執行個體類型。
- 如果您打算將競價型執行個體包含在 EC2 Fleet，請在您建立機群前檢閱 [Spot 最佳實務](#)。在您規劃機群時使用這些最佳實務，讓您能以最低價格佈建執行個體。
- 為您的 EC2 Fleet 決定目標容量。您可在執行個體中或在自訂單位中設定目標容量。如需詳細資訊，請參閱 [EC2 Fleet 執行個體權重](#)。

- 決定 EC2 Fleet 目標容量的那些部分必須為隨需容量和 Spot 容量。您可以為隨需容量或 Spot 容量 (或兩者) 指定 0。
- 指定您每單位的價格，如您使用執行個體權重。若要計算每單位價格，請將每個執行個體小時的價格除以此執行個體代表的單位數 (或權重)。如您不使用執行個體權重，預設的每單位價格即為每執行個體小時。
- 決定您願意為機群支付的每個小時金額上限。如需詳細資訊，請參閱 [控制花費](#)。
- 檢視您 EC2 Fleet 的可能選項。如需機群參數的相關資訊，請參閱 AWS CLI 命令參考中的 [create-fleet](#)。如需 EC2 Fleet 組態的範例，請參閱 [EC2 Fleet 範例組態](#)。

## 競價型執行個體的分配策略

您的啟動組態會決定 EC2 機群可從中啟動 Spot 執行個體的所有可能 Spot 容量集區 (執行個體類型和可用區域)。但是，啟動執行個體時，EC2 機群會使用您指定的配置策略，從所有可能的集區中挑選特定的集區。

### Note

(僅限 Linux 執行個體) 如果您將競價型執行個體設定為在 [AMD SEV-SNP](#) 開啟的情況下啟動，則系統會變更額外的小時使用費，相當於所選執行個體類型的 [隨需小時費率](#) 10%。如果配置策略使用價格作為輸入，則 EC2 機群不會包含此額外費用；僅使用 Spot 價格。

## 分配策略

您可以為 Spot 執行個體指定下列其中一個配置策略：

### price-capacity-optimized(建議使用)

EC2 機群會識別擁有啟動中執行個體數目之最高容量可用性的集區。這表示我們將從我們認為在短期內中斷機會最低的集區請求 Spot 執行個體。EC2 機群接著會從這些最低價集區中請求 Spot 執行個體。

price-capacity-optimized 分配策略是大多數 Spot 工作負載的最佳選擇，例如無狀態容器化應用程式、微服務、Web 應用程式、資料和分析作業以及批次處理。

### capacity-optimized

EC2 機群會識別擁有啟動中執行個體數目之最高容量可用性的集區。這表示我們將從我們認為在短期內中斷機會最低的集區請求 Spot 執行個體。您可以使用 capacity-optimized-

`prioritized` 選擇性地設定機群中每個執行個體類型的優先順序。EC2 Fleet 會首先最佳化容量，但會盡力實現執行個體類型的優先順序。

使用 競價型執行個體，定價會依據長期的供需趨勢隨時間慢慢改變，但是容量會即時波動。`capacity-optimized` 策略會查看即時容量資料並預測哪些資料為最多可用，從而自動將 Spot 執行個體啟動到最可用的集區。這適用於可能具有較高的重新啟動工作中斷成本的工作負載，例如持續整合 (CI)、影像和媒體轉譯、深度學習以及可能具有較高的重新啟動工作中斷成本的高效能運算 (HPC) 工作負載。`capacity-optimized` 策略可實現盡可能少的中斷，因此可降低您工作負載的整體成本。

或者，您可以使用具有優先順序參數的 `capacity-optimized-prioritized` 分配策略，以從最高到最低的優先順序對執行個體類型排序。您可以為不同的執行個體類型設定相同的優先順序。EC2 Fleet 將首先針對容量進行最佳化，但會盡力實現執行個體類型的優先順序 (例如，若實現優先順序不會顯著影響 EC2 Fleet 佈建最佳容量的能力)。對於必須將中斷可能性降至最低的工作負載來說，這是一個很好的選擇，而且某些執行個體類型的偏好也很重要。請注意，當您設定 `capacity-optimized-prioritized` 的優先順序時，如果隨需執行個體 `AllocationStrategy` 設定為 `prioritized`，則相同的優先順序也會套用至您的隨需執行個體。

`diversified`

競價型執行個體 會分佈在所有 Spot 容量集區。

`lowest-price`(不建議使用)

#### Warning

我們不建議使用 `lowest-price` 配置策略，因為它具有最高的 Spot 執行個體中斷風險。

Spot 執行個體來自價格最低且還有可用容量的集區。這是預設策略。不過，建議您透過指定 `price-capacity-optimized` 分配策略來覆寫預設。

如果價格最低的集區沒有可用容量，Spot 執行個體來自下一個價格最低且還有可用容量的集區。

如果集區在滿足您的所需容量之前用完容量，EC2 機群將繼續從下一個價格最低的集區中選取集區來滿足您的請求。為確保達到所需的容量，您可能會從數個集區中接收 Spot 執行個體。

由於此策略僅考慮執行個體價格，而不考慮容量可用性，因此可能會導致高中斷率。

## InstancePoolsToUseCount

要配置目標 Spot 容量的 Spot 集區數量。僅在配置策略設定為 lowest-price 時有效。EC2 機群會選取價格最低的 Spot 集區，並將目標 Spot 容量平均配置到您指定數量的 Spot 集區上。

請注意，EC2 Fleet 會盡最大努力從您指定的集區數量中提取 Spot 執行個體。如果集區在滿足您的目標容量之前用完 Spot 容量，則 EC2 機群將透過從下一個價格最低的集區中進行提取來繼續滿足您的請求。為確保達到目標容量，您可能會從超過您指定的集區數量中接收 Spot 執行個體。同樣地，如果大多數集區沒有 Spot 容量，您可能會從少於您指定的集區數量中接收完整目標容量。

### 選擇適當的分配策略

您可以選擇適當的 Spot 分配策略，進而針對您的使用案例最佳化機群。對於隨需執行個體目標容量，EC2 機群一律會根據公有隨需價格選取最低價的執行個體類型，同時遵循 Spot 執行個體的分配策略 – price-capacity-optimized、capacity-optimized、diversified 或 lowest-price。

### 平衡最低價格與容量可用性

為了平衡價格最低的 Spot 容量集區與具有最高容量可用性的 Spot 容量集區之間的利弊，建議您使用 price-capacity-optimized 分配策略。此策略會根據集區的價格和這些集區中 Spot 執行個體的容量可用性，決定要從哪些集區請求 Spot 執行個體的集區。這表示我們將從我們認為在短期內中斷機會最低的集區請求 Spot 執行個體，同時我們仍會將價格納入考慮。

如果您的機群執行彈性且無狀態的工作負載，包括容器化應用程式、微服務、Web 應用程式、資料和分析作業以及批次處理，則使用 price-capacity-optimized 分配策略，以達到最佳的成本節省和容量可用性。

如果您的機群執行的工作負載有較高的重新啟動工作中斷的成本，則您應該實作檢查點作業，以便應用程式可以在中斷時從該點重新啟動。透過使用檢查點作業，您可以讓 price-capacity-optimized 分配策略非常適合這些工作負載，因為其可以從價格最低的集區中分配容量，而這些集區也會提供較低的 Spot 執行個體中斷率。

如需使用 price-capacity-optimized 分配策略的範例組態，請參閱 [範例 10：在 price-capacity-optimized 叢集中啟動 Spot 執行個體](#)。

### 當工作負載具有較高的中斷成本時

如果您執行的工作負載使用價格相似的執行個體類型，或是中斷成本非常之高，以致於任何成本節省都不足以彌補中斷的邊際增加，則您可以選擇是否使用該 capacity-optimized 策略。該策略可從最可用的 Spot 容量集區分配容量，而這些集區會提供較少中斷的可能性，進而降低您工作負載的整體成

本。如需使用 capacity-optimized 分配策略的範例組態，請參閱 [範例 8：在容量最佳化叢集中啟動 Spot 執行個體](#)。

當必須將中斷可能性降至最低，而且對於某些執行個體類型的偏好也很重要時，您可以使用 capacity-optimized-prioritized 分配策略，然後設定執行個體類型的順序，來表現您的集區優先順序，以從最高到最低優先順序使用。如需範例組態，請參閱 [範例 9：在具有優先順序的容量最佳化叢集中啟動 Spot 執行個體](#)。

請注意，當您設定 capacity-optimized-prioritized 的優先順序時，如果隨需執行個體 AllocationStrategy 設定為 prioritized，則相同的優先順序也會套用至您的隨需執行個體。

當您的工作負載具有時間彈性且容量可用性並非相關因素時

如果您的機群很小或執行時間很短，您仍可在考慮容量可用性的同時，使用 price-capacity-optimized 來實現成本節省最大化。

當您的機群很龐大或執行時間較長時

如果您的機群很龐大或執行時間較長，則可以使用 diversified 策略，透過將多個競價型執行個體分散到多個集區中來提高機群的可用性。例如，如果 EC2 Fleet 指定了 10 個集區和 100 個執行個體的目標容量，則該機群會在每個集區中啟動 10 個競價型執行個體。如果某個集區的 Spot 價格超過該集區的最高價格，則只有 10% 的機群會受到影響。隨著時間，使用這種策略也會使您的機群對任何一個集區 Spot 價格上漲較不敏感。藉由 diversified 策略，EC2 Fleet 不會將競價型執行個體啟動至任何等於或高於 [隨需價格](#) 的 Spot 價格集區中。

維護目標容量

在競價型執行個體因 Spot 價格或 Spot 容量集區可用容量的變更而終止之後，類型為 maintain 的 EC2 Fleet 會啟動替代競價型執行個體。分配策略會確定從中啟動替代執行個體的集區，如下所示：

- 如果分配策略是 price-capacity-optimized，則機群會在擁有最多 Spot 執行個體容量可用性的集區中啟動替代執行個體，同時將價格納入考慮並識別價格最低且具有高容量可用性的執行個體。
- 如果分配策略為 capacity-optimized，則機群會在擁有最多 Spot 執行個體容量可用性的集區中啟動替代執行個體。
- 如果分配策略為 diversified，則機群會將替代競價型執行個體分配到剩餘的集區中。

## EC2 Fleet 的屬性型執行個體類型選擇

建立 EC2 Fleet 時，您必須指定一或多個執行個體類型，以在機群中設定隨需執行個體和 Spot 執行個體。做為手動指定執行個體類型的替代方式，您可以指定執行個體必須具有的屬性，然後 Amazon

EC2 會識別具有這些屬性的所有執行個體類型。這就是所謂的屬性型執行個體類型選項。例如，您可以指定執行個體所需的 vCPU 數目下限和上限，然後 EC2 Fleet 將使用符合這些 vCPU 需求的任何可用執行個體類型來啟動執行個體。

屬性型執行個體類型選擇非常適合可對其使用哪些執行個體類型具有彈性的工作負載和架構，例如執行容器或 Web 機群、處理大數據，以及實作持續整合和部署 (CI/CD) 工具時。

## 優勢

屬性型執行個體類型選擇具有下列優勢：

- 輕鬆使用正確的執行個體類型 — 有這麼多可用的執行個體類型，因此尋找適合您工作負載的執行個體類型可能非常耗時。當您指定執行個體類型屬性時，執行個體類型會自動具有工作負載所需的屬性。
- 簡化的組態 — 若要為 EC2 叢集手動指定多個執行個體類型，您必須為每個執行個體類型建立個別的啟動範本覆寫。但是，有了屬性型執行個體類型選擇，若要提供多個執行個體類型，您只需要在啟動範本或在啟動範本覆寫中指定執行個體屬性。
- 自動使用新的執行個體類型 — 當您指定執行個體屬性而非執行個體類型時，您的叢集可以在發行時使用新一代執行個體類型，即「future 校對」叢集的配置。
- 執行個體類型彈性 — 當您指定執行個體屬性而非執行個體類型時，EC2 Fleet 可以從各種執行個體類型中選取以啟動 Spot 執行個體，這符合 [Spot 執行個體類型彈性的最佳實務](#)。

## 主題

- [屬性型執行個體類型選擇的運作方式](#)
- [價格保護](#)
- [考量事項](#)
- [建立具有屬性型執行個體類型選擇的 EC2 Fleet](#)
- [有效和無效的組態範例](#)
- [預覽具有指定屬性的執行個體類型](#)

## 屬性型執行個體類型選擇的運作方式

若要在機群組態中使用屬性型執行個體類型選擇，請將執行個體類型清單取代為執行個體所需的執行個體屬性清單。EC2 Fleet 將在具有指定執行個體屬性的任何可用執行個體類型上啟動執行個體。

## 主題



- [執行個體屬性的類型](#)
- [要在何處設定屬性型執行個體類型選擇](#)
- [EC2 Fleet 在佈建機群時如何使用屬性型執行個體類型選擇](#)

## 執行個體屬性的類型

您可以指定數個執行個體屬性來表示運算需求，例如：

- vCPU 計數 — 每個執行個體的 vCPUs 數目下限和上限。
- 記憶體 — 每個執行個體 GiBs 的記憶體下限和最大記憶體。
- 本機儲存 — 是否使用 EBS 或執行個體儲存磁碟區進行本機儲存。
- 高載效能 — 是否使用 T 執行個體系列，包括 T4G、T3A、T3 和 T2 類型。

如需每個屬性和預設值的說明，請參閱 Amazon EC2 API 參考 [InstanceRequirements](#) 中的。

## 要在何處設定屬性型執行個體類型選擇

根據您使用主控台還是 AWS CLI，您可以指定以屬性為基礎之執行個體類型選取的執行個體屬性，如下所示：

在主控台中，您可以在下列機群組態元件中指定執行個體屬性：

- 在啟動範本中，則參考機群請求中的啟動範本

在中 AWS CLI，您可以在下列一或所有叢集組態元件中指定執行個體屬性：

- 在啟動範本中，則參考機群請求中的啟動範本
- 在啟動範本覆寫中

如果想要混合使用不同 AMI 的執行個體，您可以在多個啟動範本覆寫中指定執行個體屬性。例如，不同的執行個體類型可以使用 x86 和 ARM 型處理器。

## EC2 Fleet 在佈建機群時如何使用屬性型執行個體類型選擇

EC2 Fleet 以下列方式佈建機群：

- EC2 Fleet 可識別具有所指定屬性的執行個體類型。
- EC2 Fleet 會使用價格保護來決定要排除哪些執行個體類型。

- EC2 叢集會根據具有相符執行個體類型的區域或可用區域，決定 AWS 要考慮從中啟動執行個體的容量集區。
- EC2 Fleet 會套用指定的分配策略，以決定要從哪些容量集區中啟動執行個體。

請注意，屬性型執行個體類型選擇不會挑選要從中佈建機群的容量集區；這是分配策略的任務。

如果您指定分配策略，EC2 Fleet 將根據指定的分配策略啟動執行個體。

- 若為 Spot 執行個體，屬性型執行個體類型選擇支援 price-capacity-optimized、capacity-optimized 和 lowest-price 分配策略。請注意，我們不建議使用 lowest-price Spot 配置策略，因為 Spot 執行個體的中斷風險最高。
- 若為隨需執行個體，屬性型執行個體類型選擇支援 lowest-price 分配策略。
- 如果具有所指定執行個體屬性的執行個體類型沒有容量，則無法啟動任何執行個體，而且機群會傳回錯誤。

## 價格保護

價格保護是一項功能，可防止 EC2 Fleet 使用您認為過於昂貴的執行個體類型，即使它們剛好符合您指定的屬性也一樣。要使用價格保護，請設置價格閾值。然後，當 Amazon EC2 選取具有屬性的執行個體類型時，就會排除定價高於閾值的執行個體類型。

Amazon EC2 計算價格閾值的方式如下：

- Amazon EC2 會先從符合屬性的執行個體類型中識別出價格最低的執行個體類型。
- 然後，Amazon EC2 會採用您為價格保護參數指定的值（以百分比表示），並將其與已識別執行個體類型的價格相乘。結果是用作價格閾值的價格。

隨需執行個體和 Spot 執行個體有不同的價格門檻。

當您使用以屬性為基礎的執行個體類型選項建立叢集時，預設會啟用價格保護。您可以保留預設值，也可以指定自己的值。

您也可以關閉價格保護。若要指出沒有價格保護閾值，請指定較高的百分比值，例如999999。

## 主題

- [如何識別最低價執行個體類型](#)
- [隨需執行個體價格保](#)
- [競價型實例價格保護](#)

## • [指定價格保護門檻](#)

### 如何識別最低價執行個體類型

Amazon EC2 透過識別與指定屬性相符的執行個體類型中最低價格的執行個體類型，來決定要依據價格閾值的價格。它通過以下方式執行此操作：

- 它首先查看與您的屬性匹配的當前一代 C，M 或 R 實例類型。如果找到任何相符項目，就會識別價格最低的執行個體類型。
- 如果沒有相符項目，則會查看符合屬性的任何目前產生的執行個體類型。如果找到任何相符項目，就會識別價格最低的執行個體類型。
- 如果沒有相符項目，就會查看符合屬性的任何上一代執行個體類型，並識別價格最低的執行個體類型。

### 隨需執行個體價格保

隨需執行個體類型的價格保護閾值是以比識別的最低價隨需執行個體類型

(*OnDemandMaxPricePercentageOverLowestPrice*) 高的百分比來計算。您可以指定您願意支付的更高百分比。如果您未指定此參數，則系統會使用預設值來計算比識別價格高出 20% 的價格保護閾值。20

例如，如果識別的隨需執行個體價格為且您指定25，則價格閾值會高出 25% 0.4271。0.4271它的計算方式如下： $0.4271 * 1.25 = 0.533875$ 計算的價格是您願意為隨需執行個體支付的最高價格，在此範例中，Amazon EC2 將排除任何成本超過的隨需執行個體類型0.533875。

### 競價型實例價格保護

根據預設，Amazon EC2 會自動套用最佳競價型執行個體價格保護，以持續從各種執行個體類型中進行選取。您也可以自行手動設定價格保護。但是，讓 Amazon EC2 為您執行此操作可以提高 Spot 容量滿足的可能性。

您可以使用下列其中一個選項手動指定價格保護。如果您手動設定價格保護，建議您使用第一個選項。

- 已識別最低價格隨需執行個體類型的百分比  
[*MaxSpotPriceAsPercentageOfOptimalOnDemandPrice*]

例如，如果識別的隨需執行個體類型價格為且您指定60，則價格閾值為 60% 0.4271。0.4271它的計算方式如下： $0.4271 * 0.60 = 0.25626$ 計算的價格是您願意為競價型執行個體支付的最高價格，在此範例中，Amazon EC2 將排除任何成本超過的競價型執行個體類型0.25626。

- 高於識別的最低價 Spot 執行個體類型 [**SpotMaxPricePercentageOverLowestPrice**] 的百分比

例如，如果已識別的競價型執行個體類型價格為且您指定 25，則價格閾值高於 25%

0.1808。0.1808 它的計算方式如下： $0.1808 * 1.25 = 0.226$  計算的價格是您願意為競價型執行個體支付的最高價格，在此範例中，Amazon EC2 將排除任何成本超過的競價型執行個體類型 0.266。我們不建議使用此參數，因為現貨價格可能會波動，因此您的價格保護閾值也可能會波動。

## 指定價格保護門檻

### 如要指定價格保護閾值

在建立 EC2 機群時，為屬性型執行個體類型選擇設定機群，然後執行下列作業：

- 如要指定隨需執行個體價格保護閾值，請於 JSON 組態檔案的 InstanceRequirements 結構中，對於 OnDemandMaxPricePercentageOverLowestPrice，以百分比形式輸入價格保護閾值。
- 若要指定競價型執行個體價格保護閾值，請在 JSON 設定檔的 InstanceRequirements 結構中，指定下列其中一個參數：
  - 對於 MaxSpotPriceAsPercentageOfOptimalOnDemandPrice，以百分比輸入價格保護閾值。
  - 對於 SpotMaxPricePercentageOverLowestPrice，以百分比輸入價格保護閾值。

如需有關建立機群的詳細資訊，請參閱 [建立具有屬性型執行個體類型選擇的 EC2 Fleet](#)。

### Note

建立 EC2 機群時，若您將 TargetCapacityUnitType 設定為 vcpu 或 memory-mib，則價格保護閾值會根據每個 vCPU 或每個記憶體的价格進行套用，而非每個執行個體的价格。

## 考量事項

- 您可以在 EC2 Fleet 中指定執行個體類型或執行個體屬性，但不能同時指定兩者。

使用 CLI 時，啟動範本覆寫將覆寫啟動範本。例如，如果啟動範本包含執行個體類型，且啟動範本覆寫包含執行個體屬性，則執行個體屬性所識別的執行個體將覆寫啟動範本中的執行個體類型。

- 使用 CLI 時，若將執行個體屬性指定為覆寫，您也無法指定權重或優先順序。
- 您最多可以在請求組態中指定四個 InstanceRequirements 結構。

## 建立具有屬性型執行個體類型選擇的 EC2 Fleet

您可以使用 AWS CLI，將機群設定為使用屬性型執行個體類型選擇。

若要建立具有屬性型執行個體類型選取的 EC2 機群 (AWS CLI)

使用 [create-fleet](#) (AWS CLI) 命令建立 EC2 Fleet。指定 JSON 檔案中的機群組態。

```
aws ec2 create-fleet \
 --region us-east-1 \
 --cli-input-json file://file_name.json
```

範例 *file\_name*.json 檔案

下列範例包含將 EC2 機群設定為使用屬性型執行個體類型選取的參數，其後是文字說明。

```
{
 "SpotOptions": {
 "AllocationStrategy": "price-capacity-optimized"
 },
 "LaunchTemplateConfigs": [{
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "my-launch-template",
 "Version": "1"
 },
 "Overrides": [{
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 2
 },
 "MemoryMiB": {
 "Min": 4
 }
 }
 }
]
},
 "TargetCapacitySpecification": {
 "TotalTargetCapacity": 20,
 "DefaultTargetCapacityType": "spot"
 },
 "Type": "instant"
}
```

屬性型執行個體類型選取的屬性在 InstanceRequirements 結構中予以指定。在此範例中，會指定兩個屬性：

- VCpuCount – 至少指定 2 個 vCPU。由於未指定最大值，因此沒有最大限制。
- MemoryMiB – 至少指定 4 MiB 的記憶體。由於未指定最大值，因此沒有最大限制。

將會識別任何具有 2 個或更多 vCPU 和 4 MiB 或更多記憶體的執行個體類型。但是，當 [EC2 機群佈建機群](#)時，價格保護和分配策略可能會排除某些執行個體類型。

如需可指定之所有可能屬性的清單和說明，請參閱 Amazon EC2 API 參考 [InstanceRequirements](#) 中的。

#### Note

當 InstanceRequirements 併入機群組態中時，必須排除 InstanceType 和 WeightedCapacity；它們無法與執行個體屬性同時決定機群組態。

JSON 也包含下列機群組態：

- "AllocationStrategy": "*price-capacity-optimized*" – 機群中 Spot 執行個體的分配策略。
- "LaunchTemplateName": "*my-launch-template*", "Version": "*1*" – 啟動範本包含一些執行個體組態資訊，但如果指定了任何執行個體類型，則會由 InstanceRequirements 中指定的屬性進行覆寫。
- "TotalTargetCapacity": *20* – 目標容量為 20 個執行個體。
- "DefaultTargetCapacityType": "*spot*" – 預設容量為 Spot 執行個體。
- "Type": "*instant*" – 機群的要求類型為 instant。

#### 有效和無效的組態範例

如果您使 AWS CLI 用建立 EC2 叢集，則必須確定叢集組態有效。下列範例顯示有效和無效的組態。

當組態包含下列項目時，視為無效：

- 同時具有 Overrides 和 InstanceRequirements 的單一 InstanceType
- 兩個 Overrides 結構，一個具有 InstanceRequirements，而另一個具有 InstanceType

- 兩個在同一 InstanceRequirements 內具有重疊屬性值的 LaunchTemplateSpecification 結構

### 範例組態

- [有效組態：具有覆寫的單一啟動範本](#)
- [有效組態：具有多個的單一啟動範本 InstanceRequirements](#)
- [有效組態：兩個啟動範本，每個都具有覆寫](#)
- [有效組態：僅限指定的 InstanceRequirements，沒有重疊的屬性值](#)
- [組態無效：Overrides 包含 InstanceRequirements 和 InstanceType](#)
- [組態無效：兩個 Overrides 包含 InstanceRequirements 和 InstanceType](#)
- [組態無效：重疊屬性值](#)

有效組態：具有覆寫的單一啟動範本

下列組態有效。其包含一個啟動範本和一個 Overrides 結構 (其中包含一個 InstanceRequirements 結構)。範例組態的文字說明如下。

```
{
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "My-launch-template",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 2,
 "Max": 8
 },
 "MemoryMib": {
 "Min": 0,
 "Max": 10240
 },
 "MemoryGiBPerVCpu": {
 "Max": 10000
 },
 "RequireHibernateSupport": true
 }
 }
]
 }
]
}
```

```

 }
 }
]
},
"TargetCapacitySpecification": {
 "TotalTargetCapacity": 5000,
 "DefaultTargetCapacityType": "spot",
 "TargetCapacityUnitType": "vcpu"
}
}
}

```

## InstanceRequirements

若要使用屬性型執行個體選擇，您必須在組態中包含 InstanceRequirements 結構，並為機群中的執行個體指定所需的屬性。

在上述範例中，指定了下列執行個體屬性：

- VCpuCount – 執行個體類型必須具有最少 2 個，最多 8 個的 vCPU。
- MemoryMiB – 執行個體類型必須具有最多 10240 MiB 的記憶體。最小值若為 0，表示沒有最小限制。
- MemoryGiBPerVCpu – 執行個體類型必須具有每個 vCPU 最多 10,000 MiB 的記憶體。Min 為選用參數。若省略它，表示沒有最小限制。

## TargetCapacityUnitType

TargetCapacityUnitType 參數會指定目標容量的單位。在範例中，目標容量為 5000，而目標容量單位類型為 vcpu，它們一起指定了所需的目標容量，即 5,000 個 vCPU。EC2 Fleet 將啟動足夠的執行個體，以便機群中的 vCPU 總數為 5,000 個 vCPU。

有效組態：具有多個的單一啟動範本 InstanceRequirements

下列組態有效。其包含一個啟動範本和一個 Overrides 結構 (其中包含兩個 InstanceRequirements 結構)。InstanceRequirements 中指定的屬性是有效的，因為這些值不重疊—第一個 InstanceRequirements 結構會指定 0-2 個 vCPU 的 VCpuCount，而第二個 InstanceRequirements 結構則會指定 4-8 個 vCPU。

```
{
```



```
"LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "MyLaunchTemplate",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 0,
 "Max": 2
 },
 "MemoryMiB": {
 "Min": 0
 }
 }
 },
 {
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 4,
 "Max": 8
 },
 "MemoryMiB": {
 "Min": 0
 }
 }
 }
]
 }
],
"TargetCapacitySpecification": {
 "TotalTargetCapacity": 1,
 "DefaultTargetCapacityType": "spot"
}
}
```

有效組態：兩個啟動範本，每個都具有覆寫

下列組態有效。其包含兩個啟動範本，每個都具有一個 `Overrides` 結構 (其中包含一個 `InstanceRequirements` 結構)。這個組態有助於同一個機群中的 `arm` 和 `x86` 架構支援。

```
{
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "armLaunchTemplate",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 0,
 "Max": 2
 },
 "MemoryMiB": {
 "Min": 0
 }
 }
 }
],
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "x86LaunchTemplate",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 0,
 "Max": 2
 },
 "MemoryMiB": {
 "Min": 0
 }
 }
 }
]
 }
],
 "TargetCapacitySpecification": {
 "TotalTargetCapacity": 1,
 "DefaultTargetCapacityType": "spot"
 }
 }
}
```

```

 }
}

```

有效組態：僅限指定的 **InstanceRequirements**，沒有重疊的屬性值

下列組態有效。其包含兩個 `LaunchTemplateSpecification` 結構，每個都有一個啟動範本和一個 `Overrides` 結構 (其中包含一個 `InstanceRequirements` 結構)。 `InstanceRequirements` 中指定的屬性是有效的，因為這些值不重疊—第一個 `InstanceRequirements` 結構會指定 0-2 個 vCPU 的 `VCpuCount`，而第二個 `InstanceRequirements` 結構則會指定 4-8 個 vCPU。

```

{
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "MyLaunchTemplate",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 0,
 "Max": 2
 },
 "MemoryMiB": {
 "Min": 0
 }
 }
 }
]
 },
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "MyOtherLaunchTemplate",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 4,
 "Max": 8
 }
 }
 }
]
 }
]
}

```

```

 "MemoryMiB": {
 "Min": 0
 }
],
 "TargetCapacitySpecification": {
 "TotalTargetCapacity": 1,
 "DefaultTargetCapacityType": "spot"
 }
}

```

### 組態無效：Overrides 包含 InstanceRequirements 和 InstanceType

下列組態無效。Overrides 結構同時包含 InstanceRequirements 和 InstanceType。針對 Overrides，您可以指定 InstanceRequirements 或 InstanceType，但不能同時指定兩者。

```

{
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "MyLaunchTemplate",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 0,
 "Max": 2
 },
 "MemoryMiB": {
 "Min": 0
 }
 }
 },
 {
 "InstanceType": "m5.large"
 }
]
 }
]
}

```

```
 }
],
 "TargetCapacitySpecification": {
 "TotalTargetCapacity": 1,
 "DefaultTargetCapacityType": "spot"
 }
}
```

### 組態無效：兩個 **Overrides** 包含 **InstanceRequirements** 和 **InstanceType**

下列組態無效。Overrides 結構同時包含 InstanceRequirements 和 InstanceType。您可以指定 InstanceRequirements 或 InstanceType，但不能同時指定兩者，即使它們處於不同的 Overrides 結構也一樣。

```
{
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "MyLaunchTemplate",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 0,
 "Max": 2
 },
 "MemoryMiB": {
 "Min": 0
 }
 }
 }
]
 },
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "MyOtherLaunchTemplate",
 "Version": "1"
 },
 "Overrides": [

```

```

 "InstanceType": "m5.large"
 }
]
 },
 "TargetCapacitySpecification": {
 "TotalTargetCapacity": 1,
 "DefaultTargetCapacityType": "spot"
 }
}

```

### 組態無效：重疊屬性值

下列組態無效。兩個 InstanceRequirements 結構每個都包含 "VCpuCount": {"Min": 0, "Max": 2}。這些屬性的值重疊，這會導致容量集區重複。

```

{
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "MyLaunchTemplate",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 0,
 "Max": 2
 },
 "MemoryMiB": {
 "Min": 0
 }
 },
 {
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 0,
 "Max": 2
 },
 "MemoryMiB": {
 "Min": 0
 }
 }
 }
]
 }
 }
]
}

```

```

 }
 }
}
],
"TargetCapacitySpecification": {
 "TotalTargetCapacity": 1,
 "DefaultTargetCapacityType": "spot"
}
}
}

```

### 預覽具有所指定屬性的執行個體類型

您可以使用[從執行個體需求取得執行個體類型 AWS CLI 指令來預覽符合指定屬性的例證類型](#)。這特別有助於解決在不啟動任何執行個體的情況下要在請求組態中指定哪些屬性。請注意，該命令不會考慮可用容量。

### 透過使用指定屬性來預覽例證類型清單 AWS CLI

1. (選用) 若要產生可指定的所有可能屬性，請使用 [get-instance-types-from-instance-requirements](#) 命令和 `--generate-cli-skeleton` 參數。您可以選擇性地將輸出導向至檔案來儲存它，方法是使用 `input > attributes.json`。

```

aws ec2 get-instance-types-from-instance-requirements \
 --region us-east-1 \
 --generate-cli-skeleton input > attributes.json

```

### 預期的輸出結果

```

{
 "DryRun": true,
 "ArchitectureTypes": [
 "i386"
],
 "VirtualizationTypes": [
 "hvm"
],
 "InstanceRequirements": {
 "VCpuCount": {

```

```
 "Min": 0,
 "Max": 0
 },
 "MemoryMiB": {
 "Min": 0,
 "Max": 0
 },
 "CpuManufacturers": [
 "intel"
],
 "MemoryGiBPerVCpu": {
 "Min": 0.0,
 "Max": 0.0
 },
 "ExcludedInstanceTypes": [
 ""
],
 "InstanceGenerations": [
 "current"
],
 "SpotMaxPricePercentageOverLowestPrice": 0,
 "OnDemandMaxPricePercentageOverLowestPrice": 0,
 "BareMetal": "included",
 "BurstablePerformance": "included",
 "RequireHibernateSupport": true,
 "NetworkInterfaceCount": {
 "Min": 0,
 "Max": 0
 },
 "LocalStorage": "included",
 "LocalStorageTypes": [
 "hdd"
],
 "TotalLocalStorageGB": {
 "Min": 0.0,
 "Max": 0.0
 },
 "BaselineEbsBandwidthMbps": {
 "Min": 0,
 "Max": 0
 },
 "AcceleratorTypes": [
 "gpu"
],
],
```



```
 "AcceleratorCount": {
 "Min": 0,
 "Max": 0
 },
 "AcceleratorManufacturers": [
 "nvidia"
],
 "AcceleratorNames": [
 "a100"
],
 "AcceleratorTotalMemoryMiB": {
 "Min": 0,
 "Max": 0
 },
 "NetworkBandwidthGbps": {
 "Min": 0.0,
 "Max": 0.0
 },
 "AllowedInstanceTypes": [
 ""
]
 },
 "MaxResults": 0,
 "NextToken": ""
}
```

2. 使用上一個步驟的輸出來建立 JSON 組態檔，並將其設定如下：

#### Note

您必須提供 ArchitectureTypes、VirtualizationTypes、VCpuCount 和 MemoryMiB 的值。您可以省略其他屬性；省略時，就會使用預設值。如需每個屬性及其預設值的說明，請參閱 Amazon EC2 命令列參考中的 [get-instance-types-from-instance-requirements](#)。

- a. 針對 ArchitectureTypes，指定一或多種類型的處理器架構。
- b. 針對 VirtualizationTypes，指定一或多種類型的虛擬化。
- c. 針對 VCpuCount，指定 vCPU 數目下限和上限。若要指定沒有下限，請針對 Min，指定 0。若要指定沒有上限，請省略 Max 參數。

- d. 針對 MemoryMiB，指定記憶體的数量下限和上限 (以 MiB 為單位)。若要指定沒有下限，請針對 Min，指定 0。若要指定沒有上限，請省略 Max 參數。
  - e. 您可以選擇性地指定一或多個其他屬性，以進一步限制傳回的執行個體類型清單。
3. 若要預覽具有您在 JSON 檔案中指定之屬性的執行個體類型，請使用 [get-instance-types-from-instance-requirements](#) 命令，並使用 `--cli-input-json` 參數指定 JSON 檔案的名稱和路徑。您可以選擇性地格式化要以表格格式出現的輸出。

```
aws ec2 get-instance-types-from-instance-requirements \
 --cli-input-json file://attributes.json \
 --output table
```

### 範例 *attributes.json* 文件

在此範例中，JSON 檔案中包含必要的屬性。它們是 ArchitectureTypes、VirtualizationTypes、VCpuCount 和 MemoryMiB。此外，選用 InstanceGenerations 屬性也包含在內。請注意，對於 MemoryMiB，可以省略 Max，以表示沒有限制。

```
{
 "ArchitectureTypes": [
 "x86_64"
],
 "VirtualizationTypes": [
 "hvm"
],
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 4,
 "Max": 6
 },
 "MemoryMiB": {
 "Min": 2048
 },
 "InstanceGenerations": [
 "current"
]
 }
}
```

## 範例輸出

```

|GetInstanceTypesFromInstanceRequirements|
+-----+
|| InstanceTypes ||
|+-----+|
|| InstanceType ||
|+-----+|
|| c4.xlarge ||
|| c5.xlarge ||
|| c5a.xlarge ||
|| c5ad.xlarge ||
|| c5d.xlarge ||
|| c5n.xlarge ||
|| d2.xlarge ||
|| ... ||

```

4. 在識別符合您需求的執行個體類型之後，請記下您使用的執行個體屬性，以便在設定機群請求時可以使用它們。

## 針對隨需備份設定 EC2 Fleet

如果您有緊急、無法預測的擴展需求，例如新聞網站必須在重大新聞事件或遊戲啟動期間擴展，我們建議為您的隨需執行個體指定備用的執行個體類型，以防您的首要選項沒有足夠的可用容量。例如，您可能偏好 c5.2xlarge 隨需執行個體，但如果可用容量不足，則在高峰負載期間您會願意使用一些 c4.2xlarge 執行個體。在此情況下，EC2 Fleet 會嘗試使用 c5.2xlarge 執行個體來實現所有目標容量，但如果容量不足，則會自動啟動 c4.2xlarge 執行個體以實現目標容量。

### 主題

- [針對隨需容量排定執行個體類型的優先順序](#)
- [針對隨需執行個體使用容量預留](#)

### 針對隨需容量排定執行個體類型的優先順序

EC2 機群嘗試實現您的隨需容量時，其會預設為先啟動價格最低的執行個體類型。若 `AllocationStrategy` 設為 `prioritized`，EC2 Fleet 會使用優先順序來決定在實現隨需容量時要先使用的執行個體類型。優先順序會指派至啟動範本覆寫，優先順序最高的將會先啟動。

## 範例：排定執行個體類型的優先順序

在本例中，您設定三個啟動範本覆寫，每個範本均具有不同的執行個體類型。

執行個體類型的隨需價格在價格範圍內。以下是本範例中使用的執行個體類型，依價格順序列出，從最便宜的執行個體類型開始：

- m4.large – 最便宜
- m5.large
- m5a.large

若您不使用優先順序來決定順序，則機群會以最便宜的執行個體類型開始來實現隨需容量。

但是，假設您有要首先使用且未使用的 m5.large 預留執行個體。您可設定啟動範本覆寫，以便依優先順序使用執行個體類型，如下所示：

- m5.large – 優先順序 1
- m4.large – 優先順序 2
- m5a.large – 優先順序 3

## 針對 隨需執行個體 使用 容量預留

隨需容量預留可讓您在特定的可用區域中，為隨需執行個體預訂任何期間的運算容量。您可將 EC2 Fleet 設定為在啟動隨需執行個體時先使用容量預留。

容量預留設定為 open 或 targeted。EC2 Fleet 可以按照 open 或 targeted 的容量預留啟動隨需執行個體：

- 如果容量預留為 open，則具有相符屬性的隨需執行個體會以預留容量自動執行。
- 如果容量預留為 targeted，則隨需執行個體必須特別針對它使用預留容量執行。這對於用完特定容量預留或控制何時該使用特定容量預留的情況下十分有用。

如果您在 EC2 Fleet 中使用 targeted 容量預留，必須有足夠的容量預留以滿足目標隨需容量，否則啟動會失敗。為了避免啟動失敗，最好將 targeted 容量預留新增至資源群組，然後針對資源群組為目標來執行。資源群組不需要有足夠的容量預留；如果在滿足目標隨需容量之前用完容量預留，則機群可以將剩餘的目標容量啟動為一般隨需容量。

## 在 EC2 Fleet 中使用容量預留

1. 將機群配置為 `instant` 類型。您無法針對其他類型的機群使用容量預留。
2. 將容量預留的使用策略配置為 `use-capacity-reservations-first`。
3. 在啟動範本的 Capacity reservation (容量保留) 中，選擇 `Open` (開放) 或 `Target by group` (依群組分類的目標)。如果您選擇 `Target by group` (依群組分類的目標)，請指定容量預留資源群組 ID。

當機群嘗試滿足隨需容量時，如果發現多個執行個體集區具有相符的未使用容量預留，則會根據隨需分配策略 (`lowest-price` 或 `prioritized`) 決定啟動隨需執行個體的集區。

如需設定機群以使用容量預留來滿足隨需容量的範例，請參閱 [EC2 Fleet 範例組態](#) 中的範例 5 至 7。

如需設定容量預留的相關資訊，請參閱 [On-Demand Capacity Reservations](#) 與 [隨需容量預留常見問答集](#)。

## 容量重新平衡

您可以設定 EC2 Fleet，以便在 Amazon EC2 發出重新平衡建議時啟動替代 Spot 執行個體，以通知您 Spot 執行個體的中斷風險提高。容量重新平衡可協助您維持工作負載可用性，方法是在執行中的執行個體受到 Amazon EC2 的中斷之前，使用新的 Spot 執行個體主動擴增機群。如需詳細資訊，請參閱 [EC2 執行個體重新平衡建議](#)。

若要設定 EC2 Fleet 以啟動取代 Spot 執行個體，請使用 [create-fleet](#) (AWS CLI) 命令和 `MaintenanceStrategies` 結構中的相關參數。如需詳細資訊，請參閱 [範例啟動設定](#)。

### 限制

- 容量重新平衡僅適用於類型為 `maintain` 的機群。
- 機群執行時，您無法修改容量重新平衡設定。若要變更容量重新平衡設定，您必須刪除機群並建立新的機群。

### 組態選項

EC2 Fleet 的 `ReplacementStrategy` 支援以下兩個值：

#### `launch-before-terminate`

Amazon EC2 會在新的替代 Spot 執行個體啟動之後，終止收到重新平衡通知的 Spot 執行個體。當指定 `launch-before-terminate` 時，您亦須指定 `termination-delay` 的值。在啟動新的替

代執行個體之後，Amazon EC2 會等待 `termination-delay` 的持續時間，然後終止舊的執行個體。對於 `termination-delay`，下限為 120 秒 (2 分鐘)，上限則為 7200 秒 (2 小時)。

建議僅在您可以預測執行個體關閉程序將需要多長時間才能完成時，才使用 `launch-before-terminate`。這將確保只在關機程序完成之後，才會終止舊執行個體。請注意，Amazon EC2 可以中斷舊的執行個體，並在 `termination-delay` 之前提供兩分鐘警告。

強烈建議您不要使用 `lowest-price` 分配策略與 `launch-before-terminate` 結合，以避免替代 Spot 執行個體也處於提高的中斷風險。

## launch

針對現有 Spot 執行個體發出重新平衡通知時，Amazon EC2 會啟動替代 Spot 執行個體。Amazon EC2 不會自動終止收到重新平衡建議的執行個體。您可以終止舊的執行個體，也可以讓它們保持執行中狀態。所有執行個體在執行時，您需要支付所有執行個體的費用。

## 考量事項

如果您設定容量重新平衡的 EC2 Fleet，請考慮下列事項：

盡可能在請求中提供更多 Spot 容量集區

設定您的 EC2 Fleet，以使用多個執行個體類型和可用區域。這提供了在各種 Spot 容量集區中啟動 Spot 執行個體的彈性。如需詳細資訊，請參閱 [對執行個體類型和可用區域具有彈性](#)。

避免中斷替代 Spot 執行個體的風險提高

如果您使用 `lowest-price` 分配策略，替代 Spot 執行個體的中斷風險可能會提高。這是因為 Amazon EC2 總是會在當時具有可用容量的最低價集區中啟動執行個體，即使您的替代 Spot 執行個體在啟動後很快就會中斷也一樣。為了避免中斷風險的提高，強烈建議您不要使用 `lowest-price` 分配策略，而是改用 `capacity-optimized` 或 `capacity-optimized-prioritized` 分配策略。這些策略可確保在最佳 Spot 容量集區中啟動替代 Spot 執行個體，因此不太可能在不久的將來中斷。如需詳細資訊，請參閱 [使用價格和容量最佳化分配策略](#)。

只有在可用性相同或更好時，Amazon EC2 才會啟動新執行個體

容量重新平衡的其中一個目標是改善 Spot 執行個體的可用性。如果現有的 Spot 執行個體收到重新平衡建議，只有新執行個體提供與現有執行個體相同或更好的可用性時，Amazon EC2 才會啟動新執行個體。如果新執行個體的中斷風險比現有執行個體更大，則 Amazon EC2 將不會啟動新執行個體。但是，Amazon EC2 將繼續評估 Spot 容量集區，並在可用性改善時啟動新執行個體。

如果 Amazon EC2 不主動啟動新執行個體，現有執行個體可能會中斷。發生這種情況時，無論新執行個體是否存在高中斷風險，Amazon EC2 都會嘗試啟動新執行個體。

## 容量重新平衡不會增加 Spot 執行個體中斷率

當您啟用容量重新平衡時，不會增加 [Spot 執行個體中斷率](#) (當 Amazon EC2 需要恢復容量時回收的 Spot 執行個體數量)。但是，如果容量重新平衡偵測到執行個體有中斷風險，Amazon EC2 將立即嘗試啟動新的執行個體。因此，與在有風險的執行個體中斷後等待 Amazon EC2 啟動新的執行個體相比，可能會取代更多執行個體。

雖然您可以在啟用容量重新平衡的情況下取代更多執行個體，但您可以從主動式而非被動式中受益，因為在執行個體中斷之前您有更多時間採取動作。透過 [Spot 執行個體中斷通知](#)，您通常最多只有兩分鐘時間來正常關閉您的執行個體。透過容量重新平衡預先啟動新的執行個體，可讓現有程序更有可能在有風險的執行個體上完成，您可以啟動執行個體關閉程序，並防止在有風險的執行個體上排程新的工作。您也可以開始準備新啟動的執行個體以接管應用程式。透過容量重新平衡的主動式取代，您可以從正常連續性中獲益。

作為示範使用容量重新平衡的風險和益處的理論範例，請考慮下列案例：

- 下午 2:00 – 收到執行個體 A 的重新平衡建議，Amazon EC2 會立即開始嘗試啟動取代執行個體 B，讓您有時間開始關機程序。\*
- 下午 2:30 – 收到執行個體 B 的重新平衡建議，會取代為執行個體 C，讓您有時間開始關機程序。\*
- 下午 2:32 – 如果未啟用容量重新平衡，並且如果在下午 2:32 收到執行個體 A 的 Spot 執行個體中斷通知，則您最多只有兩分鐘時間採取動作，但是執行個體 A 會一直執行到目前。

\* 如果指定了 `launch-before-terminate`，Amazon EC2 將在取代執行個體連線後終止有風險的執行個體。

Amazon EC2 可以啟動新的替代 Spot 執行個體，直到滿足的容量為目標容量雙倍

設定 EC2 Fleet 為容量重新平衡時，機群會嘗試為每個收到重新平衡建議的 Spot 執行個體啟動新的替代 Spot 執行個體。Spot 執行個體收到重新平衡建議後，該執行個體不再計入以滿足容量的一部分。根據替代策略，Amazon EC2 會在預先設定的終止延遲之後終止執行個體，或讓執行個體繼續執行。這讓您有機會在執行個體上執行 [重新平衡動作](#)。

如果您的機群達到目標容量的兩倍，即使替代執行個體本身收到重新平衡建議，它仍會停止啟動新的替代執行個體。

例如，您建立目標容量為 100 個 Spot 執行個體的 EC2 Fleet。所有 Spot 執行個體都會收到重新平衡建議，這會導致 Amazon EC2 啟動 100 個替代 Spot 執行個體。這會將已滿足 Spot 執行個體的數目提高至 200 個，這是目標容量的兩倍。某些替代執行個體會收到重新平衡建議，但不會再啟動替代執行個體，因為機群不能超過目標容量的兩倍。

請注意，所有執行個體在執行時都需支付這些執行個體的費用。

建議您將 EC2 Fleet 設定為終止收到重新平衡建議的 Spot 執行個體

如果您設定 EC2 Fleet 進行容量重新平衡，建議您只在可以預測執行個體關閉程序需要多長時間才能完成時，才選擇具有適當終止延遲的 `launch-before-terminate`。這將確保只在關機程序完成之後，才會終止舊執行個體。

如果您選擇自行終止建議用於重新平衡的執行個體，建議您監控機群中 Spot 執行個體收到的重新平衡建議訊號。透過監控訊號，您可以在 Amazon EC2 中斷受影響的執行個體之前，快速對受影響的執行個體執行[重新平衡動作](#)，然後手動終止它們。如果您未終止執行個體，則會在執行時繼續為這些執行個體付費。Amazon EC2 不會自動終止收到重新平衡建議的執行個體。

您可以使用 Amazon EventBridge 或執行個體中繼資料設定通知。如需詳細資訊，請參閱[監控重新平衡建議訊號](#)。

EC2 Fleet 在縮減或擴展期間計算已滿足容量時，不會計算收到重新平衡建議的執行個體

如果您的 EC2 Fleet 已設定容量重新平衡，且您將目標容量變更為縮減或擴展，則機群不會將標示要重新平衡的執行個體計算為已滿足容量一部分，如下所示：

- 縮減 – 如果您減少所需的目標容量，Amazon EC2 會終止未標示要重新平衡的執行個體，直到達到所需容量為止。標示要重新平衡的執行個體不會計入已滿足的容量。

例如，您建立目標容量為 100 個 Spot 執行個體的 EC2 機群。10 個執行個體會收到重新平衡建議，因此 Amazon EC2 會啟動 10 個新的替代執行個體，從而產生 110 個執行個體的滿足容量。然後，您可以將目標容量減少為 50 (縮減)，但已滿足容量實際上為 60 個執行個體，因為標示要重新平衡的 10 個執行個體不會被 Amazon EC2 終止。您需要手動終止這些執行個體，或者您也可以讓它們保持運行。

- 橫向擴展 – 如果您增加想要的目標容量，Amazon EC2 會啟動新執行個體，直到達到所需容量為止。標示要重新平衡的執行個體不會計入已滿足的容量。

例如，您建立目標容量為 100 個 Spot 執行個體的 EC2 Fleet。10 個執行個體會收到重新平衡建議，因此機群會啟動 10 個新的替代執行個體，從而產生 110 個執行個體的滿足容量。然後，您將目標容量增加至 200 (擴展)，但已滿足容量實際上是 210 個執行個體，因為標示要重新平衡的 10 個執行個體不會被機群計為目標容量的一部分。您需要手動終止這些執行個體，或者您也可以讓它們保持運行。



## 最高價格覆寫

每個 EC2 Fleet 可以包含全球最高價格，或使用預設值 (隨需價格)。機群將此用作每個啟動規格的預設最高價格。

您可以選擇性指定一或多個啟動規格中的最高價格。此價格特定於啟動規格。如果啟動規格包含特定價格，則 EC2 Fleet 會使用這個最高價格來覆寫全球最高價格。不包括特定最高價格的任何其他啟動規格仍會使用全球最高價格。

## 控制花費

當它符合下列其中一個參數時，EC2 Fleet 會停止啟動執行個體：TotalTargetCapacity 或 MaxTotalPrice (您願意支付的最大金額)。若要控制您為機群支付的每個小時金額，您可以指定 MaxTotalPrice。達到總價格上限時，EC2 Fleet 會停止啟動執行個體，即使它未符合目標容量亦然。

以下範例顯示兩個不同的案例。在第一個範例中，EC2 Fleet 會在符合目標容量時停止啟動執行個體。在第二個範例中，EC2 Fleet 會在達到您願意支付的金額上限 (MaxTotalPrice) 時停止啟動執行個體。

### 範例：達到目標容量時停止啟動執行個體

假設有 m4.large 隨需執行個體的請求，其中：

- 隨需價格：每小時 0.10 USD
- OnDemandTargetCapacity：10
- MaxTotalPrice：1.50 USD

EC2 Fleet 會啟動 10 個隨需執行個體，因為 1.00 USD 的總計 (10 個執行個體 x 0.10 USD) 未超出隨需執行個體的 1.50 USD 的 MaxTotalPrice。

### 範例：達到總價格上限時停止啟動執行個體

假設有 m4.large 隨需執行個體的請求，其中：

- 隨需價格：每小時 0.10 USD
- OnDemandTargetCapacity：10
- MaxTotalPrice：0.80 USD

如果 EC2 Fleet 啟動了隨需目標容量 (10 個隨需執行個體)，每個小時的總成本會是 1.00 USD。這超出為隨需執行個體指定的 MaxTotalPrice 的金額 (0.80 USD)。為了防止花費超出您願意支付的金額，EC2 Fleet 只會啟動 8 個隨需執行個體 (低於隨需目標容量)，因為啟動更多會超出隨需執行個體的 MaxTotalPrice。

## EC2 Fleet 執行個體權重

建立 EC2 Fleet 時，您可以定義每個執行個體類型可對應用程式的效能貢獻的容量單位。然後可以使用執行個體加權來調整每個啟動規格的價格上限。

根據預設，您指定的價格是每執行個體小時。使用執行個體權重功能時，您指定的價格是每單位小時。您可以透過將執行個體類型價格除以它所代表的單位數，來計算每單位小時的價格。EC2 Fleet 會將目標容量除以執行個體的權重，來計算出要啟動的執行個體數量。如果結果不是整數，則機群會將其捨入到下一個整數，以便您的機群大小不低於其目標容量。即使所啟動的執行個體容量超過所請求的目標容量，該機群也可以選取您在啟動規格中指定的任何集區。

下表包含決定目標容量為 10 的 EC2 Fleet 每單位價格之計算範例。

執行個體類型	執行個體權重	目標容量	已啟動的執行個體數目	每個執行個體小時的價格	每個單位小時的價格
r3.xlarge	2	10	5 (10 除以 2)	0.05 USD	\$0.025 (.05 除以 2)
r3.8xlarge	8	10	2 (10 除以 8，將結果四捨五入)	0.10 USD	\$0.0125 (.10 除以 8)

按照下列方式，在實現請求時使用 EC2 Fleet 執行個體權重，以在完成要求時，使用每單位最低價在集區中佈建所需目標容量：

1. 在您所選擇的執行個體 (預設) 或裝置中 (例如虛擬 CPU、記憶體、儲存空間或傳輸量)，設定 EC2 Fleet 的目標容量。

2. 設定每單位價格
3. 為每個啟動規格指定權重，即執行個體類型代表目標容量的單位數。

### 執行個體權重範例

請考慮下列組態的 EC2 Fleet 請求：

- 目標容量為 24
- 執行個體類型為 r3.2xlarge 且權重為 6 的啟動規格
- 執行個體類型為 c3.xlarge 且權重為 5 的啟動規格

權重表示執行個體類型代表目標容量的單位數。如果第一個啟動規格提供最低的每單位價格 (r3.2xlarge 的每小時執行個體價格除以 6)，EC2 Fleet 將啟動其中四個執行個體 (24 除以 6)。

如果第二個啟動規格提供了最低的每單位價格 (c3.xlarge 的每小時執行個體價格除以 5)，則 EC2 Fleet 會啟動這些執行個體的其中 5 個 (24 除以 5，將結果四捨五入)。

### 執行個體權重與分配策略

請考慮下列組態的 EC2 Fleet 請求：

- 目標容量為 30 個 競價型執行個體
- 執行個體類型為 c3.2xlarge 且權重為 8 的啟動規格
- 執行個體類型為 m3.xlarge 且權重為 8 的啟動規格
- 執行個體類型為 r3.xlarge 且權重為 8 的啟動規格

EC2 Fleet 會啟動 4 個執行個體 (30 除以 8，將結果四捨五入)。採用 diversified 策略，機群會在三個集區的每一個集區啟動一個執行個體，並且三個集區任一個中的第四個執行個體會提供最低的每單位價格。

## 使用 EC2 Fleet

若要開始使用 EC2 Fleet，您需要建立一個請求，其中包含總目標容量、隨需容量、Spot 容量、執行個體的一或多個啟動規格，以及您願意支付的最高價格。機群請求必須包含啟動範本，該範本定義機群啟動執行個體所需的資訊，例如 AMI、執行個體類型、子網或可用區域，以及一或多個安全群組。您可以為執行個體類型、子網、可用區域和您願意支付的最高價格指定啟動規格覆寫，並且可以為每個啟動規格覆寫指派加權容量。

EC2 Fleet 會在有可用容量時啟動 隨需執行個體，並在最高價格超過 Spot 價格和容量可用時啟動 競價型執行個體。

如果您的叢集包含 競價型執行個體，則 Amazon EC2 會在 Spot 價格變更時嘗試維持您的叢集目標容量。

maintain 或 request 類型的 EC2 Fleet 請求會一直保持作用中狀態直到過期或您將其刪除。當您刪除 maintain 或 request 類型的機群時，您可以指定該刪除是否會終止該機群中的執行個體。否則，隨需執行個體 會一直執行直到您將其終止為止，而 Spot 執行個體會一直執行直到中斷或您將其終止為止。

## 目錄

- [EC2 Fleet 請求狀態](#)
- [EC2 Fleet 先決條件](#)
- [EC2 Fleet 運作狀態檢查](#)
- [產生 EC2 Fleet JSON 組態檔案](#)
- [建立 EC2 Fleet](#)
- [標記 EC2 Fleet](#)
- [說明您的 EC2 機群](#)
- [修改 EC2 Fleet](#)
- [刪除 EC2 Fleet](#)

## EC2 Fleet 請求狀態

EC2 Fleet 請求可以是下列任一狀態：

### submitted

正在評估 EC2 Fleet 請求且 Amazon EC2 正準備啟動執行個體的目標數量。請求可以包含 隨需執行個體、競價型執行個體 或兩者皆包含。超出機群限制的請求會立即刪除。

### active

EC2 Fleet 請求已驗證且 Amazon EC2 正在嘗試維持執行中的執行個體目標數目。此請求會一直維持在該狀態，直到修改或刪除。

## modifying

正在修改 EC2 Fleet 請求。此請求會一直維持在該狀態，直到已完成修改或已刪除請求。只有 `maintain` 機群類型可以修改。此狀態不適用於其他請求類型。

## deleted\_running

EC2 Fleet 請求已刪除，且不會啟動其他執行個體。其現有執行個體將繼續執行直到中斷或手動終止。此請求會一直維持在該狀態，直到所有執行個體中斷或終止。刪除 EC2 Fleet 請求後，只有類型 `maintain` 或 `request` 的 EC2 Fleet 可具有執行中的執行個體。不支援具有執行中執行個體的已刪除 `instant` 機群。此狀態不適用於 `instant` 機群。

## deleted\_terminating

EC2 Fleet 請求已刪除，且其執行個體已終止。此請求會一直維持在該狀態，直到所有執行個體終止。

## deleted

EC2 Fleet 已刪除，且無執行中的執行個體。請求會在其執行個體終止之後兩天刪除。

## EC2 Fleet 先決條件

若要建立 EC2 Fleet，必須具備下列必要條件：

- [啟動範本](#)
- [EC2 Fleet 的服務連結角色](#)
- [授予客戶受管金鑰存取權，以便與加密的 AMI 和 EBS 快照搭配使用](#)
- [EC2 Fleet 使用者的許可](#)

### 啟動範本

啟動範本包含關於要啟動的執行個體資訊，例如執行個體類型、可用區域以及您願意支付的最高價格。如需詳細資訊，請參閱 [從啟動範本啟動執行個體](#)。


### EC2 Fleet 的服務連結角色

`AWSServiceRoleForEC2Fleet` 角色會授與 EC2 Fleet 許可，以代表您請求、啟動、終止和標記執行個體。Amazon EC2 會使用此服務連結角色來完成下列動作：

- `ec2:RunInstances` – 啟動執行個體。
- `ec2:RequestSpotInstances` – 請求 競價型執行個體。

- `ec2:TerminateInstances` – 終止執行個體。
- `ec2:DescribeImages` – 針對 競價型執行個體 描述 Amazon Machine Images (AMI)。
- `ec2:DescribeInstanceStatus` – 描述 競價型執行個體 的狀態。
- `ec2:DescribeSubnets` – 描述 競價型執行個體 的子網。
- `ec2:CreateTags` – 將標籤新增至 EC2 Fleet、執行個體和磁碟區。

在使用 AWS CLI 或 API 建立 EC2 叢集之前，請確保此角色存在。

 Note

instant EC2 Fleet 不需要此角色。

若要建立角色，請依照下列所示使用 IAM 主控台。

若要建立 EC2 叢集的 `AWSServiceRoleForEC2Fleet` 角色

1. 在以下網址開啟 IAM 主控台：<https://console.aws.amazon.com/iam/>。
2. 在導覽窗格中，選擇 Roles (角色)，然後選擇 Create role (建立角色)。
3. 在 [選取信任實體的類型] 頁面上，執行下列動作：
  - a. 針對信任的實體類型，請選擇 AWS 服務。
  - b. 在 [使用案例] 下，針對服務或使用案例，選擇 [EC2-叢集]。

 Tip

一定要選擇 EC2-艦隊。如果選擇 EC2，EC2 -叢集使用案例不會出現在使用案例清單中。EC2-叢集使用案例會自動建立具有所需 IAM 許可的政策，並將建議 `AWSServiceRoleForEC2Fleet` 作為角色名稱。

- c. 選擇下一步。
4. 在 Add permissions (新增許可) 頁面上，選擇 Next (下一步)。
  5. 在命名、檢閱和建立頁面上，選擇建立角色。

如果您不再需要使用 EC2 叢集，建議您刪除 `AWSServiceRoleForEC2Fleet` 角色。從您的帳戶中刪除此角色後，如果您建立其他機群，則可以再次建立此角色。

如需詳細資訊，請參閱《IAM 使用者指南》中的[使用服務連結角色](#)。

授予客戶受管金鑰存取權，以便與加密的 AMI 和 EBS 快照搭配使用

如果您在 EC2 叢集中指定加密的 [AMI](#) 或加密的 Amazon EBS 快照，並使用 AWS KMS 金鑰進行加密，則必須授與 `AWSServiceRoleForEC2Fleet` 角色使用客戶受管金鑰的權限，以便 Amazon EC2 可以代表您啟動執行個體。若要執行此動作，您必須將授予新增至客戶受管金鑰，如下列程序所示。

提供許可時，授與為金鑰政策的替代方案。如需詳細資訊，請參閱《AWS Key Management Service 開發人員指南》中的[使用授與](#)和[在 AWS KMS 中使用金鑰政策](#)。

授與 `AWSServiceRoleForEC2Fleet` 角色使用客戶管理金鑰的權限

- 使用 [create-grant](#) 命令可將授權新增至客戶管理的金鑰，並指定授與執行授權之作業的主體 (`AWSServiceRoleForEC2Fleet` 服務連結角色)。客戶受管金鑰由 `key-id` 參數和客戶受管金鑰的 ARN 指定。主體是由 `grantee-principal` 參數和 `AWSServiceRoleForEC2Fleet` 服務連結角色的 ARN 所指定。

```
aws kms create-grant \
 --region us-east-1 \
 --key-id arn:aws:kms:us-east-1:444455556666:key/1234abcd-12ab-34cd-56ef-1234567890ab \
 --grantee-principal arn:aws:iam::111122223333:role/AWSServiceRoleForEC2Fleet \
 --operations "Decrypt" "Encrypt" "GenerateDataKey" "GenerateDataKeyWithoutPlaintext" "CreateGrant" "DescribeKey" "ReEncryptFrom" "ReEncryptTo"
```

## EC2 Fleet 使用者的許可

如果使用者將會建立或管理 EC2 Fleet，請務必授予這些使用者所需的許可。

### 為 EC2 Fleet 建立政策

1. 前往 <https://console.aws.amazon.com/iam/> 開啟 IAM 主控台。
2. 在導覽窗格中，選擇政策。
3. 選擇 Create policy (建立政策)。
4. 在 Create policy (建立政策) 頁面上，選擇 JSON 索引標籤，接著使用下列內容來取代其文字，然後選擇 Review policy (檢閱政策)。

```
{
```

```
"Version": "2012-10-17",
"Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:*"
],
 "Resource": "*"
 },
 {
 "Effect": "Allow",
 "Action": [
 "iam:ListRoles",
 "iam:PassRole",
 "iam:ListInstanceProfiles"
],
 "Resource": "arn:aws:iam::123456789012:role/DevTeam*"
 }
]
```

`ec2:*` 會授與使用者權限，以呼叫所有 Amazon EC2 API 動作。若要將使用者限制於特定的 Amazon EC2 API 動作，請改為指定這些動作。

使用者必須具有呼叫下列動作的權限：`iam:ListRoles` 動作 (以列舉現有的 IAM 角色)、`iam:PassRole` 動作 (以指定 EC2 Fleet 角色) 和 `iam:ListInstanceProfiles` 動作 (以列舉現有的執行個體設定檔)。

(選用) 若要讓使用者能夠使用 IAM 主控台建立角色或執行個體設定檔，您也必須將下列動作新增至政策中：

- `iam:AddRoleToInstanceProfile`
- `iam:AttachRolePolicy`
- `iam:CreateInstanceProfile`
- `iam:CreateRole`
- `iam:GetRole`
- `iam:ListPolicies`

5. 在 Review policy (檢閱政策) 頁面上，輸入政策名稱和說明，然後選擇 Create policy (建立政策)。
6. 若要提供存取權，請新增權限至您的使用者、群組或角色：



- 使用者和群組位於 AWS IAM Identity Center :

建立權限合集。請按照 AWS IAM Identity Center 使用者指南 中的 [建立權限合集](#) 說明進行操作。

- 透過身分提供者在 IAM 中管理的使用者 :

建立聯合身分的角色。請按照 IAM 使用者指南 的 [為第三方身分提供者 \(聯合\) 建立角色](#) 中的指示進行操作。

- IAM 使用者 :

- 建立您的使用者可擔任的角色。請按照 IAM 使用者指南 的 [為 IAM 使用者建立角色](#) 中的指示進行操作。
- (不建議) 將政策直接附加至使用者，或將使用者新增至使用者群組。請遵循 IAM 使用者指南的 [新增許可到使用者 \(主控台\)](#) 中的指示。

## EC2 Fleet 運作狀態檢查

EC2 Fleet 會每兩分鐘定期檢查機群中執行個體的運作狀態。執行個體的運作狀態會是 healthy 或 unhealthy。

EC2 Fleet 會根據 Amazon EC2 提供的狀態檢查資訊，來判定執行個體的運作狀態。如果執行個體狀態檢查或系統狀態檢查的運作狀態檢查，出現連續三次為 impaired，則該執行個體即為 unhealthy。如需詳細資訊，請參閱 [您的執行個體的状态檢查](#)。

您可以設定機群以取代運作狀態不佳的 競價型執行個體。ReplaceUnhealthyInstances 設為 true 以後，在報告為 unhealthy 時，則 Spot 執行個體會被取代。當取代一個狀態不佳的 Spot 執行個體時，機群可能會低於其目標容量長達幾分鐘。

### 要求

- 運作狀態檢查取代僅支援維持目標容量的 EC2 Fleet (類型 maintain 的機群)，不支援類型 request 或 instant 的機群。
- 運作狀態檢查取代僅支援 競價型執行個體。此功能不支援 隨需執行個體。
- 您可以設定 EC2 Fleet 只在建立執行個體時取代狀況不良的執行個體。
- 使用者只有在擁有呼叫 ec2:DescribeInstanceStatus 動作的許可時，才能使用運作狀態檢查取代功能。

## 設定 EC2 Fleet 以取代運作狀態不佳的 競價型執行個體

1. 遵循下列步驟來建立 EC2 Fleet。如需詳細資訊，請參閱 [建立 EC2 Fleet](#)。
2. 若要設定機群以取代運作狀態不佳的 競價型執行個體，請在 JSON 檔案 `ReplaceUnhealthyInstances` 中輸入 `true`。

## 產生 EC2 Fleet JSON 組態檔案

若要檢視 EC2 Fleet 組態參數的完整清單，您可以產生 JSON 檔案。如需每個參數的說明，請參閱 AWS CLI 命令參考中的 [create-fleet](#)。

使用命令列產生所有可能 EC2 Fleet 參數的 JSON 檔案

- 使用 [create-fleet](#) (AWS CLI) 命令和 `--generate-cli-skeleton` 參數來產生 EC2 Fleet JSON 檔案，並將輸出導向至檔案以儲存它：

```
aws ec2 create-fleet \
 --generate-cli-skeleton input > ec2createfleet.json
```

## 範例輸出

```
{
 "DryRun": true,
 "ClientToken": "",
 "SpotOptions": {
 "AllocationStrategy": "capacity-optimized",
 "MaintenanceStrategies": {
 "CapacityRebalance": {
 "ReplacementStrategy": "launch"
 }
 },
 "InstanceInterruptionBehavior": "hibernate",
 "InstancePoolsToUseCount": 0,
 "SingleInstanceType": true,
 "SingleAvailabilityZone": true,
 "MinTargetCapacity": 0,
 "MaxTotalPrice": ""
 },
 "OnDemandOptions": {
 "AllocationStrategy": "prioritized",
 "CapacityReservationOptions": {
```

```
 "UsageStrategy": "use-capacity-reservations-first"
 },
 "SingleInstanceType": true,
 "SingleAvailabilityZone": true,
 "MinTargetCapacity": 0,
 "MaxTotalPrice": ""
},
"ExcessCapacityTerminationPolicy": "termination",
"LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateId": "",
 "LaunchTemplateName": "",
 "Version": ""
 },
 "Overrides": [
 {
 "InstanceType": "r5.metal",
 "MaxPrice": "",
 "SubnetId": "",
 "AvailabilityZone": "",
 "WeightedCapacity": 0.0,
 "Priority": 0.0,
 "Placement": {
 "AvailabilityZone": "",
 "Affinity": "",
 "GroupName": "",
 "PartitionNumber": 0,
 "HostId": "",
 "Tenancy": "dedicated",
 "SpreadDomain": "",
 "HostResourceGroupArn": ""
 }
 },
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 0,
 "Max": 0
 },
 "MemoryMiB": {
 "Min": 0,
 "Max": 0
 },
 "CpuManufacturers": [
 "amd"
```

```
],
 "MemoryGiBPerVCpu": {
 "Min": 0.0,
 "Max": 0.0
 },
],
 "ExcludedInstanceTypes": [
 ""
],
],
 "InstanceGenerations": [
 "previous"
],
],
 "SpotMaxPricePercentageOverLowestPrice": 0,
 "OnDemandMaxPricePercentageOverLowestPrice": 0,
 "BareMetal": "included",
 "BurstablePerformance": "required",
 "RequireHibernateSupport": true,
 "NetworkInterfaceCount": {
 "Min": 0,
 "Max": 0
 },
],
 "LocalStorage": "excluded",
 "LocalStorageTypes": [
 "ssd"
],
],
 "TotalLocalStorageGB": {
 "Min": 0.0,
 "Max": 0.0
 },
],
 "BaselineEbsBandwidthMbps": {
 "Min": 0,
 "Max": 0
 },
],
 "AcceleratorTypes": [
 "inference"
],
],
 "AcceleratorCount": {
 "Min": 0,
 "Max": 0
 },
],
 "AcceleratorManufacturers": [
 "amd"
],
],
 "AcceleratorNames": [
 "a100"
```

```

],
 "AcceleratorTotalMemoryMiB": {
 "Min": 0,
 "Max": 0
 }
 }
}
],
"TargetCapacitySpecification": {
 "TotalTargetCapacity": 0,
 "OnDemandTargetCapacity": 0,
 "SpotTargetCapacity": 0,
 "DefaultTargetCapacityType": "on-demand",
 "TargetCapacityUnitType": "memory-mib"
},
"TerminateInstancesWithExpiration": true,
"Type": "instant",
"ValidFrom": "1970-01-01T00:00:00",
"ValidUntil": "1970-01-01T00:00:00",
"ReplaceUnhealthyInstances": true,
"TagSpecifications": [
 {
 "ResourceType": "fleet",
 "Tags": [
 {
 "Key": "",
 "Value": ""
 }
]
 }
]
},
"Context": ""
}

```

## 建立 EC2 Fleet

若要建立 EC2 Fleet，您只需指定下列參數：

- `LaunchTemplateId` 或 `LaunchTemplateName` – 指定要使用的啟動範本 (其中包含要啟動的執行個體參數，例如執行個體類型、可用區域和您願意支付的最高價格)

- TotalTargetCapacity – 指定機群的總目標容量
- DefaultTargetCapacityType – 指定預設購買選項是隨需還是 Spot

您可指定多個覆寫啟動範本的啟動規格。啟動規格因執行個體類型、可用區域、子網和最高價格而異，且可以包含不同的加權容量。或者，您可以指定執行個體必須具有的屬性，然後 Amazon EC2 會識別具有這些屬性的所有執行個體類型。如需詳細資訊，請參閱 [EC2 Fleet 的屬性型執行個體類型選擇](#)。

如果您未指定參數，則機群會使用參數的預設值。

指定 JSON 檔案中的機群參數。如需詳細資訊，請參閱 [產生 EC2 Fleet JSON 組態檔案](#)。

目前沒有主控台支援建立 EC2 Fleet。

若要建立 EC2 Fleet (AWS CLI)

- 使用 [create-fleet](#) (AWS CLI) 命令來建立 EC2 Fleet，並指定包含機群組態參數的 JSON 檔案。

```
aws ec2 create-fleet --cli-input-json file://file_name.json
```

如需組態檔案範例，請參閱 [EC2 Fleet 範例組態](#)。

以下是類型 request 或 maintain 的叢集輸出範例。

```
{
 "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"
}
```

以下是啟動目標容量之類型 instant 的叢集輸出範例。

```
{
 "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE",
 "Errors": [],
 "Instances": [
 {
 "LaunchTemplateAndOverrides": {
 "LaunchTemplateSpecification": {
 "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
 "Version": "1"
 },
 },
 "Overrides": {
```

```

 "InstanceType": "c5.large",
 "AvailabilityZone": "us-east-1a"
 }
},
"Lifecycle": "on-demand",
"InstanceIds": [
 "i-1234567890abcdef0",
 "i-9876543210abcdef9"
],
"InstanceType": "c5.large",
"Platform": null
},
{
 "LaunchTemplateAndOverrides": {
 "LaunchTemplateSpecification": {
 "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
 "Version": "1"
 },
 "Overrides": {
 "InstanceType": "c4.large",
 "AvailabilityZone": "us-east-1a"
 }
 },
 "Lifecycle": "on-demand",
 "InstanceIds": [
 "i-5678901234abcdef0",
 "i-5432109876abcdef9"
]
}
]
}

```

以下是啟動部分目標容量並附有未啟動之執行個體錯誤之類型 `instant` 的叢集輸出範例。

```

{
 "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE",
 "Errors": [
 {
 "LaunchTemplateAndOverrides": {
 "LaunchTemplateSpecification": {
 "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
 "Version": "1"
 },
 "Overrides": {

```

```

 "InstanceType": "c4.xlarge",
 "AvailabilityZone": "us-east-1a",
 }
},
"Lifecycle": "on-demand",
"ErrorCode": "InsufficientInstanceCapacity",
"ErrorMessage": ""
},
],
"Instances": [
 {
 "LaunchTemplateAndOverrides": {
 "LaunchTemplateSpecification": {
 "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
 "Version": "1"
 },
 "Overrides": {
 "InstanceType": "c5.large",
 "AvailabilityZone": "us-east-1a"
 }
 },
 "Lifecycle": "on-demand",
 "InstanceIds": [
 "i-1234567890abcdef0",
 "i-9876543210abcdef9"
]
 }
]
}

```

以下是未啟動任何執行個體之類型 `instant` 的叢集輸出範例。

```

{
 "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE",
 "Errors": [
 {
 "LaunchTemplateAndOverrides": {
 "LaunchTemplateSpecification": {
 "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
 "Version": "1"
 },
 "Overrides": {
 "InstanceType": "c4.xlarge",
 "AvailabilityZone": "us-east-1a",

```



```
 }
 },
 "Lifecycle": "on-demand",
 "ErrorCode": "InsufficientCapacity",
 "ErrorMessage": ""
},
{
 "LaunchTemplateAndOverrides": {
 "LaunchTemplateSpecification": {
 "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
 "Version": "1"
 },
 "Overrides": {
 "InstanceType": "c5.large",
 "AvailabilityZone": "us-east-1a",
 }
 },
 "Lifecycle": "on-demand",
 "ErrorCode": "InsufficientCapacity",
 "ErrorMessage": ""
},
],
"Instances": []
}
```

## 標記 EC2 Fleet

為便於將 EC2 Fleet 請求分類和進行管理，您可以利用自訂中繼資料來標記這些請求。您可以在建立 EC2 Fleet 請求時，或建立之後，將標籤指派給該請求。

當您標記機群請求時，由機群啟動的執行個體和磁碟區不會自動加上標記。您需要明確標記由機群啟動的執行個體和磁碟區。您可以選擇僅將標籤指派給機群請求，或僅指派給機群啟動的執行個體，或僅指派給連接至機群啟動之執行個體的磁碟區，或指派給前述三項。

### Note

針對 `instant` 機群類型，您可以為連接至 隨需執行個體 和 競價型執行個體 的磁碟區加標籤。針對 `request` 或 `maintain` 機群類型，您只可以為連接至 隨需執行個體 的磁碟區加標籤。

如需有關標籤運作方式的詳細資訊，請參閱 [標記您的 Amazon EC2 資源](#)。

## 必要條件

授予使用者標記資源的許可。如需詳細資訊，請參閱 [範例：標籤資源](#)。

### 授予使用者標記資源的許可

建立包含下列項目的 IAM 政策：

- `ec2:CreateTags` 動作。這會授予使用者建立標籤的許可。
- `ec2:CreateFleet` 動作。這會授予使用者建立 EC2 Fleet 請求的許可。
- 針對 `Resource`，建議您指定 `"*"`。這可讓使用者標記所有資源類型。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "TagEC2FleetRequest",
 "Effect": "Allow",
 "Action": [
 "ec2:CreateTags",
 "ec2:CreateFleet"
],
 "Resource": "*"
 }
]
}
```

### Important

我們目前不支援 `create-fleet` 資源的資源層級許可。如果您指定 `create-fleet` 作為資源，當您嘗試標記機群時，會收到未經授權的例外狀況。下列範例說明如何不要設定政策。

```
{
 "Effect": "Allow",
 "Action": [
 "ec2:CreateTags",
 "ec2:CreateFleet"
],
 "Resource": "arn:aws:ec2:us-east-1:111122223333:create-fleet/*"
}
```

若要提供存取權，請新增權限至您的使用者、群組或角色：

- 使用者和群組位於 AWS IAM Identity Center：

建立權限合集。請按照 AWS IAM Identity Center 使用者指南 中的 [建立權限合集](#) 說明進行操作。

- 透過身分提供者在 IAM 中管理的使用者：

建立聯合身分的角色。請按照 IAM 使用者指南 的 [為第三方身分提供者 \(聯合\) 建立角色](#) 中的指示進行操作。

- IAM 使用者：

- 建立您的使用者可擔任的角色。請按照 IAM 使用者指南 的 [為 IAM 使用者建立角色](#) 中的指示進行操作。

- (不建議) 將政策直接附加至使用者，或將使用者新增至使用者群組。請遵循 IAM 使用者指南的 [新增許可到使用者 \(主控台\)](#) 中的指示。

## 標記新的 EC2 Fleet 請求

若要在建立 EC2 Fleet 請求時加以標記，請在建立機群時所用的 [JSON 檔案](#) 中指定金鑰值對。ResourceType 的值必須為 fleet。若您指定其他值，機群請求將會失敗。

## 標記由 EC2 Fleet 啟動的執行個體和磁碟區

若要在機群啟動執行個體時標記這些執行個體和磁碟區，請在 EC2 Fleet 請求中參考的 [啟動範本](#) 中指定標籤。

### Note

您無法標記連接至 競價型執行個體、由 request 或 maintain 機群類型啟動的磁碟區。

若要標記現有的 EC2 Fleet 請求、執行個體和磁碟區 (AWS CLI)

使用 [建立標籤](#) 命令來標記現有資源。

```
aws ec2 create-tags \
 --resources fleet-12a34b55-67cd-8ef9-
ba9b-9208dEXAMPLE i-1234567890abcdef0 vol-1234567890EXAMPLE \
 --tags Key=purpose,Value=test
```

## 說明您的 EC2 機群

您可說明您的 EC2 機群組態、您 EC2 機群中的執行個體，及您 EC2 機群的事件歷史記錄。

如要說明您的 EC2 機群 (AWS CLI)

使用 [describe-fleets](#) 命令來說明您的 EC2 Fleet。

```
aws ec2 describe-fleets
```

### Important

若機群類型為 `instant`，則必須指定機群 ID，否則其不會顯示於回應中。包括 `--fleet-ids`，如下所示：

```
aws ec2 describe-fleets --fleet-ids fleet-8a22eee4-f489-ab02-06b8-832a7EXAMPLE
```

## 範例輸出

```
{
 "Fleets": [
 {
 "ActivityStatus": "fulfilled",
 "CreateTime": "2022-02-09T03:35:52+00:00",
 "FleetId": "fleet-364457cd-3a7a-4ed9-83d0-7b63e51bb1b7",
 "FleetState": "active",
 "ExcessCapacityTerminationPolicy": "termination",
 "FulfilledCapacity": 2.0,
 "FulfilledOnDemandCapacity": 0.0,
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "my-launch-template",
 "Version": "$Latest"
 }
 }
],
 "TargetCapacitySpecification": {
 "TotalTargetCapacity": 2,

```

```

 "OnDemandTargetCapacity": 0,
 "SpotTargetCapacity": 2,
 "DefaultTargetCapacityType": "spot"
 },
 "TerminateInstancesWithExpiration": false,
 "Type": "maintain",
 "ReplaceUnhealthyInstances": false,
 "SpotOptions": {
 "AllocationStrategy": "capacity-optimized",
 "InstanceInterruptionBehavior": "terminate"
 },
 "OnDemandOptions": {
 "AllocationStrategy": "lowestPrice"
 }
}
]
}

```

使用 [describe-fleet-instances](#) 命令來說明特定 EC2 Fleet 的執行個體。執行中執行個體的傳回清單會定期重新整理且可能過期。

```
aws ec2 describe-fleet-instances --fleet-id fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE
```

### 範例輸出

```

{
 "ActiveInstances": [
 {
 "InstanceId": "i-09cd595998cb3765e",
 "InstanceHealth": "healthy",
 "InstanceType": "m4.large",
 "SpotInstanceRequestId": "sir-86k84j6p"
 },
 {
 "InstanceId": "i-09cf95167ca219f17",
 "InstanceHealth": "healthy",
 "InstanceType": "m4.large",
 "SpotInstanceRequestId": "sir-dvxi7fsm"
 }
],
 "FleetId": "fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE"
}

```

使用 [describe-fleet-history](#) 命令來說明特定時間之特定 EC2 Fleet 的歷史記錄。

```
aws ec2 describe-fleet-history --fleet-id fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE --start-time 2018-04-10T00:00:00Z
```

### 範例輸出

```
{
 "HistoryRecords": [
 {
 "EventInformation": {
 "EventSubType": "submitted"
 },
 "EventType": "fleetRequestChange",
 "Timestamp": "2020-09-01T18:26:05.000Z"
 },
 {
 "EventInformation": {
 "EventSubType": "active"
 },
 "EventType": "fleetRequestChange",
 "Timestamp": "2020-09-01T18:26:15.000Z"
 },
 {
 "EventInformation": {
 "EventDescription": "t2.small, ami-07c8bc5c1ce9598c3, ...",
 "EventSubType": "progress"
 },
 "EventType": "fleetRequestChange",
 "Timestamp": "2020-09-01T18:26:17.000Z"
 },
 {
 "EventInformation": {
 "EventDescription": "{\"instanceType\":\"t2.small\", ...}",
 "EventSubType": "launched",
 "InstanceId": "i-083a1c446e66085d2"
 },
 "EventType": "instanceChange",
 "Timestamp": "2020-09-01T18:26:17.000Z"
 },
 {
 "EventInformation": {
 "EventDescription": "{\"instanceType\":\"t2.small\", ...}",
```

```
 "EventSubType": "launched",
 "InstanceId": "i-090db02406cc3c2d6"
 },
 "EventType": "instanceChange",
 "Timestamp": "2020-09-01T18:26:17.000Z"
}
],
"FleetId": "fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
>LastEvaluatedTime": "1970-01-01T00:00:00.000Z",
"StartTime": "2018-04-09T23:53:20.000Z"
}
```

## 修改 EC2 Fleet

您可以修改處於 submitted 或 active 狀態中的 EC2 Fleet。當您修改機群時，機群會進入 modifying 狀態。

您僅可以修改 maintain 類型的 EC2 Fleet。您無法修改 request 或 instant 類型的 EC2 Fleet。

您可以修改 EC2 Fleet 的下列參數：

- `target-capacity-specification` – 增加或減少 `TotalTargetCapacity`、`OnDemandTargetCapacity` 和 `SpotTargetCapacity` 的目標容量。
- `excess-capacity-termination-policy` – 如果機群的總目標容量減少到低於 EC2 Fleet 的目前大小，是否應終止執行中的執行個體。有效值為 `no-termination` 和 `termination`。

當您增加目標容量時，EC2 Fleet 會根據為 `DefaultTargetCapacityType` 所指定的執行個體購買選項，來啟動其他的執行個體，這些選項為 隨需執行個體 或 競價型執行個體。

如果 `DefaultTargetCapacityType` 是 `spot`，EC2 叢集會根據其 [配置策略](#) 啟動額外的 Spot 執行個體。

當您降低目標容量時，EC2 Fleet 會刪除超過新目標容量的任何開放請求。您可以請求機群終止執行個體，直到機群大小達到新的目標容量。如果分配策略為 `lowest-price`，則機群會以最高每單位價格終止執行個體。如果分配策略為 `diversified`，則機群會在集區中終止執行個體。或者，您可以請求該 EC2 Fleet 將機群保持為目前的大小，但不能取代任何中斷的 競價型執行個體 或任何您手動終止的執行個體。

當 EC2 Fleet 因為目標容量減少而終止 Spot 執行個體時，執行個體會收到 Spot 執行個體中斷通知。

## 若要修改 EC2 Fleet (AWS CLI)

使用 [modify-fleet](#) 命令來更新特定 EC2 Fleet 的目標容量。

```
aws ec2 modify-fleet \
 --fleet-id fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
 --target-capacity-specification TotalTargetCapacity=20
```

如果您要降低目標容量，但希望保持目前的大小，則可以按照下列方式修改前一個命令。

```
aws ec2 modify-fleet \
 --fleet-id fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
 --target-capacity-specification TotalTargetCapacity=10 \
 --excess-capacity-termination-policy no-termination
```

## 刪除 EC2 Fleet

如果您不再需要某個 EC2 Fleet，您可以將其刪除。刪除機群後，也會取消與機群相關的所有 Spot 請求，因此不會啟動新的 Spot 執行個體。

刪除 EC2 Fleet 時，必須指定是否也要終止其所有執行個體。這包含隨需執行個體和 Spot 執行個體。對於 instant 叢集，EC2 叢集必須在刪除叢集時終止執行個體。不支援具有執行中執行個體的已刪除 instant 機群。

如果指定在刪除機群時必須終止執行個體，機群將進入 `deleted_terminating` 狀態。否則會進入 `deleted_running` 狀態，且執行個體會一直執行直到中斷或您手動終止。

### 限制

- 您可以在單一要求中刪除多達 25 個類型 instant 的叢集。
- 您最多可以刪除 100 個類型的叢集，也可以 `request` 在 `maintain` 單一要求中刪除。
- 您可以在單一要求中刪除多達 125 個叢集，前提是您不超過每個叢集類型的配額 (如上所述)。
- 如果超過要刪除的叢集指定數目，則不會刪除任何叢集。
- 在刪除 instant 機群的單一要求中，最多可終止 1000 個執行個體。

## 若要刪除 EC2 Fleet 並終止其執行個體 (AWS CLI)

使用 [delete-fleets](#) 命令和 `--terminate-instances` 參數來刪除指定的 EC2 Fleet 並終止其相關執行個體。



```
aws ec2 delete-fleets \
 --fleet-ids fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
 --terminate-instances
```

### 範例輸出

```
{
 "UnsuccessfulFleetDeletions": [],
 "SuccessfulFleetDeletions": [
 {
 "CurrentFleetState": "deleted_terminating",
 "PreviousFleetState": "active",
 "FleetId": "fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE"
 }
]
}
```

### 若要刪除 EC2 Fleet 但不終止其執行個體 (AWS CLI)

您可以使用 `--no-terminate-instances` 參數來修改前一個命令以刪除指定的 EC2 Fleet，而不需終止其相關執行個體。

#### Note

`--no-terminate-instances` 不支援 instant 機群。

```
aws ec2 delete-fleets \
 --fleet-ids fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
 --no-terminate-instances
```

### 範例輸出

```
{
 "UnsuccessfulFleetDeletions": [],
 "SuccessfulFleetDeletions": [
 {
 "CurrentFleetState": "deleted_running",
 "PreviousFleetState": "active",
 "FleetId": "fleet-4b8aaae8-dfb5-436d-a4c6-3dafa4c6b7dcEXAMPLE"
 }
]
}
```

```

 }
]
}

```

### 當機群無法刪除時進行故障診斷

如果刪除 EC2 Fleet 失敗，則輸出中的 `UnsuccessfulFleetDeletions` 會傳回的 EC2 Fleet ID、錯誤碼和錯誤訊息。

錯誤代碼為：

- `ExceededInstantFleetNumForDeletion`
- `fleetIdDoesNotExist`
- `fleetIdMalformed`
- `fleetNotInDeletableState`
- `NoTerminateInstancesNotSupported`
- `UnauthorizedOperation`
- `unexpectedError`

### `ExceededInstantFleetNumForDeletion` 疑難排解

如果您在單一請求中嘗試刪除超過 25 個 instant 機群，則會傳回 `ExceededInstantFleetNumForDeletion` 錯誤。下列為此錯誤的範例輸出。

```

{
 "UnsuccessfulFleetDeletions": [
 {
 "FleetId": " fleet-5d130460-0c26-bfd9-2c32-0100a098f625",
 "Error": {
 "Message": "Can't delete more than 25 instant fleets in a single
request.",
 "Code": "ExceededInstantFleetNumForDeletion"
 }
 },
 {
 "FleetId": "fleet-9a941b23-0286-5bf4-2430-03a029a07e31",
 "Error": {
 "Message": "Can't delete more than 25 instant fleets in a single
request.",
 "Code": "ExceededInstantFleetNumForDeletion"
 }
 }
]
}

```

```

 }
 }
 .
 .
 .
],
 "SuccessfulFleetDeletions": []
}

```

## NoTerminateInstancesNotSupported 疑難排解

如果您指定在刪除 instant 機群時不得終止該機群中的執行個體，則會傳回

NoTerminateInstancesNotSupported 錯誤。--no-terminate-instances 不支援 instant 機群。下列為此錯誤的範例輸出。

```

{
 "UnsuccessfulFleetDeletions": [
 {
 "FleetId": "fleet-5d130460-0c26-bfd9-2c32-0100a098f625",
 "Error": {
 "Message": "NoTerminateInstances option is not supported for
instant fleet",
 "Code": "NoTerminateInstancesNotSupported"
 }
 }
],
 "SuccessfulFleetDeletions": []
}

```

## UnauthorizedOperation 疑難排解

如果沒有終止執行個體的許可，您在刪除必須終止其執行個體的機群時，會收到

UnauthorizedOperation 錯誤。以下是錯誤回應。

```

<Response><Errors><Error><Code>UnauthorizedOperation</Code><Message>You are not
authorized to perform this
operation. Encoded authorization failure message: VvuncIxj7Z_CPGNYXWqnuFV-
YjByeAU66Q9752NtQ-I3-qnDLWs6JLFd
KnSMmiq5s6cGqjjPtEDpsnGHzyHasFH0aRYJpaDVravoW25azn6KNkUQQ1FwhJyujt2dtNCdduJfrqcFYAj1EiRMkfDht7
BHturzDK6A560Y2nDSUiMmAB1y9UNTqaZJ9SNe5sNxKMqZaqKtjRbk02RZu5V2vn9VMk6fm2aMVHbY9JhLvGypLcMUjtJ76
VPiU5v2s-
UgZ7h0p2yth6ysUdh10Ng6dBYu8_y_HtEI54invCj4CoK0qawqzMNe6rcmCQHvtCxtXsbkgyaEbcwmrm2m01-
EMhekLFZeJLr

```

```
DtY0pYcE14_nWFX1wtQDCnNNCmxnJZAoJvb3VMDYpDTsxjQv1Px0DZuqWHS23YXWVywzgnLtHeRf2o4lUhGBw17mXsS07k7
PT9vrHtQiILor5VVTsjSPWg7edj__1rsnXhwPSu8gI48ZLRGrPQqFq0RmK0_QIE8N8s6NWzCK4yoX-9gDcheur0GpkprPIC
</Message></Error></Errors><RequestID>89b1215c-7814-40ae-a8db-41761f43f2b0</
RequestID></Response>
```

若要解決錯誤，您必須將 `ec2:TerminateInstances` 動作新增至 IAM 政策，如下列範例所示。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "DeleteFleetsAndTerminateInstances",
 "Effect": "Allow",
 "Action": [
 "ec2:DeleteFleets",
 "ec2:TerminateInstances"
],
 "Resource": "*"
 }
]
}
```

# Spot 叢集

Spot 機群是一組 Spot 執行個體和可選的隨需執行個體 (根據您指定的條件啟動)。Spot 執行個體會選取符合您需求的 Spot 容量集區，並啟動 Spot 執行個體以滿足該機群的目標容量。根據預設，Spot Fleets 會設定為維持目標容量，維持的方式是在機群中的競價型執行個體終止後，啟動替換的執行個體。您可以用一次性請求的形式來提交 Spot Fleet，在執行個體終止之後，此等請求也不會存續。您可以在 Spot Fleet 請求中包含隨需執行個體請求。

## Note

如果想要使用主控台建立包含 Spot 執行個體的機群，則建議使用 Auto Scaling 群組而非 Spot 機群。如需詳細資訊，請參閱《Amazon EC2 Auto Scaling 使用者指南》中的[具備多個執行個體類型及購買選項的 Auto Scaling 群組](#)。

如果您想要使用建立包含 Spot 執行個體的叢集，建議您使用 Auto Scaling 群組或 EC2 叢集而非 Spot 叢集。AWS CLI Spot [RequestSpot](#) 艦隊所基於的艦隊 API 是一種傳統 API，無需計劃投資。

如需有關要使用的建議 API 的詳細資訊，請參閱 [使用哪種 Spot 請求方法最好？](#)

## 主題

- [Spot Fleet 請求類型](#)
- [Spot Fleet 組態策略](#)
- [使用 Spot Fleet](#)
- [CloudWatch 競價型艦隊的指標](#)
- [Spot Fleet 的自動擴展](#)

## Spot Fleet 請求類型

Spot Fleet 請求有兩種類型：

### request

如果您將請求類型設定為 request，Spot Fleet 會針對您所需容量送出非同步的一次性請求。如果容量因為 Spot 中斷而減少，機群不會嘗試補充 Spot 執行個體；如果無法使用容量，也不會在替代的 Spot 容量集區中提交請求。

## maintain

如果您將請求類型設定為 `maintain`，Spot Fleet 會針對您所需容量送出非同步請求，並自動補充所有中斷的 Spot 執行個體以維持容量。

若要在 Amazon EC2 主控台中指定請求類型，請在建立 Spot Fleet 請求時執行下列操作：

- 若要建立 `request` 類型 Spot Fleet，請清除 `Maintain target capacity` (維護目標容量) 核取方塊。
- 若要建立 `maintain` 類型 Spot Fleet，請選取 `Maintain target capacity` (維護目標容量) 核取方塊。

如需詳細資訊，請參閱 [使用已定義的參數建立 Spot Fleet 請求 \(主控台\)](#)。

這兩種請求類型都受益於分配策略。如需詳細資訊，請參閱 [競價型執行個體的分配策略](#)。

## Spot Fleet 組態策略

Spot Fleet 是 Spot 執行個體和可選隨需執行個體的集合或機群。

Spot Fleet 會嘗試啟動 Spot 執行個體和隨需執行個體的數量，以符合您在 Spot Fleet 請求中指定的目標容量。如果有可用容量且您在請求中指定的價格上限超過目前 Spot 價格，則會實現競價型執行個體的請求。如果您的 Spot 執行個體中斷，Spot Fleet 也會嘗試維持其目標容量機群。

您也可以設定您願意為機群支付的每個小時金額上限，而 Spot Fleet 會啟動執行個體，直到達到金額上限為止。達到您願意支付的金額上限時，機群會停止啟動執行個體，即使它未符合目標容量亦然。

Spot 容量集區是一組未使用的 EC2 執行個體，這些執行個體擁有相同的執行個體類型 (例如 `m5.large`)、作業系統、可用區域和網路平台。建立 Spot Fleet 請求時，您可以加入多個啟動規格，這些規格會隨執行個體類型、AMI、可用區域或子網而有不同。根據 Spot Fleet 請求中包含的啟動規格以及 Spot Fleet 請求的組態，Spot Fleet 會選取用於實現請求的 Spot 容量集區。競價型執行個體來自選取的集區。

### 目錄

- [規劃 Spot Fleet 請求](#)
- [競價型執行個體的分配策略](#)
- [Spot Fleet 的屬性型執行個體類型選擇](#)
- [Spot Fleet 中的隨需](#)
- [容量重新平衡](#)

- [Spot 價格覆寫](#)
- [控制花費](#)
- [Spot Fleet 執行個體權重](#)

## 規劃 Spot Fleet 請求

建立 Spot Fleet 請求之前，請參閱 [Spot 最佳實務](#)。在規劃 Spot Fleet 請求時採用這些最佳實務做法，如此就能夠以最低價格來佈建您所要的執行個體類型。我們也建議您進行下列動作：

- 決定是否要建立 Spot Fleet，它針對所需的目標容量提交一次性請求，或是隨時間維持目標容量。
- 決定滿足您應用程式需求的執行個體類型。
- 決定 Spot Fleet 請求的目標容量。您可在執行個體或自訂單位中設定目標容量。如需詳細資訊，請參閱 [Spot Fleet 執行個體權重](#)。
- 決定 Spot Fleet 目標容量的哪些部分必須是隨需容量。隨需容量可以指定 0。
- 指定您每單位的價格，如您使用執行個體權重。若要計算每單位價格，請將每個執行個體小時的價格除以執行個體代表的單位數 (或權重)。如您不使用執行個體權重，預設的每單位價格即為每執行個體小時。
- 檢視 Spot Fleet 請求可能的選項。如需詳細資訊，請參閱 AWS CLI 命令參考中的 [request-spot-fleet](#) 命令。如需額外的範例，請參閱 [Spot Fleet 範例組態](#)。

## 競價型執行個體 的分配策略

您的啟動組態會決定 Spot 機群可從中啟動 Spot 執行個體的所有可能 Spot 容量集區 (執行個體類型和可用區域)。但是，啟動執行個體時，Spot 機群會使用您指定的配置策略，從所有可能的集區中挑選特定的集區。

### Note

(僅限 Linux 執行個體) 如果您將競價型執行個體設定為在 [AMD SEV-SNP](#) 開啟的情況下啟動，則系統會變更額外的小時使用費，相當於所選執行個體類型的 [隨需小時費率](#) 10%。如果配置策略使用價格作為輸入，則 EC2 機群不會包含此額外費用；僅使用 Spot 價格。

## 分配策略

您可以為 Spot 執行個體指定下列其中一個配置策略：

## priceCapacityOptimized(建議使用)

Spot 機群會識別擁有啟動中執行個體數目之最高容量可用性的集區。這表示我們將從我們認為在短期內中斷機會最低的集區請求 Spot 執行個體。Spot 機群接著會從這些最低價集區中請求 Spot 執行個體。

priceCapacityOptimized 分配策略是大多數 Spot 工作負載的最佳選擇，例如無狀態容器化應用程式、微服務、Web 應用程式、資料和分析作業以及批次處理。

## capacityOptimized

Spot 機群會識別擁有啟動中執行個體數目之最高容量可用性的集區。這表示我們將從我們認為在短期內中斷機會最低的集區請求 Spot 執行個體。您可以使用 capacityOptimizedPrioritized 選擇性地設定機群中每個執行個體類型的優先順序。Spot Fleet 會首先最佳化容量，但會盡力實現執行個體類型的優先順序。

使用 競價型執行個體，定價會依據長期的供需趨勢隨時間慢慢改變，但是容量會即時波動。capacityOptimized 策略會查看即時容量資料並預測哪些資料為最多可用，從而自動將 Spot 執行個體啟動到最可用的集區。這適用於可能具有較高的重新啟動工作中斷成本的工作負載，例如持續整合 (CI)、影像和媒體轉譯、深度學習以及可能具有較高的重新啟動工作中斷成本的高效能運算 (HPC) 工作負載。capacityOptimized 策略可實現盡可能少的中斷，因此可降低您工作負載的整體成本。

或者，您可以使用具有優先順序參數的 capacityOptimizedPrioritized 分配策略，以從最高到最低的優先順序對執行個體類型排序。您可以為不同的執行個體類型設定相同的優先順序。Spot Fleet 會先針對容量進行最佳化，但會盡力遵守執行個體類型的優先順序 (例如，如果遵守優先順序不會顯著影響 Spot Fleet 佈建最佳容量的能力)。對於必須將中斷可能性降至最低的工作負載來說，這是一個很好的選擇，而且某些執行個體類型的偏好也很重要。只有在您的機群使用啟動範本時，才支援使用優先順序。請注意，當您設定 capacityOptimizedPrioritized 的優先順序時，如果隨需執行個體 AllocationStrategy 設定為 prioritized，則相同的優先順序也會套用至您的隨需執行個體。

## diversified

競價型執行個體分佈在所有集區中。

## 選擇適當的分配策略

您可以選擇適當的 Spot 分配策略，進而針對您的使用案例最佳化機群。對於隨需執行個體目標容量，Spot 叢集一律會根據公有隨需價格選擇最便宜的執行個體類型，同時遵循 Spot 執行個體的配置策略 priceCapacityOptimized diversified (或)。capacityOptimized



## 平衡最低價格與容量可用性

為了平衡價格最低的 Spot 容量集區與具有最高容量可用性的 Spot 容量集區之間的利弊，建議您使用 `priceCapacityOptimized` 分配策略。此策略會根據集區的價格和這些集區中 Spot 執行個體的容量可用性，決定要從哪些集區請求 Spot 執行個體的集區。這表示我們將從我們認為在短期內中斷機會最低的集區請求 Spot 執行個體，同時我們仍會將價格納入考慮。

如果您的機群執行彈性且無狀態的工作負載，包括容器化應用程式、微服務、Web 應用程式、資料和分析作業以及批次處理，則使用 `priceCapacityOptimized` 分配策略，以達到最佳的成本節省和容量可用性。

如果您的機群執行的工作負載有較高的重新啟動工作中斷的成本，則您應該實作檢查點作業，以便應用程式可以在中斷時從該點重新啟動。透過使用檢查點作業，您可以讓 `priceCapacityOptimized` 分配策略非常適合這些工作負載，因為其可以從價格最低的集區中分配容量，而這些集區也會提供較低的 Spot 執行個體中斷率。

如需使用 `priceCapacityOptimized` 分配策略的範例組態，請參閱 [範例 9：在具有優先順序的容量最佳化叢集中啟動 Spot 執行個體](#)。

### 當工作負載具有較高的中斷成本時

如果您執行的工作負載使用價格相似的執行個體類型，或是中斷成本非常之高，以致於任何成本節省都不足以彌補中斷的邊際增加，則您可以選擇是否使用該 `capacityOptimized` 策略。該策略可從最可用的 Spot 容量集區分配容量，而這些集區會提供較少中斷的可能性，進而降低您工作負載的整體成本。如需使用 `capacityOptimized` 分配策略的範例組態，請參閱 [範例 7：設定容量重新平衡以啟動替代 Spot 執行個體](#)。

當必須將中斷可能性降至最低，而且對於某些執行個體類型的偏好也很重要時，您可以使用 `capacityOptimizedPrioritized` 分配策略，然後設定執行個體類型的順序，來表現您的集區優先順序，以從最高到最低優先順序使用。如需範例組態，請參閱 [範例 8：在容量最佳化叢集中啟動 Spot 執行個體](#)。

請注意，只有在您的機群使用啟動範本時，才支援使用優先順序。另請注意，當您設定 `capacityOptimizedPrioritized` 的優先順序時，如果隨需執行個體 `AllocationStrategy` 設定為 `prioritized`，則相同的優先順序也會套用至您的隨需執行個體。

### 當您的工作負載具有時間彈性且容量可用性並非相關因素時

如果您的機群很小或執行時間很短，您仍可在考慮容量可用性的同時，使用 `priceCapacityOptimized` 來實現成本節省最大化。

## 當您的機群很龐大或執行時間較長時

如果您的機群很龐大或執行時間較長，則可以使用 `diversified` 策略，透過將多個競價型執行個體分散到多個集區中來提高機群的可用性。例如，如果 Spot 機群指定了 10 個集區和 100 個執行個體的目標容量，則該機群會在每個集區中啟動 10 個 Spot 執行個體。如果某個集區的 Spot 價格超過該集區的最高價格，則只有 10% 的機群會受到影響。隨著時間，使用這種策略也會使您的機群對任何一個集區 Spot 價格上漲較不敏感。使用 `diversified` 策略，Spot Fleet 不會將 Spot 執行個體啟動至任何等於或高於 [隨需價格](#) 的 Spot 價格集區中。

## 維護目標容量

在 Spot 執行個體因 Spot 價格或 Spot 容量集區可用容量的變更而終止之後，類型為 `maintain` 的 Spot Fleet 會啟動替代 Spot 執行個體。分配策略會確定從中啟動替代執行個體的集區，如下所示：

- 如果分配策略是 `priceCapacityOptimized`，則機群會在擁有最多 Spot 執行個體容量可用性的集區中啟動替代執行個體，同時將價格納入考慮並識別價格最低且具有高容量可用性的執行個體。
- 如果分配策略為 `capacityOptimized`，則機群會在擁有最多 Spot 執行個體容量可用性的集區中啟動替代執行個體。
- 如果分配策略為 `diversified`，則機群會將替代競價型執行個體分配到剩餘的集區中。

## Spot Fleet 的屬性型執行個體類型選擇

建立 Spot Fleet 時，您必須指定一或多個執行個體類型，以在機群中設定隨需執行個體和 Spot 執行個體。做為手動指定執行個體類型的替代方式，您可以指定執行個體必須具有的屬性，然後 Amazon EC2 會識別具有這些屬性的所有執行個體類型。這就是所謂的屬性型執行個體類型選項。例如，您可以指定執行個體所需的 vCPU 數目下限和上限，然後 Spot Fleet 將使用符合這些 vCPU 需求的任何可用執行個體類型來啟動執行個體。

屬性型執行個體類型選擇非常適合可對其使用哪些執行個體類型具有彈性的工作負載和架構，例如執行容器或 Web 機群、處理大數據，以及實作持續整合和部署 (CI/CD) 工具時。

## 優勢

屬性型執行個體類型選擇具有下列優勢：

- 輕鬆使用正確的執行個體類型 — 有這麼多可用的執行個體類型，因此尋找適合您工作負載的執行個體類型可能非常耗時。當您指定執行個體類型屬性時，執行個體類型會自動具有工作負載所需的屬性。

- 簡化的組態 — 若要為 Spot 叢集手動指定多個執行個體類型，您必須為每個執行個體類型建立個別的啟動範本覆寫。但是，有了屬性型執行個體類型選擇，若要提供多個執行個體類型，您只需要在啟動範本或在啟動範本覆寫中指定執行個體屬性。
- 自動使用新的執行個體類型 — 當您指定執行個體屬性而非執行個體類型時，您的叢集可以在發行時使用新一代執行個體類型，即「future 校對」叢集的配置。
- 執行個體類型彈性 — 當您指定執行個體屬性而非執行個體類型時，Spot Fleet 可以從各種各樣的執行個體類型中進行選擇，以啟動 [Spot 執行個體類型彈性的 Spot 最佳實務](#)。

## 主題

- [屬性型執行個體類型選擇的運作方式](#)
- [價格保護](#)
- [考量事項](#)
- [建立具有屬性型執行個體類型選擇的 Spot Fleet](#)
- [有效和無效的組態範例](#)
- [預覽具有所指定屬性的執行個體類型](#)

## 屬性型執行個體類型選擇的運作方式

若要在機群組態中使用屬性型執行個體類型選擇，請將執行個體類型清單取代為執行個體所需的執行個體屬性清單。Spot Fleet 將在具有所指定執行個體屬性的任何可用執行個體類型上啟動執行個體。

## 主題

- [執行個體屬性的類型](#)
- [要在何處設定屬性型執行個體類型選擇](#)
- [Spot Fleet 在佈建機群時如何使用屬性型執行個體類型選擇](#)

## 執行個體屬性的類型

您可以指定數個執行個體屬性來表示運算需求，例如：

- vCPU 計數 — 每個執行個體的 vCPUs 數目下限和上限。
- 記憶體 — 每個執行個體 GiBs 的記憶體下限和最大記憶體。
- 本機儲存 — 是否使用 EBS 或執行個體儲存磁碟區進行本機儲存。

- 高載效能 — 是否使用 T 執行個體系列，包括 T4G、T3A、T3 和 T2 類型。

如需每個屬性和預設值的說明，請參閱 Amazon EC2 API 參考 [InstanceRequirements](#) 中的。

### 要在何處設定屬性型執行個體類型選擇

根據您使用主控台還是 AWS CLI，您可以指定以屬性為基礎之執行個體類型選取的執行個體屬性，如下所示：

在主控台中，您可以在下列一個或兩個機群組態元件中指定執行個體屬性：

- 在啟動範本中，則參考機群請求中的啟動範本
- 在機群請求中

在中 AWS CLI，您可以在下列一或所有叢集組態元件中指定執行個體屬性：

- 在啟動範本中，參考機群請求中的啟動範本
- 在啟動範本覆寫中

如果想要混合使用不同 AMI 的執行個體，您可以在多個啟動範本覆寫中指定執行個體屬性。例如，不同的執行個體類型可以使用 x86 和 ARM 型處理器。

- 在啟動規格中

### Spot Fleet 在佈建機群時如何使用屬性型執行個體類型選擇

Spot Fleet 以下列方式佈建機群：

- Spot Fleet 可識別具有所指定屬性的執行個體類型。
- Spot Fleet 會使用價格保護來決定要排除哪些執行個體類型。
- Spot 叢集會根據具有相符執行個體類型的區域或可用區域，決定 AWS 要考慮從中啟動執行個體的容量集區。
- Spot Fleet 會套用指定的分配策略，以決定要從哪些容量集區中啟動執行個體。

請注意，屬性型執行個體類型選擇不會挑選要從中佈建機群的容量集區；這是分配策略的任務。可能有大量的執行個體類型具有指定的屬性，而其中一些可能很昂貴。

如果您指定分配策略，Spot Fleet 將根據指定的分配策略啟動執行個體。

- 若為 Spot 執行個體，屬性型執行個體類型選擇支援 capacityOptimizedPrioritized 和 capacityOptimized 分配策略。
- 對於隨需執行個體，屬性型執行個體類型選擇支援 lowestPrice 配置策略，從而保證 Spot 叢集將從最便宜的容量集區啟動隨需執行個體。
- 如果具有所指定執行個體屬性的執行個體類型沒有容量，則無法啟動任何執行個體，而且機群會傳回錯誤。

## 價格保護

價格保護是一項功能，可防止 Spot Fleet 使用您認為過於昂貴的執行個體類型，即使它們剛好符合您指定的屬性也一樣。要使用價格保護，請設置價格閾值。然後，當 Amazon EC2 選取具有屬性的執行個體類型時，就會排除定價高於閾值的執行個體類型。

Amazon EC2 計算價格閾值的方式如下：

- Amazon EC2 會先從符合屬性的執行個體類型中識別出價格最低的執行個體類型。
- 然後，Amazon EC2 會採用您為價格保護參數指定的值 (以百分比表示)，並將其與已識別執行個體類型的價格相乘。結果是用作價格閾值的價格。

隨需執行個體和 Spot 執行個體有不同的價格門檻。

當您使用以屬性為基礎的執行個體類型選項建立叢集時，預設會啟用價格保護。您可以保留預設值，也可以指定自己的值。

您也可以關閉價格保護。若要指出沒有價格保護閾值，請指定較高的百分比值，例如 999999。

## 主題

- [如何識別最低價執行個體類型](#)
- [隨需執行個體價格保](#)
- [競價型實例價格保護](#)
- [指定價格保護門檻](#)

## 如何識別最低價執行個體類型

Amazon EC2 透過識別與指定屬性相符的執行個體類型中最低價格的執行個體類型，來決定要依據價格閾值的價格。它通過以下方式執行此操作：

- 它首先查看與您的屬性匹配的當前一代 C，M 或 R 實例類型。如果找到任何相符項目，就會識別價格最低的執行個體類型。
- 如果沒有相符項目，則會查看符合屬性的任何目前產生的執行個體類型。如果找到任何相符項目，就會識別價格最低的執行個體類型。
- 如果沒有相符項目，就會查看符合屬性的任何上一代執行個體類型，並識別價格最低的執行個體類型。

## 隨需執行個體價格保

隨需執行個體類型的價格保護閾值是以比識別的最低價隨需執行個體類型 (*OnDemandMaxPricePercentageOverLowestPrice*) 高的百分比來計算。您可以指定您願意支付的更高百分比。如果您未指定此參數，則系統會使用預設值來計算比識別價格高出 20% 的價格保護閾值。20

例如，如果識別的隨需執行個體價格為且您指定25，則價格閾值會高出 25% 0.4271。0.4271它的計算方式如下： $0.4271 * 1.25 = 0.533875$ 計算的價格是您願意為隨需執行個體支付的最高價格，在此範例中，Amazon EC2 將排除任何成本超過的隨需執行個體類型0.533875。

## 競價型實例價格保護

根據預設，Amazon EC2 會自動套用最佳競價型執行個體價格保護，以持續從各種執行個體類型中進行選取。您也可以自行手動設定價格保護。但是，讓 Amazon EC2 為您執行此操作可以提高 Spot 容量滿足的可能性。

您可以使用下列其中一個選項手動指定價格保護。如果您手動設定價格保護，建議您使用第一個選項。

- 已識別最低價格隨需執行個體類型的百分比  
[*MaxSpotPriceAsPercentageOfOptimalOnDemandPrice*]

例如，如果識別的隨需執行個體類型價格為且您指定60，則價格閾值為 60% 0.4271。0.4271它的計算方式如下： $0.4271 * 0.60 = 0.25626$ 計算的價格是您願意為競價型執行個體支付的最高價格，在此範例中，Amazon EC2 將排除任何成本超過的競價型執行個體類型0.25626。

- 高於識別的最低價 Spot 執行個體類型 [*SpotMaxPricePercentageOverLowestPrice*] 的百分比

例如，如果已識別的競價型執行個體類型價格為且您指定25，則價格閾值高於 25% 0.1808。0.1808它的計算方式如下： $0.1808 * 1.25 = 0.226$ 計算的價格是您願意為競價型執行個體支付的最高價格，在此範例中，Amazon EC2 將排除任何成本超過的競價型執行個體類型0.266。我們不建議使用此參數，因為現貨價格可能會波動，因此您的價格保護閾值也可能會波動。

## 指定價格保護門檻

### 如要指定價格保護閾值

在建立 Spot 機群時，為屬性型執行個體類型選擇設定機群，然後執行下列作業：

- 主控台

如要指定隨需執行個體價格保護閾值，請在 Additional instance attribute (其他執行個體屬性) 下，選擇 On-demand price protection (隨需價格保護)，然後選擇 Add attribute (新增屬性)。對於 On-Demand price protection percentage (隨需價格保護百分比)，請以百分比形式輸入價格保護閾值。

如要指定 Spot 執行個體價格保護閾值，請在 Additional instance attribute (其他執行個體屬性) 下，選擇 Spot price protection (Spot 價格保護)，然後選擇 Add attribute (新增屬性)。選擇參數並以百分比輸入價格保護閾值。

- AWS CLI

如要指定隨需執行個體價格保護閾值，請於 JSON 組態檔案的 InstanceRequirements 結構中，對於 OnDemandMaxPricePercentageOverLowestPrice，以百分比形式輸入價格保護閾值。

若要指定競價型執行個體價格保護閾值，請在 JSON 設定檔的 InstanceRequirements 結構中，指定下列其中一個參數：

- 對於 MaxSpotPriceAsPercentageOfOptimalOnDemandPrice，以百分比輸入價格保護閾值。
- 對於 SpotMaxPricePercentageOverLowestPrice，以百分比輸入價格保護閾值。

如需有關建立機群的詳細資訊，請參閱 [建立具有屬性型執行個體類型選擇的 Spot Fleet](#)。

#### Note

建立 Spot 機群時，若您將 Total target capacity (總目標容量) 類型設定為 vCPUs 或 Memory (MiB) (記憶體 (MiB))(主控台) 或 TargetCapacityUnitType 為 vcpu 或 memory-mib (AWS CLI)，則價格保護閾值會根據每個 vCPU 或每個記憶體的價格進行套用，而非每個執行個體的價格。

### 考量事項

- 您可以在 Spot Fleet 中指定執行個體類型或執行個體屬性，但不能同時指定兩者。

使用 CLI 時，啟動範本覆寫將覆寫啟動範本。例如，如果啟動範本包含執行個體類型，且啟動範本覆寫包含執行個體屬性，則執行個體屬性所識別的執行個體將覆寫啟動範本中的執行個體類型。

- 使用 CLI 時，若將執行個體屬性指定為覆寫，您也無法指定權重或優先順序。
- 您最多可以在請求組態中指定四個 InstanceRequirements 結構。

## 建立具有屬性型執行個體類型選擇的 Spot Fleet

您可以使用 Amazon EC2 主控台或 AWS CLI，將機群設定為使用屬性型執行個體類型選擇。

### 主題

- [使用主控台建立 Spot Fleet](#)
- [使用建立競價型叢集 AWS CLI](#)

## 使用主控台建立 Spot Fleet

### 為屬性型執行個體類型選擇配置 Spot Fleet (主控台)

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Spot Requests (Spot 請求)，然後選擇 Request Spot Instances (請求 Spot 執行個體)。
3. 遵循步驟來建立 Spot Fleet。如需詳細資訊，請參閱 [使用已定義的參數建立 Spot Fleet 請求 \(主控台\)](#)。

在建立 Spot Fleet 時，為屬性型執行個體類型選擇設定機群，如下所示：

- a. 對於 Instance type requirements (執行個體類型請求)，選擇 Specify instance attributes that match your compute requirements (指定符合運算需求的執行個體屬性)。
- b. 對於 vCPUs，輸入所需的 vCPU 數量下限和上限。若要指定無限制，請選取 No minimum (無下限)、No maximum (無上限)，或兩者。
- c. 對於 Memory (GiB) (記憶體 (GiB))，輸入所需記憶體數量的下限和上限。若要指定無限制，請選取 No minimum (無下限)、No maximum (無上限)，或兩者。
- d. (選用) 對於 Additional instance attributes (其他執行個體屬性)，您可以選擇性地指定一或多個屬性，以更詳細地表達您的運算需求。每個額外屬性都會將進一步的限制新增至您的請求。
- e. (選用) 展開 Preview matching instance types (預覽相符的執行個體類型)，以檢視具有所指定屬性的執行個體類型。



## 使用建立競價型叢集 AWS CLI

若要為屬性型執行個體類型選取設定 Spot Fleet (AWS CLI)

使用 [request-spot-fleet](#) (AWS CLI) 命令建立 Spot Fleet。指定 JSON 檔案中的機群組態。

```
aws ec2 request-spot-fleet \
 --region us-east-1 \
 --spot-fleet-request-config file://file_name.json
```

### 範例 *file\_name*.json 檔案

下列範例包含將 Spot 機群設定為使用屬性型執行個體類型選取的參數，其後是文字說明。

```
{
 "AllocationStrategy": "priceCapacityOptimized",
 "TargetCapacity": 20,
 "Type": "request",
 "LaunchTemplateConfigs": [{
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "my-launch-template",
 "Version": "1"
 },
 "Overrides": [{
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 2
 },
 "MemoryMiB": {
 "Min": 4
 }
 }
 }
]
}
```

屬性型執行個體類型選取的屬性在 InstanceRequirements 結構中予以指定。在此範例中，會指定兩個屬性：

- VCpuCount – 至少指定 2 個 vCPU。由於未指定最大值，因此沒有最大限制。
- MemoryMiB – 至少指定 4 MiB 的記憶體。由於未指定最大值，因此沒有最大限制。

將會識別任何具有 2 個或更多 vCPU 和 4 MiB 或更多記憶體之執行個體類型。但是，當 [Spot 機群佈建機群](#) 時，價格保護和分配策略可能會排除某些執行個體類型。

如需可指定之所有可能屬性的清單和說明，請參閱 Amazon EC2 API 參考 [InstanceRequirements](#) 中的。

#### Note

當 InstanceRequirements 併入機群組態中時，必須排除 InstanceType 和 WeightedCapacity；它們無法與執行個體屬性同時決定機群組態。

JSON 也包含下列機群組態：

- "AllocationStrategy": "*priceCapacityOptimized*" – 機群中 Spot 執行個體的分配策略。
- "LaunchTemplateName": "*my-launch-template*", "Version": "*1*" – 啟動範本包含一些執行個體組態資訊，但如果指定了任何執行個體類型，則會由 InstanceRequirements 中指定的屬性進行覆寫。
- "TargetCapacity": *20* – 目標容量為 20 個執行個體。
- "Type": "*request*" – 機群的要求類型為 request。

#### 有效和無效的組態範例

如果您使用 AWS CLI 建立 Spot 叢集，則必須確定您的叢集組態有效。下列範例顯示有效和無效的組態。

當組態包含下列項目時，視為無效：

- 同時具有 Overrides 和 InstanceRequirements 的單一 InstanceType
- 兩個 Overrides 結構，一個具有 InstanceRequirements，而另一個具有 InstanceType
- 兩個在同一 InstanceRequirements 內具有重疊屬性值的 LaunchTemplateSpecification 結構

#### 範例組態

- [有效組態：具有覆寫的單一啟動範本](#)
- [有效組態：具有多個的單一啟動範本 InstanceRequirements](#)

- [有效組態：兩個啟動範本，每個都具有覆寫](#)
- [有效組態：僅限指定的 InstanceRequirements，沒有重疊的屬性值](#)
- [組態無效：Overrides 包含 InstanceRequirements 和 InstanceType](#)
- [組態無效：兩個 Overrides 包含 InstanceRequirements 和 InstanceType](#)
- [組態無效：重疊屬性值](#)

有效組態：具有覆寫的單一啟動範本

下列組態有效。其包含一個啟動範本和一個 Overrides 結構 (其中包含一個 InstanceRequirements 結構)。範例組態的文字說明如下。

```
{
 "SpotFleetRequestConfig": {
 "AllocationStrategy": "priceCapacityOptimized",
 "ExcessCapacityTerminationPolicy": "default",
 "IamFleetRole": "arn:aws:iam::000000000000:role/aws-ec2-spot-fleet-tagging-
role",
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "My-launch-template",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 2,
 "Max": 8
 },
 "MemoryMib": {
 "Min": 0,
 "Max": 10240
 },
 "MemoryGiBPerVCpu": {
 "Max": 10000
 },
 "RequireHibernateSupport": true
 }
 }
]
 }
]
 }
}
```

```
 }
],
 "TargetCapacity": 5000,
 "OnDemandTargetCapacity": 0,
 "TargetCapacityUnitType": "vcpu"
}
}
```

## InstanceRequirements

若要使用屬性型執行個體選擇，您必須在組態中包含 InstanceRequirements 結構，並為機群中的執行個體指定所需的屬性。

在上述範例中，指定了下列執行個體屬性：

- VCpuCount – 執行個體類型必須具有最少 2 個，最多 8 個的 vCPU。
- MemoryMiB – 執行個體類型必須具有最多 10240 MiB 的記憶體。最小值若為 0，表示沒有最小限制。
- MemoryGiBPerVCpu – 執行個體類型必須具有每個 vCPU 最多 10,000 MiB 的記憶體。Min 為選用參數。若省略它，表示沒有最小限制。

## TargetCapacityUnitType

TargetCapacityUnitType 參數會指定目標容量的單位。在範例中，目標容量為 5000，而目標容量單位類型為 vcpu，它們一起指定了所需的目標容量，即 5,000 個 vCPU。Spot Fleet 將啟動足夠的執行個體，以便機群中的 vCPU 總數為 5,000 個 vCPU。

有效組態：具有多個的單一啟動範本 InstanceRequirements

下列組態有效。其包含一個啟動範本和一個 Overrides 結構 (其中包含兩個 InstanceRequirements 結構)。InstanceRequirements 中指定的屬性是有效的，因為這些值不重疊—第一個 InstanceRequirements 結構會指定 0-2 個 vCPU 的 VCpuCount，而第二個 InstanceRequirements 結構則會指定 4-8 個 vCPU。

```
{
 "SpotFleetRequestConfig": {
 "AllocationStrategy": "priceCapacityOptimized",
 "ExcessCapacityTerminationPolicy": "default",
 "IamFleetRole": "arn:aws:iam::000000000000:role/aws-ec2-spot-fleet-tagging-
role",
 "LaunchTemplateConfigs": [
```

```

 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "MyLaunchTemplate",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 0,
 "Max": 2
 },
 "MemoryMiB": {
 "Min": 0
 }
 }
 },
 {
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 4,
 "Max": 8
 },
 "MemoryMiB": {
 "Min": 0
 }
 }
 }
]
 },
 "TargetCapacity": 1,
 "OnDemandTargetCapacity": 0,
 "Type": "maintain"
 }
}

```

有效組態：兩個啟動範本，每個都具有覆寫

下列組態有效。其包含兩個啟動範本，每個都具有一個 `Overrides` 結構 (其中包含一個 `InstanceRequirements` 結構)。這個組態有助於同一個機群中的 `arm` 和 `x86` 架構支援。

```
{
```

```
"SpotFleetRequestConfig": {
 "AllocationStrategy": "priceCapacityOptimized",
 "ExcessCapacityTerminationPolicy": "default",
 "IamFleetRole": "arn:aws:iam::000000000000:role/aws-ec2-spot-fleet-tagging-
role",
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "armLaunchTemplate",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 0,
 "Max": 2
 },
 "MemoryMiB": {
 "Min": 0
 }
 }
 }
],
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "x86LaunchTemplate",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 0,
 "Max": 2
 },
 "MemoryMiB": {
 "Min": 0
 }
 }
 }
]
 }
],
 "TargetCapacity": 1,
```

```

 "OnDemandTargetCapacity": 0,
 "Type": "maintain"
 }
}

```

有效組態：僅限指定的 **InstanceRequirements**，沒有重疊的屬性值

下列組態有效。其包含兩個 `LaunchTemplateSpecification` 結構，每個都有一個啟動範本和一個 `Overrides` 結構 (其中包含一個 `InstanceRequirements` 結構)。 `InstanceRequirements` 中指定的屬性是有效的，因為這些值不重疊—第一個 `InstanceRequirements` 結構會指定 0-2 個 vCPU 的 `VCpuCount`，而第二個 `InstanceRequirements` 結構則會指定 4-8 個 vCPU。

```

{
 "SpotFleetRequestConfig": {
 "AllocationStrategy": "priceCapacityOptimized",
 "ExcessCapacityTerminationPolicy": "default",
 "IamFleetRole": "arn:aws:iam::000000000000:role/aws-ec2-spot-fleet-tagging-
role",
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "MyLaunchTemplate",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 0,
 "Max": 2
 },
 "MemoryMiB": {
 "Min": 0
 }
 }
 }
]
 },
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "MyOtherLaunchTemplate",
 "Version": "1"
 },

```

```

 "Overrides": [
 {
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 4,
 "Max": 8
 },
 "MemoryMiB": {
 "Min": 0
 }
 }
 }
]
 },
 "TargetCapacity": 1,
 "OnDemandTargetCapacity": 0,
 "Type": "maintain"
}
}

```

### 組態無效：Overrides 包含 InstanceRequirements 和 InstanceType

下列組態無效。Overrides 結構同時包含 InstanceRequirements 和 InstanceType。針對 Overrides，您可以指定 InstanceRequirements 或 InstanceType，但不能同時指定兩者。

```

{
 "SpotFleetRequestConfig": {
 "AllocationStrategy": "priceCapacityOptimized",
 "ExcessCapacityTerminationPolicy": "default",
 "IamFleetRole": "arn:aws:iam::000000000000:role/aws-ec2-spot-fleet-tagging-
role",
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "MyLaunchTemplate",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 0,

```



```

 "Max": 2
 },
 "MemoryMiB": {
 "Min": 0
 }
 },
 {
 "InstanceType": "m5.large"
 }
]
},
"TargetCapacity": 1,
"OnDemandTargetCapacity": 0,
"Type": "maintain"
}
}

```

### 組態無效：兩個 **Overrides** 包含 **InstanceRequirements** 和 **InstanceType**

下列組態無效。Overrides 結構同時包含 InstanceRequirements 和 InstanceType。您可以指定 InstanceRequirements 或 InstanceType，但不能同時指定兩者，即使它們處於不同的 Overrides 結構也一樣。

```

{
 "SpotFleetRequestConfig": {
 "AllocationStrategy": "priceCapacityOptimized",
 "ExcessCapacityTerminationPolicy": "default",
 "IamFleetRole": "arn:aws:iam::000000000000:role/aws-ec2-spot-fleet-tagging-
role",
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "MyLaunchTemplate",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 0,
 "Max": 2
 }
 }
 }
]
 }
]
 }
}

```

```

 },
 "MemoryMiB": {
 "Min": 0
 }
 }
}
],
},
{
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "MyOtherLaunchTemplate",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceType": "m5.large"
 }
]
}
],
"TargetCapacity": 1,
"OnDemandTargetCapacity": 0,
"Type": "maintain"
}
}

```

### 組態無效：重疊屬性值

下列組態無效。兩個 InstanceRequirements 結構每個都包含 "VCpuCount": {"Min": 0, "Max": 2}。這些屬性的值重疊，這會導致容量集區重複。

```

{
 "SpotFleetRequestConfig": {
 "AllocationStrategy": "priceCapacityOptimized",
 "ExcessCapacityTerminationPolicy": "default",
 "IamFleetRole": "arn:aws:iam::000000000000:role/aws-ec2-spot-fleet-tagging-
role",
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "MyLaunchTemplate",
 "Version": "1"
 },
 },
]
 }
}

```

```
 "Overrides": [
 {
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 0,
 "Max": 2
 },
 "MemoryMiB": {
 "Min": 0
 }
 },
 {
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 0,
 "Max": 2
 },
 "MemoryMiB": {
 "Min": 0
 }
 }
 }
]
 },
 "TargetCapacity": 1,
 "OnDemandTargetCapacity": 0,
 "Type": "maintain"
 }
}
```

## 預覽具有所指定屬性的執行個體類型

您可以使用[從執行個體需求取得執行個體類型 AWS CLI 指令](#)來預覽符合指定屬性的例證類型。這特別有助於解決在不啟動任何執行個體的情況下要在請求組態中指定哪些屬性。請注意，該命令不會考慮可用容量。

## 透過使用指定屬性來預覽例證類型清單 AWS CLI

1. (選用) 若要產生可指定的所有可能屬性，請使用 [get-instance-types-from-instance-requirements](#) 命令和 `--generate-cli-skeleton` 參數。您可以選擇性地將輸出導向至檔案來儲存它，方法是使用 `input > attributes.json`。

```
aws ec2 get-instance-types-from-instance-requirements \
 --region us-east-1 \
 --generate-cli-skeleton input > attributes.json
```

## 預期的輸出結果

```
{
 "DryRun": true,
 "ArchitectureTypes": [
 "i386"
],
 "VirtualizationTypes": [
 "hvm"
],
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 0,
 "Max": 0
 },
 "MemoryMiB": {
 "Min": 0,
 "Max": 0
 },
 "CpuManufacturers": [
 "intel"
],
 "MemoryGiBPerVCpu": {
 "Min": 0.0,
 "Max": 0.0
 },
 "ExcludedInstanceTypes": [
 ""
],
 "InstanceGenerations": [
 "current"
],
 "SpotMaxPricePercentageOverLowestPrice": 0,
 "OnDemandMaxPricePercentageOverLowestPrice": 0,
 "BareMetal": "included",
 "BurstablePerformance": "included",
 "RequireHibernateSupport": true,
 "NetworkInterfaceCount": {
```

```
 "Min": 0,
 "Max": 0
 },
 "LocalStorage": "included",
 "LocalStorageTypes": [
 "hdd"
],
 "TotalLocalStorageGB": {
 "Min": 0.0,
 "Max": 0.0
 },
 "BaselineEbsBandwidthMbps": {
 "Min": 0,
 "Max": 0
 },
 "AcceleratorTypes": [
 "gpu"
],
 "AcceleratorCount": {
 "Min": 0,
 "Max": 0
 },
 "AcceleratorManufacturers": [
 "nvidia"
],
 "AcceleratorNames": [
 "a100"
],
 "AcceleratorTotalMemoryMiB": {
 "Min": 0,
 "Max": 0
 },
 "NetworkBandwidthGbps": {
 "Min": 0.0,
 "Max": 0.0
 },
 "AllowedInstanceTypes": [
 ""
]
},
"MaxResults": 0,
"NextToken": ""
}
```

2. 使用上一個步驟的輸出來建立 JSON 組態檔，並將其設定如下：

 Note

您必須提供 ArchitectureTypes、VirtualizationTypes、VCpuCount 和 MemoryMiB 的值。您可以省略其他屬性；省略時，就會使用預設值。

如需每個屬性及其預設值的說明，請參閱 Amazon EC2 命令列參考中的 [get-instance-types-from-instance-requirements](#)。

- a. 針對 ArchitectureTypes，指定一或多種類型的處理器架構。
  - b. 針對 VirtualizationTypes，指定一或多種類型的虛擬化。
  - c. 針對 VCpuCount，指定 vCPU 數目下限和上限。若要指定沒有下限，請針對 Min，指定 0。若要指定沒有上限，請省略 Max 參數。
  - d. 針對 MemoryMiB，指定記憶體的数量下限和上限 (以 MiB 為單位)。若要指定沒有下限，請針對 Min，指定 0。若要指定沒有上限，請省略 Max 參數。
  - e. 您可以選擇性地指定一或多個其他屬性，以進一步限制傳回的執行個體類型清單。
3. 若要預覽具有您在 JSON 檔案中指定之屬性的執行個體類型，請使用 [get-instance-types-from-instance-requirements](#) 命令，並使用 `--cli-input-json` 參數指定 JSON 檔案的名稱和路徑。您可以選擇性地格式化要以表格格式出現的輸出。

```
aws ec2 get-instance-types-from-instance-requirements \
 --cli-input-json file://attributes.json \
 --output table
```

### 範例 *attributes.json* 文件

在此範例中，JSON 檔案中包含必要的屬性。它們是 ArchitectureTypes、VirtualizationTypes、VCpuCount 和 MemoryMiB。此外，選用 InstanceGenerations 屬性也包含在內。請注意，對於 MemoryMiB，可以省略 Max，以表示沒有限制。

```
{
 "ArchitectureTypes": [
 "x86_64"
],
 "VirtualizationTypes": [
```

```

 "hvm"
],
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 4,
 "Max": 6
 },
 "MemoryMiB": {
 "Min": 2048
 },
 "InstanceGenerations": [
 "current"
]
 }
}

```

### 範例輸出

```

|GetInstanceTypesFromInstanceRequirements|
+-----+
|| InstanceTypes ||
|+-----+|
|| InstanceType ||
|+-----+|
|| c4.xlarge ||
|| c5.xlarge ||
|| c5a.xlarge ||
|| c5ad.xlarge ||
|| c5d.xlarge ||
|| c5n.xlarge ||
|| c6a.xlarge ||
|| ... ||

```

4. 在識別符合您需求的執行個體類型之後，請記下您使用的執行個體屬性，以便在設定機群請求時可以使用它們。

## Spot Fleet 中的隨需

為確保永遠有執行個體容量，您可以在 Spot Fleet 請求中加入對隨需容量的請求。在 Spot Fleet 請求中，指定需要的目標容量，以及必須為隨需的容量大小。餘額包含的 Spot 容量，會在有可用 Amazon

EC2 容量和可用性時啟動。例如，如果您在 Spot 機群請求中，將目標容量指定為 10、將隨需容量指定為 8，則 Amazon EC2 會啟動 8 個容量單位做為隨需、啟動 2 個容量單位 (10-8=2) 做為 Spot。

針對隨需容量排定執行個體類型的優先順序

Spot 機群嘗試實現您的隨需容量時，它會預設為先啟動價格最低的執行個體類型。若 `OnDemandAllocationStrategy` 設為 `prioritized`，Spot Fleet 會使用優先順序來決定在實現隨需容量時要先使用的執行個體類型。

優先順序會指派至啟動範本覆寫，優先順序最高的將會先啟動。

範例：排定執行個體類型的優先順序

在本例中，您設定三個啟動範本覆寫，每個範本均具有不同的執行個體類型。

執行個體類型的隨需價格在價格範圍內。以下是本範例中使用的執行個體類型，依價格順序列出，從最便宜的執行個體類型開始：

- `m4.large` – 最便宜
- `m5.large`
- `m5a.large`

若您不使用優先順序來決定順序，則機群會以最便宜的執行個體類型開始來實現隨需容量。

但是，假設您要首先使用且未使用的 `m5.large` 預留執行個體。您可設定啟動範本覆寫，以便依優先順序使用執行個體類型，如下所示：

- `m5.large` – 優先順序 1
- `m4.large` – 優先順序 2
- `m5a.large` – 優先順序 3

## 容量重新平衡

您可以設定 Spot Fleet，以便在 Amazon EC2 發出重新平衡建議時啟動替代 Spot 執行個體，以通知您 Spot 執行個體的中斷風險提高。容量重新平衡可協助您維持工作負載可用性，方法是在執行中的執行個體受到 Amazon EC2 的中斷之前，使用新的 Spot 執行個體主動擴增機群。如需詳細資訊，請參閱 [EC2 執行個體重新平衡建議](#)。

要設定 Spot Fleet 以啟動替代 Spot 執行個體，您可以使用 Amazon EC2 主控台或 AWS CLI。



- Amazon EC2 主控台：在您建立 Spot Fleet 時，必須選取 Capacity rebalance (容量重新平衡) 核取方塊。如需詳細資訊，請參閱 [使用已定義的參數建立 Spot Fleet 請求 \(主控台\)](#) 中的步驟 6.d。
- AWS CLI：使用 [request-spot-fleet](#) 命令和 SpotMaintenanceStrategies 結構中的相關參數。如需詳細資訊，請參閱 [範例啟動設定](#)。

## 限制

- 容量重新平衡僅適用於類型為 maintain 的機群。
- 機群執行時，您無法修改容量重新平衡設定。若要變更容量重新平衡設定，您必須刪除機群並建立新的機群。

## 組態選項

Spot Fleet 的 ReplacementStrategy 支援以下兩個值：

### launch-before-terminate

Amazon EC2 會在新的替代 Spot 執行個體啟動之後，終止收到重新平衡通知的 Spot 執行個體。當指定 launch-before-terminate 時，您亦須指定 termination-delay 的值。在啟動新的替代執行個體之後，Amazon EC2 會等待 termination-delay 的持續時間，然後終止舊的執行個體。對於 termination-delay，下限為 120 秒 (2 分鐘)，上限則為 7200 秒 (2 小時)。

建議僅在您可以預測執行個體關閉程序將需要多長時間才能完成時，才使用 launch-before-terminate。這將確保只在關機程序完成之後，才會終止舊執行個體。請注意，Amazon EC2 可以中斷舊的執行個體，並在 termination-delay 之前提供兩分鐘警告。

### launch

針對現有 Spot 執行個體發出重新平衡通知時，Amazon EC2 會啟動替代 Spot 執行個體。Amazon EC2 不會自動終止收到重新平衡建議的執行個體。您可以終止舊的執行個體，也可以讓它們保持執行中狀態。所有執行個體在執行時，您需要支付所有執行個體的費用。

## 考量事項

如果您設定容量重新平衡的 Spot Fleet，請考慮下列事項：

### 盡可能在請求中提供更多 Spot 容量集區

設定您的 Spot Fleet，以使用多個執行個體類型和可用區域。這提供了在各種 Spot 容量集區中啟動 Spot 執行個體的彈性。如需詳細資訊，請參閱 [對執行個體類型和可用區域具有彈性](#)。

## 避免中斷替代 Spot 執行個體的風險提高

為了避免中斷的風險增加，我們建議使用 `capacityOptimized` 或 `capacityOptimizedPrioritized` 配置策略。這些策略可確保在最佳 Spot 容量集區中啟動替代 Spot 執行個體，因此不太可能在不久的將來中斷。如需詳細資訊，請參閱 [使用價格和容量最佳化分配策略](#)。

只有在可用性相同或更好時，Amazon EC2 才會啟動新執行個體

容量重新平衡的其中一個目標是改善 Spot 執行個體的可用性。如果現有的 Spot 執行個體收到重新平衡建議，只有新執行個體提供與現有執行個體相同或更好的可用性時，Amazon EC2 才會啟動新執行個體。如果新執行個體的中斷風險比現有執行個體更大，則 Amazon EC2 將不會啟動新執行個體。但是，Amazon EC2 將繼續評估 Spot 容量集區，並在可用性改善時啟動新執行個體。

如果 Amazon EC2 不主動啟動新執行個體，現有執行個體可能會中斷。發生這種情況時，無論新執行個體是否存在高中斷風險，Amazon EC2 都會嘗試啟動新執行個體。

容量重新平衡不會增加 Spot 執行個體中斷率

當您啟用容量重新平衡時，不會增加 [Spot 執行個體中斷率](#) (當 Amazon EC2 需要恢復容量時回收的 Spot 執行個體數量)。但是，如果容量重新平衡偵測到執行個體有中斷風險，Amazon EC2 將立即嘗試啟動新的執行個體。因此，與在有風險的執行個體中斷後等待 Amazon EC2 啟動新的執行個體相比，可能會取代更多執行個體。

雖然您可以在啟用容量重新平衡的情況下取代更多執行個體，但您可以從主動式而非被動式中受益，因為在執行個體中斷之前您有更多時間採取動作。透過 [Spot 執行個體中斷通知](#)，您通常最多只有兩分鐘時間來正常關閉您的執行個體。透過容量重新平衡預先啟動新的執行個體，可讓現有程序更有可能在有風險的執行個體上完成，您可以啟動執行個體關閉程序，並防止在有風險的執行個體上排程新的工作。您也可以開始準備新啟動的執行個體以接管應用程式。透過容量重新平衡的主動式取代，您可以從正常連續性中獲益。

作為示範使用容量重新平衡的風險和益處的理論範例，請考慮下列案例：

- 下午 2:00 – 收到執行個體 A 的重新平衡建議，Amazon EC2 會立即開始嘗試啟動取代執行個體 B，讓您有時間開始關機程序。\*
- 下午 2:30 – 收到執行個體 B 的重新平衡建議，會取代為執行個體 C，讓您有時間開始關機程序。\*
- 下午 2:32 – 如果未啟用容量重新平衡，並且如果在下午 2:32 收到執行個體 A 的 Spot 執行個體中斷通知，則您最多只有兩分鐘時間採取動作，但是執行個體 A 會一直執行到目前。

\* 如果指定了 `launch-before-terminate`，Amazon EC2 將在取代執行個體連線後終止有風險的執行個體。

## Amazon EC2 可以啟動新的替代 Spot 執行個體，直到滿足的容量為目標容量雙倍

設定 Spot 機群為容量重新平衡時，Amazon EC2 會嘗試為每個收到重新平衡建議的 Spot 執行個體啟動新的替代 Spot 執行個體。Spot 執行個體收到重新平衡建議後，該執行個體不再計入以滿足容量的一部分。根據替代策略，Amazon EC2 會在預先設定的終止延遲之後終止執行個體，或讓執行個體繼續執行。這讓您有機會在執行個體上執行[重新平衡動作](#)。

如果您的機群達到目標容量的兩倍，即使替代執行個體本身收到重新平衡建議，它仍會停止啟動新的替代執行個體。

例如，您建立目標容量為 100 個 Spot 執行個體的 Spot Fleet。所有 Spot 執行個體都會收到重新平衡建議，這會導致 Amazon EC2 啟動 100 個替代 Spot 執行個體。這會將已滿足 Spot 執行個體的數目提高至 200 個，這是目標容量的兩倍。某些替代執行個體會收到重新平衡建議，但不會再啟動替代執行個體，因為機群不能超過目標容量的兩倍。

請注意，所有執行個體在執行時都需支付這些執行個體的費用。

建議您將 Spot Fleet 設定為終止收到重新平衡建議的 Spot 執行個體

如果您設定 Spot Fleet 進行容量重新平衡，建議您只在可以預測執行個體關閉程序需要多長時間才能完成時，才選擇具有適當終止延遲的 `launch-before-terminate`。這將確保只在關機程序完成之後，才會終止舊執行個體。

如果您選擇自行終止建議用於重新平衡的執行個體，建議您監控機群中 Spot 執行個體收到的重新平衡建議訊號。透過監控訊號，您可以在 Amazon EC2 中斷受影響的執行個體之前，快速對受影響的執行個體執行[重新平衡動作](#)，然後手動終止它們。如果您未終止執行個體，則會在執行時繼續為這些執行個體付費。Amazon EC2 不會自動終止收到重新平衡建議的執行個體。

您可以使用 Amazon EventBridge 或執行個體中繼資料設定通知。如需詳細資訊，請參閱[監控重新平衡建議訊號](#)。

Spot Fleet 在縮減或擴展期間計算已滿足容量時，不會計算收到重新平衡建議的執行個體

如果您的 Spot Fleet 已設定容量重新平衡，且您將目標容量變更為縮減或擴展，則機群不會將標示要重新平衡的執行個體計算為已滿足容量一部分，如下所示：

- 縮減 – 如果您減少所需的目標容量，Amazon EC2 會終止未標示要重新平衡的執行個體，直到達到所需容量為止。標示要重新平衡的執行個體不會計入已滿足的容量。

例如，您建立目標容量為 100 個 Spot 執行個體的 Spot 機群。10 個執行個體會收到重新平衡建議，因此 Amazon EC2 會啟動 10 個新的替代執行個體，從而產生 110 個執行個體的滿足容量。

然後，您可以將目標容量減少為 50 (縮減)，但已滿足容量實際上為 60 個執行個體，因為標示要重新平衡的 10 個執行個體不會被 Amazon EC2 終止。您需要手動終止這些執行個體，或者您也可以讓它們保持運行。

- 橫向擴展 – 如果您增加想要的目標容量，Amazon EC2 會啟動新執行個體，直到達到所需容量為止。標示要重新平衡的執行個體不會計入已滿足的容量。

例如，您建立目標容量為 100 個 Spot 執行個體的 Spot 機群。10 個執行個體會收到重新平衡建議，因此 Amazon EC2 會啟動 10 個新的替代執行個體，從而產生 110 個執行個體的滿足容量。然後，您將目標容量增加至 200 (擴展)，但已滿足容量實際上是 210 個執行個體，因為標示要重新平衡的 10 個執行個體不會被機群計為目標容量的一部分。您需要手動終止這些執行個體，或者您也可以讓它們保持運行。

## Spot 價格覆寫

每個 Spot Fleet 請求可以包含全球最高價格，或使用預設值 (隨需價格)。Spot Fleet 將此用作每個啟動規格的預設最高價格。

您可以選擇性指定一或多個啟動規格中的最高價格。此價格特定於啟動規格。如果啟動規格包含特定價格，則 Spot Fleet 會使用這個最高價格來覆寫全球最高價格。不包括特定最高價格的任何其他啟動規格仍會使用全球最高價格。

## 控制花費

在達到目標容量或您願意支付的最大金額時，Spot Fleet 會停止啟動執行個體。若要控制您為機群支付的每個小時金額，您可以為 競價型執行個體 指定 `SpotMaxTotalPrice`，以及為 隨需執行個體 指定 `OnDemandMaxTotalPrice`。達到總價格上限時，Spot Fleet 會停止啟動執行個體，即使它未符合目標容量亦然。

以下範例顯示兩個不同的案例。在第一個範例中，Spot Fleet 會在符合目標容量時停止啟動執行個體。在第二個範例中，Spot Fleet 會在達到您願意支付的金額上限時停止啟動執行個體。

範例：達到目標容量時停止啟動執行個體

假設有 `m4.large` 隨需執行個體 的請求，其中：

- 隨需價格：每小時 0.10 USD
- `OnDemandTargetCapacity`：10
- `OnDemandMaxTotalPrice`：1.50 USD

Spot Fleet 會啟動 10 個隨需執行個體，因為 1.00 USD 的總計 (10 個執行個體 x 0.10 USD) 未超出 1.50 USD 的 `OnDemandMaxTotalPrice`。

範例：達到總價格上限時停止啟動執行個體

假設有 `m4.large` 隨需執行個體的請求，其中：

- 隨需價格：每小時 0.10 USD
- `OnDemandTargetCapacity`：10
- `OnDemandMaxTotalPrice`：0.80 USD

如果 Spot Fleet 啟動了隨需目標容量 (10 個隨需執行個體)，每個小時的總成本會是 1.00 USD。這超出為 `OnDemandMaxTotalPrice` 指定的金額 (0.80 USD)。為了防止花費超出您願意支付的金額，Spot Fleet 只會啟動 8 個隨需執行個體 (低於隨需目標容量)，因為啟動更多會超出 `OnDemandMaxTotalPrice`。

## Spot Fleet 執行個體權重

在請求 競價型執行個體 的機群時，您可以定義每個執行個體類型對應用程式效能所貢獻的容量單位，並利用執行個體權重來據以調整每個 Spot 容量集區的最高價。

根據預設，您指定的價格是每執行個體小時。使用執行個體權重功能時，您指定的價格是每單位小時。您可以透過將執行個體類型價格除以它所代表的單位數，來計算每單位小時的價格。Spot Fleet 會將目標容量除以執行個體的權重，來計算出要啟動的 Spot Fleet 數量。如果結果不是整數，則 Spot Fleet 會將其捨入到下一個整數，以便您的機群大小不低於其目標容量。即使所啟動的執行個體容量超過所請求的目標容量，Spot Fleet 也可以選取您在啟動規格中指定的任何集區。

下表提供計算範例，針對目標容量為 10 的 Spot Fleet 請求得出每單位價格。

執行個體類型	執行個體權重	每個執行個體小時的價格	每個單位小時的價格	已啟動的執行個體數目
<code>r3.xlarge</code>	2	0.05 USD	.025 (.05 除以 2)	5 (10 除以 2)

執行個體類型	執行個體權重	每個執行個體小時的價格	每個單位小時的價格	已啟動的執行個體數目
r3.8xlarge	8	0.10 USD	.0125	2
			(.10 除以 8)	(10 除以 8, 將結果四捨五入)

按照下列方式，在實現請求時使用 Spot Fleet 執行個體權重，以在完成要求時，使用每單位最低價在集區中佈建所需目標容量：

1. 在您所選擇的執行個體 (預設) 或裝置中 (例如虛擬 CPU、記憶體、儲存空間或傳輸量)，設定 Spot Fleet 的目標容量。
2. 設定每單位價格
3. 為每個啟動組態指定權重，也就是執行個體類型代表的目標容量單位數。

### 執行個體權重範例

請考慮下列組態的 Spot Fleet 請求：

- 目標容量為 24
- 執行個體類型為 r3.2xlarge 且權重為 6 的啟動規格
- 執行個體類型為 c3.xlarge 且權重為 5 的啟動規格

權重表示執行個體類型代表目標容量的單位數。如果第一個啟動規格提供了最低的每單位價格 (r3.2xlarge 的每小時執行個體價格除以 6)，則 Spot Fleet 會啟動這些執行個體的其中 4 個 (24 除以 6)。

如果第二個啟動規格提供了最低的每單位價格 (c3.xlarge 的每小時執行個體價格除以 5)，則 Spot Fleet 會啟動這些執行個體的其中 5 個 (24 除以 5，將結果四捨五入)。

### 執行個體權重與分配策略

請考慮下列組態的 Spot Fleet 請求：

- 目標容量為 30
- 執行個體類型為 c3.2xlarge 且權重為 8 的啟動規格

- 執行個體類型為 m3.xlarge 且權重為 8 的啟動規格
- 執行個體類型為 r3.xlarge 且權重為 8 的啟動規格

Spot Fleet 會啟動 4 個執行個體 (30 除以 8，將結果四捨五入)。遵循 diversified 策略，Spot Fleet 在 3 個集區的每個集區中皆啟動 1 個執行個體，而任一集區中的第 4 個執行個體會提供每單位最低價格。

## 使用 Spot Fleet

若要開始使用 Spot Fleet，您需要建立一個 Spot Fleet 請求，其中包含目標容量、選用的隨需部分、執行個體的一個或多個啟動規格，以及您願意支付的最高價格。機群請求必須包含啟動規格，該範本定義機群啟動執行個體所需的資訊，例如 AMI、執行個體類型、子網或可用區域，以及一或多個安全群組。

如果您的叢集包含競價型執行個體，則 Amazon EC2 會在 Spot 價格變更時嘗試維持您的叢集目標容量。

提交之後，即無法修改一次性請求的目標容量。若要變更一次性請求的目標容量，請刪除該請求，並提交新的請求。

Spot Fleet 請求會持續有效作用直到過期，或直到您取消該請求。取消機群請求時，您可以指定取消該請求是否終止該機群中的 Spot 執行個體。

### 目錄

- [Spot Fleet 請求狀態](#)
- [Spot Fleet 運作狀態檢查](#)
- [Spot Fleet 許可](#)
- [建立 Spot Fleet 請求](#)
- [標記 Spot Fleet](#)
- [說明您的 Spot 機群](#)
- [修改 Spot Fleet 請求](#)
- [取消 Spot Fleet 請求](#)

## Spot Fleet 請求狀態

Spot Fleet 請求可以是下列任一狀態：

- **submitted** – 正在評估 Spot Fleet 請求且 Amazon EC2 正準備啟動執行個體的目標數量。超出 Spot 機群限制的請求會立即取消。
- **active** – 已驗證 Spot Fleet 且 Amazon EC2 正在嘗試維持執行中的 Spot 執行個體目標數量。此請求會一直維持在該狀態，直到修改或取消。
- **modifying** – 正在修改 Spot Fleet 請求。請求會一直維持在此狀態，直到已完成修改或已取消 Spot Fleet。一次性的 request 無法修改，而且其狀態不適用此種 Spot 請求。
- **cancelled\_running** – 已取消 Spot Fleet，而且不會啟動額外的 Spot 執行個體。其現有 競價型執行個體將繼續執行直到中斷或終止。此請求會一直維持在該狀態，直到所有執行個體中斷或終止。
- **cancelled\_terminating** – 已取消 Spot Fleet，並且其 Spot 執行個體正在終止。此請求會一直維持在該狀態，直到所有執行個體終止。
- **cancelled** – 已取消 Spot Fleet 且沒有執行中的 Spot 執行個體。Spot Fleet 請求會在其執行個體終止之後兩天刪除。

## Spot Fleet 運作狀態檢查

Spot Fleet 會每兩分鐘定期檢查機群中 Spot 執行個體的運作狀態。執行個體的運作狀態會是 **healthy** 或 **unhealthy**。

Spot Fleet 會根據 Amazon EC2 提供的狀態檢查資訊，來判定執行個體的運作狀態。如果執行個體狀態檢查或系統狀態檢查等運作狀態檢查，出現連續三次為 **unhealthy**，則該執行個體即為 **impaired**。如需詳細資訊，請參閱 [您的執行個體的状态檢查](#)。

您可以設定機群以取代運作狀態不佳的 競價型執行個體。啟用運作狀態檢查取代後，Spot 執行個體會在被報告為 **unhealthy** 時被取代。當取代一個狀態不佳的 Spot 執行個體時，機群可能會低於其目標容量長達幾分鐘。

### 要求

- 運作狀態檢查替代僅支援維持目標容量的 Spot Fleets (類型 **maintain** 的機群)，不支援一次性的 Spot Fleets (類型 **request** 的機群)。
- 運作狀態檢查取代僅支援 競價型執行個體。此功能不支援 隨需執行個體。
- 您可以設定 Spot Fleet，只在建立執行個體時取代狀況不良的執行個體。
- 使用者只有在擁有呼叫 `ec2:DescribeInstanceStatus` 動作的許可時，才能使用運作狀態檢查取代功能。



## Console

若要使用主控台來設定 Spot Fleet 以取代運作狀態不佳的 Spot 執行個體

1. 遵循下列步驟，建立 Spot Fleet。如需詳細資訊，請參閱 [使用已定義的參數建立 Spot Fleet 請求 \(主控台\)](#)。
2. 若要設定機群以取代運作狀態不佳的 競價型執行個體，請在 [運作狀態檢查](#) 中選擇 取代運作狀態不佳的執行個體。若要啟用此選項，首先必須選擇維護目標容量。

## AWS CLI

若要使用 AWS CLI來設定 Spot Fleet 以取代運作狀態不佳的 Spot 執行個體

1. 遵循下列步驟，建立 Spot Fleet。如需詳細資訊，請參閱 [使用建立競價型叢集 AWS CLI](#)。
2. 若要設定機群以取代運作狀態不佳的 競價型執行個體，請在 `ReplaceUnhealthyInstances` 中輸入 `true`。

## Spot Fleet 許可

如果使用者將會建立或管理 Spot 機群，您必須授予這些使用者所需的許可。

如果使用 Amazon EC2 主控台來建立 Spot Fleet，它會建立名為 `AWSServiceRoleForEC2SpotFleet` 和 `AWSServiceRoleForEC2Spot` 的兩個服務連結角色，以及名為 `aws-ec2-spot-fleet-tagging-role` 的角色來授與 Spot Fleet 許可，以代表您請求、啟動、終止和標記資源。如果使用 AWS CLI 或 API，您必須確認這些角色存在。

請使用下列指示，授與必要的許可並建立角色。

### 許可和角色

- [將 Spot 機群的許可授予使用者](#)
- [Spot Fleet 的服務連結角色](#)
- [Spot 執行個體的服務連結角色](#)
- [標記 Spot Fleet 的 IAM 角色](#)

### 將 Spot 機群的許可授予使用者

如果使用者將會建立或管理 Spot 機群，請務必授予這些使用者所需的許可。

## 為 Spot 機群建立政策

1. 在以下網址開啟 IAM 主控台：<https://console.aws.amazon.com/iam/>。
2. 在導覽窗格中，選擇政策、建立政策。
3. 在建立政策頁面上，選擇 JSON，使用下列內容來替代其文字。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:RunInstances",
 "ec2:CreateTags",
 "ec2:RequestSpotFleet",
 "ec2:ModifySpotFleetRequest",
 "ec2:CancelSpotFleetRequests",
 "ec2:DescribeSpotFleetRequests",
 "ec2:DescribeSpotFleetInstances",
 "ec2:DescribeSpotFleetRequestHistory"
],
 "Resource": "*"
 },
 {
 "Effect": "Allow",
 "Action": "iam:PassRole",
 "Resource": "arn:aws:iam::*:role/aws-ec2-spot-fleet-tagging-role"
 },
 {
 "Effect": "Allow",
 "Action": [
 "iam:CreateServiceLinkedRole",
 "iam:ListRoles",
 "iam:ListInstanceProfiles"
],
 "Resource": "*"
 }
]
}
```

上述範例政策會向使用者授予大多數 Spot Fleet 使用案例所需的許可。若要將使用者限制於特定的 API 動作，請改為僅指定這些 API 動作。

## 需要的 EC2 和 IAM API

必須在原則中包含下列 API：

- `ec2:RunInstances` – 在 Spot Fleet 中啟動執行個體所需
- `ec2:CreateTags` – 標記 Spot Fleet 請求、執行個體或磁碟區所需
- `iam:PassRole` – 指定 Spot Fleet 角色所需
- `iam:CreateServiceLinkedRole` – 建立服務連結角色所需
- `iam:ListRoles` – 列舉現有 IAM 角色所需
- `iam:ListInstanceProfiles` – 列舉現有執行個體描述檔所需

### Important

如果您在啟動規格或啟動範本中為 IAM 執行個體設定檔指定角色，您必須授予使用者許可，以將角色傳遞到服務。若要執行此操作，請在 IAM 政策中包含 `"arn:aws:iam::*:role/IamInstanceProfile-role"`，做為 `iam:PassRole` 動作的資源。如需詳細資訊，請參閱 [《IAM 使用者指南》中的授與使用者將角色傳遞給 AWS 服務的權限](#)。

## Spot Fleet API

視需要將下列 Spot Fleet API 動作新增至您的政策：

- `ec2:RequestSpotFleet`
- `ec2:ModifySpotFleetRequest`
- `ec2:CancelSpotFleetRequests`
- `ec2:DescribeSpotFleetRequests`
- `ec2:DescribeSpotFleetInstances`
- `ec2:DescribeSpotFleetRequestHistory`

## 選用的 IAM API

(選用) 若要讓使用者能夠使用 IAM 主控台建立角色或執行個體設定檔，您必須將下列動作新增至政策中：

- iam:AddRoleToInstanceProfile
  - iam:AttachRolePolicy
  - iam:CreateInstanceProfile
  - iam:CreateRole
  - iam:GetRole
  - iam:ListPolicies
4. 選擇檢閱政策。
  5. 在 Review policy (檢閱政策) 頁面上，輸入政策名稱和說明，然後選擇 Create policy (建立政策)。
  6. 若要提供存取權，請新增權限至您的使用者、群組或角色：
    - 使用者和群組位於 AWS IAM Identity Center：

建立權限合集。請按照 AWS IAM Identity Center 使用者指南 中的 [建立權限合集](#) 說明進行操作。
    - 透過身分提供者在 IAM 中管理的使用者：

建立聯合身分的角色。請按照 IAM 使用者指南 的 [為第三方身分提供者 \(聯合\) 建立角色](#) 中的指示進行操作。
    - IAM 使用者：
      - 建立您的使用者可擔任的角色。請按照 IAM 使用者指南 的 [為 IAM 使用者建立角色](#) 中的指示進行操作。
      - (不建議) 將政策直接附加至使用者，或將使用者新增至使用者群組。請遵循 IAM 使用者指南的 [新增許可到使用者 \(主控台\)](#) 中的指示。

## Spot Fleet 的服務連結角色

Amazon EC2 使用許可的服務連結角色，它需要代表您呼叫其他 AWS 服務。服務連結角色是直接連結至 AWS 服務的唯一 IAM 角色類型。服務連結角色提供將權限委派給 AWS 服務的安全方式，因為只有連結的服務可以擔任服務連結角色。如需詳細資訊，請參閱《IAM 使用者指南》中的 [使用服務連結角色](#)。

Amazon EC2 使用名為AWSServiceRoleForEC2SpotFleet的服務連結角色代表您啟動和管理執行個體。

### ⚠ Important

如果您在競價型叢集中指定加密的 [AMI](#) 或加密的 Amazon EBS 快照，則必須授與該 `AWSServiceRoleForEC2SpotFleet` 角色使用 CMK 的權限，以便 Amazon EC2 可以代表您啟動執行個體。如需詳細資訊，請參閱 [授與 CMK 的存取權，以便與加密的 AMI 和 EBS 快照搭配使用](#)。

## 授予的權限 `AWSServiceRoleForEC2SpotFleet`

Amazon EC2 用 `AWSServiceRoleForEC2SpotFleet` 於完成以下操作：

- `ec2:RequestSpotInstances` - 請求 競價型執行個體
- `ec2:RunInstances` - 啟動執行個體
- `ec2:TerminateInstances` - 終止執行個體
- `ec2:DescribeImages` - 描述 Amazon Machine Images (AMI) 的執行個體
- `ec2:DescribeInstanceStatus` - 監控執行個體的狀態
- `ec2:DescribeSubnets` - 描述執行個體的字網路
- `ec2:CreateTags` - 將標籤新增至 Spot Fleet 請求、執行個體和磁碟區
- `elasticloadbalancing:RegisterInstancesWithLoadBalancer` - 將指定執行個體新增到指定的負載平衡器
- `elasticloadbalancing:RegisterTargets` - 向指定的目標群組登錄指定的目標

## 建立服務連結角色

在大部分情況下，您不需要手動建立服務連結角色。當您第一次使用主控台建立競價型叢集時，Amazon EC2 會建立 `AWSServiceRoleForEC2SpotFleet` 服務連結角色。

如果您在 2017 年 10 月之前有一個有效的競價型叢集請求，則當 Amazon EC2 開始支援此服務連結角色時，Amazon EC2 會在您的 AWS 帳戶中建立該 `AWSServiceRoleForEC2SpotFleet` 角色。如需詳細資訊，請參閱 IAM 使用者指南 [中出現在我的 AWS 帳戶中的新角色](#)。

如果您使用 AWS CLI 或 API 建立競價型叢集，則必須先確定此角色存在。

`AWSServiceRoleForEC2SpotFleet` 使用主控台建立

1. 在以下網址開啟 IAM 主控台：<https://console.aws.amazon.com/iam/>。

2. 在導覽窗格中，選擇角色。
3. 選擇 Create Role (建立角色)。
4. 在 Select trusted entity (選取信任的實體) 頁面上，執行以下作業：
  - a. 針對信任的實體類型，請選擇 AWS 服務。
  - b. 在使用案例下，對於服務或使用案例，選擇 EC2。
  - c. 對於使用案例，請選擇 EC2-競價型叢集。
  - d. 選擇下一步。
5. 在 Add permissions (新增許可) 頁面上，選擇 Next (下一步)。
6. 在命名、檢閱和建立頁面上，選擇建立角色。

若要 AWSServiceRoleForEC2SpotFleet 使用 AWS CLI

使用如下所示的 [create-service-linked-role](#) 命令。

```
aws iam create-service-linked-role --aws-service-name spotfleet.amazonaws.com
```

如果您不再需要使用 Spot 叢集，建議您刪除該AWSServiceRoleForEC2SpotFleet角色。從您的帳戶中刪除此角色後，如果您使用主控台請求 Spot Fleet，Amazon EC2 會再次建立此角色。如需詳細資訊，請參閱《IAM 使用者指南》中的[刪除服務連結角色](#)。

授與 CMK 的存取權，以便與加密的 AMI 和 EBS 快照搭配使用

如果您在 Spot 叢集請求中指定加密的 [AMI](#) 或加密的 Amazon EBS 快照，並使用客戶受管金鑰進行加密，則必須授與該AWSServiceRoleForEC2SpotFleet角色使用 CMK 的權限，以便 Amazon EC2 可以代表您啟動執行個體。若要執行此動作，您必須新增授與至 CMK，如以下列程序所示。

提供許可時，授與為金鑰政策的替代方案。如需詳細資訊，請參閱 [開發人員指南](#)中的[使用授予 AWS KMS](#)和在 AWS Key Management Service 中使用金鑰政策。

授與 AWSServiceRoleForEC2SpotFleet 角色使用 CMK 的權限

- 使用 [create-grant](#) 命令可將授權新增至 CMK，並指定授與執行授權作業之權限的主體 (AWSServiceRoleForEC2SpotFleet服務連結角色)。CMK 是透過 key-id 參數和 CMK 的 ARN 指定。主體是由grantee-principal參數和AWSServiceRoleForEC2SpotFleet服務連結角色的 ARN 所指定。

```
aws kms create-grant \
```

```
--region us-east-1 \
--key-id arn:aws:kms:us-
east-1:444455556666:key/1234abcd-12ab-34cd-56ef-1234567890ab \
--grantee-principal arn:aws:iam::111122223333:role/
AWSServiceRoleForEC2SpotFleet \
--operations "Decrypt" "Encrypt" "GenerateDataKey"
"GenerateDataKeyWithoutPlaintext" "CreateGrant" "DescribeKey" "ReEncryptFrom"
"ReEncryptTo"
```

## Spot 執行個體的服務連結角色

Amazon EC2 使用名為AWSServiceRoleForEC2Spot的服務連結角色代表您啟動和管理 Spot 執行個體。如需詳細資訊，請參閱 [Spot 執行個體請求的服務連結角色](#)。

## 標記 Spot Fleet 的 IAM 角色

aws-ec2-spot-fleet-tagging-role IAM 角色授予 Spot Fleet 許可，以標記 Spot Fleet 請求、執行個體和磁碟區。如需詳細資訊，請參閱 [標記 Spot Fleet](#)。

### Important

如果您選擇標記機群中的執行個體，並也選擇維護目標容量 (Spot 機群請求為類型 maintain)，為使用者與 IamFleetRole 所設定的許可差異可能會導致機群中執行個體標記行為的不一致。如果 IamFleetRole 不包含 CreateTags 許可，則可能無法標記由機群啟動的某些執行個體。雖然我們正在努力修正此不一致，但為了確保標記由機群啟動的所有執行個體，建議您針對 IamFleetRole 使用 aws-ec2-spot-fleet-tagging-role 角色。或者，若要使用現有角色，請將受 AmazonEC2SpotFleetTaggingRole AWS 管理策略附加至現有角色。否則，您需要手動將 CreateTags 許可新增至現有政策。

## 若要建立用於標記 Spot Fleet 的 IAM 角色

1. 在以下網址開啟 IAM 主控台：<https://console.aws.amazon.com/iam/>。
2. 在導覽窗格中，選擇角色。
3. 選擇 Create Role (建立角色)。
4. 在 Select trusted entity (選取受信任實體) 頁面，於 Trusted entity type (受信任實體的類型) 下，選擇 AWS service (服務)。
5. 在使用案例下，從其他 AWS 服務的使用案例中選擇 EC2，然後選擇 EC2-競價型叢集標記。

6. 選擇下一步。
7. 在 Add permissions (新增許可) 頁面上，選擇 Next (下一步)。
8. 在 Name, review, and create (名稱，檢閱和建立) 頁面上，針對 Role name (角色名稱) 輸入角色的名稱 (例如 **aws-ec2-spot-fleet-tagging-role**)。
9. 檢視頁面上的資訊，然後選擇 Create role (建立角色)。

## 預防跨服務混淆代理人

[混淆代理人問題](#)屬於安全性議題，其中沒有執行動作許可的實體可以強制具有更多權限的實體執行該動作。若要限制 Spot 機群為資源提供另一項服務的許可，我們建議在 `aws-ec2-spot-fleet-tagging-role` 信任政策中使用 [aws:SourceArn](#) 和 [aws:SourceAccount](#) 全域條件內容金鑰。

將 `aws:SourceArn` 和 `aws:SourceAccount` 條件金鑰新增至 **aws-ec2-spot-fleet-tagging-role** 信任政策

1. 在以下網址開啟 IAM 主控台：<https://console.aws.amazon.com/iam/>。
2. 在導覽窗格中，選擇角色。
3. 尋找您之前建立的 `aws-ec2-spot-fleet-tagging-role` 並選擇連結 (而非核取方塊)。
4. 在 Summary (摘要) 下，選擇 Trust relationships (信任關係) 索引標籤，然後選擇 Edit trust policy (編輯信任政策)。
5. 在 JSON 陳述式中，加入一個 Condition 元素，其中包含 `aws:SourceAccount` 和 `aws:SourceArn` 全域條件內容金鑰來防止[混淆代理問題](#)，如下所式：

```
"Condition": {
 "ArnLike": {
 "aws:SourceArn": "arn:aws:ec2:us-east-1:account_id:spot-fleet-request/sfr-
**
 },
 "StringEquals": {
 "aws:SourceAccount": "account_id"
 }
}
```



**Note**

如果 `aws:SourceArn` 值包含帳戶 ID，且同時使用這兩個全域條件內容金鑰，則在相同政策陳述式中使用 `aws:SourceAccount` 值和 `aws:SourceArn` 值中的帳戶時，必須使用相同的帳戶 ID。

最終信任政策如下所示：

```
{
 "Version": "2012-10-17",
 "Statement": {
 "Sid": "ConfusedDeputyPreventionExamplePolicy",
 "Effect": "Allow",
 "Principal": {
 "Service": "spotfleet.amazonaws.com"
 },
 "Action": "sts:AssumeRole",
 "Condition": {
 "ArnLike": {
 "aws:SourceArn": "arn:aws:ec2:us-east-1:account_id:spot-fleet-request/sfr-
*"
 },
 "StringEquals": {
 "aws:SourceAccount": "account_id"
 }
 }
 }
}
```

## 6. 選擇更新政策。

下表提供了 `aws:SourceArn` 的潛在值，以不同程度地限制 `aws-ec2-spot-fleet-tagging-role` 的範圍。

API 操作	呼叫的服務	範圍	<code>aws:SourceArn</code>
RequestSpot艦隊	AWS STS (AssumeRole )	將AssumeRole 功能限制aws-ec2-	arn:aws:ec2:*: <i>123456789</i>

API 操作	呼叫的服務	範圍	aws:SourceArn
		spot-fleet-tagging-role spot-fleet-requests 在指定帳戶中。	<i>012</i> :spot-fleet-request/sfr-*
RequestSpot艦隊	AWS STS (AssumeRole )	aws-ec2-spot-fleet-tagging-role 將AssumeRole 功能限制spot-fleet-requests 在指定帳戶和指定區域中。請注意，此角色在其他區域中無法使用。	arn:aws:ec2: <i>us-east-1</i> : <i>123456789</i> <i>012</i> :spot-fleet-request/sfr-*
RequestSpot艦隊	AWS STS (AssumeRole )	將 aws-ec2-spot-fleet-tagging-role 上的 AssumeRole 功能限制為僅限影響機群 sfr-1111111-1111-1111-1111-1111-111111111111 的動作。請注意，此角色可能無法用於其他 Spot 機群。此外，此角色無法用於透過 request-spot-fleet 啟動任何新的 Spot 機群。	arn:aws:ec2: <i>us-east-1</i> : <i>123456789</i> <i>012</i> :spot-fleet-request/sfr- <i>11111111-1111-1111-1111-1111</i>

## 建立 Spot Fleet 請求

使用 AWS Management Console，只選擇您的應用程式或工作需求以及最低運算規格，以快速建立 Spot Fleet 請求。Amazon EC2 會設定最符合您的需求並遵循 Spot 最佳實務的機群。如需詳細資訊，

請參閱 [快速建立 Spot Fleet 請求 \(主控台\)](#)。否則，您可以修改任何預設設定。如需詳細資訊，請參閱 [使用已定義的參數建立 Spot Fleet 請求 \(主控台\)](#) 及 [使用建立競價型叢集 AWS CLI](#)。

用於建立 Spot Fleet 的選項

- [快速建立 Spot Fleet 請求 \(主控台\)](#)
- [使用已定義的參數建立 Spot Fleet 請求 \(主控台\)](#)
- [使用建立競價型叢集 AWS CLI](#)

快速建立 Spot Fleet 請求 (主控台)

依照這些步驟快速建立 Spot Fleet 請求。

若要使用建議的設定建立 Spot Fleet 請求 (主控台)

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 如果您是第一次使用 Spot，將會顯示歡迎頁面；請選擇開始使用。否則，請選擇請求競價型執行個體。
4. 在 Launch parameters (啟動參數) 下，選擇 Manually configure launch parameters (手動設定啟動參數)。
5. 對於 AMI，選擇 AMI。
6. 在 Target capacity (目標容量) 下，針對 Total target capacity (總目標容量)，指定要請求的單位數目。對於單位類型，您可以選擇 Instances (執行個體)、vCPUs, 或 Memory (MiB) (記憶體 (MiB))。
7. 對於 Your fleet request at a glance (您的機群請求一覽)，檢閱您的機群組態，然後選擇 Launch (啟動)。


使用已定義的參數建立 Spot Fleet 請求 (主控台)

您可以使用您定義的參數建立 Spot Fleet。

若要使用已定義的參數建立 Spot Fleet 請求 (主控台)

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot 請求。

3. 如果您是第一次使用 Spot，將會顯示歡迎頁面；請選擇開始使用。否則，請選擇請求競價型執行個體。
4. 對於 Launch parameters (啟動參數)，請執行下列動作：
  - a. 若要在 Spot 主控台中定義啟動參數，請選擇 Manually configure launch parameters (手動設定啟動參數)。
  - b. 對於 AMI，請選擇由提供的基本 AMI 之一 AWS，或選擇 [搜尋 AMI] 以使用我們使用者社群 AWS Marketplace、或您自己的使用者社群中的 AMI。

 Note

如果啟動參數中指定的 AMI 已取消註冊或停用，則無法從 AMI 啟動新執行個體。對於設定為維持目標容量的車隊，將不會維持目標容量。

- c. (選用) 針對金鑰對名稱，選擇現有的金鑰對或建立新的金鑰對。

[現有的金鑰對] 選擇金鑰對。

[新的金鑰對] 選擇 Create new key pair (建立新的金鑰對) 以前往 Key Pairs (金鑰對) 頁面。當您完成時，請返回 Spot Requests (Spot 請求) 頁面並重新整理清單。

- d. (選用) 展開 Additional launch parameters (其他啟動參數)，並執行下列動作：
  - i. (選用) 若要啟用 Amazon EBS 最佳化，請針對 EBS-optimized (EBS 最佳化)，選取 Launch EBS-optimized instances (啟動 EBS 最佳化的執行個體)。
  - ii. (選用) 若要新增執行個體的暫時區塊層級儲存，請針對執行個體存放區，選擇連接啟動。
  - iii. (選用) 若要新增儲存空間，請選擇 Add new volume (新增磁碟區)，並指定其他的執行個體存放區磁碟區或 Amazon EBS 磁碟區，取決於執行個體類型。
  - iv. (選用) 執行個體預設為啟用基本監控。若要啟用詳細監視，對於監視，請選取啟用 CloudWatch 詳細監視。
  - v. (選用) 若要執行專用 Spot 執行個體，對於租用，請選擇 Dedicated - run a dedicated instance (專用 - 執行專用執行個體)。
  - vi. (選用) 針對安全群組，選擇一或多個安全群組或建立新的安全群組。

[現有的安全群組] 選擇一或多個安全群組。

[新的安全群組] 選擇 Create new security group (建立新安全群組) 以前往 Security Groups (安全群組) 頁面。當您完成時，請返回 Spot Requests (Spot 請求) 並重新整理清單。

- vii. (選用) 若要從網際網路連線到您的執行個體，請針對自動指派 IPv4 公有 IP，選擇啟用。
- viii. (選用) 若要啟動包含 IAM 角色的 競價型執行個體，請針對 IAM 執行個體描述檔，選擇該角色。
- ix. (選用) 若要執行啟動指令碼，請將指令碼複製到使用者資料。
- x. (選用) 若要新增標籤，請選擇 Create tag (建立標籤) 並輸入標籤的鍵和值，然後選擇 Create (建立)。針對每個標籤，重複執行上述動作。

對於每個標籤，若要使用相同標籤標記執行個體和 Spot Fleet 請求，請確定同時選取 Instances (執行個體) 和 Fleet (機群)。若只要標記由機群啟動的執行個體，請清除 Fleet (機群)。若只要標記 Spot Fleet 請求，請清除 Instances (執行個體)。

5. 針對其他請求詳細資訊，執行下列操作：

- a. 檢閱其他的請求詳細資訊。若要進行變更，請清除套用預設值。
- b. (選用) 針對 IAM 叢集角色，您可使用預設角色或選擇不同的角色。若要在變更角色後使用預設角色，請選擇使用預設角色。
- c. (選用) 針對最高價格，您可以使用預設的最高價格 (隨需價格)，或指定您願意支付的最高價格。如果最高價格低於您所選執行個體類型的 Spot 價格，則 競價型執行個體不會啟動。
- d. (選用) 若要建立僅在特定期間內有效的請求，請編輯請求有效性起始日期和請求有效性結束日期。
- e. (選用) 根據預設，我們會在 Spot Fleet 請求到期時終止 Spot 執行個體。若要在請求過期後繼續執行這些執行個體，請清除在請求過期時終止執行個體。
- f. (選用) 若要登錄 競價型執行個體與負載平衡器，請選擇從一個或多個負載平衡器接收流量，然後選擇一個或多個 Classic Load Balancer 或目標群組。

6. 針對最小運算單位，選擇應用程式或任務所需之最低硬體規格 (vCPU、記憶體和儲存)，再選擇依規格或依執行個體類型。

- 針對依規格，指定所需 vCPU 數目和記憶體數量。
- 針對依執行個體類型，接受預設的執行個體類型，或選擇變更執行個體類型選擇不同的執行個體類型。


7. 針對 Target capacity (目標容量)，執行下列動作：

- a. 針對 Total target capacity (總目標容量)，指定目標容量要請求的單位數目。對於單位類型，您可以選擇 Instances (執行個體)、vCPUs, 或 Memory (MiB) (記憶體 (MiB))。指定目標容量為 0，以便稍後可新增容量，選擇維持目標容量。
- b. (選用) 針對 Include On-Demand base capacity (包含隨需基礎容量)，指定要請求的隨需單位數量。數量必須小於總目標容量。Amazon EC2 會計算差額，並將差額分配給要請求的 Spot 單位。

 Important

若要指定選用的隨需容量，您必須先選擇啟動範本。

- c. (選擇性) Amazon EC2 預設會在競價型執行個體中斷時終止。若要維持目標容量，請選取維持目標容量。然後，您可以指定 Amazon EC2 在 Spot 執行個體中斷時終止、停止或休眠。若要這麼做，請在中斷行為中選擇對應的選項。

 Note

如果啟動參數中指定的 AMI 已取消註冊或停用，則無法從 AMI 啟動新執行個體。對於設定為維持目標容量的車隊，將不會維持目標容量。

- d. (選用) 若要允許 Spot Fleet 在機群中為現有的 Spot 執行個體發出執行個體重新平衡通知時啟動替代 Spot 執行個體，請選取 Capacity rebalance (容量重新平衡)，然後選擇執行個體取代策略。如果選擇 Launch before terminate (終止前啟動)，請指定 Spot Fleet 終止舊執行個體之前的延遲 (以秒為單位)。如需詳細資訊，請參閱 [容量重新平衡](#)。
  - e. (選用) 若要控制您為了在機群中所有的 Spot 執行個體所支付的每小時金額，請選取 Set maximum cost for Spot Instances (設定 Spot 執行個體的最大成本)，然後輸入您願意支付的每小時金額上限。達到總額上限時，Spot Fleet 會停止啟動 Spot 執行個體，即使它未符合目標容量亦然。如需詳細資訊，請參閱 [控制花費](#)。
8. 針對 Network (網路)，請執行下列動作：
- a. 針對網路，選擇現有的 VPC 或建立新的 VPC。

[現有的 VPC] 選擇 VPC。

[新的 VPC] 選擇建立新的 VPC 以進入 Amazon VPC 主控台。當您完成時，請返回精靈並重新整理清單。

- b. (選用) 對於可用區域，讓 AWS 為您的 Spot 執行個體選擇可用區域，或指定一個或多個可用區域。

如果您在可用區域中擁有一或多個子網，請從子網選擇適當的子網。若要新增子網，請選擇建立新的子網，以進入 Amazon VPC 主控台。當您完成時，請返回精靈並重新整理清單。
9. 針對 Instance type requirements (執行個體類型需求)，您可以指定執行個體屬性，並讓 Amazon EC2 利用這些屬性識別最佳的執行個體類型，或者您可以指定執行個體清單。如需詳細資訊，請參閱 [Spot Fleet 的屬性型執行個體類型選擇](#)。
  - a. 如果選擇 Specify instance attributes that match your compute requirements (指定符合運算需求的執行個體屬性)，請指定您的執行個體屬性，如下所示：
    - i. 對於 vCPUs，輸入所需的 vCPU 數量下限和上限。若要指定無限制，請選取 No minimum (無下限)、No maximum (無上限)，或兩者。
    - ii. 對於 Memory (GiB) (記憶體 (GiB))，輸入所需記憶體數量的下限和上限。若要指定無限制，請選取 No minimum (無下限)、No maximum (無上限)，或兩者。
    - iii. (選用) 對於 Additional instance attributes (其他執行個體屬性)，您可以選擇性地指定一個或多個屬性，以更詳細地表達您的運算需求。每個額外屬性都會將進一步的限制新增至您的請求。您可以省略額外屬性；省略時，就會使用預設值。如需每個屬性及其預設值的描述，請參閱 Amazon EC2 命令列參考中的 [get-spot-placement-scores](#)。
    - iv. (選用) 若要檢視具有指定屬性的執行個體類型，請展開 Preview matching instance types (預覽相符的執行個體類型)。若要排除執行個體類型，使其不在請求中使用，請選取執行個體，然後選擇 Exclude selected instance types (排除選取的執行個體類型)。
  - b. 如果選擇 Manually select instance types (手動選取執行個體類型)，Spot Fleet 會提供預設的執行個體類型清單。若要選取更多執行個體類型，請選擇 Add instance types (新增執行個體類型)、選取要在請求中使用的執行個體類型，然後選擇 Select (選取)。若要刪除執行個體類型，請選取執行個體類型並選擇 Delete (刪除)。
10. 針對 Allocation strategy (分配策略) 選擇符合您需求的策略。如需詳細資訊，請參閱 [競價型執行個體的分配策略](#)。
11. 針對 Your fleet request at a glance (您的機群請求一覽)，檢閱您的機群配置，並視需要進行任何調整。
12. (選擇性) 若要下載啟動組態的複本以搭配使用 AWS CLI，請選擇 JSON 設定。
13. 選擇 Launch (啟動)。

Spot 機群請求類型為 fleet。請求處理完成時，會新增類型為 instance 的請求，其狀態為 active 和 fulfilled。

## 使用建立競價型叢集 AWS CLI

若要使用 AWS CLI

- 請使用 [request-spot-fleet](#) 命令建立 Spot 機群請求。

```
aws ec2 request-spot-fleet --spot-fleet-request-config file://config.json
```

如需組態檔案範例，請參閱 [Spot Fleet 範例組態](#)。

下列為範例輸出：

```
{
 "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE"
}
```

## 標記 Spot Fleet

為便於將 Spot Fleet 請求分類和進行管理，您可以利用自訂中繼資料來標記這些請求。您可以在建立 Spot Fleet 請求時或之後將標記指派給請求。您可以使用 Amazon EC2 主控台或命令列工具來指派標籤。

當您標記 Spot Fleet 請求時，由 Spot Fleet 啟動的執行個體和磁碟區不會自動加上標記。您需要明確標記由 Spot Fleet 啟動的執行個體和磁碟區。您可以選擇僅將標籤指派給 Spot Fleet 請求，或僅指派給機群啟動的執行個體，或僅指派給連接至機群啟動之執行個體的磁碟區，或指派給前述三項。

### Note

磁碟區標籤僅支援連接至 隨需執行個體 的磁碟區。您無法為連接至 競價型執行個體 的磁碟區加標籤。

如需標籤運作方式的詳細資訊，請參閱 [標記您的 Amazon EC2 資源](#)。

## 目錄

- [必要條件](#)
- [標記新的 Spot Fleet](#)
- [標記新的 Spot Fleet 以及執行個體和其啟動的磁碟區](#)



- [標記現有 Spot Fleet](#)
- [檢視 Spot Fleet 請求標籤](#)

## 必要條件

授予使用者標記資源的許可。如需詳細資訊，請參閱 [範例：標籤資源](#)。

授予使用者標記資源的許可

建立包含下列項目的 IAM 政策：

- `ec2:CreateTags` 動作。這會授予使用者建立標籤的許可。
- `ec2:RequestSpotFleet` 動作。這會授予使用者建立 Spot 機群請求的許可。
- 對於 `Resource`，您必須指定 `"*"`。這可讓使用者標記所有資源類型。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "TagSpotFleetRequest",
 "Effect": "Allow",
 "Action": [
 "ec2:CreateTags",
 "ec2:RequestSpotFleet"
],
 "Resource": "*"
 }
]
}
```

### Important

我們目前不支援 `spot-fleet-request` 資源的資源層級許可。如果您指定 `spot-fleet-request` 作為資源，當您嘗試標記機群時，會收到未經授權的例外狀況。下列範例說明如何不要設定政策。

```
{
 "Effect": "Allow",
 "Action": [
```

```
 "ec2:CreateTags",
 "ec2:RequestSpotFleet"
],
 "Resource": "arn:aws:ec2:us-east-1:111122223333:spot-fleet-request/*"
}
```

若要提供存取權，請新增權限至您的使用者、群組或角色：

- 使用者和群組位於 AWS IAM Identity Center：

建立權限合集。請按照 AWS IAM Identity Center 使用者指南 中的 [建立權限合集](#) 說明進行操作。

- 透過身分提供者在 IAM 中管理的使用者：

建立聯合身分的角色。請按照 IAM 使用者指南 的 [為第三方身分提供者 \(聯合\) 建立角色](#) 中的指示進行操作。

- IAM 使用者：

- 建立您的使用者可擔任的角色。請按照 IAM 使用者指南 的 [為 IAM 使用者建立角色](#) 中的指示進行操作。
- (不建議) 將政策直接附加至使用者，或將使用者新增至使用者群組。請遵循 IAM 使用者指南的 [新增許可到使用者 \(主控台\)](#) 中的指示。

## 標記新的 Spot Fleet

若要使用主控台標記新的 Spot Fleet 請求

1. 遵循 [使用已定義的參數建立 Spot Fleet 請求 \(主控台\)](#) 程序。
2. 若要新增標籤，請展開其他組態，選擇新增標籤，然後輸入標籤的金鑰和值。針對每個標籤，重複執行上述動作。

對於每個標籤，您可以標記 Spot Fleet 請求和具有相同標籤的執行個體。若要標記兩者，請確定已同時選取執行個體標籤和機群標籤。若只要標記 Spot Fleet 請求，請清除 Instance tags (執行個體標籤)。若只要標記由機群啟動的執行個體，請清除機群標籤。

3. 完成建立 Spot Fleet 請求的必要欄位，然後選擇 Launch (啟動)。如需詳細資訊，請參閱 [使用已定義的參數建立 Spot Fleet 請求 \(主控台\)](#)。

若要使用標記新的 Spot 叢集請求 AWS CLI

若要在建立 Spot Fleet 請求時加上標記，請依下列方式設定 Spot Fleet 請求組態：

- 在 SpotFleetRequestConfig 中指定 Spot Fleet 請求的標籤。
- 對於 ResourceType，請指定 spot-fleet-request。若指定其他值，機群請求將會失敗。
- 對於 Tags，指定金鑰值對。您可以指定多個金鑰值對。

在下面的例子中，Spot Fleet 請求標記了兩個標籤：Key=Environment and Value=Production 和 Key=Cost-Center and Value=123。

```
{
 "SpotFleetRequestConfig": {
 "AllocationStrategy": "priceCapacityOptimized",
 "ExcessCapacityTerminationPolicy": "default",
 "IamFleetRole": "arn:aws:iam::111122223333:role/aws-ec2-spot-fleet-tagging-
role",
 "LaunchSpecifications": [
 {
 "ImageId": "ami-0123456789EXAMPLE",
 "InstanceType": "c4.large"
 }
],
 "SpotPrice": "5",
 "TargetCapacity": 2,
 "TerminateInstancesWithExpiration": true,
 "Type": "maintain",
 "ReplaceUnhealthyInstances": true,
 "InstanceInterruptionBehavior": "terminate",
 "InstancePoolsToUseCount": 1,
 "TagSpecifications": [
 {
 "ResourceType": "spot-fleet-request",
 "Tags": [
 {
 "Key": "Environment",
 "Value": "Production"
 },
 {
 "Key": "Cost-Center",
 "Value": "123"
 }
]
 }
]
 }
}
```

```
]
 }
}
```

標記新的 Spot Fleet 以及執行個體和其啟動的磁碟區

標記新的 Spot 叢集請求及其啟動的執行個體和磁碟區 AWS CLI

若要在建立 Spot Fleet 請求時加上標記，並在機群啟動執行個體和磁碟區時給它們加上標記，請依下列方式設定 Spot Fleet 請求組態：

Spot Fleet 請求標籤：

- 在 `SpotFleetRequestConfig` 中指定 Spot Fleet 請求的標籤。
- 對於 `ResourceType`，請指定 `spot-fleet-request`。若指定其他值，機群請求將會失敗。
- 對於 `Tags`，指定金鑰值對。您可以指定多個金鑰值對。

執行個體標籤：

- 在 `LaunchSpecifications` 中指定執行個體的標籤。
- 對於 `ResourceType`，請指定 `instance`。若指定其他值，機群請求將會失敗。
- 對於 `Tags`，指定金鑰值對。您可以指定多個金鑰值對。

或者，您可以在 Spot Fleet 請求中參照的[啟動範本](#)中，指定執行個體的標籤。

磁碟區標籤：

- 在 Spot Fleet 請求中參照的[啟動範本](#)中，指定磁碟區的標籤。不支援在 `LaunchSpecifications` 中標記磁碟區。

在下面的例子中，Spot Fleet 請求標記了兩個標籤：Key=Environment and Value=Production 和 Key=Cost-Center and Value=123。由機群啟動的執行個體標記了一個標籤 (這與 Spot Fleet 請求的標籤之一相同)：Key=Cost-Center and Value=123。

```
{
 "SpotFleetRequestConfig": {
 "AllocationStrategy": "priceCapacityOptimized",
 "ExcessCapacityTerminationPolicy": "default",
```

```
 "IamFleetRole": "arn:aws:iam::111122223333:role/aws-ec2-spot-fleet-tagging-
role",
 "LaunchSpecifications": [
 {
 "ImageId": "ami-0123456789EXAMPLE",
 "InstanceType": "c4.large",
 "TagSpecifications": [
 {
 "ResourceType": "instance",
 "Tags": [
 {
 "Key": "Cost-Center",
 "Value": "123"
 }
]
 }
]
 }
],
 "SpotPrice": "5",
 "TargetCapacity": 2,
 "TerminateInstancesWithExpiration": true,
 "Type": "maintain",
 "ReplaceUnhealthyInstances": true,
 "InstanceInterruptionBehavior": "terminate",
 "InstancePoolsToUseCount": 1,
 "TagSpecifications": [
 {
 "ResourceType": "spot-fleet-request",
 "Tags": [
 {
 "Key": "Environment",
 "Value": "Production"
 },
 {
 "Key": "Cost-Center",
 "Value": "123"
 }
]
 }
]
 }
}
```

## 若要使用標記 Spot 叢集啟動的執行個體 AWS CLI

若要在機群啟動執行個體時加上標記，您可以在 Spot Fleet 請求中參照的[啟動範本](#)中指定標籤，或者您可以指定 Spot Fleet 請求組態中的標籤，如下所示：

- 在 LaunchSpecifications 中指定執行個體的標籤。
- 對於 ResourceType，請指定 instance。若指定其他值，機群請求將會失敗。
- 對於 Tags，指定金鑰值對。您可以指定多個金鑰值對。

在下列範例中，機群啟動的執行個體均標記了一個標籤：Key=Cost-Center and Value=123。

```
{
 "SpotFleetRequestConfig": {
 "AllocationStrategy": "priceCapacityOptimized",
 "ExcessCapacityTerminationPolicy": "default",
 "IamFleetRole": "arn:aws:iam::111122223333:role/aws-ec2-spot-fleet-tagging-
role",
 "LaunchSpecifications": [
 {
 "ImageId": "ami-0123456789EXAMPLE",
 "InstanceType": "c4.large",
 "TagSpecifications": [
 {
 "ResourceType": "instance",
 "Tags": [
 {
 "Key": "Cost-Center",
 "Value": "123"
 }
]
 }
]
 }
]
 },
 "SpotPrice": "5",
 "TargetCapacity": 2,
 "TerminateInstancesWithExpiration": true,
 "Type": "maintain",
 "ReplaceUnhealthyInstances": true,
 "InstanceInterruptionBehavior": "terminate",
 "InstancePoolsToUseCount": 1
}
```

```
}
```

若要使用 AWS CLI 來標記連接至由 Spot Fleet 啟動的隨需執行個體的磁碟區

若要在機群建立磁碟區時標記磁碟區，請在 Spot Fleet 請求中參考的[啟動範本](#)中指定標籤。

#### Note

磁碟區標籤僅支援連接至 隨需執行個體 的磁碟區。您無法為連接至 競價型執行個體 的磁碟區加標籤。

不支援在 LaunchSpecifications 中標記磁碟區。

## 標記現有 Spot Fleet

若要使用主控台標記現有的 Spot Fleet 請求

建立 Spot Fleet 請求後，您可以使用主控台為機群請求新增標籤。

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取您的 Spot Fleet 請求。
4. 選擇標籤索引標籤，然後選擇建立標籤。

若要使用標記現有的 Spot 叢集請求 AWS CLI

使用 [create-tags](#) 命令來標記現有資源。在下面的例子中，現有的 Spot Fleet 請求會使用 Key=purpose 和 Value=test 標籤。

```
aws ec2 create-tags \
 --resources sfr-11112222-3333-4444-5555-66666EXAMPLE \
 --tags Key=purpose,Value=test
```

## 檢視 Spot Fleet 請求標籤

若要使用主控台檢視 Spot Fleet 請求標籤

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot 請求。

### 3. 選取您的 Spot Fleet 請求並選擇 Tags (標籤)。

若要描述 Spot Fleet 請求標籤

使用 [describe-tag](#) 命令檢視指定資源的標籤。在下面的範例中，會描述指定 Spot Fleet 請求的標籤。

```
aws ec2 describe-tags \
 --filters "Name=resource-id,Values=sfr-11112222-3333-4444-5555-66666EXAMPLE"
```

```
{
 "Tags": [
 {
 "Key": "Environment",
 "ResourceId": "sfr-11112222-3333-4444-5555-66666EXAMPLE",
 "ResourceType": "spot-fleet-request",
 "Value": "Production"
 },
 {
 "Key": "Another key",
 "ResourceId": "sfr-11112222-3333-4444-5555-66666EXAMPLE",
 "ResourceType": "spot-fleet-request",
 "Value": "Another value"
 }
]
}
```

您也可以透過描述 Spot Fleet 請求來檢視 Spot Fleet 請求的標籤。

使用 [describe-spot-fleet-requests](#) 命令來檢視指定 Spot Fleet 請求的組態，這包括針對機群請求指定的任何標籤。

```
aws ec2 describe-spot-fleet-requests \
 --spot-fleet-request-ids sfr-11112222-3333-4444-5555-66666EXAMPLE
```

```
{
 "SpotFleetRequestConfigs": [
 {
 "ActivityStatus": "fulfilled",
 "CreateTime": "2020-02-13T02:49:19.709Z",
 "SpotFleetRequestConfig": {
 "AllocationStrategy": "capacityOptimized",

```



```
 "OnDemandAllocationStrategy": "lowestPrice",
 "ExcessCapacityTerminationPolicy": "Default",
 "FulfilledCapacity": 2.0,
 "OnDemandFulfilledCapacity": 0.0,
 "IamFleetRole": "arn:aws:iam::111122223333:role/aws-ec2-spot-fleet-
tagging-role",
 "LaunchSpecifications": [
 {
 "ImageId": "ami-0123456789EXAMPLE",
 "InstanceType": "c4.large"
 }
],
 "TargetCapacity": 2,
 "OnDemandTargetCapacity": 0,
 "Type": "maintain",
 "ReplaceUnhealthyInstances": false,
 "InstanceInterruptionBehavior": "terminate"
 },
 "SpotFleetRequestId": "sfr-11112222-3333-4444-5555-66666EXAMPLE",
 "SpotFleetRequestState": "active",
 "Tags": [
 {
 "Key": "Environment",
 "Value": "Production"
 },
 {
 "Key": "Another key",
 "Value": "Another value"
 }
]
}
]
```

## 說明您的 Spot 機群

當您的最高價超過 Spot 價格，而且有可用的容量時，Spot Fleet 會啟動 Spot 執行個體。競價型執行個體會持續運作直到中斷或您自行將其終止。

如要說明您的 Spot 機群 (主控台)

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot 請求。

3. 選取您的 Spot Fleet 請求。若要查看組態的詳細資訊，請選擇描述。
4. 若要列出 Spot Fleet 的 Spot 執行個體，請選擇 Instances (執行個體)。
5. 若要檢視 Spot Fleet 的歷史記錄，請選擇 History (歷史記錄)。

如要說明您的 Spot 機群 (AWS CLI)

請使用 [describe-spot-fleet-requests](#) 命令來描述 Spot Fleet 請求。

```
aws ec2 describe-spot-fleet-requests
```

請使用 [describe-spot-fleet-instances](#) 命令來描述所指定 Spot Fleet 的 Spot 執行個體。

```
aws ec2 describe-spot-fleet-instances \
 --spot-fleet-request-id sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE
```

請使用 [describe-spot-fleet-request-history](#) 命令來描述所指定 Spot Fleet 請求的歷史記錄。

```
aws ec2 describe-spot-fleet-request-history \
 --spot-fleet-request-id sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
 --start-time 2015-05-18T00:00:00Z
```

## 修改 Spot Fleet 請求

您可以修改作用中的 Spot Fleet 請求，來完成下列任務：

- 增加目標容量和隨需部分
- 減少目標容量和隨需部分

### Note

您無法修改一次性的 Spot Fleet 請求。當您建立 Spot Fleet 請求時，如果已選取 Maintain target capacity (維護目標容量)，則只能修改 Spot Fleet 請求。

當您增加目標容量時，Spot Fleet 會啟動額外的 Spot 執行個體。當您增加隨需部分時，Spot Fleet 會啟動額外的隨需執行個體。

當您增加目標容量時，Spot 叢集會根據其 Spot 叢集請求的[配置策略](#)啟動額外的 Spot 執行個體。

當您減少目標容量時，Spot Fleet 會取消超出新目標容量的所有開放要求。您可以請求 Spot Fleet 終止 Spot 執行個體，直到機群大小達到新的目標容量。如果分配策略為 `diversified`，則 Spot Fleet 會在集區中終止執行個體。或者，您可以請求該 Spot Fleet 將機群保持為目前的大小，但不能取代任何中斷的或您手動終止的 Spot 執行個體。

當 Spot Fleet 因為目標容量減少而終止執行個體時，執行個體會收到 Spot 執行個體中斷通知。

若要修改 Spot Fleet 請求 (主控台)

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取您的 Spot Fleet 請求。
4. 選擇動作和修改目標容量。
5. 在修改目標容量中，執行下列動作：
  - a. 輸入新的目標容量和隨需部分。
  - b. (選用) 如果您正在減少目標容量，但希望叢集保持目前的大小，請清除終止執行個體。
  - c. 選擇提交。

若要使用修改 Spot 叢集請求 AWS CLI

請使用 [modify-spot-fleet-request](#) 命令更新指定 Spot Fleet 請求的目標容量。

```
aws ec2 modify-spot-fleet-request \
 --spot-fleet-request-id sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
 --target-capacity 20
```

如果要減少所指定 Spot Fleet 的目標容量，而不會因此終止任何 Spot 執行個體，您可以如下修改先前的命令：

```
aws ec2 modify-spot-fleet-request \
 --spot-fleet-request-id sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
 --target-capacity 10 \
 --excess-capacity-termination-policy NoTermination
```

## 取消 Spot Fleet 請求

如果不再需要 Spot 機群，則可取消 Spot 機群請求。取消機群請求後，也會取消與機群相關的所有 Spot 請求，因此不會啟動新的 Spot 執行個體。

刪除時 Spot 機群請求時，必須指定是否也要終止其全部執行個體。這包含隨需執行個體和 Spot 執行個體。

如果指定在刪除機群請求時必須終止執行個體，則機群請求會進入 `cancelled_terminating` 狀態。否則，機群請求會進入 `cancelled_running` 狀態，而執行個體會持續執行直到中斷，或直到您手動終止這些執行個體。

## 限制

- 您可以在單一要求中刪除多達 100 個叢集。如果超過指定的數目，則不會刪除任何叢集。

## 若要取消 Spot Fleet 請求 (主控台)

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取您的 Spot Fleet 請求。
4. 選擇動作、取消請求。
5. 在取消 Spot 請求對話方塊中，執行下列操作：
  - a. 若要在取消 Spot 機群請求的同時終止關聯的執行個體，請將終止執行個體核取方塊保持為選取狀態。若要取消 Spot 機群請求，而不終止關聯的執行個體，請清除終止執行個體核取方塊。
  - b. 選擇確認。

## 若要取消 Spot 叢集請求並使用 AWS CLI

請使用 [cancel-spot-fleet-requests](#) 命令取消指定的 Spot 機群請求並終止其隨需執行個體和 Spot 執行個體。

```
aws ec2 cancel-spot-fleet-requests \
 --spot-fleet-request-ids sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
 --terminate-instances
```

## 範例輸出

```
{
 "SuccessfulFleetRequests": [
 {
```

```

 "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
 "CurrentSpotFleetRequestState": "cancelled_terminating",
 "PreviousSpotFleetRequestState": "active"
 }
],
"UnsuccessfulFleetRequests": []
}

```

使用 AWS CLI 取消 Spot 機群請求，而不終止其執行個體

您可以使用 `--no-terminate-instances` 參數來修改前一個命令，以取消指定的 Spot 機群請求，而無需終止其隨需執行個體和 Spot 執行個體。

```

aws ec2 cancel-spot-fleet-requests \
 --spot-fleet-request-ids sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
 --no-terminate-instances

```

範例輸出

```

{
 "SuccessfulFleetRequests": [
 {
 "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
 "CurrentSpotFleetRequestState": "cancelled_running",
 "PreviousSpotFleetRequestState": "active"
 }
],
 "UnsuccessfulFleetRequests": []
}

```

## CloudWatch 競價型艦隊的指標

Amazon EC2 提供可用來監控競價型叢集的 Amazon CloudWatch 指標。

### Important

為確保準確性，建議您在使用這些指標時，啟用詳細監控功能。如需詳細資訊，請參閱 [啟用或關閉執行個體的詳細監控](#)。

如需 Amazon EC2 提供 CloudWatch 指標的詳細資訊，請參閱 [使用監控執行個體 CloudWatch](#)。

## Spot 機群指標

AWS/EC2Spot命名空間包含下列指標，以及叢集中 Spot 執行個體的CloudWatch 指標。如需詳細資訊，請參閱 [執行個體指標](#)。

指標	描述
AvailableInstancePoolsCount	Spot Fleet 請求中指定的 Spot 容量集區。  單位：計數
BidsSubmittedForCapacity	Amazon EC2 已提交 Spot Fleet 請求的容量。  單位：計數
EligibleInstancePoolCount	在 Spot Fleet 請求中指定的 Spot 容量集區，Amazon EC2 可在其中實現請求。在您願意為 Spot 執行個體支付的最高價格低於 Spot 價格或 Spot 價格高於隨需執行個體價格的集區中，Amazon EC2 不會實現請求。  單位：計數
FulfilledCapacity	Amazon EC2 已履行的容量。  單位：計數
MaxPercentCapacityAllocation	在 Spot Fleet 請求中指定的所有 Spot Fleet 集區的 PercentCapacityAllocation 的最大值。  單位：百分比
PendingCapacity	TargetCapacity 與 FulfilledCapacity 之間的差異。  單位：計數

指標	描述
PercentCapacityAllocation	已為指定維度的 Spot 容量集區配置的容量。若要取得所有 Spot 容量集區所記錄的最大值，請使用 MaxPercentCapacityAllocation。  單位：百分比
TargetCapacity	Spot Fleet 請求的目標容量。  單位：計數
TerminatingCapacity	由於佈建容量而終止的容量大於目標容量。  單位：計數

如果指標的測量單位是 Count，則最有用的統計資訊是 Average。

## Spot Fleet 維度

若要篩選 Spot Fleet 的資料，請使用下列維度。

維度	描述
AvailabilityZone	依可用區域篩選資料。
FleetRequestId	依 Spot Fleet 請求篩選資料。
InstanceType	依執行個體類型篩選資料。

## 檢視 Spot 叢集的 CloudWatch 指標

您可以使用 Amazon CloudWatch 主控台檢視競價型叢集的 CloudWatch 指標。這些指標會以監控圖表的形式顯示。如果 Spot Fleet 正在使用中，這些圖表會顯示資料點。

指標會先依據命名空間進行分組，然後再根據各命名空間內不同的維度組合來分組。例如，您可以按 Spot Fleet 請求 ID、執行個體類型或可用區域來檢視所有 Spot Fleet 指標或 Spot Fleet 指標群組。

### 若要檢視 Spot Fleet 指標

1. 開啟主 CloudWatch 控制台，網址為 <https://console.aws.amazon.com/cloudwatch/>。
2. 在導覽窗格中，選擇 Metrics (指標)。
3. 選擇 EC2 Spot 命名空間。

#### Note

若未顯示 EC2 Spot 命名空間，這有兩個原因。您尚未使用 Spot 叢集，只有您使用的 AWS 服務會將指標傳送至 Amazon。CloudWatch 或是因為您過去內兩週未曾使用 Spot Fleet，因此沒有顯示此命名空間。

4. (選用) 若要根據維度來篩選指標，請選擇下列其中一個選項：
  - 機群請求指標 – 依 Spot Fleet 請求分組
  - 根據可用區域 – 依 Spot Fleet 請求和可用區域分組
  - 根據執行個體類型 – 依 Spot Fleet 請求和執行個體類型分組
  - 根據可用區域/執行個體類型 – 依 Spot Fleet 請求、可用區域和執行個體類型分組
5. 若要檢視指標的資料，請勾選指標旁的核取方塊。

EC2 Spot Search Metrics 1 to 18 of 18 Metrics

Fleet Request Metrics By Availability Zone By Instance Type By Availability Zone/Instance Type

Showing all results (18) for EC2 Spot > Fleet Request Metrics. For more results expand your search to All EC2 Spot Metrics.

Select All | Clear

FleetRequestId	Metric Name
<input type="checkbox"/> sfr-4a707781-8fac-459b-a5ae-4701fcee47d7	AvailableInstancePoolsCount
<input type="checkbox"/> sfr-4a707781-8fac-459b-a5ae-4701fcee47d7	BidsSubmittedForCapacity
<input checked="" type="checkbox"/> sfr-4a707781-8fac-459b-a5ae-4701fcee47d7	CPUUtilization
<input type="checkbox"/> sfr-4a707781-8fac-459b-a5ae-4701fcee47d7	DiskReadBytes



## Spot Fleet 的自動擴展

自動擴展是根據需求自動增加或減少 Spot Fleet 目標容量的能力。Spot Fleet 可以在您選擇的範圍內，啟動執行個體 (向外擴展) 或終止執行個體 (向內擴展)，以回應一個或多個擴展政策。

Spot Fleet 支援以下類型的自動擴展：

- [目標追蹤擴展](#) – 根據特定指標的目標值，來增加或減少機群目前的容量。這和您家中溫控器維持溫度的方式類似 — 您只要選擇溫度，然後溫控器就會完成其他的動作。
- [步進擴展](#) – 根據一組擴展調整 (也稱為步進調整)，來增加或減少機群目前的容量，其會根據警示違規的程度而有不同。
- [排程的擴展](#) – 根據日期和時間，來增加或減少機群目前的容量。

如果您要使用[執行個體權重](#)，請記住 Spot Fleet 可視需要超過目標容量。已實現的容量可以是浮點數，但目標容量必須是整數，因此 Spot Fleet 會四捨五入為下一個整數。如果檢視警示觸發時擴展政策的結果，您必須將這些行為列入考量。例如，假設目標容量為 30、已實現的容量為 30.1，則規模調整政策會減掉 1。當警示觸發時，自動擴展程序會從 30.1 減掉 1 而得出 29.1，然後再四捨五入為 30，因此不需採取規模調整動作。在另一個範例中，假設選擇了 2、4 和 8 的執行個體權重，而目標容量為 10，但沒有可用的權重 2 執行個體，因此 Spot Fleet 針對已實現的容量 12 佈建權重為 4 和 8 的執行個體。如果規模調整政策將目標容量減少了 20%，而且警示已觸發，則自動擴展程序會從 12 減掉  $12 * 0.2$  而得出 9.6，然後再四捨五入為 10，因此不需採取規模調整動作。

您為 Spot Fleet 建立的擴展政策支援冷卻時間。這是在擴展活動完成之後，先前的觸發相關擴展活動還能影響未來擴展活動的秒數。如果使用向外擴展政策，則在冷卻時間仍有效時，由先前起始冷卻時間的向外擴展事件所新增的容量，會在計算時列入下次向外擴展所需容量的一部分。這種做法的目的是連續的向外擴展 (但並非過度)。如果使用向內擴展政策，在冷卻時間尚未到期之前，會利用此期間來封鎖請求中後續的擴展。這種做法的目的是保守地進行擴展，以保障應用程式的可用性。不過，如果在向內擴展後的冷卻時間內，有另一個警示觸發了向外擴展政策，則自動擴展功能會立即擴展可擴展的目標。

建議您依據 1 分鐘頻率的基於執行個體指標進行擴展，因為這樣能確保更快速地回應使用率變更。5 分鐘頻率的擴展指標可能會導致較慢的回應時間和用到過時的指標資料進行擴展。若要 CloudWatch 在 1 分鐘內將執行個體的指標資料傳送至，您必須特別啟用詳細監控。如需詳細資訊，請參閱 [啟用或關閉執行個體的詳細監控](#) 及 [使用已定義的參數建立 Spot Fleet 請求 \(主控台\)](#)。

如需設定 Spot Fleet 擴展的詳細資訊，請參閱以下資源：

- AWS CLI 命令參考中的 [應用程式自動擴展](#) 一節
- [Application Auto Scaling API 參考](#)

- [Application Auto Scaling 使用者指南](#)

## Spot Fleet 自動擴展所需的 IAM 許可

透過 Amazon EC2、Amazon CloudWatch 和應用程式自動擴展 API 的組合，可實現競價型叢集的自動擴展。Spot 叢集請求是使用 Amazon EC2 建立的，使用建立警示 CloudWatch，並使用應用程式自動擴展建立擴展政策。

除了 [Spot 機群](#) 和 Amazon EC2 的 IAM 許可之外，存取機群擴展設定的使用者必須擁有支援動態擴展之服務的適當許可。使用者必須具備使用以下範例政策中各項動作的許可。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "application-autoscaling:*",
 "ec2:DescribeSpotFleetRequests",
 "ec2:ModifySpotFleetRequest",
 "cloudwatch:DeleteAlarms",
 "cloudwatch:DescribeAlarmHistory",
 "cloudwatch:DescribeAlarms",
 "cloudwatch:DescribeAlarmsForMetric",
 "cloudwatch:GetMetricStatistics",
 "cloudwatch:ListMetrics",
 "cloudwatch:PutMetricAlarm",
 "cloudwatch:DisableAlarmActions",
 "cloudwatch:EnableAlarmActions",
 "iam:CreateServiceLinkedRole",
 "sns:CreateTopic",
 "sns:Subscribe",
 "sns:Get*",
 "sns:List*"
],
 "Resource": "*"
 }
]
}
```

您也可以建立自己的 IAM 政策，以便具有更精細的 Application Auto Scaling API 呼叫許可。如需詳細資訊，請參閱 Application Auto Scaling 使用者指南中的身分 [身分驗證與存取控制](#)。

應用程式 Auto Scaling 服務還需要權限來描述您的競價型叢集和 CloudWatch 警報，以及代表您修改 Spot 叢集目標容量的權限。如果您為 Spot Fleet 啟用自動擴展，則會建立名為 `AWSServiceRoleForApplicationAutoScaling_EC2SpotFleetRequest` 的服務連結角色。此服務連結角色會授與 Application Auto Scaling 許可以描述政策的警示、監控機群目前的容量，以及修改機群的容量。Application Auto Scaling 的原始受管 Spot Fleet 角色為 `aws-ec2-spot-fleet-autoscale-role`，但已不再需要。服務連結的角色預設為 Application Auto Scaling。如需詳細資訊，請參閱 Application Auto Scaling 使用者指南中的 [服務連結角色](#)。

## 使用目標追蹤政策來擴展 Spot Fleet

使用目標追蹤擴展政策，您可以選取指標及設定目標值。Spot Fleet 會建立和管理觸發擴展政策的 CloudWatch 警示，並根據指標和目標值計算擴展調整。規模調整政策會視需要新增或移除容量，以讓指標保持在等於或接近指定目標值。除了讓指標保持在接近目標值之外，目標追蹤規模調整政策也會配合指標中，因為負載模式波動所造成的波動調整，並將機群容量中的快速波動降到最低。

您可以為 Spot Fleet 建立多個目標追蹤規模調整政策，但前提是每個政策都使用不同的指標。機群會根據提供最大機群容量的政策，來進行擴展。這可涵蓋多種情況，確保您永遠有足夠的容量可處理您的應用程式工作負載。

為了確保應用程式的可用性，機群可以配合指標按比例快速地擴展規模，但是以更漸進的方式縮減規模。

當 Spot Fleet 因為目標容量減少而終止執行個體時，執行個體會收到 Spot 執行個體中斷通知。

請勿編輯或刪除 Spot 叢集針對目標追蹤擴展政策管理的 CloudWatch 警示。當您刪除目標追蹤擴展政策時，Spot Fleet 會自動刪除警示。

### 限制

Spot Fleet 請求必須具有 `maintain` 的請求類型。類型 `request` 的請求不支援自動擴展功能。

### 設定目標追蹤政策 (主控台)

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取 Spot Fleet 請求，然後選擇 Auto Scaling。
4. 如果未設定自動擴展，請選擇設定。
5. 利用容量擴展的範圍來設定機群的最低和最高容量。自動擴展功能不會將機群的規模擴展到小於最低容量或大於最高容量。
6. 針對 Policy Name (政策名稱)，請輸入政策的名稱。

7. 選擇目標指標。
8. 輸入指標的目標值。
9. 針對冷卻時間，請指定一個新值 (以秒為單位) 或保留預設值。
10. (選用) 選擇停用向內擴展，來略過根據目前的組態建立向內擴展政策。您可以使用不同的組態來建立向內擴展政策。
11. 選擇 Save (儲存)。

## 使用設定目標追蹤原則 AWS CLI

1. 使用 [register-scalable-target](#) 命令，將 Spot Fleet 請求註冊為可擴充的目標。
2. 使用 [put-scaling-policy](#) 命令，來建立擴展政策。

## 使用步進擴展政策來擴展 Spot Fleet

使用步驟擴展政策，您可以指定 CloudWatch 警示以觸發擴展程序。例如，如果想在 CPU 的利用率達到某個水準時擴展，請使用 Amazon EC2 所提供的 CPUUtilization 指標來建立警示。

建立步進規模調整政策時，您必須指定下列規模調整類型的其中一種：

- 新增 – 根據指定的容量單位數量或指定的目前容量百分比，來增加機群的目標容量。
- 移除 – 根據指定的容量單位數量或指定的目前容量百分比，來減少機群的目標容量。
- 設定為 – 將機群的目標容量設定為指定的容量單位數量。

警示觸發時，自動擴展程序會利用已實現的容量和規模調整政策，來計算出新的目標容量，然後據以更新目標容量。例如，假設目標容量為和已實現的容量為 10，而且規模調整政策會增加 1。當警示觸發時，自動擴展程序會將 10 加上 1 而得出 11，因此 Spot Fleet 會啟動 1 個執行個體。

當 Spot Fleet 因為目標容量減少而終止執行個體時，執行個體會收到 Spot 執行個體中斷通知。

### 限制

Spot Fleet 請求必須具有 `maintain` 的請求類型。類型 `request` 的請求或 Spot 區塊不支援自動擴展功能。

### 必要條件

- 請考慮哪些 CloudWatch 指標對您的應用程式很重要。您可以根據提供的指標 AWS 或您自己的自訂指標建立 CloudWatch 警示。

- 對於您將在資源調度 AWS 度政策中使用的指標，如果提供 CloudWatch 指標的服務預設未啟用指標，請啟用指標收集。

## 建立 CloudWatch 鬧鐘

1. 開啟主 CloudWatch 控制台，網址為 <https://console.aws.amazon.com/cloudwatch/>。
2. 在導覽窗格中，選擇 Alarms (警示)。
3. 選擇建立警示。
4. 在指定指標和條件頁面上，選擇選取指標。
5. 選擇 EC2 Spot、叢集請求指標、選取量度 (例如 TargetCapacity)，然後選擇選取量度。

指定指標和條件頁面隨即出現，顯示您選取指標的圖形及其他資訊。

6. 針對期間，選擇警示的評估期間，例如，1 分鐘。評估警示時，每個期間都會彙整為一個資料點。

### Note

期間越短會建立更敏感的警示。

7. 針對條件，透過定義閾值條件來定義警示。例如，您可以定義一個閾值，設定每當指標值大於或等於 80% 時，即會觸發警示。
8. 在其他組態下，針對要產生警示的資料點，指定多少資料點 (評估期間) 必須處於 ALARM 狀態才會觸發警示，例如，1 評估期間或 3 個中的 2 個評估期間。這樣建立的警示會在連續超過這麼多期間時進入 ALARM 狀態。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的 [評估警示](#)。
9. 對於遺失資料處理，選擇其中之一個選項 (或保留預設的 將遺失資料視為遺失)。如需詳細資訊，請參閱 [Amazon CloudWatch 使用者指南中的設定 CloudWatch 警示如何處理遺失的資料](#)。
10. 選擇下一步。
11. (選用) 若要接收擴展事件的通知，針對通知，您可以選擇或建立您要用於接收通知的 Amazon SNS 主題。否則，您可以立即刪除通知，並根據需要在之後新增通知。
12. 選擇下一步。
13. 在新增描述下，輸入警示的名稱和描述，然後選擇下一步。
14. 選擇建立警示。

## 為您的 Spot Fleet 設定步進擴展政策 (主控台)

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取 Spot Fleet 請求，然後選擇 Auto Scaling。
4. 如果未設定自動擴展，請選擇設定。
5. 利用容量擴展的範圍來設定機群的最低和最高容量。擴展政策不會將機群的規模擴展到小於最低容量或大於最高容量。
6. 針對擴展政策、政策類型，選擇步驟擴展政策。
7. 一開始，擴展政策包含名為 ScaleUp 和 ScaleDown 的步驟擴展政策。您可以完成這些政策，或是選擇移除政策來刪除政策。您也可以選擇新增政策。
8. 若要定義政策，請執行下列的動作：
  - a. 針對 Policy Name (政策名稱)，請輸入政策的名稱。
  - b. 對於政策觸發器，選取現有警示或選擇建立警示以開啟 Amazon CloudWatch 主控台並建立警示。
  - c. 針對修改容量，請定義要擴展的量以及步驟調整的下限與上限。您可以新增或移除特定數量的執行個體，或現有機群大小的百分比，或將機群設為確切大小。

例如，若要建立可將機群容量增加 30% 的步驟擴展政策，請選擇 Add，在下一個欄位中輸入 30，然後選擇 percent。根據預設，新增政策的下限為警示閾值，而上限為無限大正數 (+) 值。根據預設，移除政策的上限為警示閾值，而下限為無限小負數 (-) 值。
  - d. (選用) 若要新增另一個步驟，請選擇新增步驟。
  - e. 針對冷卻時間，請指定一個新值 (以秒為單位) 或保留預設值。
9. 選擇儲存。

若要設定 Spot 叢集的步驟擴展原則，請使用 AWS CLI

1. 使用 [register-scalable-target](#) 命令，將 Spot Fleet 請求註冊為可擴充的目標。
2. 使用 [put-scaling-policy](#) 命令，來建立擴展政策。
3. 使用 [put-metric-alarm](#) 命令，來建立觸發擴展政策的警示。

## 使用排程擴展來擴展 Spot Fleet

按照排程進行擴展讓您能夠因應可預測的需求變化，據以擴展您的應用程式。若要使用排程的擴展，請建立排程的動作以告知 Spot Fleet 在特定的時間執行擴展活動。當您建立排程動作時，可以指定現有的 Spot Fleet、應該發生擴展活動的時間、最小容量和最大容量。您可以建立僅擴展一次或按照重複排程擴展的排程動作。

您只能針對已存在的 Spot Fleets 建立已排程動作。您無法在建立 Spot Fleet 的同時建立排程的動作。

### 限制

Spot Fleet 請求必須具有 `maintain` 的請求類型。類型 `request` 的請求或 Spot 區塊不支援自動擴展功能。

### 建立一次性排程動作

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取您的 Spot Fleet 請求，然後選擇靠近畫面底部的 Scheduled Scaling (排程的擴展) 標籤。
4. 選擇建立排程動作。
5. 在名稱中，為排程動作指定名稱。
6. 為容量下限、容量上限或兩者輸入一值。
7. 針對重複，選擇一次。
8. (選用) 針對開始時間、結束時間或兩者選擇日期和時間。
9. 選擇提交。

### 按照重複排程擴展

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取您的 Spot Fleet 請求，然後選擇靠近畫面底部的 Scheduled Scaling (排程的擴展) 標籤。
4. 針對重複，選擇其中一個預先定義的排程 (例如，每天)，或選擇自訂，然後輸入 Cron 表達式。如需排程擴展支援的 Cron 運算式的詳細資訊，請參閱 Amazon CloudWatch 事件使用者指南中的 [Cron 運算式](#)。
5. (選用) 針對開始時間、結束時間或兩者選擇日期和時間。
6. 選擇提交。

## 編輯排程動作

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取您的 Spot Fleet 請求，然後選擇靠近畫面底部的 Scheduled Scaling (排程的擴展) 標籤。
4. 選取排程動作，然後選擇動作、編輯。
5. 執行需要的變更，然後選擇提交。

## 刪除排程動作

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取您的 Spot Fleet 請求，然後選擇靠近畫面底部的 Scheduled Scaling (排程的擴展) 標籤。
4. 選取排程動作，然後選擇動作、刪除。
5. 出現確認提示時，請選擇 Delete (刪除)。

若要使用管理排程的縮放 AWS CLI

使用下列命令：

- [put-scheduled-action](#)
- [describe-scheduled-actions](#)
- [delete-scheduled-action](#)

## 使用 Amazon 監控車隊事件 EventBridge

當 EC2 Fleet 或 Spot Fleet 的狀態發生變化時，機群會發出通知。該通知作為發送到 Amazon 的事件提供 EventBridge (以前稱為 Amazon CloudWatch 活動)。盡可能發出事件。

使用 Amazon EventBridge，您可以建立規則來觸發程式設計動作以回應事件。例如，您可以建立兩個 EventBridge 規則，其中一個規則在叢集狀態變更時觸發，另一個則在叢集中的執行個體終止時觸發。如果機群狀態變化，您可以設定第一個規則，以便規則可叫用 SNS 主題來向您傳送電子郵件通知。如果執行個體終止，您可以設定第二個規則，以便規則可叫用 Lambda 函數來啟動新執行個體。

主題

- [EC2 Fleet 事件類型](#)



- [Spot Fleet 事件類型](#)
- [創建 Amazon EventBridge 規則](#)

## EC2 Fleet 事件類型

### Note

只有類型 `maintain` 和 `request` 的機群會發出事件。類型 `instant` 的機群不會發出事件，因為它們提交一次性的同步請求，而且在回應中會立即知道機群的狀態。

有五種 EC2 Fleet 事件類型。每個事件類型都有數個子類型。

事件會以 JSON 格式傳送至 EventBridge。事件中的下列欄位會形成規則中定義的事件模式，以及觸發動作的事件模式：

```
"source": "aws.ec2fleet"
```

識別事件來自 EC2 Fleet。

```
"detail-type": "EC2 Fleet State Change"
```

識別事件類型。

```
"detail": { "sub-type": "submitted" }
```

識別事件子類型。

### 事件類型

- [EC2 Fleet 狀態變更](#)
- [EC2 Fleet Spot 執行個體請求變更](#)
- [EC2 Fleet 執行個體變更](#)
- [EC2 Fleet 資訊](#)
- [EC2 Fleet 錯誤](#)

## EC2 Fleet 狀態變更

EC2 叢集變更狀態 EventBridge 時，EC2 叢集會傳送 EC2 Fleet State Change 事件至 Amazon。

以下是此事件的範例資料。

```
{
 "version": "0",
 "id": "715ed6b3-b8fc-27fe-fad6-528c7b8bf8a2",
 "detail-type": "EC2 Fleet State Change",
 "source": "aws.ec2fleet",
 "account": "123456789012",
 "time": "2020-11-09T09:00:20Z",
 "region": "us-east-1",
 "resources": [
 "arn:aws:ec2:us-east-1:123456789012:fleet/fleet-598fb973-87b7-422d-be4d-6b0809bfff0a"
],
 "detail": {
 "sub-type": "active"
 }
}
```

sub-type 的可能值為：

active

EC2 Fleet 請求已驗證且 Amazon EC2 正在嘗試維持執行中的執行個體目標數目。

deleted

EC2 機群 請求已刪除，且無執行中的執行個體。EC2 機群 會在其執行個體終止之後兩天刪除。

deleted\_running

EC2 機群 請求已刪除，且不會啟動其他執行個體。其現有執行個體將繼續執行直到中斷或終止。此請求會一直維持在該狀態，直到所有執行個體中斷或終止。

deleted\_terminating

EC2 Fleet 請求已刪除，且其執行個體已終止。此請求會一直維持在該狀態，直到所有執行個體終止。

expired

EC2 Fleet 請求已過期。如果請求是使用 `TerminateInstancesWithExpiration` 集建立的，後續 `terminated` 事件會指出已終止執行個體。

## modify\_in\_progress

正在修改 EC2 Fleet 請求。請求會一直維持在此狀態，直到已完成修改。

## modify\_succeeded

EC2 Fleet 請求已修改。

## submitted

正在評估 EC2 Fleet 請求且 Amazon EC2 正準備啟動執行個體的目標數量。

## progress

EC2 Fleet 請求正在履行的過程中。

## EC2 Fleet Spot 執行個體請求變更

當叢集中的 Spot 執 EC2 Fleet Spot Instance Request Change 行個體請求變更狀態 EventBridge 時，EC2 叢集會傳送事件至 Amazon。

以下是此事件的範例資料。

```
{
 "version": "0",
 "id": "19331f74-bf4b-a3dd-0f1b-ddb1422032b9",
 "detail-type": "EC2 Fleet Spot Instance Request Change",
 "source": "aws.ec2fleet",
 "account": "123456789012",
 "time": "2020-11-09T09:00:05Z",
 "region": "us-east-1",
 "resources": [
 "arn:aws:ec2:us-east-1:123456789012:fleet/
fleet-83fd4e48-552a-40ef-9532-82a3acca5f10"
],
 "detail": {
 "spot-instance-request-id": "sir-rmqske6h",
 "description": "SpotInstanceRequestId sir-rmqske6h, PreviousState:
cancelled_running",
 "sub-type": "cancelled"
 }
}
```

sub-type 的可能值為：

## active

Spot 執行個體請求已履行，並具有關聯的 Spot 執行個體。

## cancelled

您已取消 Spot 執行個體請求，或 Spot 執行個體請求已過期。

## disabled

您已停用 Spot 執行個體。

## submitted

已提交 Spot 執行個體請求。

## EC2 Fleet 執行個體變更

當叢集中的執行 EC2 Fleet Instance Change 個體變更狀態 EventBridge 時，EC2 叢集會傳送事件至 Amazon。

以下是此事件的範例資料。

```
{
 "version": "0",
 "id": "542ce428-c8f1-0608-c015-e8ed6522c5bc",
 "detail-type": "EC2 Fleet Instance Change",
 "source": "aws.ec2fleet",
 "account": "123456789012",
 "time": "2020-11-09T09:00:23Z",
 "region": "us-east-1",
 "resources": [
 "arn:aws:ec2:us-east-1:123456789012:fleet/fleet-598fb973-87b7-422d-be4d-6b0809bfff0a"
],
 "detail": {
 "instance-id": "i-0c594155dd5ff1829",
 "description": "{\"instanceType\":\"c5.large\",\"image\":\"ami-6057e21a\", \"productDescription\":\"Linux/UNIX\", \"availabilityZone\":\"us-east-1d\"}",
 "sub-type": "launched"
 }
}
```

sub-type 的可能值為：

## launched

已啟動新執行個體。

## terminated

已終止執行個體。

## termination\_notified

Amazon EC2 在縮減規模期間終止 Spot 執行個體，也就是機群的目標容量向下修改 (例如，從目標容量 4 變為 3) 時，系統會傳送執行個體終止通知。

## EC2 Fleet 資訊

EC2 叢集會在履行期間發生錯誤 EventBridge 時，將 EC2 Fleet Information 事件傳送至 Amazon。此資訊事件不會封鎖機群嘗試履行其目標容量。

以下是此事件的範例資料。

```
{
 "version": "0",
 "id": "76529817-d605-4571-7224-d36cc1b2c0c4",
 "detail-type": "EC2 Fleet Information",
 "source": "aws.ec2fleet",
 "account": "123456789012",
 "time": "2020-11-09T08:17:07Z",
 "region": "us-east-1",
 "resources": [
 "arn:aws:ec2:us-east-1:123456789012:fleet/fleet-8becf5fe-
bb9e-415d-8f54-3fa5a8628b91"
],
 "detail": {
 "description": "c4.xlarge, ami-0947d2ba12ee1ff75, Linux/UNIX, us-east-1a,
Spot price in either SpotFleetRequestConfigData or SpotFleetLaunchSpecification or
LaunchTemplate or LaunchTemplateOverrides is less than Spot market price $0.0619",
 "sub-type": "launchSpecUnusable"
 }
}
```

sub-type 的可能值為：

## fleetProgressHalted

每個啟動規格中的價格無效，因為價格低於 Spot 價格 (所有啟動規格都已產生 launchSpecUnusable 事件)。如果 Spot 價格變動，啟動規格則可能會變成有效。

## launchSpecTemporarilyBlacklisted

組態無效，且多次嘗試啟動執行個體失敗。如需詳細資訊，請參閱事件的描述。

## launchSpecUnusable

啟動規格中的價格無效，因為價格低於 Spot 價格。

## registerWithLoadBalancersFailed

嘗試在負載平衡器失敗時註冊執行個體。如需詳細資訊，請參閱事件的描述。

## EC2 Fleet 錯誤

EC2 叢集會在履行期間發生錯誤 EventBridge 時，將 EC2 Fleet Error 事件傳送至 Amazon。錯誤事件會阻止機群嘗試履行其目標容量。

以下是此事件的範例資料。

```
{
 "version": "0",
 "id": "69849a22-6d0f-d4ce-602b-b47c1c98240e",
 "detail-type": "EC2 Fleet Error",
 "source": "aws.ec2fleet",
 "account": "123456789012",
 "time": "2020-10-07T01:44:24Z",
 "region": "us-east-1",
 "resources": [
 "arn:aws:ec2:us-east-1:123456789012:fleet/fleet-9bb19bc6-60d3-4fd2-ae47-d33e68eafa08"
],
 "detail": {
 "description": "m3.large, ami-00068cd7555f543d5, Linux/UNIX: IPv6 is not supported for the instance type 'm3.large'. ",
 "sub-type": "spotFleetRequestConfigurationInvalid"
 }
}
```

sub-type 的可能值為：

## iamFleetRoleInvalid

EC2 Fleet 沒有啟動或終止執行個體所需的許可。

## allLaunchSpecsTemporarilyBlacklisted

沒有任何組態是有效的，且多次嘗試啟動執行個體失敗。如需詳細資訊，請參閱事件的描述。

## spotInstanceCountLimitExceeded

您已達到可啟動 Spot 執行個體的數目上限。

## spotFleetRequestConfigurationInvalid

組態無效。如需詳細資訊，請參閱事件的描述。

## Spot Fleet 事件類型

有五種 Spot Fleet 事件類型。每個事件類型都有數個子類型。

事件會以 JSON 格式傳送至 EventBridge。事件中的下列欄位會形成規則中定義的事件模式，以及觸發動作的事件模式：

```
"source": "aws.ec2spotfleet"
```

識別事件來自 Spot Fleet。

```
"detail-type": "EC2 Spot Fleet State Change"
```

識別事件類型。

```
"detail": { "sub-type": "submitted" }
```

識別事件子類型。

### 事件類型

- [EC2 Spot Fleet 狀態變更](#)
- [EC2 Spot Fleet Spot 執行個體請求變更](#)
- [EC2 Spot Fleet 執行個體變更](#)
- [EC2 Spot Fleet 資訊](#)
- [EC2 Spot Fleet 錯誤](#)

## EC2 Spot Fleet 狀態變更

Spot 叢集會在 Spot 叢集變更狀態 EventBridge 時向 Amazon 傳送 EC2 Spot Fleet State Change 事件。

以下是此事件的範例資料。

```
{
 "version": "0",
 "id": "d1af1091-6cc3-2e24-203a-3b870e455d5b",
 "detail-type": "EC2 Spot Fleet State Change",
 "source": "aws.ec2spotfleet",
 "account": "123456789012",
 "time": "2020-11-09T08:57:06Z",
 "region": "us-east-1",
 "resources": [
 "arn:aws:ec2:us-east-1:123456789012:spot-fleet-request/sfr-4b6d274d-0cea-4b2c-b3be-9dc627ad1f55"
],
 "detail": {
 "sub-type": "submitted"
 }
}
```

sub-type 的可能值為：

**active**

已驗證 Spot Fleet 請求，且 Amazon EC2 正在嘗試維持執行中的執行個體目標數目。

**cancelled**

Spot Fleet 請求已取消，且無執行中的執行個體。Spot Fleet 將會在其執行個體終止後兩天刪除。

**cancelled\_running**

Spot Fleet 請求已取消，不會啟動其他執行個體。其現有執行個體將繼續執行直到中斷或終止。此請求會一直維持在該狀態，直到所有執行個體中斷或終止。

**cancelled\_terminating**

Spot Fleet 請求已取消，其執行個體正在終止。此請求會一直維持在該狀態，直到所有執行個體終止。



## expired

Spot Fleet 請求已過期。如果請求是使用 `TerminateInstancesWithExpiration` 集建立的，後續 `terminated` 事件會指出已終止執行個體。

## modify\_in\_progress

正在修改 Spot Fleet 請求。請求會一直維持在此狀態，直到已完成修改。

## modify\_succeeded

已修改 Spot Fleet 請求。

## submitted

正在評估 Spot Fleet 請求且 Amazon EC2 正準備啟動執行個體的目標數量。

## progress

正在履行 Spot Fleet 請求。

## EC2 Spot Fleet Spot 執行個體請求變更

當叢集中的 Spot 執 EC2 Spot Fleet Spot Instance Request Change 行個體請求變更狀態 EventBridge 時，Spot 叢集會傳送事件至 Amazon。

以下是此事件的範例資料。

```
{
 "version": "0",
 "id": "cd141ef0-14af-d670-a71d-fe46e9971bd2",
 "detail-type": "EC2 Spot Fleet Spot Instance Request Change",
 "source": "aws.ec2spotfleet",
 "account": "123456789012",
 "time": "2020-11-09T08:53:21Z",
 "region": "us-east-1",
 "resources": [
 "arn:aws:ec2:us-east-1:123456789012:spot-fleet-request/sfr-
a98d2133-941a-47dc-8b03-0f94c6852ad1"
],
 "detail": {
 "spot-instance-request-id": "sir-a2w9gc5h",
 "description": "SpotInstanceRequestId sir-a2w9gc5h, PreviousState:
cancelled_running",
 "sub-type": "cancelled"
 }
}
```

```
}
}
```

sub-type 的可能值為：

active

Spot 執行個體請求已履行，並具有關聯的 Spot 執行個體。

cancelled

您已取消 Spot 執行個體請求，或 Spot 執行個體請求已過期。

disabled

您已停用 Spot 執行個體。

submitted

已提交 Spot 執行個體請求。

## EC2 Spot Fleet 執行個體變更

當叢集中的執行 EC2 Spot Fleet Instance Change 個體狀態變更 EventBridge 時，Spot 叢集會傳送事件至 Amazon。

以下是此事件的範例資料。

```
{
 "version": "0",
 "id": "11591686-5bd7-bbaa-eb40-d46529c2710f",
 "detail-type": "EC2 Spot Fleet Instance Change",
 "source": "aws.ec2spotfleet",
 "account": "123456789012",
 "time": "2020-11-09T07:25:02Z",
 "region": "us-east-1",
 "resources": [
 "arn:aws:ec2:us-east-1:123456789012:spot-fleet-request/sfr-c8a764a4-bedc-4b62-af9c-0095e6e3ba61"
],
 "detail": {
 "instance-id": "i-08b90df1e09c30c9b",
 "description": "{\"instanceType\":\"r4.2xlarge\",\"image\":
\"ami-032930428bf1abbff\",\"productDescription\":\"Linux/UNIX\",\"availabilityZone\":
\"us-east-1a\"}",
```

```
 "sub-type": "launched"
 }
}
```

sub-type 的可能值為：

launched

已啟動新執行個體。

terminated

已終止執行個體。

termination\_notified

Amazon EC2 在縮減規模期間終止 Spot 執行個體，也就是機群的目標容量向下修改 (例如，從目標容量 4 變為 3) 時，系統會傳送執行個體終止通知。

## EC2 Spot Fleet 資訊

在履行期間發生錯誤 EventBridge 時，競價型叢集會傳送 EC2 Spot Fleet Information 事件至 Amazon。此資訊事件不會封鎖機群嘗試履行其目標容量。

以下是此事件的範例資料。

```
{
 "version": "0",
 "id": "73a60f70-3409-a66c-635c-7f66c5f5b669",
 "detail-type": "EC2 Spot Fleet Information",
 "source": "aws.ec2spotfleet",
 "account": "123456789012",
 "time": "2020-11-08T20:56:12Z",
 "region": "us-east-1",
 "resources": [
 "arn:aws:ec2:us-east-1:123456789012:spot-fleet-request/sfr-2531ea06-af18-4647-8757-7d69c94971b1"
],
 "detail": {
 "description": "r3.8xlarge, ami-032930428bf1abbff, Linux/UNIX, us-east-1a, Spot bid price is less than Spot market price $0.5291",
 "sub-type": "launchSpecUnusable"
 }
}
```

```
}
```

sub-type 的可能值為：

`fleetProgressHalted`

每個啟動規格中的價格無效，因為價格低於 Spot 價格 (所有啟動規格都已產生 `launchSpecUnusable` 事件)。如果 Spot 價格變動，啟動規格則可能會變成有效。

`launchSpecTemporarilyBlacklisted`

組態無效，且多次嘗試啟動執行個體失敗。如需詳細資訊，請參閱事件的描述。

`launchSpecUnusable`

啟動規格中的價格無效，因為價格低於 Spot 價格。

`registerWithLoadBalancersFailed`

嘗試在負載平衡器失敗時註冊執行個體。如需詳細資訊，請參閱事件的描述。

## EC2 Spot Fleet 錯誤

在履行期間發生錯誤 EventBridge 時，競價型叢集會傳送 EC2 Spot Fleet Error 事件至 Amazon。錯誤事件會阻止機群嘗試履行其目標容量。

以下是此事件的範例資料。

```
{
 "version": "0",
 "id": "10adc4e7-675c-643e-125c-5bfa1b1ba5d2",
 "detail-type": "EC2 Spot Fleet Error",
 "source": "aws.ec2spotfleet",
 "account": "123456789012",
 "time": "2020-11-09T06:56:07Z",
 "region": "us-east-1",
 "resources": [
 "arn:aws:ec2:us-east-1:123456789012:spot-fleet-request/
sfr-38725d30-25f1-4f30-83ce-2907c56dba17"
],
 "detail": {
 "description": "r4.2xlarge, ami-032930428bf1abbff, Linux/UNIX: The
associatePublicIPAddress parameter can only be specified for the network interface
with DeviceIndex 0. ",
```

```
 "sub-type": "spotFleetRequestConfigurationInvalid"
 }
}
```

sub-type 的可能值為：

iamFleetRoleInvalid

Spot Fleet 沒有啟動或終止執行個體所需的許可。

allLaunchSpecsTemporarilyBlacklisted

沒有任何組態是有效的，且多次嘗試啟動執行個體失敗。如需詳細資訊，請參閱事件的描述。

spotInstanceCountLimitExceeded

您已達到可啟動 Spot 執行個體的數目上限。

spotFleetRequestConfigurationInvalid

組態無效。如需詳細資訊，請參閱事件的描述。

## 創建 Amazon EventBridge 規則

當針對 EC2 叢集或競價型叢集發出狀態變更通知時，通知的事件會傳送至 Amazon EventBridge。如果 EventBridge 偵測到符合規則中定義的模式的事件模式，則會 EventBridge 叫用規則中指定的一個或多個目標。

您可以撰寫 EventBridge 規則，並在事件模式符合規則時自動執行哪些動作。

主題

- [建立 Amazon EventBridge 規則以監控 EC2 叢集事件](#)
- [建立 Amazon EventBridge 規則以監控競價型叢集事件](#)

### 建立 Amazon EventBridge 規則以監控 EC2 叢集事件

當 EC2 叢集發出狀態變更通知時，通知的事件會以 JSON 檔案 EventBridge 的形式傳送至 Amazon。您可以撰寫 EventBridge 規則來自動化事件模式與規則相符時要採取的動作。如果 EventBridge 偵測到符合規則中定義的模式的事件模式，則會 EventBridge 叫用規則中指定的目標 (或多個目標)。

下列欄位會形成規則中定義的事件模式：

```
"source": "aws.ec2fleet"
```

識別事件來自 EC2 Fleet。

```
"detail-type": "EC2 Fleet State Change"
```

識別事件類型。

```
"detail": { "sub-type": "submitted" }
```

識別事件子類型。

如需 EC2 Fleet 事件和範例事件資料的清單，請參閱 [the section called “EC2 Fleet 事件類型”](#)。

## 範例

- [建立 EventBridge 規則以傳送通知](#)
- [建立 EventBridge 規則以觸發 Lambda 函數](#)

### 建立 EventBridge 規則以傳送通知

下列範例會建立 EventBridge 規則，以便在每次 Amazon EC2 發出 EC2 叢集狀態變更通知時傳送電子郵件、文字訊息或行動推播通知。將發出此範例中的訊號作為 EC2 Fleet State Change 事件，這會觸發規則定義的動作。

在建立 EventBridge 規則之前，您必須為電子郵件、文字訊息或行動推播通知建立 Amazon SNS 主題。

### 建立 EventBridge 規則以在 EC2 叢集狀態變更時傳送通知

1. 在 <https://console.aws.amazon.com/events/> 打開 Amazon EventBridge 控制台。
2. 選擇建立規則。
3. 針對 Define rule detail (定義規則詳細資訊) 執行下列動作：
  - a. 輸入規則的 Name (名稱)，或者輸入描述。

在同一個區域和同一個事件匯流排上，規則不能與另一個規則同名。

- b. 針對 Event bus (事件匯流排) 選擇 default (預設值)。當您帳戶中的 AWS 服務產生事件時，一律會前往您帳戶的預設事件匯流排。
- c. 針對 Rule type (規則類型) 選擇 Rule with an event pattern (具有事件模式的規則)。

- d. 選擇下一步。
4. 針對 Build event pattern (建置事件模式) 執行下列動作：
    - a. 對於事件來源，請選擇AWS 事件或 EventBridge合作夥伴事件。
    - b. 針對此範例的 Event pattern (事件模式)，您需指定下列事件模式，以便與 EC2 Fleet Instance Change 事件相符。

```
{
 "source": ["aws.ec2fleet"],
 "detail-type": ["EC2 Fleet Instance Change"]
}
```

若要新增事件模式，您可以透過選擇 Event pattern form (事件模式表單) 使用範本，或者透過選擇 Custom pattern (JSON editor) (自定義模式 (JSON 編輯器)) 指定自有模式，如下：

- i. 若要使用範本建立事件模式，請執行下列動作：
    - A. 選擇 Event pattern form (事件模式表單)。
    - B. 在 Event source (事件來源) 欄位中，選擇 AWS services (服務)。
    - C. 針對 AWS Service (服務)，請選擇 EC2 Fleet (EC2 機群)。
    - D. 針對 Event type (事件類型)，選擇 EC2 Fleet Instance Change (EC2 機群執行個體變更)。
    - E. 若要自訂範本，請選擇 Edit pattern (編輯模式) 並進行變更以與範例事件模式相符。
  - ii. (替代) 若要指定自訂事件模式，請執行下列動作：
    - A. 選擇 Custom pattern (JSON editor) (自訂模式 (JSON 編輯器))。
    - B. 在 Event pattern (事件模式) 方塊中，為此範例新增事件模式。
- c. 選擇下一步。
5. 針對 Select target(s) (選取目標) 執行下列動作：
    - a. 在 Target types (目標類型) 欄位中，選擇 AWS service (服務)。
    - b. 針對 Select a target (選取目標) 選取 SNS topic (SNS 主題)，以便在事件發生時傳送電子郵件、簡訊或行動裝置推送通知。
    - c. 針對 Topic (主題)，請選擇現有的主題。您首先需要使用 Amazon SNS 主控台建立 Amazon SNS 主題。如需詳細資訊，請參閱 [Amazon 簡單通知服務開發人員指南中的使用 Amazon SNS 進行 application-to-person \(A2P\) 簡訊](#)。

- d. (選用) 在 Additional settings (其他設定) 下，您可以選擇性地設定其他設定。如需詳細資訊，請參閱 [Amazon EventBridge 使用者指南中的建立對事件做出反應的 Amazon EventBridge 規則](#) (步驟 16)。
  - e. 選擇下一步。
6. (選用) 針對 Tags (標籤)，您可以選擇性地將一或多個標籤指派給您的規則，然後選擇 Next (下一步)。
  7. 針對 Review and create (檢閱和建立) 執行下列動作：
    - a. 檢閱規則的詳細資訊，然後視需求進行修改。
    - b. 選擇建立規則。

有關更多信息，請參閱 [Amazon EventBridge 用戶指南中的 Amazon EventBridge 規則和 Amazon EventBridge 事件模式](#)

### 建立 EventBridge 規則以觸發 Lambda 函數

下列範例會建立 EventBridge 規則，以便在每次 Amazon EC2 在執行個體啟動時發出 EC2 叢集執行個體變更通知時觸發 Lambda 函數。將發出此範例中的訊號作為 EC2 Fleet Instance Change 事件 (子類型 launched)，這會觸發規則定義的動作。

在建立 EventBridge 規則之前，您必須先建立 Lambda 函數。

若要建立要在 EventBridge 規則中使用的 Lambda 函數

1. 開啟主 AWS Lambda 控制台，網址為 <https://console.aws.amazon.com/lambda/>。
2. 選擇 Create function (建立函式)。
3. 輸入函數的名稱、設定程式碼，然後選擇 Create function (建立函數)。

如需有關使用 Lambda 的詳細資訊，請參閱 AWS Lambda 開發人員指南中的 [使用主控台建立 Lambda 函數](#)。

若要建立 EventBridge 規則以在 EC2 叢集中的執行個體變更狀態時觸發 Lambda 函數

1. 在 <https://console.aws.amazon.com/events/> 打開 Amazon EventBridge 控制台。
2. 選擇建立規則。
3. 針對 Define rule detail (定義規則詳細資訊) 執行下列動作：



- a. 輸入規則的Name (名稱), 或者輸入描述。

在同一個區域和同一個事件匯流排上, 規則不能與另一個規則同名。
  - b. 針對 Event bus (事件匯流排) 選擇 default (預設值)。當您帳戶中的 AWS 服務產生事件時, 一律會前往您帳戶的預設事件匯流排。
  - c. 針對 Rule type (規則類型) 選擇 Rule with an event pattern (具有事件模式的規則)。
  - d. 選擇下一步。
4. 針對 Build event pattern (建置事件模式) 執行下列動作:
    - a. 對於事件來源, 請選擇AWS 事件或 EventBridge合作夥伴事件。
    - b. 針對此範例的 Event pattern (事件模式), 您需指定下列事件模式, 以便與 EC2 Fleet Instance Change 事件和 launched 子類型相符。

```
{
 "source": ["aws.ec2fleet"],
 "detail-type": ["EC2 Fleet Instance Change"],
 "detail": {
 "sub-type": ["launched"]
 }
}
```

若要新增事件模式, 您可以透過選擇 Event pattern form (事件模式表單) 使用範本, 或者透過選擇 Custom pattern (JSON editor) (自定義模式 (JSON 編輯器)) 指定自有模式, 如下:

- i. 若要使用範本建立事件模式, 請執行下列動作:
  - A. 選擇 Event pattern form (事件模式表單)。
  - B. 在 Event source (事件來源) 欄位中, 選擇 AWS services (服務)。
  - C. 針對 AWS Service (服務), 請選擇 EC2 Fleet (EC2 機群)。
  - D. 針對 Event type (事件類型), 選擇 EC2 Fleet Instance Change (EC2 機群執行個體變更)。
  - E. 選擇 Edit pattern (編輯模式), 並新增 "detail": {"sub-type": ["launched"]} 以符合範例事件模式。若要獲得正確的 JSON 格式, 請在前方括號 (]) 之後插入一個逗號 (,)。
- ii. (替代) 若要指定自訂事件模式, 請執行下列動作:
  - A. 選擇 Custom pattern (JSON editor) (自訂模式 (JSON 編輯器))。

- B. 在 Event pattern (事件模式) 方塊中，為此範例新增事件模式。
  - c. 選擇下一步。
5. 針對 Select target(s) (選取目標) 執行下列動作：
  - a. 在 Target types (目標類型) 欄位中，選擇 AWS service (服務)。
  - b. 針對 Select a target (選取目標) 選取 SNS topic (SNS 主題)，以便在事件發生時傳送電子郵件、簡訊或行動裝置推送通知。
  - c. 針對 Topic (主題)，請選擇 Lambda function (Lambda 函數)，然後在 Function (函數) 中選擇您建立的函數，以便在事件發生時回應。
  - d. (選用) 在 Additional settings (其他設定) 下，您可以選擇性地設定其他設定。如需詳細資訊，請參閱 [Amazon EventBridge 使用者指南中的建立對事件做出反應的 Amazon EventBridge 規則](#) (步驟 16)。
  - e. 選擇下一步。
6. (選用) 針對 Tags (標籤)，您可以選擇性地將一或多個標籤指派給您的規則，然後選擇 Next (下一步)。
7. 針對 Review and create (檢閱和建立) 執行下列動作：
  - a. 檢閱規則的詳細資訊，然後視需求進行修改。
  - b. 選擇建立規則。

如需如何建立 Lambda 函數和執行 Lambda 函數的 EventBridge 規則的教學課程，請參閱 AWS Lambda 開發人員指南 EventBridge 中的 [教學課程：使用記錄 Amazon EC2 執行個體的狀態](#)。

## 建立 Amazon EventBridge 規則以監控競價型叢集事件

當針對競價型叢集發出狀態變更通知時，通知的事件會以 JSON 檔案 EventBridge 的形式傳送至 Amazon。您可以撰寫 EventBridge 規則來自動化事件模式與規則相符時要採取的動作。如果 EventBridge 偵測到符合規則中定義的模式的事件模式，則會 EventBridge 叫用規則中指定的目標 (或多個目標)。

下列欄位會形成規則中定義的事件模式：

```
"source": "aws.ec2spotfleet"
```

識別事件來自 Spot Fleet。

```
"detail-type": "EC2 Spot Fleet State Change"
```

識別事件類型。

```
"detail": { "sub-type": "submitted" }
```

識別事件子類型。

如需 Spot Fleet 事件和範例事件資料的清單，請參閱 [the section called “Spot Fleet 事件類型”](#)。

## 範例

- [建立 EventBridge 規則以傳送通知](#)
- [建立 EventBridge 規則以觸發 Lambda 函數](#)

### 建立 EventBridge 規則以傳送通知

下列範例會建立 EventBridge 規則，以便在每次 Amazon EC2 發出 Spot 叢集狀態變更通知時傳送電子郵件、文字訊息或行動推播通知。將發出此範例中的訊號作為 EC2 Spot Fleet State Change 事件，這會觸發規則定義的動作。在建立 EventBridge 規則之前，您必須為電子郵件、文字訊息或行動推播通知建立 Amazon SNS 主題。

若要建立 EventBridge 規則，以便在 Spot 叢集狀態變更時傳送通知

1. 在 <https://console.aws.amazon.com/events/> 打開 Amazon EventBridge 控制台。
2. 選擇建立規則。
3. 針對 Define rule detail (定義規則詳細資訊) 執行下列動作：
  - a. 輸入規則的 Name (名稱)，或者輸入描述。

在同一個區域和同一個事件匯流排上，規則不能與另一個規則同名。
  - b. 針對 Event bus (事件匯流排) 選擇 default (預設值)。當您帳戶中的 AWS 服務產生事件時，一律會前往您帳戶的預設事件匯流排。
  - c. 針對 Rule type (規則類型) 選擇 Rule with an event pattern (具有事件模式的規則)。
  - d. 選擇下一步。
4. 針對 Build event pattern (建置事件模式) 執行下列動作：
  - a. 對於事件來源，請選擇 AWS 事件或 EventBridge 合作夥伴事件。

- b. 針對此範例的 Event pattern (事件模式)，您需指定下列事件模式，以便與 EC2 Spot Fleet Instance Change 事件相符。

```
{
 "source": ["aws.ec2spotfleet"],
 "detail-type": ["EC2 Spot Fleet Instance Change"]
}
```

若要新增事件模式，您可以透過選擇 Event pattern form (事件模式表單) 使用範本，或者透過選擇 Custom pattern (JSON editor) (自定義模式 (JSON 編輯器)) 指定自有模式，如下：

- i. 若要使用範本建立事件模式，請執行下列動作：
    - A. 選擇 Event pattern form (事件模式表單)。
    - B. 在 Event source (事件來源) 欄位中，選擇 AWS services (服務)。
    - C. 針對 AWS Service (服務)，請選擇 EC2 Spot Fleet (EC2 Spot 機群)。
    - D. 針對 Event type (事件類型)，選擇 EC2 Spot Fleet Instance Change (EC2 Spot 機群執行個體變更)。
    - E. 若要自訂範本，請選擇 Edit pattern (編輯模式) 並進行變更以與範例事件模式相符。
  - ii. (替代) 若要指定自訂事件模式，請執行下列動作：
    - A. 選擇 Custom pattern (JSON editor) (自訂模式 (JSON 編輯器))。
    - B. 在 Event pattern (事件模式) 方塊中，為此範例新增事件模式。
- c. 選擇下一步。

5. 針對 Select target(s) (選取目標) 執行下列動作：

- a. 在 Target types (目標類型) 欄位中，選擇 AWS service (服務)。
- b. 針對 Select a target (選取目標) 選取 SNS topic (SNS 主題)，以便在事件發生時傳送電子郵件、簡訊或行動裝置推送通知。
- c. 針對 Topic (主題)，請選擇現有的主題。您首先需要使用 Amazon SNS 主控台建立 Amazon SNS 主題。如需詳細資訊，請參閱 [Amazon 簡單通知服務開發人員指南中的使用 Amazon SNS 進行 application-to-person \(A2P\) 簡訊](#)。
- d. (選用) 在 Additional settings (其他設定) 下，您可以選擇性地設定其他設定。如需詳細資訊，請參閱 [Amazon EventBridge 使用者指南中的建立對事件做出反應的 Amazon EventBridge 規則](#) (步驟 16)。

- e. 選擇下一步。

6. (選用) 針對 Tags (標籤)，您可以選擇性地將一或多個標籤指派給您的規則，然後選擇 Next (下一步)。
7. 針對 Review and create (檢閱和建立) 執行下列動作：
  - a. 檢閱規則的詳細資訊，然後視需求進行修改。
  - b. 選擇建立規則。

有關更多信息，請參閱 [Amazon EventBridge 用戶指南中的 Amazon EventBridge 規則和 Amazon EventBridge 事件模式](#)

### 建立 EventBridge 規則以觸發 Lambda 函數

下列範例會建立 EventBridge 規則，以便在每次 Amazon EC2 在執行個體啟動時發出競價型叢集執行個體變更通知時觸發 Lambda 函數。將發出此範例中的訊號作為 EC2 Spot Fleet Instance Change 事件 (子類型 launched)，這會觸發規則定義的動作。

在建立 EventBridge 規則之前，您必須先建立 Lambda 函數。

若要建立要在 EventBridge 規則中使用的 Lambda 函數

1. 開啟主 AWS Lambda 控制台，網址為 <https://console.aws.amazon.com/lambda/>。
2. 選擇 Create function (建立函式)。
3. 輸入函數的名稱、設定程式碼，然後選擇 Create function (建立函數)。

如需有關使用 Lambda 的詳細資訊，請參閱 AWS Lambda 開發人員指南中的 [使用主控台建立 Lambda 函數](#)。

若要建立 EventBridge 規則以在競價型叢集中的執行個體變更狀態時觸發 Lambda 函數

1. 在 <https://console.aws.amazon.com/events/> 打開 Amazon EventBridge 控制台。
2. 選擇建立規則。
3. 針對 Define rule detail (定義規則詳細資訊) 執行下列動作：
  - a. 輸入規則的 Name (名稱)，或者輸入描述。

在同一個區域和同一個事件匯流排上，規則不能與另一個規則同名。
  - b. 針對 Event bus (事件匯流排) 選擇 default (預設值)。當您帳戶中的 AWS 服務產生事件時，一律會前往您帳戶的預設事件匯流排。

- c. 針對 Rule type (規則類型) 選擇 Rule with an event pattern (具有事件模式的規則)。
  - d. 選擇下一步。
4. 針對 Build event pattern (建置事件模式) 執行下列動作：
    - a. 對於事件來源，請選擇AWS 事件或 EventBridge合作夥伴事件。
    - b. 針對此範例的 Event pattern (事件模式)，您需指定下列事件模式，以便與 EC2 Spot Fleet Instance Change 事件和 launched 子類型相符。

```
{
 "source": ["aws.ec2spotfleet"],
 "detail-type": ["EC2 Spot Fleet Instance Change"],
 "detail": {
 "sub-type": ["launched"]
 }
}
```

若要新增事件模式，您可以透過選擇 Event pattern form (事件模式表單) 使用範本，或者透過選擇 Custom pattern (JSON editor) (自定義模式 (JSON 編輯器)) 指定自有模式，如下：

- i. 若要使用範本建立事件模式，請執行下列動作：
    - A. 選擇 Event pattern form (事件模式表單)。
    - B. 在 Event source (事件來源) 欄位中，選擇 AWS services (服務)。
    - C. 針對 AWS Service (服務)，請選擇 EC2 Spot Fleet (EC2 Spot 機群)。
    - D. 針對 Event type (事件類型)，選擇 EC2 Spot Fleet Instance Change (EC2 Spot 機群執行個體變更)。
    - E. 選擇 Edit pattern (編輯模式)，並新增 "detail": {"sub-type": ["launched"]} 以符合範例事件模式。若要獲得正確的 JSON 格式，請在前方括號 (]) 之後插入一個逗號 (,)。
  - ii. (替代) 若要指定自訂事件模式，請執行下列動作：
    - A. 選擇 Custom pattern (JSON editor) (自訂模式 (JSON 編輯器))。
    - B. 在 Event pattern (事件模式) 方塊中，為此範例新增事件模式。
  - c. 選擇下一步。
5. 針對 Select target(s) (選取目標) 執行下列動作：
    - a. 在 Target types (目標類型) 欄位中，選擇 AWS service (服務)。

- b. 針對 Select a target (選取目標) 選取 SNS topic (SNS 主題)，以便在事件發生時傳送電子郵件、簡訊或行動裝置推送通知。
  - c. 針對 Topic (主題)，請選擇 Lambda function (Lambda 函數)，然後在 Function (函數) 中選擇您建立的函數，以便在事件發生時回應。
  - d. (選用) 在 Additional settings (其他設定) 下，您可以選擇性地設定其他設定。如需詳細資訊，請參閱 [Amazon EventBridge 使用者指南中的建立對事件做出反應的 Amazon EventBridge 規則](#) (步驟 16)。
  - e. 選擇下一步。
6. (選用) 針對 Tags (標籤)，您可以選擇性地將一或多個標籤指派給您的規則，然後選擇 Next (下一步)。
  7. 針對 Review and create (檢閱和建立) 執行下列動作：
    - a. 檢閱規則的詳細資訊，然後視需求進行修改。
    - b. 選擇建立規則。

如需如何建立 Lambda 函數和執行 Lambda 函數的 EventBridge 規則的教學課程，請參閱AWS Lambda 開發人員指南 EventBridge中的[教學課程：使用記錄 Amazon EC2 執行個體的狀態](#)。

## EC2 Fleet 和 Spot Fleet 的教學課程

下列教學課程將帶您完成建立 EC2 Fleet 和 Spot Fleet 的常用程序。

### 教學課程

- [教學：使用 EC2 Fleet 與執行個體權重](#)
- [教學：使用 EC2 Fleet 與隨需做為主要容量](#)
- [教學課程：使用目標容量預留啟動隨需執行個體](#)
- [教學課程：在容量區塊中啟動執行個體](#)
- [教學：使用 Spot Fleet 與執行個體權重](#)

### 教學：使用 EC2 Fleet 與執行個體權重

本教學課程使用一間稱為 Example Corp 的虛構公司，說明使用執行個體權重來請求 EC2 Fleet 的程序。

## 目標

Example Corp 是一間製藥公司，希望利用 Amazon EC2 的計算能力篩選可能用於抗癌的化合物。

## 規劃

Example Corp 首先檢視了 [Spot 最佳實務](#)。接下來，Example Corp 決定他們的 EC2 Fleet 需求。

### 執行個體類型

Example Corp 擁有運算和記憶體密集型應用程式，該應用程式在至少 60 GB 記憶體和八個虛擬 CPU (vCPU) 情況下效能最佳。他們希望以最低的價格為應用程式最大限度利用這些資源。Example Corp 認為以下任一個 EC2 執行個體類型都能滿足他們的需求：

執行個體類型	記憶體 (GiB)	vCPU
r3.2xlarge	61	8
r3.4xlarge	122	16
r3.8xlarge	244	32

### 目標容量 (單位)

使用執行個體加權時，目標容量可以等於執行個體數目 (預設值) 或核心 (vCPUs)、記憶體 () 和儲存體 (GBGiBs) 等因素的組合。Example Corp 將其應用程式基礎 (60 GB RAM 和 8 個 vCPU) 做為一個單位考慮，Example Corp 決定此數量的 20 倍將可滿足他們的需求。因此，該公司將其 EC2 Fleet 請求的目標容量設定為 20。

### 執行個體權重

Example Corp 決定目標容量後，即計算執行個體權重。為了計算每個執行個體類型的執行個體權重，將決定達成目標容量所需的每個執行個體類型單位，如下所示：

- r3.2xlarge (61.0 GB, 8 vCPUs) = 1 單位 (共 20 單位)
- r3.4xlarge (122.0 GB, 16 vCPUs) = 2 單位 (共 20 單位)
- r3.8xlarge (244.0 GB, 32 vCPUs) = 4 單位 (共 20 單位)



因此，Example Corp 會針對其 EC2 Fleet 請求中的各個啟動組態，指派 1、2 和 4 的執行個體權重。

### 每個單位小時的價格

Example Corp 會使用每執行個體小時的[隨需價格](#)，來做為其價格的起點。他們也可以使用最近的 Spot 價格，或兩者的組合。為計算每單位小時價格，他們將每個執行個體小時的起始價格除以權重。例如：

執行個體類型	隨需價格	執行個體權重	每個單位小時的價格
r3.2xLarge	0.7 USD	1	0.7 USD
r3.4xLarge	1.4 USD	2	0.7 USD
r3.8xLarge	\$2.8	4	0.7 USD

Example Corp 可以使用每單位小時 0.7 USD 的全球價格，並且對於所有三種執行個體類型都具有競爭力。在 r3.8xlarge 啟動規格中，他們也可以使用每單位小時 0.7 USD 全局價格和 0.9 USD 每單位小時特定價格。

### 驗證許可

Example Corp 在建立 EC2 Fleet 之前，驗證其具有所需許可的 IAM 角色。如需詳細資訊，請參閱[EC2 Fleet 先決條件](#)。

### 建立啟動範本

接下來，範例公司會建立一個啟動範本。啟動範本 ID 會在下列步驟中使用。如需詳細資訊，請參閱[建立啟動範本](#)。

### 建立 EC2 Fleet

Example Corp 會建立一個檔案 config.json，且其 EC2 Fleet 組態如下。在下列範例中，以您自己的資源識別符取代資源識別符。

```
{
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
```

```
 "LaunchTemplateId": "lt-07b3bc7625cdab851",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceType": "r3.2xlarge",
 "SubnetId": "subnet-482e4972",
 "WeightedCapacity": 1
 },
 {
 "InstanceType": "r3.4xlarge",
 "SubnetId": "subnet-482e4972",
 "WeightedCapacity": 2
 },
 {
 "InstanceType": "r3.8xlarge",
 "MaxPrice": "0.90",
 "SubnetId": "subnet-482e4972",
 "WeightedCapacity": 4
 }
]
}
],
"TargetCapacitySpecification": {
 "TotalTargetCapacity": 20,
 "DefaultTargetCapacityType": "spot"
}
}
```

Example Corp 使用以下 [create-fleet](#) 命令來建立 EC2 Fleet。

```
aws ec2 create-fleet \
 --cli-input-json file://config.json
```

如需詳細資訊，請參閱 [建立 EC2 Fleet](#)。

## 實現

分配策略決定您的 競價型執行個體 來自哪個 Spot 容量集區。

採用 lowest-price 策略 (預設的策略)，競價型執行個體 來自實現時每單位價格最低的集區。為提供 20 個單位的容量，EC2 Fleet 將啟動 20 個 r3.2xlarge 執行個體 (20 除以 1)、10 個 r3.4xlarge 執行個體 (20 除以 2) 或 5 個 r3.8xlarge 執行個體 (20 除以 4)。

如果 Example Corp 使用 diversified 策略，競價型執行個體 將來自所有三個集區。EC2 Fleet 將啟動 6 個 r3.2xlarge 執行個體 (提供 6 個單位)、3 個 r3.4xlarge 執行個體 (提供 6 個單位) 和 2 個 r3.8xlarge 執行個體 (提供 8 個單位)，共計 20 個單位。

## 教學：使用 EC2 Fleet 與隨需做為主要容量

本教學課程使用一間稱為 ABC Online 的虛構公司來說明請求 EC2 Fleet 的隨需做為主要容量的程序，以及 Spot 容量 (如果可用)。

### 目標

餐飲外送公司 ABC Online 希望能夠跨 EC2 執行個體類型佈建 Amazon EC2 容量和購買選項，以實現該公司所要的規模、效能和成本。

### 計畫

ABC Online 需要固定容量，以在尖峰期運作，但希望以更低的價格提高容量。ABC Online 決定下列的 EC2 Fleet 需求：

- 隨需執行個體容量 – ABC Online 需要 15 個隨需執行個體來確保其可以在尖峰期間容納流量。
- Spot 執行個體容量 – ABC Online 希望透過佈建 5 個 Spot 執行個體以更低的價格提高效能。

### 驗證許可

ABC Online 在建立 EC2 Fleet 之前，驗證其具有所需許可的 IAM 角色。如需詳細資訊，請參閱 [EC2 Fleet 先決條件](#)。

### 建立啟動範本

接下來，ABC Online 會建立啟動範本。啟動範本 ID 會在下列步驟中使用。如需詳細資訊，請參閱 [建立啟動範本](#)。

### 建立 EC2 Fleet

ABC Online 會建立一個檔案 config.json，且其 EC2 Fleet 組態如下。在下列範例中，以您自己的資源識別符取代資源識別符。

```
{
 "LaunchTemplateConfigs": [

```

```
{
 "LaunchTemplateSpecification": {
 "LaunchTemplateId": "lt-07b3bc7625cdab851",
 "Version": "2"
 }
},
"TargetCapacitySpecification": {
 "TotalTargetCapacity": 20,
 "OnDemandTargetCapacity": 15,
 "DefaultTargetCapacityType": "spot"
}
}
```

ABC Online 使用以下 [create-fleet](#) 命令來建立 EC2 Fleet。

```
aws ec2 create-fleet \
 --cli-input-json file://config.json
```

如需詳細資訊，請參閱 [建立 EC2 Fleet](#)。

## 實現

分配策略決定隨需容量能始終實現，而如果存在容量和可用性，則目標容量的剩餘部分將做為 Spot 實現。

## 教學課程：使用目標容量預留啟動隨需執行個體

本教學課程將逐步引導您完成必須執行的所有步驟，以便您的 EC2 Fleet 啟動隨需執行個體至 targeted 容量預留。

您將會了解到如何設定機群，以在啟動隨需執行個體時先使用 targeted 隨需容量預留。您也會了解到如何設定機群，以便機群在隨需目標容量總計超過可用的未使用容量預留數目時，使用指定的分配策略來選取要在其中啟動剩餘目標容量的執行個體集區。

### EC2 Fleet 組態

本教學課程使用的機群組態如下：

- 目標容量：10 個隨需執行個體

- 總計未使用的 targeted 容量預留：6 (小於機群的隨需目標容量：10 個隨需執行個體)
- 容量預留集區數目：2 (us-east-1a 和 us-east-1b)
- 每個集區的容量預留數目：3
- 隨需分配策略：lowest-price (當未使用的容量預留數目小於隨需目標容量時，機群會根據隨需分配策略決定要用來啟動剩餘隨需容量的集區。)

請注意，您也可以使用 prioritized 分配策略，而不使用 lowest-price 分配策略。

若要按照 targeted 容量預留來啟動隨需執行個體，必須執行以下幾個步驟：

- [步驟 1：建立容量預留](#)
- [步驟 2：建立容量預留資源群組](#)
- [步驟 3：將容量預留新增至容量預留資源群組](#)
- (選用) [步驟 4：檢視資源群組中的容量預留](#)
- [步驟 5：建立啟動範本，此範本指定容量預留以特定資源群組為目標](#)
- (選用) [步驟 6：描述啟動範本](#)
- [步驟 7：建立 EC2 Fleet](#)
- (選用) [步驟 8：檢視剩餘未使用的容量預留數目](#)

## 步驟 1：建立容量預留

使用 [create-capacity-reservation](#) 命令來建立容量預留，其中三個用於 us-east-1a，另外三個用於 us-east-1b。除了可用區域之外，容量預留的其他屬性皆相同。

在 **us-east-1a** 中的 3 個容量預留

```
aws ec2 create-capacity-reservation \
 --availability-zone us-east-1a\
 --instance-type c5.xlarge\
 --instance-platform Linux/UNIX \
 --instance-count 3 \
 --instance-match-criteria targeted
```

最後產生的容量預留 ID 範例

```
cr-1234567890abcdef1
```

## 在 **us-east-1b** 中的 3 個容量預留

```
aws ec2 create-capacity-reservation \
 --availability-zone us-east-1b\
 --instance-type c5.xlarge\
 --instance-platform Linux/UNIX \
 --instance-count 3 \
 --instance-match-criteria targeted
```

### 最後產生的容量預留 ID 範例

```
cr-54321abcdef567890
```

## 步驟 2：建立容量預留資源群組

使用 `resource-groups` 服務和 [create-group](#) 命令來建立容量預留資源群組。在此範例中，資源群組會命名為 `my-cr-group`。如需為何必須建立資源群組的相關資訊，請參閱 [針對 隨需執行個體 使用 容量預留](#)。

```
aws resource-groups create-group \
 --name my-cr-group \
 --configuration '{"Type":"AWS::EC2::CapacityReservationPool"}'
 '{"Type":"AWS::ResourceGroups::Generic", "Parameters": [{"Name": "allowed-resource-
types", "Values": ["AWS::EC2::CapacityReservation"]}]}'
```

## 步驟 3：將容量預留新增至容量預留資源群組

使用 `resource-groups` 服務和 [group-resources](#) 命令，將您在步驟 1 中建立的容量預留新增至容量預留資源群組。請注意，您必須參考隨需容量預留的 ARN。

```
aws resource-groups group-resources \
 --group my-cr-group \
 --resource-arns \
 arn:aws:ec2:us-east-1:123456789012:capacity-reservation/cr-1234567890abcdef1 \
 arn:aws:ec2:us-east-1:123456789012:capacity-reservation/cr-54321abcdef567890
```

### 範例輸出

```
{
 "Failed": [],
```

```
"Succeeded": [
 "arn:aws:ec2:us-east-1:123456789012:capacity-reservation/cr-1234567890abcdef1",
 "arn:aws:ec2:us-east-1:123456789012:capacity-reservation/cr-54321abcdef567890"
]
```

## (選用) 步驟 4：檢視資源群組中的容量預留

使用 `resource-groups` 服務和 [list-group-resources](#) 命令可選擇性地描述資源群組以檢視其容量預留。

```
aws resource-groups list-group-resources --group my-cr-group
```

### 範例輸出

```
{
 "ResourceIdentifiers": [
 {
 "ResourceType": "AWS::EC2::CapacityReservation",
 "ResourceArn": "arn:aws:ec2:us-east-1:123456789012:capacity-reservation/
cr-1234567890abcdef1"
 },
 {
 "ResourceType": "AWS::EC2::CapacityReservation",
 "ResourceArn": "arn:aws:ec2:us-east-1:123456789012:capacity-reservation/
cr-54321abcdef567890"
 }
]
}
```

## 步驟 5：建立啟動範本，此範本指定容量預留以特定資源群組為目標

使用 [create-launch-template](#) 命令來建立啟動範本，使用此範本來指定要使用的容量預留。在此範例中，機群會使用新增至資源群組的 `targeted` 容量預留。因此，啟動範本資料會指定容量預留以特定資源群組為目標。在此範例中，啟動範本會命名為 `my-launch-template`。

```
aws ec2 create-launch-template \
 --launch-template-name my-launch-template \
 --launch-template-data \
 '{"ImageId": "ami-0123456789example",
 "CapacityReservationSpecification":
```

```
 {"CapacityReservationTarget":
 { "CapacityReservationResourceGroupArn": "arn:aws:resource-groups:us-
east-1:123456789012:group/my-cr-group" }
 }
 }'
```

## (選用) 步驟 6：描述啟動範本

使用 [describe-launch-template](#) 命令可選擇性地描述啟動範本以檢視其組態。

```
aws ec2 describe-launch-template-versions --launch-template-name my-launch-template
```

### 範例輸出

```
{
 "LaunchTemplateVersions": [
 {
 "LaunchTemplateId": "lt-01234567890example",
 "LaunchTemplateName": "my-launch-template",
 "VersionNumber": 1,
 "CreateTime": "2021-01-19T20:50:19.000Z",
 "CreatedBy": "arn:aws:iam::123456789012:user/Admin",
 "DefaultVersion": true,
 "LaunchTemplateData": {
 "ImageId": "ami-0947d2ba12ee1ff75",
 "CapacityReservationSpecification": {
 "CapacityReservationTarget": {
 "CapacityReservationResourceGroupArn": "arn:aws:resource-
groups:us-east-1:123456789012:group/my-cr-group"
 }
 }
 }
 }
]
}
```

## 步驟 7：建立 EC2 Fleet

建立 EC2 Fleet，該機群會指定將啟動之執行個體的組態資訊。下列 EC2 Fleet 組態僅顯示此範例的相關組態。啟動範本 my-launch-template 是您在步驟 5 中建立的啟動範本。有兩個執行個體集區，每個集區內的執行個體類型相同 (c5.xlarge)，但使用不同的可用區域 (us-east-1a 和 us-



east-1b)。執行個體集區的價格相同，因為定價是根據區域界定，而不是根據可用區域界定的。總目標容量為 10，預設目標容量類型為 on-demand。隨需分配策略為 lowest-price。容量預留的使用策略為 use-capacity-reservations-first。

### Note

機群類型必須為 instant。其他機群類型不支援 use-capacity-reservations-first。

```
{
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "my-launch-template",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceType": "c5.xlarge",
 "AvailabilityZone": "us-east-1a"
 },
 {
 "InstanceType": "c5.xlarge",
 "AvailabilityZone": "us-east-1b"
 }
]
 }
],
 "TargetCapacitySpecification": {
 "TotalTargetCapacity": 10,
 "DefaultTargetCapacityType": "on-demand"
 },
 "OnDemandOptions": {
 "AllocationStrategy": "lowest-price",
 "CapacityReservationOptions": {
 "UsageStrategy": "use-capacity-reservations-first"
 }
 },
 "Type": "instant"
}
```

使用上述組態建立 instant 機群之後，會啟動以下 10 個執行個體以符合目標容量：

- 會先使用容量預留來啟動 6 個隨需執行個體，如下所示：
  - 3 個隨需執行個體已在 us-east-1a 中啟動至 3 個 c5.xlarge targeted 容量預留
  - 3 個隨需執行個體已在 c5.xlarge 中啟動至 3 個 targeted us-east-1b 容量預留
- 為了滿足目標容量，會根據隨需分配策略 (在本例中為 lowest-price) 使用一般隨需容量來啟動 4 個額外的隨需執行個體。不過，由於集區的價格相同 (因為價格是按區域而非可用區域計價)，因此機群會使用任一個集區來啟動剩餘的 4 個隨需執行個體。

## (選用) 步驟 8：檢視剩餘未使用的容量預留數目

啟動機群後，您也可以執行 [describe-capacity-reservations](#)，以查看還剩餘多少個未使用的容量預留。在此範例中，您應該會看到下列回應，其中顯示所有集區中的所有 容量預留 都已使用。

```
{ "CapacityReservationId": "cr-111",
 "InstanceType": "c5.xlarge",
 "AvailableInstanceCount": 0
}

{ "CapacityReservationId": "cr-222",
 "InstanceType": "c5.xlarge",
 "AvailableInstanceCount": 0
}
```

## 教學課程：在容量區塊中啟動執行個體

本教學課程將逐步引導您完成必須執行的所有步驟，以便 EC2 Fleet 在容量區塊中啟動您的執行個體。如需容量區塊的更多資訊，請參閱 [ML 的容量區塊](#)。

您可以使用即時類型的 EC2 Fleet，在容量區塊中啟動執行個體。如需詳細資訊，請參閱 [使用「instant」類型的 EC2 Fleet](#)。

大多數情況下，EC2 Fleet 請求的目標容量應小於或等於您鎖定之容量區塊的預留可用容量。超出容量區塊預留上限的目標容量請求將無法滿足。如果目標容量請求超出容量區塊預留上限，您會收到容量不足例外狀況，告知您請求超出容量區塊的預留上限。

### Note

對於容量區塊，EC2 Fleet 將不會撤回至針對所需的目標剩餘容量啟動隨需執行個體。

如果 EC2 Fleet 無法在可用的預留容量區塊中滿足所要求的目標容量，EC2 Fleet 會盡可能滿足容量需求，並傳回能夠啟動的執行個體。您可以再次重複呼叫 EC2 Fleet，直到佈建所有執行個體為止。

設定 EC2 Fleet 請求後，您必須等待至容量區塊預留的開始日期。如果您請求 EC2 Fleet 啟動至尚未啟動的容量區塊，則會收到容量不足錯誤。

預留的容量區塊啟用後，您就可以執行 EC2 Fleet 的 API 呼叫，並根據您選取的參數將執行個體佈建到容量區塊中。在容量區塊中執行的執行個體會繼續執行，直到您透過單獨的 Amazon EC2 API 呼叫停止或終止這些執行個體，或 Amazon EC2 在容量區塊預留結束時終止這些執行個體。

### 考量事項

- 同一 CreateFleet 請求不支援多個容量區塊。
- 不支援在使用 OnDemandTargetCapacity 或 SpotTargetCapacity 時，同時將 capacity-block 設為 DefaultTargetCapacity。
- 如果將 DefaultTargetCapacityType 設為 capacity-block，則無法提供 OnDemandOptions::CapacityReservationOptions。一旦提供，便會發生例外狀況。

### 建立啟動範本

啟動範本 ID 會在下列步驟中使用。如需詳細資訊，請參閱 [建立啟動範本](#)。

若要設定啟動範本，請針對 InstanceMarketOptionsRequest 將 MarketType 設為 capacity-block。設定 CapacityReservationID 參數，指定您要鎖定的容量區塊預留 ID。

### 建立 EC2 Fleet

建立檔案 config.json，並為其 EC2 Fleet 採用以下組態。在下列範例中，以您自己的資源識別符取代資源識別符。

```
{
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "CBR-launch-template",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceType": "p5.48xlarge",
 "AvailabilityZone": "us-east-1a"
 }
]
 }
]
}
```

```
 },
],
}
],
"TargetCapacitySpecification": {
 "TotalTargetCapacity": 10,
 "DefaultTargetCapacityType": "capacity-block"
},
"Type": "instant"
}
```

使用以下 [create-fleet](#) 命令。

```
aws ec2 create-fleet \
 --cli-input-json file://config.json
```

如需詳細資訊，請參閱 [建立 EC2 Fleet](#)。

## 教學：使用 Spot Fleet 與執行個體權重

本教學課程使用一間稱為 Example Corp 的虛構公司，來說明使用執行個體權重來請求 Spot Fleet 的程序。

### 目標

Example Corp 是一間製藥公司，希望利用 Amazon EC2 的運算能力，來篩選可能用於抗癌的化合物。

### 規劃

Example Corp 首先檢視了 [Spot 最佳實務](#)。接著，Example Corp 針對其 Spot Fleet 決定了下列需求。

#### 執行個體類型

Example Corp 擁有運算和記憶體密集型應用程式，該應用程式在至少 60 GB 記憶體和八個虛擬 CPU (vCPU) 情況下效能最佳。他們希望以最低的價格為應用程式最大限度利用這些資源。Example Corp 認為以下任一個 EC2 執行個體類型都能滿足他們的需求：

執行個體類型	記憶體 (GiB)	vCPU
r3.2xlarge	61	8

執行個體類型	記憶體 (GiB)	vCPU
r3.4xlarge	122	16
r3.8xlarge	244	32

### 目標容量 (單位)

使用執行個體加權時，目標容量可以等於執行個體數目 (預設值) 或核心 (vCPUs)、記憶體 () 和儲存體 (GBGiBs) 等因素的組合。Example Corp 將其應用程式的基本需求 (60 GB 的 RAM 和 8 個 vCPU)，視為 1 個單位來考量，然後決定此數量的 20 倍可滿足其需求。因此，該公司將其 Spot Fleet 請求的目標容量設定為 20。

### 執行個體權重

Example Corp 決定目標容量後，即計算執行個體權重。為了計算每個執行個體類型的執行個體權重，將決定達成目標容量所需的每個執行個體類型單位，如下所示：

- r3.2xlarge (61.0 GB, 8 vCPUs) = 1 單位 (共 20 單位)
- r3.4xlarge (122.0 GB, 16 vCPUs) = 2 單位 (共 20 單位)
- r3.8xlarge (244.0 GB, 32 vCPUs) = 4 單位 (共 20 單位)

因此，Example Corp 會針對其 Spot Fleet 請求中的各個啟動組態，指派 1、2 和 4 的執行個體權重。

### 每個單位小時的價格

Example Corp 會使用每執行個體小時的[隨需價格](#)，來做為其價格的起點。他們也可以使用最近的 Spot 價格，或兩者的組合。為計算每單位小時價格，他們將每個執行個體小時的起始價格除以權重。例如：

執行個體類型	隨需價格	執行個體權重	每個單位小時的價格
r3.2xLarge	0.7 USD	1	0.7 USD
r3.4xLarge	1.4 USD	2	0.7 USD
r3.8xLarge	\$2.8	4	0.7 USD

Example Corp 可以使用每單位小時 0.7 USD 的全球價格，並且對於所有三種執行個體類型都具有競爭力。在 r3.8xlarge 啟動規格中，他們也可以使用每單位小時 0.7 USD 全局價格和 0.9 USD 每單位小時特定價格。

## 驗證許可

在建立 Spot Fleet 請求之前，Example Corp 會確認自己擁有 IAM 角色，而且此角色具備所需的許可。如需詳細資訊，請參閱 [Spot Fleet 許可](#)。

## 建立請求

Example Corp 會建立一個檔案，config.json，並針對其 Spot Fleet 請求使用下列組態：

```
{
 "SpotPrice": "0.70",
 "TargetCapacity": 20,
 "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
 "LaunchSpecifications": [
 {
 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "r3.2xlarge",
 "SubnetId": "subnet-482e4972",
 "WeightedCapacity": 1
 },
 {
 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "r3.4xlarge",
 "SubnetId": "subnet-482e4972",
 "WeightedCapacity": 2
 },
 {
 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "r3.8xlarge",
 "SubnetId": "subnet-482e4972",
 "SpotPrice": "0.90",
 "WeightedCapacity": 4
 }
]
}
```

Example Corp 會使用 [request-spot-fleet](#) 命令來建立 Spot Fleet 請求。

```
aws ec2 request-spot-fleet --spot-fleet-request-config file://config.json
```

如需詳細資訊，請參閱 [Spot Fleet 請求類型](#)。

## 實現

分配策略決定您的 競價型執行個體 來自哪個 Spot 容量集區。

採用 lowestPrice 策略 (預設的策略)，競價型執行個體 來自實現時每單位價格最低的集區。為提供 20 個單位的容量，Spot Fleet 將啟動 20 個 r3.2xlarge 執行個體 (20 除以 1)、10 個 r3.4xlarge 執行個體 (20 除以 2) 或 5 個 r3.8xlarge 執行個體 (20 除以 4)。

如果 Example Corp 使用 diversified 策略，競價型執行個體 將來自所有三個集區。Spot Fleet 將啟動 6 個 r3.2xlarge 執行個體 (提供 6 個單位)、3 個 r3.4xlarge 執行個體 (提供 6 個單位) 和 2 個 r3.8xlarge 執行個體 (提供 8 個單位)，共計 20 個單位。

## EC2 Fleet 和 Spot Fleet 的範例組態

以下範例為您可以用來建立 EC2 Fleet 和 Spot Fleet 的啟動組態。

### 主題

- [EC2 Fleet 範例組態](#)
- [Spot Fleet 範例組態](#)

## EC2 Fleet 範例組態

以下範例示範您可以使用 [create-fleet](#) 命令來建立 EC2 Fleet 的啟動組態。如需參數的詳細資訊，請參閱 AWS CLI 命令參考中的 [create-fleet](#)。

### 範例

- [範例 1：啟動 競價型執行個體 做為預設購買選項](#)
- [範例 2：啟動 隨需執行個體 做為預設購買選項](#)
- [範例 3：啟動 隨需執行個體 做為主要容量](#)
- [範例 4：使用多個容量保留啟動隨需執行個體](#)
- [範例 5：當目標容量總計超過未使用的容量保留數時，使用容量保留啟動隨需執行個體](#)
- [範例 6：使用目標容量保留啟動隨需執行個體](#)

- [範例 7：設定容量重新平衡以啟動替代 Spot 執行個體](#)
- [範例 8：在容量最佳化叢集中啟動 Spot 執行個體](#)
- [範例 9：在具有優先順序的容量最佳化叢集中啟動 Spot 執行個體](#)
- [範例 10：在 price-capacity-optimized 叢集中啟動 Spot 執行個體](#)
- [範例 11：設定以屬性為基礎的執行個體類型選取](#)

### 範例 1：啟動 競價型執行個體 做為預設購買選項

下列範例指定 EC2 Fleet 中所需的最小參數：啟動範本、目標容量和預設購買選項。啟動範本由其啟動範本 ID 和版本編號識別。機群的目標容量為 2 個執行個體，預設購買選項為 spot，會讓機群啟動 2 個 競價型執行個體。

```
{
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateId": "lt-0e8c754449b27161c",
 "Version": "1"
 }
 }
],
 "TargetCapacitySpecification": {
 "TotalTargetCapacity": 2,
 "DefaultTargetCapacityType": "spot"
 }
}
```

### 範例 2：啟動 隨需執行個體 做為預設購買選項

下列範例指定 EC2 Fleet 中所需的最小參數：啟動範本、目標容量和預設購買選項。啟動範本由其啟動範本 ID 和版本編號識別。機群的目標容量為 2 個執行個體，預設購買選項為 on-demand，會讓機群啟動 2 個 隨需執行個體。

```
{
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateId": "lt-0e8c754449b27161c",
```



```

 "Version": "1"
 }
 }
],
 "TargetCapacitySpecification": {
 "TotalTargetCapacity": 2,
 "DefaultTargetCapacityType": "on-demand"
 }
}

```

### 範例 3：啟動 隨需執行個體 做為主要容量

下列範例指定機群 2 個執行個體的總目標容量，以及 1 個隨需執行個體的目標容量。預設購買選項為 spot。機群依照指定啟動 1 個隨需執行個體，但需要再啟動一個執行個體才能實現總目標容量。其差值的購買選項計算式為  $TotalTargetCapacity - OnDemandTargetCapacity = DefaultTargetCapacityType$ ，這會讓機群啟動 1 個 Spot 執行個體。

```

{
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateId": "lt-0e8c754449b27161c",
 "Version": "1"
 }
 }
],
 "TargetCapacitySpecification": {
 "TotalTargetCapacity": 2,
 "OnDemandTargetCapacity": 1,
 "DefaultTargetCapacityType": "spot"
 }
}

```

### 範例 4：使用多個容量保留啟動隨需執行個體

您可以將容量預留的使用策略設為 `use-capacity-reservations-first`，以將機群設定為在啟動隨需執行個體時首先使用 隨需容量預留。本例示範當容量預留比所需目標容量更多時，機群如何選取要使用的容量預留。

本例使用的機群組態如下：

- 目標容量：12 個隨需執行個體
- 總計未使用的容量預留：15 (大於機群的目標容量：12 個隨需執行個體)
- 容量預留集區數目：3 (m5.large、m4.xlarge 和 m4.2xlarge)
- 每個集區的容量預留數目：5
- 隨需分配策略：lowest-price (當多個執行個體集區中有多個未使用的容量預留時，機群會根據隨需分配策略決定要用來啟動隨需執行個體的集區。)

請注意，您也可以使用 prioritized 分配策略，而不使用 lowest-price 分配策略。

## Capacity Reservations

帳戶在 3 個不同的集區中有以下 15 個未使用的 容量預留。每個集區中的 容量預留 數目由 AvailableInstanceCount 表示。

```
{
 "CapacityReservationId": "cr-111",
 "InstanceType": "m5.large",
 "InstancePlatform": "Linux/UNIX",
 "AvailabilityZone": "us-east-1a",
 "AvailableInstanceCount": 5,
 "InstanceMatchCriteria": "open",
 "State": "active"
}

{
 "CapacityReservationId": "cr-222",
 "InstanceType": "m4.xlarge",
 "InstancePlatform": "Linux/UNIX",
 "AvailabilityZone": "us-east-1a",
 "AvailableInstanceCount": 5,
 "InstanceMatchCriteria": "open",
 "State": "active"
}

{
 "CapacityReservationId": "cr-333",
 "InstanceType": "m4.2xlarge",
 "InstancePlatform": "Linux/UNIX",
 "AvailabilityZone": "us-east-1a",
 "AvailableInstanceCount": 5,
 "InstanceMatchCriteria": "open",
```

```
"State": "active"
}
```

## 機群組態

下列機群組態僅顯示此範例的相關組態。總目標容量為 12，預設目標容量類型為 on-demand。隨需分配策略為 lowest-price。容量預留的使用策略為 use-capacity-reservations-first。

在此範例中，隨需執行個體價格如下：

- m5.large – 每小時 0.096 美元
- m4.xlarge – 每小時 0.20 美元
- m4.2xlarge – 每小時 0.40 美元

### Note

機群類型必須為 instant。其他機群類型不支援 use-capacity-reservations-first。

```
{
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateId": "lt-abc1234567example",
 "Version": "1"
 }
 "Overrides": [
 {
 "InstanceType": "m5.large",
 "AvailabilityZone": "us-east-1a",
 "WeightedCapacity": 1
 },
 {
 "InstanceType": "m4.xlarge",
 "AvailabilityZone": "us-east-1a",
 "WeightedCapacity": 1
 },
 {
 "InstanceType": "m4.2xlarge",
 "AvailabilityZone": "us-east-1a",

```

```

 "WeightedCapacity": 1
 }
]

 },
 "TargetCapacitySpecification": {
 "TotalTargetCapacity": 12,
 "DefaultTargetCapacityType": "on-demand"
 },
 "OnDemandOptions": {
 "AllocationStrategy": "lowest-price"
 "CapacityReservationOptions": {
 "UsageStrategy": "use-capacity-reservations-first"
 }
 },
 "Type": "instant",
}

```

使用上述組態建立 instant 叢集之後，會啟動以下 12 個執行個體以符合目標容量：

- 5 個 m5.large 隨需執行個體在 us-east-1a 中 (us-east-1a 中的 m5.large 價格最低)，並且有 5 個可用的未使用 m5.large 容量預留。
- 5 個 m4.xlarge 隨需執行個體在 us-east-1a 中 – us-east-1a 中的 m4.xlarge 價格次低，並且有 5 個可用的未使用 m4.xlarge 容量預留。
- 2 個 m4.2xlarge 隨需執行個體在 us-east-1a 中 – us-east-1a 中的 m4.2xlarge 價格第三低，並且有 5 個可用的未使用 m4.2xlarge 容量預留，只需要其中 2 個就可滿足目標容量

啟動機群後，您可以執行 [describe-capacity-reservations](#)，以查看還剩餘多少個未使用的 容量預留。在本例中，您會看到下面回應，這顯示所有的 m5.large 和 m4.xlarge 容量預留都已使用，只剩 3 個 m4.2xlarge 容量預留未使用。

```

{
 "CapacityReservationId": "cr-111",
 "InstanceType": "m5.large",
 "AvailableInstanceCount": 0
}

{
 "CapacityReservationId": "cr-222",
 "InstanceType": "m4.xlarge",

```

```
 "AvailableInstanceCount": 0
 }

 {
 "CapacityReservationId": "cr-333",
 "InstanceType": "m4.2xlarge",
 "AvailableInstanceCount": 3
 }
}
```

### 範例 5：當目標容量總計超過未使用的容量保留數時，使用容量保留啟動隨需執行個體

您可以將容量預留的使用策略設為 `use-capacity-reservations-first`，以將機群設定為在啟動隨需執行個體時首先使用隨需容量預留。本例示範當總計目標容量超過可用的未使用容量預留數目時，機群會如何選取用來啟動隨需執行個體的執行個體集區。

本例使用的機群組態如下：

- 目標容量：16 個隨需執行個體
- 總計未使用的容量預留：15 (小於機群的目标容量：16 個隨需執行個體)
- 容量預留集區數目：3 (m5.large、m4.xlarge 和 m4.2xlarge)
- 每個集區的容量預留數目：5
- 隨需分配策略：lowest-price (當未使用的容量預留數目小於隨需目標容量時，機群會根據隨需分配策略決定要用來啟動剩餘隨需容量的集區。)

請注意，您也可以使用 `prioritized` 分配策略，而不使用 `lowest-price` 分配策略。

### Capacity Reservations

帳戶在 3 個不同的集區中有以下 15 個未使用的容量預留。每個集區中的容量預留數目由 `AvailableInstanceCount` 表示。

```
{
 "CapacityReservationId": "cr-111",
 "InstanceType": "m5.large",
 "InstancePlatform": "Linux/UNIX",
 "AvailabilityZone": "us-east-1a",
 "AvailableInstanceCount": 5,
 "InstanceMatchCriteria": "open",
 "State": "active"
}
```

```
{
 "CapacityReservationId": "cr-222",
 "InstanceType": "m4.xlarge",
 "InstancePlatform": "Linux/UNIX",
 "AvailabilityZone": "us-east-1a",
 "AvailableInstanceCount": 5,
 "InstanceMatchCriteria": "open",
 "State": "active"
}

{
 "CapacityReservationId": "cr-333",
 "InstanceType": "m4.2xlarge",
 "InstancePlatform": "Linux/UNIX",
 "AvailabilityZone": "us-east-1a",
 "AvailableInstanceCount": 5,
 "InstanceMatchCriteria": "open",
 "State": "active"
}
```

## 機群組態

下列機群組態僅顯示此範例的相關組態。總目標容量為 16，預設目標容量類型為 on-demand。隨需分配策略為 lowest-price。容量預留的使用策略為 use-capacity-reservations-first。

在此範例中，隨需執行個體價格如下：

- m5.large – 每小時 0.096 USD
- m4.xlarge – 每小時 0.20 USD
- m4.2xlarge – 每小時 0.40 USD

### Note

機群類型必須為 instant。其他機群類型不支援 use-capacity-reservations-first。

```
{
 "LaunchTemplateConfigs": [
 {
```

```
 "LaunchTemplateSpecification": {
 "LaunchTemplateId": "lt-0e8c754449b27161c",
 "Version": "1"
 }
 "Overrides": [
 {
 "InstanceType": "m5.large",
 "AvailabilityZone": "us-east-1a",
 "WeightedCapacity": 1
 },
 {
 "InstanceType": "m4.xlarge",
 "AvailabilityZone": "us-east-1a",
 "WeightedCapacity": 1
 },
 {
 "InstanceType": "m4.2xlarge",
 "AvailabilityZone": "us-east-1a",
 "WeightedCapacity": 1
 }
]
 }
],
"TargetCapacitySpecification": {
 "TotalTargetCapacity": 16,
 "DefaultTargetCapacityType": "on-demand"
},
"OnDemandOptions": {
 "AllocationStrategy": "lowest-price"
 "CapacityReservationOptions": {
 "UsageStrategy": "use-capacity-reservations-first"
 }
},
"Type": "instant",
}
```

使用上述組態建立 instant 叢集之後，會啟動以下 16 個執行個體以符合目標容量：

- 6 個 m5.large 隨需執行個體在 us-east-1a 中 – us-east-1a 中的 m5.large 價格最低，並且有 5 個可用的未使用 m5.large 容量預留。會先使用容量預留來啟動 5 個隨需執行個體。在使用了其餘的 m4.xlarge 和 m4.2xlarge 容量預留之後，為了滿足目標容量，會根據隨需分配策略啟動額外的隨需執行個體，在本例中為 lowest-price。

- 5 個 m4.xlarge 隨需執行個體在 us-east-1a 中 – us-east-1a 中的 m4.xlarge 價格次低，並且有 5 個可用的未使用 m4.xlarge 容量預留。
- 5 個 m4.2xlarge 隨需執行個體在 us-east-1a 中 – us-east-1a 中的 m4.2xlarge 價格第三低，並且有 5 個可用的未使用 m4.2xlarge 容量預留。

啟動機群後，您可以執行 [describe-capacity-reservations](#)，以查看還剩餘多少個未使用的 容量預留。在此範例中，您應該會看到下列回應，其中顯示所有集區中的所有 容量預留 都已使用。

```
{
 "CapacityReservationId": "cr-111",
 "InstanceType": "m5.large",
 "AvailableInstanceCount": 0
}

{
 "CapacityReservationId": "cr-222",
 "InstanceType": "m4.xlarge",
 "AvailableInstanceCount": 0
}

{
 "CapacityReservationId": "cr-333",
 "InstanceType": "m4.2xlarge",
 "AvailableInstanceCount": 0
}
```

## 範例 6：使用目標容量保留啟動隨需執行個體

您可設定機群，使其在啟動隨需執行個體時首先使用 targeted 隨需容量預留，設定方法是將容量預留的使用策略設為 use-capacity-reservations-first。本例示範如何使用 targeted 容量預留來啟動隨需執行個體，其中容量預留的屬性皆相同，但其可用區域 (us-east-1a 和 us-east-1b) 不同。本例亦示範當總計目標容量超過可用的未使用容量預留數目時，機群會如何選取用來啟動隨需執行個體的執行個體集區。

本例使用的機群組態如下：

- 目標容量：10 個隨需執行個體
- 總計未使用的 targeted 容量預留：6 (小於機群的隨需目標容量：10 個隨需執行個體)
- 容量預留集區數目：2 (us-east-1a 和 us-east-1b)



- 每個集區的容量預留數目：3
- 隨需分配策略：lowest-price (當未使用的容量預留數目小於隨需目標容量時，機群會根據隨需分配策略決定要用來啟動剩餘隨需容量的集區。)

請注意，您也可以使用 prioritized 分配策略，而不使用 lowest-price 分配策略。

如需完成此範例所須執程序的逐步演練，請參閱 [教學課程：使用目標容量預留啟動隨需執行個體](#)。

## Capacity Reservations

帳戶在 2 個不同的集區中有以下 6 個未使用的容量預留。在本例中，集區的可用區域有所不同。每個集區中的 容量預留 數目由 AvailableInstanceCount 表示。

```
{
 "CapacityReservationId": "cr-111",
 "InstanceType": "c5.xlarge",
 "InstancePlatform": "Linux/UNIX",
 "AvailabilityZone": "us-east-1a",
 "AvailableInstanceCount": 3,
 "InstanceMatchCriteria": "open",
 "State": "active"
}

{
 "CapacityReservationId": "cr-222",
 "InstanceType": "c5.xlarge",
 "InstancePlatform": "Linux/UNIX",
 "AvailabilityZone": "us-east-1b",
 "AvailableInstanceCount": 3,
 "InstanceMatchCriteria": "open",
 "State": "active"
}
```

## 機群組態

下列機群組態僅顯示此範例的相關組態。總目標容量為 10，預設目標容量類型為 on-demand。隨需分配策略為 lowest-price。容量預留的使用策略為 use-capacity-reservations-first。

在本例中，us-east-1 中的 c5.xlarge 隨需執行個體價格為每小時 0.17 美元。

**Note**

機群類型必須為 `instant`。其他機群類型不支援 `use-capacity-reservations-first`。

```
{
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "my-launch-template",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceType": "c5.xlarge",
 "AvailabilityZone": "us-east-1a"
 },
 {
 "InstanceType": "c5.xlarge",
 "AvailabilityZone": "us-east-1b"
 }
]
 }
],
 "TargetCapacitySpecification": {
 "TotalTargetCapacity": 10,
 "DefaultTargetCapacityType": "on-demand"
 },
 "OnDemandOptions": {
 "AllocationStrategy": "lowest-price",
 "CapacityReservationOptions": {
 "UsageStrategy": "use-capacity-reservations-first"
 }
 },
 "Type": "instant"
}
```

使用上述組態建立 `instant` 機群之後，會啟動以下 10 個執行個體以符合目標容量：

- 會先使用容量預留來啟動 6 個隨需執行個體，如下所示：
  - 3 個隨需執行個體已在 `us-east-1a` 中啟動至 3 個 `c5.xlarge` targeted 容量預留

- 3 個隨需執行個體已在 c5.xlarge 中啟動至 3 個 targeted us-east-1b 容量預留
- 為了滿足目標容量，會根據隨需分配策略 (在本例中為 lowest-price) 使用一般隨需容量來啟動 4 個額外的隨需執行個體。不過，由於集區的價格相同 (因為價格是按區域而非可用區域計價)，因此機群會使用任一個集區來啟動剩餘的 4 個隨需執行個體。

啟動機群後，您可以執行 [describe-capacity-reservations](#)，以查看還剩餘多少個未使用的 容量預留。在此範例中，您應該會看到下列回應，其中顯示所有集區中的所有 容量預留 都已使用。

```
{
 "CapacityReservationId": "cr-111",
 "InstanceType": "c5.xlarge",
 "AvailableInstanceCount": 0
}

{
 "CapacityReservationId": "cr-222",
 "InstanceType": "c5.xlarge",
 "AvailableInstanceCount": 0
}
```

### 範例 7：設定容量重新平衡以啟動替代 Spot 執行個體

下列範例會設定 EC2 Fleet，以便 Amazon EC2 在機群中對 Spot 執行個體發出重新平衡建議時啟動替代 Spot 執行個體。若要為 ReplacementStrategy 設定 競價型執行個體 的自動替代，請指定 launch-before-terminate。若要設定從啟動新替代 Spot 執行個體到自動刪除舊 Spot 執行個體的時間延遲，請針對 termination-delay，指定一值 (以秒為單位)。如需詳細資訊，請參閱 [組態選項](#)。

#### Note

建議您僅在可以預測執行個體關閉程序需要多長時間才能完成時，才使用 launch-before-terminate，以便只在這些程序完成後才會終止舊執行個體。所有執行個體在執行時，您需要支付所有執行個體的費用。

容量重新平衡策略的有效性取決於 EC2 Fleet 請求中指定的 Spot 容量集區數目。我們建議您使用多樣化的執行個體類型和可用區域的集合來設定機群，若要 AllocationStrategy，請指定 capacity-optimized。如需設定 EC2 Fleet 容量重新平衡時應考量哪些項目的詳細資訊，請參閱 [容量重新平衡](#)。

```
{
 "ExcessCapacityTerminationPolicy": "termination",
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "LaunchTemplate",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceType": "c3.large",
 "WeightedCapacity": 1,
 "Placement": {
 "AvailabilityZone": "us-east-1a"
 }
 },
 {
 "InstanceType": "c4.large",
 "WeightedCapacity": 1,
 "Placement": {
 "AvailabilityZone": "us-east-1a"
 }
 },
 {
 "InstanceType": "c5.large",
 "WeightedCapacity": 1,
 "Placement": {
 "AvailabilityZone": "us-east-1a"
 }
 }
]
 }
],
 "TargetCapacitySpecification": {
 "TotalTargetCapacity": 5,
 "DefaultTargetCapacityType": "spot"
 },
 "SpotOptions": {
 "AllocationStrategy": "capacity-optimized",
 "MaintenanceStrategies": {
 "CapacityRebalance": {
 "ReplacementStrategy": "launch-before-terminate",
 "TerminationDelay": "720"
 }
 }
 }
}
```

```

 }
 }
}
}

```

## 範例 8：在容量最佳化叢集中啟動 Spot 執行個體

下列範例示範如何使用可最佳化容量的 Spot 分配策略來設定 EC2 Fleet。若要最佳化容量，您必須將 `AllocationStrategy` 設為 `capacity-optimized`。

在下列範例中，三個啟動規格指定三個 Spot 容量集區。目標容量為 50 個 Spot 執行個體。EC2 Fleet 嘗試將 50 個 Spot 執行個體啟動到 Spot 容量集區中，且擁有啟動中執行個體數目之最佳容量。

```

{
 "SpotOptions": {
 "AllocationStrategy": "capacity-optimized",
 },
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "my-launch-template",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceType": "r4.2xlarge",
 "Placement": {
 "AvailabilityZone": "us-west-2a"
 }
 },
 {
 "InstanceType": "m4.2xlarge",
 "Placement": {
 "AvailabilityZone": "us-west-2b"
 }
 },
 {
 "InstanceType": "c5.2xlarge",
 "Placement": {
 "AvailabilityZone": "us-west-2b"
 }
 }
]
 }
]
}

```

```
 }
],
 "TargetCapacitySpecification": {
 "TotalTargetCapacity": 50,
 "DefaultTargetCapacityType": "spot"
 }
}
```

## 範例 9：在具有優先順序的容量最佳化叢集中啟動 Spot 執行個體

下列範例示範在盡最大努力使用優先順序時，如何使用可最佳化容量的 Spot 分配策略來設定 EC2 Fleet。

當您使用 `capacity-optimized-prioritized` 分配策略時，您可以使用 `Priority` 參數來指定 Spot 容量集區的優先順序，其中數字越小，優先順序越高。如果您對它們一視同仁，也可以為數個 Spot 容量集區設定相同的優先順序。如果您沒有設定集區的優先順序，集區將被視為最後一個優先順序。

若要排定 Spot 容量集區的優先順序，您必須將 `AllocationStrategy` 設為 `capacity-optimized-prioritized`。EC2 Fleet 將首先針對容量進行最佳化，但會盡力實現優先順序 (例如，若實現優先順序不會顯著影響 EC2 Fleet 佈建最佳容量的能力)。對於必須將中斷可能性降至最低的工作負載來說，這是一個很好的選擇，而且某些執行個體類型的偏好也很重要。

在下列範例中，三個啟動規格指定三個 Spot 容量集區。每個集區都有優先順序，其中數字越小，優先順序越高。目標容量為 50 個 Spot 執行個體。EC2 Fleet 嘗試以最高優先順序在 Spot 容量集區中啟動 50 個 Spot 執行個體，但首先針對容量進行最佳化。

```
{
 "SpotOptions": {
 "AllocationStrategy": "capacity-optimized-prioritized"
 },
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "my-launch-template",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceType": "r4.2xlarge",
 "Priority": 1,

```

```

 "Placement": {
 "AvailabilityZone": "us-west-2a"
 },
 },
 {
 "InstanceType": "m4.2xlarge",
 "Priority": 2,
 "Placement": {
 "AvailabilityZone": "us-west-2b"
 },
 },
 {
 "InstanceType": "c5.2xlarge",
 "Priority": 3,
 "Placement": {
 "AvailabilityZone": "us-west-2b"
 }
 }
]
}
],
"TargetCapacitySpecification": {
 "TotalTargetCapacity": 50,
 "DefaultTargetCapacityType": "spot"
}
}

```

## 範例 10：在 price-capacity-optimized 叢集中啟動 Spot 執行個體

下列範例示範如何使用可最佳化容量且價格最低的 Spot 分配策略來設定 EC2 機群。若要在將價格納入考慮的同時最佳化容量，您必須將 Spot AllocationStrategy 設定為 price-capacity-optimized。

在下列範例中，三個啟動規格指定三個 Spot 容量集區。目標容量為 50 個 Spot 執行個體。EC2 機群嘗試將 50 個 Spot 執行個體啟動到 Spot 容量集區中，且擁有啟動中執行個體數目之最佳容量，同時還會選擇價格最低的集區。

```

{
 "SpotOptions": {
 "AllocationStrategy": "price-capacity-optimized",
 "MinTargetCapacity": 2,
 "SingleInstanceType": true
 },
 "OnDemandOptions": {

```

```

 "AllocationStrategy": "lowest-price"
 },
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "my-launch-template",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceType": "r4.2xlarge",
 "Placement": {
 "AvailabilityZone": "us-west-2a"
 },
 },
 {
 "InstanceType": "m4.2xlarge",
 "Placement": {
 "AvailabilityZone": "us-west-2b"
 },
 },
 {
 "InstanceType": "c5.2xlarge",
 "Placement": {
 "AvailabilityZone": "us-west-2b"
 }
 }
]
 }
],
 "TargetCapacitySpecification": {
 "TotalTargetCapacity": 50,
 "OnDemandTargetCapacity": 0,
 "SpotTargetCapacity": 50,
 "DefaultTargetCapacityType": "spot"
 },
 "Type": "instant"
}

```

## 範例 11：設定以屬性為基礎的執行個體類型選取

下列範例示範如何將 EC2 機群設定為使用屬性型執行個體類型選取來識別執行個體類型。若要指定必要的執行個體屬性，您可以在 InstanceRequirements 結構中指定屬性。



在下列範例中，指定了兩個執行個體屬性：

- VCpuCount – 至少指定 2 個 vCPU。由於未指定最大值，因此沒有最大限制。
- MemoryMiB – 至少指定 4 MiB 的記憶體。由於未指定最大值，因此沒有最大限制。

將會識別任何具有 2 個或更多 vCPU 和 4 MiB 或更多記憶體的執行個體類型。但是，當 [EC2 機群佈建機群](#) 時，價格保護和分配策略可能會排除某些執行個體類型。

如需可指定之所有可能屬性的清單和說明，請參閱 Amazon EC2 API 參考 [InstanceRequirements](#) 中的。

```
{
 "SpotOptions": {
 "AllocationStrategy": "price-capacity-optimized"
 },
 "LaunchTemplateConfigs": [{
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "my-launch-template",
 "Version": "1"
 },
 "Overrides": [{
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 2
 },
 "MemoryMiB": {
 "Min": 4
 }
 }
]
 }
],
 "TargetCapacitySpecification": {
 "TotalTargetCapacity": 20,
 "DefaultTargetCapacityType": "spot"
 },
 "Type": "instant"
}
```

## Spot Fleet 範例組態

下列範例顯示可搭配 [request-spot-fleet](#) 命令使用的啟動組態，以建立 Spot Fleet 請求。如需詳細資訊，請參閱 [建立 Spot Fleet 請求](#)。

### Note

對於 Spot 機群，您無法在啟動範本或啟動規格中指定網路介面 ID。確保省略啟動範本或啟動規格中的 `NetworkInterfaceID` 參數。

### 範例

- [範例 1：使用區域中最低價格的可用區域或子網來啟動 競價型執行個體](#)
- [範例 2：使用指定清單中最低價格的可用區域或子網來啟動 競價型執行個體](#)
- [範例 3：使用指定清單中最低價格的執行個體類型來啟動 競價型執行個體](#)
- [範例 4。覆寫請求的價格](#)
- [範例 5：使用多樣化分配策略來啟動 Spot Fleet](#)
- [範例 6：使用執行個體權重來啟動 Spot Fleet](#)
- [範例 7：使用隨需容量來啟動 Spot Fleet](#)
- [範例 8：設定容量重新平衡以啟動替代 競價型執行個體](#)
- [範例 9：在容量最佳化機群中啟動 Spot 執行個體](#)
- [範例 10：使用優先順序在容量最佳化機群中啟動 Spot 執行個體](#)
- [範例 11：在 `priceCapacityOptimized` 叢集中啟動 Spot 執行個體](#)
- [範例 12：設定屬性型執行個體類型選取](#)

### 範例 1：使用區域中最低價格的可用區域或子網來啟動 競價型執行個體

下列的範例指定了單一啟動規格，其中未包含可用區域或子網。Spot Fleet 會在具有預設子網的最低價格可用區域中，啟動執行個體。您所支付的價格不會超過隨需價格。

```
{
 "TargetCapacity": 20,
 "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
 "LaunchSpecifications": [
```

```

 {
 "ImageId": "ami-1a2b3c4d",
 "KeyName": "my-key-pair",
 "SecurityGroups": [
 {
 "GroupId": "sg-1a2b3c4d"
 }
],
 "InstanceType": "m3.medium",
 "IamInstanceProfile": {
 "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
 }
 }
]
}

```

## 範例 2：使用指定清單中最低價格的可用區域或子網來啟動 競價型執行個體

下列的範例指定了兩種啟動規格，各包含不同的可用區域或子網，但具有相同的執行個體類型和 AMI。

### 可用區域

Spot Fleet 會在您所指定最低價格可用區域的預設子網中，啟動執行個體。

```

{
 "TargetCapacity": 20,
 "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
 "LaunchSpecifications": [
 {
 "ImageId": "ami-1a2b3c4d",
 "KeyName": "my-key-pair",
 "SecurityGroups": [
 {
 "GroupId": "sg-1a2b3c4d"
 }
],
 "InstanceType": "m3.medium",
 "Placement": {
 "AvailabilityZone": "us-west-2a, us-west-2b"
 },
 "IamInstanceProfile": {
 "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
 }
 }
]
}

```

```

 }
 }
]
}

```

## 子網

您可以指定預設子網或非預設子網，而非預設子網可以是來自預設 VPC 或非預設 VPC。Spot 服務會在最低價格可用區域的任何可用子網中，啟動執行個體。

您不能在 Spot Fleet 請求中指定來自同一個可用區域的不同子網。

```

{
 "TargetCapacity": 20,
 "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
 "LaunchSpecifications": [
 {
 "ImageId": "ami-1a2b3c4d",
 "KeyName": "my-key-pair",
 "SecurityGroups": [
 {
 "GroupId": "sg-1a2b3c4d"
 }
],
 "InstanceType": "m3.medium",
 "SubnetId": "subnet-a61dafcf, subnet-65ea5f08",
 "IamInstanceProfile": {
 "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
 }
 }
]
}

```

如果執行個體是在預設的 VPC 中啟動，則其預設會接收公有 IPv4 位址。如果執行個體是在非預設的 VPC 中啟動，則其預設不會接收公有 IPv4 位址。在預設規格中啟動網路介面，來指派公有 IPv4 位址給在非預設 VPC 中啟動的執行個體。指定網路介面時，必須加入使用網路介面的子網 ID 和安全群組 ID。

```

...
{
 "ImageId": "ami-1a2b3c4d",
 "KeyName": "my-key-pair",

```

```

 "InstanceType": "m3.medium",
 "NetworkInterfaces": [
 {
 "DeviceIndex": 0,
 "SubnetId": "subnet-1a2b3c4d",
 "Groups": ["sg-1a2b3c4d"],
 "AssociatePublicIpAddress": true
 }
],
 "IamInstanceProfile": {
 "Arn": "arn:aws:iam::880185128111:instance-profile/my-iam-role"
 }
 }
 ...

```

### 範例 3：使用指定清單中最低價格的執行個體類型來啟動 競價型執行個體

下列的範例指定了兩種啟動組態，各具有不同的執行個體類型，但包含相同的 AMI 和可用區域或子網。Spot Fleet 會使用具有最低價格的指定執行個體類型，來啟動執行個體。

#### 可用區域

```

{
 "TargetCapacity": 20,
 "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
 "LaunchSpecifications": [
 {
 "ImageId": "ami-1a2b3c4d",
 "SecurityGroups": [
 {
 "GroupId": "sg-1a2b3c4d"
 }
],
 "InstanceType": "c5.4xlarge",
 "Placement": {
 "AvailabilityZone": "us-west-2b"
 }
 },
 {
 "ImageId": "ami-1a2b3c4d",
 "SecurityGroups": [
 {
 "GroupId": "sg-1a2b3c4d"
 }
]
 }
]
}

```

```

 }
],
 "InstanceType": "r3.8xlarge",
 "Placement": {
 "AvailabilityZone": "us-west-2b"
 }
}
]
}

```

## 子網

```

{
 "TargetCapacity": 20,
 "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
 "LaunchSpecifications": [
 {
 "ImageId": "ami-1a2b3c4d",
 "SecurityGroups": [
 {
 "GroupId": "sg-1a2b3c4d"
 }
],
 "InstanceType": "c5.4xlarge",
 "SubnetId": "subnet-1a2b3c4d"
 },
 {
 "ImageId": "ami-1a2b3c4d",
 "SecurityGroups": [
 {
 "GroupId": "sg-1a2b3c4d"
 }
],
 "InstanceType": "r3.8xlarge",
 "SubnetId": "subnet-1a2b3c4d"
 }
]
}

```

## 範例 4。覆寫請求的價格

我們建議您使用預設的最高價格，也就是隨需價格。如果想要的話，您可以指定機群請求的最高價格，和個別啟動規格的最高價格。

下列的範例指定了機群請求的最高價格，和三種啟動規格中其中兩種的最高價格。對於未指定最高價格的任何啟動規格，則會使用機群請求的最高價格。Spot Fleet 會使用具有最低價格的執行個體類型，來啟動執行個體。

### 可用區域

```
{
 "SpotPrice": "1.00",
 "TargetCapacity": 30,
 "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
 "LaunchSpecifications": [
 {
 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "c3.2xlarge",
 "Placement": {
 "AvailabilityZone": "us-west-2b"
 },
 "SpotPrice": "0.10"
 },
 {
 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "c3.4xlarge",
 "Placement": {
 "AvailabilityZone": "us-west-2b"
 },
 "SpotPrice": "0.20"
 },
 {
 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "c3.8xlarge",
 "Placement": {
 "AvailabilityZone": "us-west-2b"
 }
 }
]
}
```

### 子網

```
{
 "SpotPrice": "1.00",
 "TargetCapacity": 30,
 "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
```

```
"LaunchSpecifications": [
 {
 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "c3.2xlarge",
 "SubnetId": "subnet-1a2b3c4d",
 "SpotPrice": "0.10"
 },
 {
 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "c3.4xlarge",
 "SubnetId": "subnet-1a2b3c4d",
 "SpotPrice": "0.20"
 },
 {
 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "c3.8xlarge",
 "SubnetId": "subnet-1a2b3c4d"
 }
]
```

## 範例 5：使用多樣化分配策略來啟動 Spot Fleet

下列範例使用 `diversified` 分配策略。這些啟動規格各具有不同的執行個體類型，但包含相同的 AMI 和可用區域或子網。Spot Fleet 會將 30 個執行個體分配給 3 種啟動規格，如此每種規格就有 10 個執行個體。如需詳細資訊，請參閱 [競價型執行個體的分配策略](#)。

### 可用區域

```
{
 "SpotPrice": "0.70",
 "TargetCapacity": 30,
 "AllocationStrategy": "diversified",
 "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
 "LaunchSpecifications": [
 {
 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "c4.2xlarge",
 "Placement": {
 "AvailabilityZone": "us-west-2b"
 }
 },
 {
```



```

 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "m3.2xlarge",
 "Placement": {
 "AvailabilityZone": "us-west-2b"
 }
},
{
 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "r3.2xlarge",
 "Placement": {
 "AvailabilityZone": "us-west-2b"
 }
}
]
}

```

## 子網

```

{
 "SpotPrice": "0.70",
 "TargetCapacity": 30,
 "AllocationStrategy": "diversified",
 "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
 "LaunchSpecifications": [
 {
 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "c4.2xlarge",
 "SubnetId": "subnet-1a2b3c4d"
 },
 {
 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "m3.2xlarge",
 "SubnetId": "subnet-1a2b3c4d"
 },
 {
 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "r3.2xlarge",
 "SubnetId": "subnet-1a2b3c4d"
 }
]
}

```

最佳實務為增加 EC2 容量在其中一個可用區域停機時可以履行 Spot 請求的機會。此最佳做法可以讓區域之間多元化。針對此案例，請將可供您使用的各可用區域納入啟動規格之中。不要每次都使用相同的子網，而是使用三個唯一的子網 (各個都映射至不同的區域)。

## 可用區域

```
{
 "SpotPrice": "0.70",
 "TargetCapacity": 30,
 "AllocationStrategy": "diversified",
 "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
 "LaunchSpecifications": [
 {
 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "c4.2xlarge",
 "Placement": {
 "AvailabilityZone": "us-west-2a"
 }
 },
 {
 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "m3.2xlarge",
 "Placement": {
 "AvailabilityZone": "us-west-2b"
 }
 },
 {
 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "r3.2xlarge",
 "Placement": {
 "AvailabilityZone": "us-west-2c"
 }
 }
]
}
```

## 子網

```
{
 "SpotPrice": "0.70",
 "TargetCapacity": 30,
 "AllocationStrategy": "diversified",
 "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
```

```
"LaunchSpecifications": [
 {
 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "c4.2xlarge",
 "SubnetId": "subnet-1a2b3c4d"
 },
 {
 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "m3.2xlarge",
 "SubnetId": "subnet-2a2b3c4d"
 },
 {
 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "r3.2xlarge",
 "SubnetId": "subnet-3a2b3c4d"
 }
]
```

## 範例 6：使用執行個體權重來啟動 Spot Fleet

下列範例使用執行個體權重，這表示價格為每單位小時而非每執行個體小時的價格。每個啟動組態都列出了不同的執行個體類型和不同的權重。Spot Fleet 會使用具有最低每單位小時價格的執行個體類型。Spot Fleet 會將目標容量除以執行個體的權重，來計算出要啟動的 Spot Fleet 數量。如果結果不是整數，則 Spot Fleet 會將其捨入到下一個整數，以便您的機群大小不低於其目標容量。

如果 r3.2xlarge 的請求成功，Spot 會佈建 4 個這類執行個體。將 20 除以 6，會得出總共 3.33 個執行個體，接著再四捨五入為 4 個執行個體。

如果 c3.xlarge 的請求成功，Spot 會佈建 7 個這類執行個體。將 20 除以 3，會得出總共 6.66 個執行個體，接著再四捨五入為 7 個執行個體。

如需詳細資訊，請參閱 [Spot Fleet 執行個體權重](#)。

### 可用區域

```
{
 "SpotPrice": "0.70",
 "TargetCapacity": 20,
 "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
 "LaunchSpecifications": [
 {
 "ImageId": "ami-1a2b3c4d",
```

```

 "InstanceType": "r3.2xlarge",
 "Placement": {
 "AvailabilityZone": "us-west-2b"
 },
 "WeightedCapacity": 6
 },
 {
 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "c3.xlarge",
 "Placement": {
 "AvailabilityZone": "us-west-2b"
 },
 "WeightedCapacity": 3
 }
]
}

```

## 子網

```

{
 "SpotPrice": "0.70",
 "TargetCapacity": 20,
 "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
 "LaunchSpecifications": [
 {
 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "r3.2xlarge",
 "SubnetId": "subnet-1a2b3c4d",
 "WeightedCapacity": 6
 },
 {
 "ImageId": "ami-1a2b3c4d",
 "InstanceType": "c3.xlarge",
 "SubnetId": "subnet-1a2b3c4d",
 "WeightedCapacity": 3
 }
]
}

```

## 範例 7：使用隨需容量來啟動 Spot Fleet

為確保永遠有執行個體容量，您可以在 Spot Fleet 請求中加入對隨需容量的請求。如果有容量，則隨需請求一律已履行。如有容量和可用性，則目標容量的餘額已履行做為 Spot 使用。

下列範例將所要的目標容量指定為 10，其中 5 個必須為隨需容量。未指定 Spot 容量；它暗含在目標容量減去隨需容量的餘額中。如果有可用的 Amazon EC2 容量和可用性，Amazon EC2 會啟動 5 個容量單位作為隨需容量，5 個容量單位 (10-5=5) 作為 Spot 容量。

如需詳細資訊，請參閱 [Spot Fleet 中的隨需](#)。

```
{
 "IamFleetRole": "arn:aws:iam::781603563322:role/aws-ec2-spot-fleet-tagging-role",
 "AllocationStrategy": "lowestPrice",
 "TargetCapacity": 10,
 "SpotPrice": null,
 "ValidFrom": "2018-04-04T15:58:13Z",
 "ValidUntil": "2019-04-04T15:58:13Z",
 "TerminateInstancesWithExpiration": true,
 "LaunchSpecifications": [],
 "Type": "maintain",
 "OnDemandTargetCapacity": 5,
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateId": "lt-0dbb04d4a6cca5ad1",
 "Version": "2"
 },
 "Overrides": [
 {
 "InstanceType": "t2.medium",
 "WeightedCapacity": 1,
 "SubnetId": "subnet-d0dc51fb"
 }
]
 }
]
}
```

## 範例 8：設定容量重新平衡以啟動替代 競價型執行個體

下列範例會設定 Spot Fleet，以便 Amazon EC2 在機群中對 Spot 執行個體發出重新平衡建議時啟動替代 Spot 執行個體。若要為 ReplacementStrategy 設定 競價型執行個體 的自動替代，請指定 launch-before-terminate。若要設定從啟動新替代 Spot 執行個體到自動刪除舊 Spot 執行個體的時間延遲，請針對 termination-delay，指定一值 (以秒為單位)。如需詳細資訊，請參閱 [組態選項](#)。

**Note**

建議僅在您可以預測執行個體關閉程序將需要多長時間才能完成時，才使用 `launch-before-terminate`。這可確保只在關機程序完成之後，才會終止舊執行個體。所有執行個體在執行時，您需要支付所有執行個體的費用。

容量重新平衡策略的有效性取決於 Spot Fleet 請求中指定的 Spot 容量集區數量。我們建議您使用多樣化的執行個體類型和可用區域的集合來設定機群，若要 `AllocationStrategy`，請指定 `capacityOptimized`。如需當設定 Spot Fleet 進行容量重新平衡時應考量哪些項目的詳細資訊，請參閱 [容量重新平衡](#)。

```
{
 "SpotFleetRequestConfig": {
 "AllocationStrategy": "capacityOptimized",
 "IamFleetRole": "arn:aws:iam::000000000000:role/aws-ec2-spot-fleet-tagging-
role",
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "LaunchTemplate",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceType": "c3.large",
 "WeightedCapacity": 1,
 "Placement": {
 "AvailabilityZone": "us-east-1a"
 }
 },
 {
 "InstanceType": "c4.large",
 "WeightedCapacity": 1,
 "Placement": {
 "AvailabilityZone": "us-east-1a"
 }
 },
 {
 "InstanceType": "c5.large",
 "WeightedCapacity": 1,
 "Placement": {
```

```

 "AvailabilityZone": "us-east-1a"
 }
]
 },
 "TargetCapacity": 5,
 "SpotMaintenanceStrategies": {
 "CapacityRebalance": {
 "ReplacementStrategy": "launch-before-terminate",
 "TerminationDelay": "720"
 }
 }
}

```

### 範例 9：在容量最佳化機群中啟動 Spot 執行個體

下列範例示範如何使用可最佳化容量的 Spot 分配策略來設定 Spot Fleet。若要最佳化容量，您必須將 `AllocationStrategy` 設為 `capacityOptimized`。

在下列範例中，三個啟動規格指定三個 Spot 容量集區。目標容量為 50 個 Spot 執行個體。Spot Fleet 嘗試將 50 個 Spot 執行個體啟動到 Spot 容量集區中，且擁有啟動中執行個體數目之最佳容量。

```

{
 "TargetCapacity": "50",
 "SpotFleetRequestConfig": {
 "AllocationStrategy": "capacityOptimized",
 },
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "my-launch-template",
 "Version": "1"
 },
 "Overrides": [
 {
 "InstanceType": "r4.2xlarge",
 "AvailabilityZone": "us-west-2a"
 },
 {
 "InstanceType": "m4.2xlarge",
 "AvailabilityZone": "us-west-2b"
 }
]
 }
]
}

```

```

 },
 {
 "InstanceType": "c5.2xlarge",
 "AvailabilityZone": "us-west-2b"
 }
]
}
]
}
}

```

## 範例 10：使用優先順序在容量最佳化機群中啟動 Spot 執行個體

下列範例示範在盡最大努力使用優先順序時，如何使用可最佳化容量的 Spot 分配策略來設定 Spot Fleet。

當您使用 `capacityOptimizedPrioritized` 分配策略時，您可以使用 `Priority` 參數來指定 Spot 容量集區的優先順序，其中數字越小，優先順序越高。如果您對它們一視同仁，也可以為數個 Spot 容量集區設定相同的優先順序。如果您沒有設定集區的優先順序，集區將被視為最後一個優先順序。

若要排定 Spot 容量集區的優先順序，您必須將 `AllocationStrategy` 設為 `capacityOptimizedPrioritized`。Spot Fleet 會先針對容量進行最佳化，但會盡力遵守優先順序 (例如，如果遵守優先順序不會顯著影響 Spot Fleet 佈建最佳容量的能力)。對於必須將中斷可能性降至最低的工作負載來說，這是一個很好的選擇，而且某些執行個體類型的偏好也很重要。

在下列範例中，三個啟動規格指定三個 Spot 容量集區。每個集區都有優先順序，其中數字越小，優先順序越高。目標容量為 50 個 Spot 執行個體。Spot Fleet 嘗試以最高優先順序在 Spot 容量集區中啟動 50 個 Spot 執行個體，但首先針對容量進行最佳化。

```

{
 "TargetCapacity": "50",
 "SpotFleetRequestConfig": {
 "AllocationStrategy": "capacityOptimizedPrioritized"
 },
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateName": "my-launch-template",
 "Version": "1"
 },
 "Overrides": [
 {

```



```

 "InstanceType": "r4.2xlarge",
 "Priority": 1,
 "AvailabilityZone": "us-west-2a"
 },
 {
 "InstanceType": "m4.2xlarge",
 "Priority": 2,
 "AvailabilityZone": "us-west-2b"
 },
 {
 "InstanceType": "c5.2xlarge",
 "Priority": 3,
 "AvailabilityZone": "us-west-2b"
 }
]
}
]
}

```

### 範例 11：在 priceCapacityOptimized 叢集中啟動 Spot 執行個體

下列範例示範如何使用可最佳化容量且價格最低的 Spot 分配策略來設定 Spot 機群。若要在將價格納入考慮的同時最佳化容量，您必須將 Spot AllocationStrategy 設定為 priceCapacityOptimized。

在下列範例中，三個啟動規格指定三個 Spot 容量集區。目標容量為 50 個 Spot 執行個體。Spot 機群嘗試將 50 個 Spot 執行個體啟動到 Spot 容量集區中，且擁有啟動中執行個體數目之最佳容量，同時還會選擇價格最低的集區。

```

{
 "SpotFleetRequestConfig": {
 "AllocationStrategy": "priceCapacityOptimized",
 "OnDemandAllocationStrategy": "lowestPrice",
 "ExcessCapacityTerminationPolicy": "default",
 "IamFleetRole": "arn:aws:iam::111111111111:role/aws-ec2-spot-fleet-tagging-
role",
 "LaunchTemplateConfigs": [
 {
 "LaunchTemplateSpecification": {
 "LaunchTemplateId": "lt-0123456789example",
 "Version": "1"
 },
 "Overrides": [

```

```

 {
 "InstanceType": "r4.2xlarge",
 "AvailabilityZone": "us-west-2a"
 },
 {
 "InstanceType": "m4.2xlarge",
 "AvailabilityZone": "us-west-2b"
 },
 {
 "InstanceType": "c5.2xlarge",
 "AvailabilityZone": "us-west-2b"
 }
]
},
"TargetCapacity": 50,
"Type": "request"
}
}

```

## 範例 12：設定屬性型執行個體類型選取

下列範例示範如何將 Spot 機群設定為使用屬性型執行個體類型選取來識別執行個體類型。若要指定必要的執行個體屬性，您可以在 InstanceRequirements 結構中指定屬性。

在下列範例中，指定了兩個執行個體屬性：

- VCpuCount – 至少指定 2 個 vCPU。由於未指定最大值，因此沒有最大限制。
- MemoryMiB – 至少指定 4 MiB 的記憶體。由於未指定最大值，因此沒有最大限制。

將會識別任何具有 2 個或更多 vCPU 和 4 MiB 或更多記憶體的執行個體類型。但是，當 [Spot 機群佈建機群](#)時，價格保護和分配策略可能會排除某些執行個體類型。

如需可指定之所有可能屬性的清單和說明，請參閱 Amazon EC2 API 參考 [InstanceRequirements](#) 中的。

```

{
 "AllocationStrategy": "priceCapacityOptimized",
 "TargetCapacity": 20,
 "Type": "request",
 "LaunchTemplateConfigs": [{
 "LaunchTemplateSpecification": {

```

```

 "LaunchTemplateName": "my-launch-template",
 "Version": "1"
 },
 "Overrides": [{
 "InstanceRequirements": {
 "VCpuCount": {
 "Min": 2
 },
 "MemoryMiB": {
 "Min": 4
 }
 }
]
}

```

## 機群配額

通常的 Amazon EC2 配額 (之前稱為限制) 適用於由 EC2 機群或 Spot 機群啟動的執行個體，例如 [Spot 執行個體限制](#) 和 [磁碟區限制](#)。

此外，也適用下列配額：

配額說明	配額
每個區域類型以及 active、deleted_running 和州 request 中的 EC2 叢集 maintain 和競價型叢集數量 cancelled_running	1,000 <sup>1 2 3</sup>
類型的 EC2 叢集數 instant	無限制
EC2 叢集和 Spot 叢集類型和 Spot 叢集的 Spot 容量集區數量 (執行個體類型和子網路的唯一組合) maintain request	300 <sup>1</sup>
EC2 叢集類型的 Spot 容量集區數量 (執行個體類型和子網路的唯一組合) instant	無限制
啟動規格中使用者資料的大小	16 KB <sup>2</sup>

配額說明	配額
每 EC2 機群或 Spot 機群的目標容量	10,000
區域中涵蓋所有 EC2 機群和 Spot 機群的目標容量	100,000 <sup>1</sup>
EC2 Fleet 請求或 Spot Fleet 請求不能跨區域。	
EC2 Fleet 請求或 Spot Fleet 請求不能橫跨同一個可用區域的不同子網。	

<sup>1</sup> 這些配額同時適用於您的 EC2 機群和您的 Spot 機群。

<sup>2</sup> 這些是硬性配額。您不能對這些配額請求提高配額。

<sup>3</sup> 刪除 EC2 機群或取消 Spot 機群請求後，如果您指定該機群不得在刪除或取消請求時終止其 Spot 執行個體，則機群請求會進入 `deleted_running` (EC2 機群) 或 `cancelled_running` (Spot 機群) 狀態，而執行個體會繼續執行直至中斷，或直至您手動將其終止。如果終止執行個體，則機群請求會進入 `deleted_terminating` (EC2 機群) 或 `cancelled_terminating` (Spot 機群) 狀態，而不會計入此配額。如需詳細資訊，請參閱 [刪除 EC2 Fleet](#) 及 [取消 Spot Fleet 請求](#)。

## 請求增加目標容量的配額

如果您需要的容量超過目標容量的預設配額，則可以請求增加配額。

### 請求增加目標容量的配額

1. 開啟「中 AWS Support 心 [建立案例](#)」表單。
2. 選擇提高服務限制。
3. 對於 Limit type (限制類型)，請選擇 EC2 Fleet (EC2 機群)。
4. 在「地區」中，選擇要求提高配額的 AWS 地區。
5. 對於 Limit (限制)，請選擇 Target Fleet Capacity per Fleet (in units) (每個機群的目標機群容量 (單位))，或者 Target Fleet Capacity per Region (in units) (每個區域的目標機群容量 (單位))，具體取決於您要增加的配額。
6. 針對 New limit value (新的限值)，輸入新的配額值。
7. 若要請求增加另一個配額，請選擇 Add another request (新增另一個請求)，然後重複步驟 4–6。

8. 對於 Use case description (使用案例描述)，輸入您請求增加配額的原因。
9. 在 Contact options (聯絡選項)，請指定您偏好的聯絡語言和聯絡方式。
10. 選擇提交。

## 監控 Amazon EC2

監控是維護 Amazon 彈性運算雲端 (Amazon EC2) 執行個體和 AWS 解決方案的可靠性、可用性和效能的重要組成部分。您應該從 AWS 解決方案中的所有零件收集監視資料，以便在發生多點故障時更輕鬆地偵錯。不過，在您開始監控 Amazon EC2 之前，應該建立包含下列項目的監控計畫：

- 監控目標是什麼？
- 要監控哪些資源？
- 監控這些資源的頻率為何？
- 要使用哪些監控工具？
- 誰將執行監控任務？
- 發生問題時應該通知誰？

在您定義監控目標並建立監控計畫之後，下一步是建立您環境中的正常 Amazon EC2 效能基準。您應該在不同的時間及負載條件下測量 Amazon EC2 效能。當您監控 Amazon EC2 時，應該存放所收集的監控資料的歷史紀錄。您可以比較目前的 Amazon EC2 效能資料與歷史資料，協助您辨識正常效能模式和效能異常狀況，並規劃其處理方式。例如，您可以監控 EC2 執行個體的 CPU 使用率、磁碟 I/O 和網路使用率。若效能不符合您所建立的基準，您可能需要重新設定或將執行個體最佳化，以降低 CPU 使用率、改善磁碟 I/O、降低網路流量。

若要建立基準，您至少必須監控下列項目：

要監控的項目	Amazon EC2 指標	監控代理程式/記錄檔 CloudWatch
CPU 使用率	<a href="#">CPUUtilization</a>	
網路使用率	<a href="#">NetworkIn</a> <a href="#">NetworkOut</a>	
磁碟效能	<a href="#">DiskRead</a> 行動 <a href="#">DiskWrite</a> 行動	
磁碟讀取/寫入	<a href="#">DiskRead</a> 位元組	

要監控的項目	Amazon EC2 指標	監控代理程式/記錄檔 CloudWatch
	<a href="#">DiskWrite位元組</a>	
記憶體使用率、磁碟置換使用率、磁碟空間使用率、分頁檔使用率、日誌收集		[Linux 和 Windows 伺服器執行個體] <a href="#">使用 CloudWatch 代理程式從 Amazon EC2 執行個體和現場部署伺服器收集指標和日誌</a>  [從 Windows 伺服器執行個體上的先前 CloudWatch 記錄代理程式移轉] <a href="#">將 Windows 伺服器執行個體記錄收集移轉至 CloudWatch 代理</a>

## 自動和手動監控

AWS 提供各種可用來監控 Amazon EC2 的工具。您可以設定其中一些工具來進行監控，但有些工具需要手動介入。

### 監控工具

- [自動化監控工具](#)
- [手動監控工具](#)

## 自動化監控工具

您可以使用下列自動化監控工具來監看 Amazon EC2，並在發生錯誤時向您回報：

- **系統狀態檢查** — 監控使用執行個體所需的 AWS 系統，以確保它們正常運作。這些檢查會偵測執行個體需要 AWS 參與修復的問題。系統狀態檢查失敗時，您可以選擇等待 AWS 修正問題，也可以自行解決問題 (例如停止並重新啟動，或終止並取代執行個體)。導致系統狀態檢查失敗的問題範例包括：
  - 網路連線中斷
  - 系統電力中斷

- 實體主機的軟體問題
- 實體主機上會影響網路連線的硬體問題

如需詳細資訊，請參閱 [您的執行個體的狀態檢查](#)。

- Instance status checks (執行個體狀態檢查) – 監控個別執行個體的軟體和網路組態。這些檢查會偵測需要您介入修復的問題。一般來說，當執行個體狀態檢查失敗時，您需要自行處理問題 (例如重新啟動執行個體，或在作業系統中進行修改)。可能導致執行個體狀態檢查失敗的問題範例包括：
  - 系統狀態檢查失敗
  - 網路或啟動組態的設定不正確
  - 記憶體用盡
  - 檔案系統毀損
  - 核心不相容

如需詳細資訊，請參閱 [您的執行個體的狀態檢查](#)。

- Amazon CloudWatch 警示 — 觀看指定時段內的單一指標，並根據指定臨界值在多個時段內相對於指定閾值的指標值執行一或多個動作。此動作是傳送到 Amazon Simple Notification Service (Amazon SNS) 主題或 Amazon EC2 Auto Scaling 政策的通知。警示只會呼叫持續狀態變更的動作。CloudWatch 警示不會僅因為處於特定狀態而叫用動作；狀態必須已變更並維持指定數目的期間。如需詳細資訊，請參閱 [使用監控執行個體 CloudWatch](#)。
- Amazon EventBridge — 自動化您的 AWS 服務並自動回應系統事件。AWS 服務中的事件會以近乎即時 EventBridge 的方式傳送到，您可以指定當事件符合您撰寫的規則時要採取的自動化動作。如需詳細資訊，請參閱 [什麼是 Amazon EventBridge ?](#)。
- Amazon CloudWatch 日誌 — 從 Amazon EC2 執行個體或其他來源監控 AWS CloudTrail、存放和存取您的日誌檔。如需詳細資訊，請參閱 [Amazon CloudWatch 日誌使用者指南](#)。
- CloudWatch 代理程式 — 從 EC2 執行個體和現場部署伺服器上的主機和來賓收集日誌和系統層級指標。如需詳細資訊，請參閱 Amazon 使用者指南中的使用 [CloudWatch 代理程式從 Amazon EC2 執行個體和現場部署伺服器收集 CloudWatch 指標和日誌](#)。

## 手動監控工具

監控 Amazon EC2 的另一個重要部分是手動監控監控指令碼、狀態檢查和 CloudWatch 警示未涵蓋的項目。Amazon EC2 和 CloudWatch 主控台儀表板可讓您 at-a-glance 檢視 Amazon EC2 環境的狀態。

- Amazon EC2 儀表板會顯示：



- 依區域的服務運作狀態和已排程事件
- 執行個體狀態
- 狀態檢查
- 警示狀態
- 執行個體指標詳細資訊 (在導覽窗格中，選擇 Instances (執行個體)，選取某個執行個體，然後選擇 Monitoring (監控) 標籤)
- 磁碟區指標詳細資訊 (在導覽窗格中，選擇 Volumes (磁碟區)，選取某個磁碟區，然後選擇 Monitoring (監控) 標籤)
- Amazon CloudWatch 儀表板顯示：
  - 目前警示與狀態
  - 警示與資源的圖表
  - 服務運作狀態

此外，您可以使用執行 CloudWatch 以下操作：

- 繪製 Amazon EC2 監控資料圖表，以便對問題進行故障診斷並探索趨勢
- 搜尋並瀏覽所有資 AWS 源指標
- 建立與編輯要通知發生問題的警示
- 查看警報和 AWS 資源的 at-a-glance 概述

## 監控最佳實務

使用下列監控最佳實務，協助您執行 Amazon EC2 監控任務。

- 讓監控優先攔截小問題，以免問題越形嚴重。
- 建立並實作監視計劃，以收集 AWS 解決方案中所有部分的監視資料，以便在發生多點失敗時，您可以更輕鬆地偵錯多點失敗。您的監控計畫應該至少處理下列問題：
  - 監控目標是什麼？
  - 要監控哪些資源？
  - 監控這些資源的頻率為何？
  - 要使用哪些監控工具？
  - 誰將執行監控任務？

- 盡可能自動化監控任務。
- 查看 EC2 執行個體上的日誌檔案。

## 監控您的執行個體狀態

您可以透過檢視您執行個體的状态檢查和排程事件，監控您執行個體的状态。

状态檢查可為您提供由 Amazon EC2 進行之自動化檢查的結果資訊。這些自動化檢查會偵測是否有特定問題正在影響您的執行個體。状态檢查資訊以及 Amazon 提供的資料 CloudWatch，為您提供每個執行個體的詳細操作能見度。

您也可以查看為您的執行個體排程之特定事件的状态。事件的状态提供為您的執行個體計畫之近期活動的相關資訊，例如重新開機或淘汰。也提供每個事件的排程開始時間和結束時間。

### 內容

- [您的執行個體的状态檢查](#)
- [執行個體的状态變更事件](#)
- [您的執行個體的排程事件](#)

## 您的執行個體的状态檢查

透過執行個體状态監控，您可以快速判斷 Amazon EC2 是否已偵測到任何可能會防止執行個體執行應用程式的問題。Amazon EC2 會在每一次執行 EC2 執行個體時執行自動化檢查，以識別硬體和軟體問題。您可以檢視這些状态檢查的結果，以找出特定及可偵測的問題。事件状态資料增強了 Amazon EC2 已提供的每個執行個體状态 (例如 pending、running、stopping) 以及 Amazon CloudWatch 監控的使用率指標 (CPU 使用率、網路流量和磁碟活動) 的資訊。

状态檢查會每分鐘執行一次，並會傳回通過或失敗状态。如果所有檢查都通過，執行個體的整體状态即為 OK (正常)。若一或多個檢查失敗，整體状态便會是 impaired (故障)。状态檢查內建於 Amazon EC2 中，因此無法停用或刪除。

當状态檢查失敗時，状态檢查的對應 CloudWatch 測量結果就會增加。如需詳細資訊，請參閱 [状态檢查指標](#)。您可以使用這些指標來建立根據状态檢查結果觸發的 CloudWatch 警示。例如，您可以建立警示，在状态檢查於特定執行個體上失敗時警告您。如需詳細資訊，請參閱 [建立和編輯状态檢查警示](#)。

您也可以建立 Amazon CloudWatch 警示來監控 Amazon EC2 執行個體，並在執行個體因潛在問題而受損時自動復原該執行個體。如需詳細資訊，請參閱 [執行個體恢復](#)。

## 內容

- [狀態檢查類型](#)
- [使用狀態檢查](#)

## 狀態檢查類型

有三種類型的狀態檢查。

- [系統狀態檢查](#)
- [執行個體狀態檢查](#)
- [附接的 EBS 狀態檢查](#)

### 系統狀態檢查

系統狀態檢查會監控執行個體執行的 AWS 系統。這些檢查會偵測您執行個體需要 AWS 介入修復的基礎問題。系統狀態檢查失敗時，您可以選擇等 AWS 待修正問題，也可以自行解決。針對 Amazon EBS 後端執行個體，您可以自行停止並啟動執行個體，多數情況下，這會將它遷移到新的主機。針對執行個體存放區支援的 Linux 執行個體，您可以終止並取代執行個體。針對 Windows 執行個體，根磁碟區必須是 Amazon EBS 磁碟區；根磁碟區不支援執行個體存放區。請注意，執行個體存放區是暫時的，當執行個體停止時，所有資料都會丟失。

下列為可能導致系統狀態檢查失敗的問題範例：

- 網路連線中斷
- 系統電力中斷
- 實體主機的軟體問題
- 實體主機上會影響網路連線的硬體問題

如果系統狀態檢查失敗，我們會遞增 [StatusCheckFailed\\_](#) System 量度。

### 裸機執行個體

如果您在裸機執行個體上從作業系統執行重新啟動，則系統狀態檢查可能會暫時傳回失敗狀態。當執行個體變得可用時，系統狀態檢查應傳回通過狀態。

## 執行個體狀態檢查

Instance status checks (執行個體狀態檢查) 監控個別執行個體的軟體和網路組態。Amazon EC2 會將地址解析協定 (ARP) 請求傳送至網路介面 (NIC)，以便檢查執行個體的運作狀態。這些檢查會偵測需要您介入修復的問題。當執行個體狀態檢查失敗時，通常您需要自行處理問題 (例如重新開機執行個體或對執行個體的組態進行變更)。

### Note

最近用systemd-networkd於網路配置的 Linux 發行版可能會與早期發行版不同，報告健康狀態檢查。在開機程序期間，這種類型的網路可能會提前啟動，並且可能會在其他也會影響執行個體健康狀態的啟動工作之前完成。依賴網路可用性的狀態檢查可在其他工作完成之前報告狀況良好的狀態。

下列為可能導致執行個體狀態檢查失敗的問題範例：

- 系統狀態檢查失敗
- 網路或啟動組態不正確
- 記憶體用盡
- 檔案系統毀損
- 核心不相容
- [Windows 執行個體] 在執行個體重新啟動期間或搭配 Windows 執行個體存放區支援的執行個體時，執行個體狀態檢查會報告失敗，直到執行個體再次可用為止。

如果執行個體狀態檢查失敗，我們會增加 [StatusCheckFailed](#) Instance 量度。

### 裸機執行個體

如果您在裸機執行個體上從作業系統執行重新啟動，則執行個體狀態檢查可能會暫時傳回失敗狀態。當執行個體變得可用時，執行個體狀態檢查應傳回通過狀態。

### 附接的 EBS 狀態檢查

附接的 EBS 狀態檢查可監控附接至執行個體的 Amazon EBS 磁碟區是否可連線且能夠完成 I/O 操作。此 StatusCheckFailed\_AttachedEBS 指標是二進位值，如果附接至執行個體的一個或多個 EBS 磁碟區無法完成 I/O 操作，則表示損壞。這些狀態檢查會對運算或 Amazon EBS 基礎設施的潛在問題

進行偵測。當連接的 EBS 狀態檢查指標失敗時，您可以等待 AWS 解決問題，也可以採取動作，例如更換受影響的磁碟區或停止並重新啟動執行個體。

以下是可能導致附接的 EBS 狀態檢查失敗的問題範例：

- EBS 磁碟區之下儲存子系統上的硬體或軟體問題
- 實體主機上會影響 EBS 磁碟區連線的硬體問題
- 執行個體與 EBS 磁碟區之間的連線問題

可以使用 `StatusCheckFailed_AttachedEBS` 指標來協助改善工作負載的彈性。您可以使用此指標建立根據狀態檢查結果觸發的 Amazon CloudWatch 警示。例如，偵測到長期影響時，可容錯移轉至次要執行個體或可用區域。或者，您可以使用 EBS CloudWatch 指標來監控每個連接磁碟區的 I/O 效能，以偵測並更換受損的磁碟區。如果您的工作負載沒有將 I/O 驅動到執行個體所附接的任何 EBS 磁碟區，並且附接的 EBS 狀態檢查顯示有損壞，則可以停止並啟動執行個體，以解決影響 EBS 磁碟區連線能力之實體主機的問題。如需詳細資訊，請參閱 [Amazon EBS 的 Amazon CloudWatch 指標](#)

#### Note

- 附接的 EBS 狀態檢查指標僅適用於 Nitro 執行個體。
- 您可以根據指標 [建立 CloudWatch 警示](#)，來監視連接的 EBS 狀態檢查 `StatusCheckFailed_AttachedEBS` 量度。您無法使用 [描述執行個體](#) AWS CLI 狀態命令來檢視此狀態檢查。

## 使用狀態檢查

可以使用主控台和命令列工具來處理狀態檢查，例如 AWS CLI。

### 主題

- [檢視狀態檢查](#)
- [建立和編輯狀態檢查警示](#)

### 檢視狀態檢查

若要檢視狀態檢查，請使用以下方法之一。

## Console

### 檢視狀態檢查

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 在執行個體頁面上，狀態檢查資料欄會列出每個執行個體的操作狀態。
4. 若要檢視特定執行個體的状态，請選取執行個體，然後選擇狀態和警示索引標籤。

<input type="checkbox"/>	Name <input type="text"/>	Instance ID	Instance state <input type="text"/>	Instance type <input type="text"/>	Status check	Alarm status	Availi
<input checked="" type="checkbox"/>	spot-instance-2	i-01aeed690c9fb5322	<span style="color: green;">✔</span> Running	t3.nano	<span style="color: orange;">⚠</span> 1/2 checks ...	<a href="#">View alarms +</a>	eu-w
<input type="checkbox"/>	spot-instance-1	i-0ba5e5bbc9d634fa6	<span style="color: gray;">⊖</span> Stopped	t3.nano	-	<a href="#">View alarms +</a>	eu-w
<input type="checkbox"/>	EIC-RHEL	i-08e66e73da739c7f4	<span style="color: green;">✔</span> Running	t2.micro	<span style="color: green;">✔</span> 2/2 checks passed	<a href="#">View alarms +</a>	eu-w
<input type="checkbox"/>	Windows	i-0cb952751a0d8388b	<span style="color: green;">✔</span> Running	t3.nano	<span style="color: green;">✔</span> 2/2 checks passed	<a href="#">View alarms +</a>	eu-w

**Instance: i-01aeed690c9fb5322 (spot-instance-2)**

Details | **Status and alarms New** | Monitoring | Security | Networking | Storage | Tags

**Status checks Info**

Status checks detect problems that may impair i-01aeed690c9fb5322 (spot-instance-2) from running your applications.

System status checks

✔ System reachability check passed

▶ Metrics

▼ Alarms

Instance status checks

⊗ Instance reachability check failed

Check failure at

2020/12/16 17:30 GMT+2 (about 1 month)

Name	State	Description	Metric name	State reason
Instance has no associated alarms				

如果執行個體狀態檢查失敗，通常就需要您自行處理問題 (例如重新啟動執行個體或對執行個體的組態進行變更)。若要排解 Linux 執行個體上的系統或執行個體狀態檢查失敗，請參閱[使用失敗狀態檢查進行 Linux 執行個體](#)。

5. 若要檢閱狀態檢查的 CloudWatch 量度，請在狀態與警示索引標籤上展開量度以查看下列量度的圖表：
  - 系統狀態檢查失敗
  - 執行個體狀態檢查失敗

如需詳細資訊，請參閱 [the section called “狀態檢查指標”](#)。

## Command line

您可以使用 [describe-instance-status](#) (AWS CLI) 命令，檢視執行中執行個體的狀態檢查。

若要檢視所有執行個體的状态，請使用下列命令。

```
aws ec2 describe-instance-status
```

若要取得執行個體狀態為 `impaired` 的所有執行個體狀態，請使用下列命令。

```
aws ec2 describe-instance-status \
 --filters Name=instance-status.status,Values=impaired
```

若要取得單一執行個體的状态，請使用下列命令。

```
aws ec2 describe-instance-status \
 --instance-ids i-1234567890abcdef0
```

或者，使用下列命令：

- [Get-EC2InstanceStatus](#) (AWS Tools for Windows PowerShell)
- [DescribeInstance狀態](#) (Amazon EC2 查詢 API)

如果您的 Linux 執行個體具有失敗狀態檢查，請參閱 [使用失敗狀態檢查進行 Linux 執行個體](#)。

## 建立和編輯狀態檢查警示

您可以使用 [狀態檢查測量結果](#) 來建立 CloudWatch 警示，以便在執行處理狀態檢查失敗時通知您。

### Important

如果缺少指標資料點，狀態檢查和狀態檢查警示可能會暫時進入不足的資料狀態。雖然很少見，但是當指標報告系統發生中斷時，即使執行個體健康狀態良好，也可能發生這種情況。我們建議您將此狀態視為遺失資料，而非狀態檢查失敗或警示洩漏，尤其是在執行個體上執行停止、終止、重新開機或復原動作以回應時。

若要建立狀態檢查警示，請使用以下方法之一：

## Console

使用下列程序來設定警示，在執行個體的狀態檢查失敗時，透過電子郵件通知您、或者停止、終止或回復執行個體。

### 建立狀態檢查警示

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體，選擇狀態檢查標籤，然後選擇動作 > 建立狀態檢查警示。
4. 在 [管理 CloudWatch 鬧鐘] 頁面的 [新增或編輯鬧鐘] 底下，選擇 [建立鬧鐘]。
5. 針對警示通知，開啟切換開關以設定 Amazon Simple Notification Service (Amazon SNS) 通知。選取現有 Amazon SNS 主題或輸入名稱以建立新主題。

若您將電子郵件地址新增到收件人清單中，或是建立了新的主題，Amazon SNS 會傳送訂閱確認電子郵件訊息給每個新的地址。每個收件人都必須選擇該訊息中包含的連結以確認訂閱。僅會寄送提醒通知至確認過的地址。

6. 針對警示動作，開啟切換開關，以指定觸發警示時要採取的動作。選取動作。
7. 針對 Alarm thresholds (警示閾值)，指定警示的指標和準則。

您可以保留 Group samples by (分組範例條件) (Average (平均)) 和 Type of data to sample (要作為範例的資料類型) (Status check failed:either (狀態檢查失敗：其中一個)) 的預設設定，或者您可以根據您的需求進行變更。

針對連續期間，設定要評估的期間數，然後在期間中，輸入在觸發提醒及傳送電子郵件前的評估期間長度。

8. (選用) 針對 Sample metric data (範例指標資料)，選擇 Add to dashboard (新增至儀表板)。
9. 選擇建立。

若您需要對執行個體狀態警示進行任何變更，您可以編輯它。

### 編輯狀態檢查警示

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體並選擇 [動作]、[監控]、[管理 CloudWatch 警示]
4. 在「管理 CloudWatch 鬧鐘」頁面的「新增或編輯鬧鐘」下，選擇「編輯鬧鐘」。



5. 針對 Search for alarm (搜尋警示)，選擇警示。
6. 完成變更後，請選擇 Update (更新)。

## Command line

在下列範例中，當執行個體的執行個體檢查或系統檢查在至少兩個連續期間中失敗時，警示便會發佈通知到 SNS 主題，arn:aws:sns:us-west-2:111122223333:my-sns-topic。使用的 CloudWatch 量度為 StatusCheckFailed。

若要使用建立狀態檢查警示 AWS CLI

1. 選取現有的 SNS 主題，或建立新主題。如需詳細資訊，請參閱[使 AWS CLI 用 AWS Command Line Interface 者指南中的搭配 Amazon SNS 使用](#)。
2. 使用下列清單指標命令來檢視適用於 Amazon EC2 的可用 Amazon CloudWatch 指標。

```
aws cloudwatch list-metrics --namespace AWS/EC2
```

3. 使用下列 [put-metric-alarm](#) 命令建立警示。

```
aws cloudwatch put-metric-alarm \
 --alarm-name StatusCheckFailed-Alarm-for-i-1234567890abcdef0 \
 --metric-name StatusCheckFailed \
 --namespace AWS/EC2 \
 --statistic Maximum \
 --dimensions Name=InstanceId,Value=i-1234567890abcdef0 \
 --unit Count \
 --period 300 \
 --evaluation-periods 2 \
 --threshold 1 \
 --comparison-operator GreaterThanOrEqualToThreshold \
 --alarm-actions arn:aws:sns:us-west-2:111122223333:my-sns-topic
```

此期間是收集 Amazon CloudWatch 指標的時間範圍 (以秒為單位)。此範例使用 300，即 60 秒乘以 5 分鐘。評估期間為針對指標的值與閾值進行比較的連續期間數。此範例使用 2。警示動作為觸發此警示時要執行的動作。此範例會設定警示，使其使用 Amazon SNS 傳送電子郵件。

## 執行個體的狀態變更事件

Amazon EC2 會在執行 EC2 Instance State-change Notification 個體狀態變更 EventBridge 時傳送事件至 Amazon。

以下是此事件的範例資料。在本範例中，執行個體進入 pending 狀態。

```
{
 "id": "7bf73129-1428-4cd3-a780-95db273d1602",
 "detail-type": "EC2 Instance State-change Notification",
 "source": "aws.ec2",
 "account": "123456789012",
 "time": "2021-11-11T21:29:54Z",
 "region": "us-east-1",
 "resources": [
 "arn:aws:ec2:us-east-1:123456789012:instance/i-abcd1111"
],
 "detail": {
 "instance-id": "i-abcd1111",
 "state": "pending"
 }
}
```

state 的可能值為：

- pending
- running
- stopping
- stopped
- shutting-down
- terminated

啟動或開始執行個體時，其會變成 pending 狀態，然後再變成 running 狀態。停止執行個體時，其會變成 stopping 狀態，然後再變成 stopped 狀態。終止執行個體時，其會變成 shutting-down 狀態，然後再變成 terminated 狀態。

## 當執行個體變更狀態時收到電子郵件通知

若要在執行個體狀態變更時接收電子郵件通知，請建立 Amazon SNS 主題，然後為 EC2 Instance State-change Notification 事件建立 EventBridge 規則。

### 建立 SNS 主題

1. 在 <https://console.aws.amazon.com/sns/v3/home> 開啟 Amazon SNS 主控台。
2. 在導覽窗格中，選擇主題。
3. 請選擇建立主題。
4. 針對類型，選擇標準。
5. 在 Name (名稱) 中，輸入主題名稱。
6. 請選擇建立主題。
7. 選擇建立訂閱。
8. 對於通訊協定，選擇電子郵件。
9. 在 Endpoint (端點) 中，輸入接收通知的電子郵件地址。
10. 選擇建立訂閱。
11. 您會收到帶有下列主旨行的電子郵件訊息：AWS Notification - Subscription Confirmation。請依照指示來確認訂閱。

### 若要建立 EventBridge 規則

1. 在以下位置打開 Amazon EventBridge 控制台 <https://console.aws.amazon.com/events/>。
2. 選擇建立規則。
3. 在 Name (名稱) 中，輸入規則名稱。
4. 針對規則類型，選擇具有事件模式的規則。
5. 選擇下一步。
6. 針對 Event pattern (事件模式)，請執行下列動作：
  - a. 在 Event source (事件來源)，選擇 AWS 服務。
  - b. 針對 AWS 服務，選擇 EC2。
  - c. 在 Event Type (事件類型) 中，選擇 EC2 Instance State-change Notification (EC2 執行個體狀態變更通知)。

- d. 根據預設，我們會傳送任何執行個體的任何狀態變更通知。如果願意，可以選擇特定狀態或特定執行個體。
7. 選擇下一步。
  8. 如下所示指定目標：
    - a. 對於 Target types (目標類型)，選擇 AWS 服務。
    - b. 對於 Select a target (選取目標)，選擇 SNS topic (SNS 主題)。
    - c. 針對 Topic (主題)，選擇您在先前程序中建立的 SNS 主題。
  9. 選擇下一步。
  10. (選用) 將標籤新增至您的規則。
  11. 選擇下一步。
  12. 選擇建立規則。
  13. 為了測試您的規則，請啟動狀態變更。例如，開始已停止的執行個體、停止執行中的執行個體或啟動執行個體。您會收到帶有下列主旨行的電子郵件訊息：AWS Notification Message。電子郵件內文包含事件資料。

## 您的執行個體的排程事件

AWS 可以為您的執行個體排程事件，例如重新開機、停止/啟動或淘汰。這些事件不會頻繁發生。如果您的其中一個執行個體會受到排程事件的影響，請在排定的事件發生前，將電子郵件 AWS 傳送至與您 AWS 帳戶相關聯的電子郵件地址。此電子郵件提供事件的詳細資訊，包括開始和結束日期。視事件而定，您可能可以採取行動來控制事件的時間。AWS 還會傳送 AWS Health 事件，您可以使用 Amazon 事件監控和管理 CloudWatch 事件。如需有關使用監視 AWS Health 事件的詳細資訊 CloudWatch，請參閱[使用 AWS Health 事件監視 CloudWatch 事件](#)。

排程事件由管理 AWS；您無法為執行個體排程事件。您可以檢視排程的事件 AWS、自訂排定的事件通知以包含或移除電子郵件通知中的標籤，以及在執行個體排定為重新開機、淘汰或停止時執行動作。

若要更新您帳戶的聯絡資訊，確保您會收到排程事件的通知，請前往[帳戶設定](#)頁面。

### Note

當執行個體受到排程事件的影響，並且屬於 Auto Scaling 群組時，Amazon EC2 Auto Scaling 最終會在運作狀態檢查期間將其取代，而您無需採取任何進一步動作。如需 Amazon EC2 Auto

Scaling 執行運作狀態檢查的詳細資訊，請參閱《Amazon EC2 Auto Scaling 使用者指南》中的 [Auto Scaling 執行個體的運作狀態檢查](#)。

## 目錄

- [排程事件類型](#)
- [檢視排程事件](#)
- [自訂排程的事件通知](#)
- [使用已排程停止或淘汰的執行個體](#)
- [使用已排程重新開機的執行個體](#)
- [使用已排程維護的執行個體](#)
- [重新安排排定的事件](#)
- [定義排程事件的事件時段](#)

## 排程事件類型

Amazon EC2 可建立以下類型的執行個體事件，其中事件會在排程的時間發生：

- Instance stop (執行個體停止)：在排程時間，執行個體停止。當您再次啟動時，它會遷移到新的主機。只適用於 Amazon EBS 後端執行個體。
- Instance retirement (執行個體淘汰)：在排程時間，如果執行個體由 Amazon EBS 支援，會將其停止；若由執行個體存放區支援，則會將其終止。
- Instance reboot (執行個體重新開機)：在排程時間，執行個體重新開機。
- System reboot (系統重新開機)：在排程時間，執行個體的主機重新開機。
- System maintenance (系統維護)：在排程時間，執行個體可能會因網路維護或電源維護而暫時受到影響。

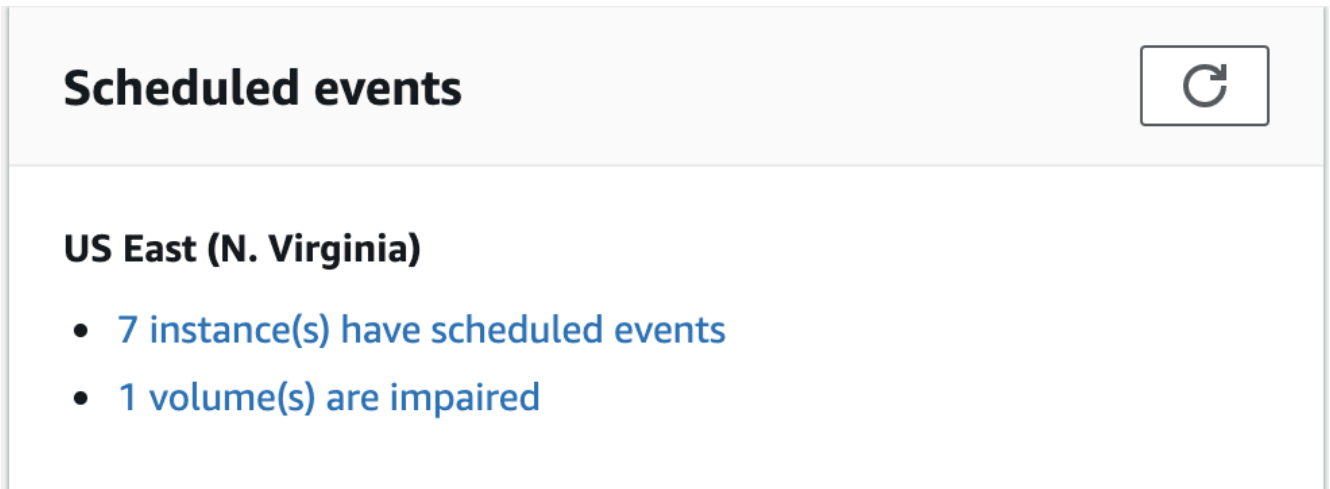
## 檢視排程事件

除了在電子郵件中接收排程事件的通知，您可以使用以下其中一個方法檢查排程事件。

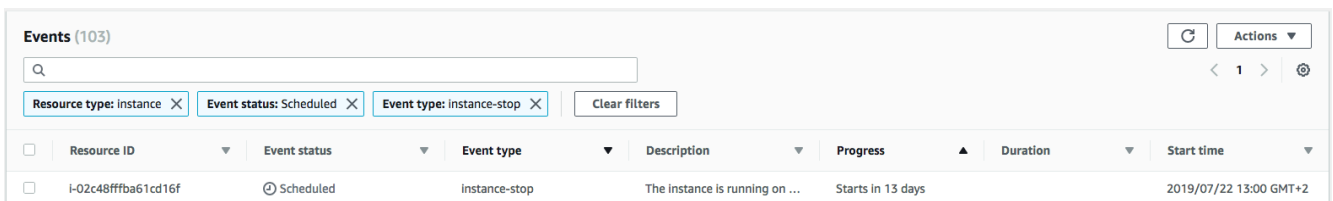
## Console

### 檢視執行個體的排程事件

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 儀表板會在排程事件下方顯示任何與事件相關聯的資源。



3. 如需更多詳細資訊，在導覽窗格中選擇事件。此時會顯示任何與事件相關聯的資源。您可以依特性 (例如事件類型、資源類型和可用區域) 進行篩選。



## AWS CLI

### 檢視執行個體的排程事件

使用 [describe-instance-status](#) 命令。

```
aws ec2 describe-instance-status \
 --instance-id i-1234567890abcdef0 \
 --query "InstanceStatuses[[]].Events"
```

下列範例輸出顯示重新開機事件。

```
[
 "Events": [
```

```

 {
 "InstanceEventId": "instance-event-0d59937288b749b32",
 "Code": "system-reboot",
 "Description": "The instance is scheduled for a reboot",
 "NotAfter": "2019-03-15T22:00:00.000Z",
 "NotBefore": "2019-03-14T20:00:00.000Z",
 "NotBeforeDeadline": "2019-04-05T11:00:00.000Z"
 }
]
]

```

下列為顯示執行個體淘汰事件的範例輸出。

```

[
 "Events": [
 {
 "InstanceEventId": "instance-event-0e439355b779n26",
 "Code": "instance-stop",
 "Description": "The instance is running on degraded hardware",
 "NotBefore": "2015-05-23T00:00:00.000Z"
 }
]
]

```

## PowerShell

使用 AWS Tools for Windows PowerShell 檢視您執行個體的排程事件

使用下列 [Get-EC2InstanceStatus](#) 命令。

```
PS C:\> (Get-EC2InstanceStatus -InstanceId i-1234567890abcdef0).Events
```

下列為顯示執行個體淘汰事件的範例輸出。

```

Code : instance-stop
Description : The instance is running on degraded hardware
NotBefore : 5/23/2015 12:00:00 AM

```

## Instance metadata

使用執行個體中繼資料檢視您執行個體的排程事件

您可以使用執行個體中繼資料服務第 2 版或執行個體中繼資料服務第 1 版，從[執行個體中繼資料](#)擷取有關執行個體之作用中維護事件的資訊。

## IMDSV2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/events/maintenance/scheduled
```

## IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/events/maintenance/scheduled
```

以下範例輸出包含排程系統重新開機事件的資訊 (採用 JSON 格式)。

```
[
 {
 "NotBefore" : "21 Jan 2019 09:00:43 GMT",
 "Code" : "system-reboot",
 "Description" : "scheduled reboot",
 "EventId" : "instance-event-0d59937288b749b32",
 "NotAfter" : "21 Jan 2019 09:17:23 GMT",
 "State" : "active"
 }
]
```

使用執行個體中繼資料檢視執行個體已完成或已取消事件的事件歷史記錄

您可以使用執行個體中繼資料服務第 2 版或執行個體中繼資料服務第 1 版，從[執行個體中繼資料](#)擷取有關已完成或已取消之執行個體事件的資訊。

## IMDSV2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/events/maintenance/history
```

## IMDSv1



```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/events/maintenance/history
```

以下範例輸出包含已取消之系統重新開機事件和已完成之系統重新開機事件的資訊 (採用 JSON 格式)。

```
[
 {
 "NotBefore" : "21 Jan 2019 09:00:43 GMT",
 "Code" : "system-reboot",
 "Description" : "[Canceled] scheduled reboot",
 "EventId" : "instance-event-0d59937288b749b32",
 "NotAfter" : "21 Jan 2019 09:17:23 GMT",
 "State" : "canceled"
 },
 {
 "NotBefore" : "29 Jan 2019 09:00:43 GMT",
 "Code" : "system-reboot",
 "Description" : "[Completed] scheduled reboot",
 "EventId" : "instance-event-0d59937288b749b32",
 "NotAfter" : "29 Jan 2019 09:17:23 GMT",
 "State" : "completed"
 }
]
```

## AWS Health

您可以使用 AWS Health Dashboard 來瞭解可能會影響執行個體的事件。會將 AWS Health Dashboard 問題分為三個群組：未結問題、排程變更及其他通知。排定的變更包含進行中或即將進行的項目。

如需詳細資訊，請參閱《指南》中的《AWS Health [AWS Health Dashboard 入門](#)》。

## 自訂排程的事件通知

您可以自訂排程事件通知，以便在電子郵件通知中包含標籤。如此可以更輕鬆地識別受影響的資源 (執行個體或專用執行個體)，並為即將到來的事件排定動作的優先順序。

當您自訂事件通知以包含標籤時，您可以選擇包含：

- 與受影響資源關聯的所有標籤

- 僅與受影響資源關聯的特定標籤

例如，假設您將 `application`、`costcenter`、`project`，和 `owner` 標籤指派給所有執行個體。您可以選擇在事件通知中包含所有標籤。或者，如果您只想在事件通知中看到 `owner` 和 `project` 標籤，則可以選擇只包含這些標籤。

選取要包含的標籤之後，事件通知將包含資源 ID (執行個體 ID 或 專用執行個體 ID)，以及與受影響資源關聯的標籤金鑰和值組。

## 任務

- [在事件通知中包含標籤](#)
- [從事件通知中移除標籤](#)
- [檢視要包含在事件通知中的標籤](#)

## 在事件通知中包含標籤

您選擇要包含的標籤會套用至所選區域的所有資源 (執行個體和 專用執行個體)。若要自訂其他區域中的事件通知，請先選取所需的區域，然後執行下列步驟。

您可以使用下列其中一種方法在事件通知中包含標籤。

## Console

### 在事件通知中包含標籤

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Events (事件)。
3. 選擇 Actions (動作)、Manage event notifications (管理事件通知)。
4. 開啟在事件通知中包含標籤。
5. 根據您要包含在事件通知中的標籤，執行下列其中一項動作：
  - 若要包含與受影響執行個體或專用執行個體相關聯的所有標籤，請選取包含所有資源標籤。
  - 如需選取要包含的標籤，請選取選擇要包含的標籤，然後選取或輸入標籤索引鍵。
6. 選擇 Save (儲存)。

## AWS CLI

在事件通知中包含所有標籤

使用 [register-instance-event-notification-attributes](#) AWS CLI 命令並設 `IncludeAllTagsOfInstance` 參數為 `true`。

```
aws ec2 register-instance-event-notification-attributes \
 --instance-tag-attribute "IncludeAllTagsOfInstance=true"
```

在事件通知中包含特定標籤

使用 [register-instance-event-notification-attributes](#) AWS CLI 命令，並使用 `InstanceTagKeys` 參數指定要包含的標籤。

```
aws ec2 register-instance-event-notification-attributes \
 --instance-tag-attribute 'InstanceTagKeys=["tag_key_1", "tag_key_2",
 "tag_key_3"]'
```

### 從事件通知中移除標籤

您可以使用下列其中一種方法將標籤從事件通知中移除。

#### Console

從事件通知中移除標籤

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Events (事件)。
3. 選擇 Actions (動作)、Manage event notifications (管理事件通知)。
4. 若要從事件通知中移除所有標籤，請關閉在事件通知中包含資源標籤。
5. 若要從事件通知中移除特定標籤，請為對應的標籤索引鍵選擇 X)。
6. 選擇 Save (儲存)。

## AWS CLI

從事件通知中移除所有標籤

使用 [deregister-instance-event-notification-attributes](#) AWS CLI 命令，並設 `IncludeAllTagsOfInstance` 參數為 `false`。

```
aws ec2 deregister-instance-event-notification-attributes \
 --instance-tag-attribute "IncludeAllTagsOfInstance=false"
```

從事件通知中移除特定標籤

使用 [deregister-instance-event-notification-attributes](#) 指 AWS CLI 令，並使用 `InstanceTagKeys` 參數指定要移除的標籤。

```
aws ec2 deregister-instance-event-notification-attributes \
 --instance-tag-attribute 'InstanceTagKeys=["tag_key_1", "tag_key_2",
 "tag_key_3"]'
```

檢視要包含在事件通知中的標籤

您可以使用下列其中一種方法來檢視要包含在事件通知中的標籤。

Console

檢視要包含在事件通知中的標籤

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Events (事件)。
3. 選擇 Actions (動作)、Manage event notifications (管理事件通知)。

AWS CLI

檢視要包含在事件通知中的標籤

使用 [describe-instance-event-notification-屬性](#) AWS CLI 命令。

```
aws ec2 describe-instance-event-notification-attributes
```

## 使用已排程停止或淘汰的執行個體

當 AWS 偵測到執行個體的基礎主機無法修復的故障時，會根據執行個體的根裝置類型，排程執行個體停止或終止執行個體。若根設備為 EBS 磁碟區，則執行個體會排程停止。若根設備為執行個體存放區磁碟區，則執行個體會排程終止。如需詳細資訊，請參閱 [執行個體淘汰](#)。

### Important

任何存放在執行個體存放區磁碟區的資料都會在執行個體停止、休眠或終止時遺失。這包含連接到根設備為 EBS 磁碟區之執行個體的執行個體存放區磁碟區。請確認您已在執行個體停止、休眠或終止前，從您的執行個體存放區磁碟區儲存您稍後需要的資料。

### Amazon EBS 後端執行個體動作

您可以等待執行個體依照排程停止。或者，您也可以自行停止並啟動執行個體，這會將它遷移到新的主機。如需停止執行個體的詳細資訊 (除了在停止時執行個體組態有何變更的資訊)，請參閱 [停止並啟動 Amazon EC2 執行個體](#)。

您可以將立即停止並啟動的作業自動化，以回應排程的執行個體停止事件。如需詳細資訊，請參閱 AWS Health 使用者指南中的 [自動化 Amazon EC2 執行個體的動作](#)。

### 執行個體存放區後端執行個體的動作

我們建議您在執行個體排程終止之前，從您最近的 AMI 啟動取代用執行個體，並將所有必要的資料遷移到取代用執行個體。接著，您便可以終止原始執行個體，或是等待執行個體依照排程終止。

## 使用已排程重新開機的執行個體

當 AWS 必須執行諸如安裝更新或維護基礎主機之類的工作時，它可以排程執行個體或基礎主機進行重新開機。您可以 [重新排程大部分重新開機事件](#)，讓您的執行個體在適合您的特定日期和時間重新開機。

### 檢視重新開機事件類型

您可以使用下列其中一種方法，檢視重新開機事件是執行個體重新開機還是系統重新開機。

#### Console

##### 檢視排程重新開機事件的類型

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。

2. 在導覽窗格中，選擇 Events (事件)。
3. 從篩選清單中選擇 Resource type: instance (資源類型：執行個體)。
4. 對於每個執行個體，檢視 Event type (事件類型) 欄中的值。此值為 system-reboot 或 instance-reboot。

## AWS CLI

檢視排程重新開機事件的類型

使用 [describe-instance-status](#) 命令。

```
aws ec2 describe-instance-status \
 --instance-id i-1234567890abcdef0
```

對於排程的重新開機事件，Code 的值為 system-reboot 或 instance-reboot。下列範例輸出顯示 system-reboot 事件。

```
[
 "Events": [
 {
 "InstanceEventId": "instance-event-0d59937288b749b32",
 "Code": "system-reboot",
 "Description": "The instance is scheduled for a reboot",
 "NotAfter": "2019-03-14T22:00:00.000Z",
 "NotBefore": "2019-03-14T20:00:00.000Z",
 "NotBeforeDeadline": "2019-04-05T11:00:00.000Z"
 }
]
]
```

## 執行個體重新開機動作

在其排程維護時段內，您可以等待執行個體重新啟動。[重新排程](#)執行個體重新啟動至適合您的日期和時間，或在方便時親自[重新啟動](#)執行個體。

在您的執行個體重新開機後，排程事件便會取消，事件的描述也會更新。基礎主機的擱置維護已完成，您可以在完全啟動後再次開始使用您的執行個體。

## 系統重新開機動作

您無法自行重新開機系統。您可以等待在排程的維護時段視窗內，讓系統重新開機，或將系統重新開機 [重新排程](#) 在適合您的日期和時間。系統重新開機通常在幾分鐘內完成。系統重新開機發生後，執行個體會保留其 IP 地址和 DNS 名稱，且任何位於本機執行個體存放區磁碟區上的資料都會保留。在系統重新開機完成後，執行個體的排程事件便會清除，並且您可以確認您執行個體上的軟體正以預期的方式運作中。

或者，若必須在不同時間維護執行個體，而且您無法重新排程系統重新開機，則您可以停止並啟動 Amazon EBS 後端執行個體，使其遷移到新的主機。但是，本機執行個體存放區磁碟區上的資料不會保留。您也可以將立即執行個體停止並啟動的作業自動化，以回應排程的系統重新啟動事件。如需詳細資訊，請參閱 AWS Health 使用者指南中的 [自動化 EC2 執行個體的動作](#)。對於執行個體存放區支援的執行個體，如果您無法重新排程系統重新開機，則可以從您最近的 AMI 啟動取代用執行個體，在排程維護時段視窗前將所有必要資料遷移到取代用執行個體，然後終止原始執行個體。

## 使用已排程維護的執行個體

當 AWS 必須維護執行個體的基礎主機時，就會排程執行個體進行維護。有兩種維護事件類型：網路維護及電源維護。

在網路維護期間，排程執行個體會暫時遺失網路連線能力。維護完成後，便會還原您執行個體的一般網路連線能力。

在電源維護期間，排程執行個體會暫時離線，然後重新開機。當執行重新開機時，會保留所有您執行個體的組態設定。

在您的執行個體重新開機後 (通常需要數分鐘)，確認您的應用程式正以預期的方式正常運作。此時，您的執行個體應該不會再有任何相關聯的排程事件，或若是如此，則排程事件描述的開頭為 [Completed] ([已完成])。有時需要最多 1 個小時，才會更新執行個體狀態描述。完成的維護事件會在 Amazon EC2 主控台儀表板上顯示最多一週。

### Amazon EBS 後端執行個體動作

您可以等待維護依照排程執行。或者，您可以停止並啟動執行個體，這會將它遷移到新的主機。如需停止執行個體的詳細資訊 (除了在停止時執行個體組態有何變更的資訊)，請參閱 [停止並啟動 Amazon EC2 執行個體](#)。

您可以將立即停止並啟動的作業自動化，以回應排程的維護事件。如需詳細資訊，請參閱 AWS Health 使用者指南中的 [自動化 EC2 執行個體的動作](#)。

### 執行個體存放區後端執行個體的動作

您可以等待維護依照排程執行。或者，若您希望在排程維護視窗期間維持一般操作，您可以從您最近的 AMI 啟動取代用執行個體，在排程維護視窗前將所有必要資料遷移到取代用執行個體，然後終止原始執行個體。

## 重新安排排定的事件

您可以重新排程事件，使其發生在適合您的特定日期和時間。只有具有截止日期的活動才能重新排程。有其他[重新排程事件的限制](#)。

您可以使用下列其中一種方法來重新排程事件。

### Console

#### 重新排程事件

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Events (事件)。
3. 從篩選清單中選擇 Resource type: instance (資源類型：執行個體)。
4. 選取一或多個執行個體，然後選擇 Actions (動作)、Schedule event (排程事件)。

只能重新排程有事件截止日期 (由 Deadline (截止日期) 的值指出) 的事件。如果其中一個選取的事件沒有截止日期，則會停用 Actions (動作)、Schedule event (排程事件)。

5. 對於 New start time (新增開始時間)，輸入事件的新日期和時間。新的日期和時間必須發生在 Event deadline (事件截止日期) 之前。
6. 選擇 Save (儲存)。

更新後的事件開始時間可能需要一兩分鐘才會反映在主控台中。

### AWS CLI

#### 重新排程事件

1. 只能重新排程有事件截止日期 (由 NotBeforeDeadline 的值指出) 的事件。使用 [describe-instance-status](#) 命令來檢視 NotBeforeDeadline 參數值。

```
aws ec2 describe-instance-status \
 --instance-id i-1234567890abcdef0
```



以下範例輸出顯示 `system-reboot` 事件，因為 `NotBeforeDeadline` 有值，所以可重新排程此事件。

```
[
 "Events": [
 {
 "InstanceEventId": "instance-event-0d59937288b749b32",
 "Code": "system-reboot",
 "Description": "The instance is scheduled for a reboot",
 "NotAfter": "2019-03-14T22:00:00.000Z",
 "NotBefore": "2019-03-14T20:00:00.000Z",
 "NotBeforeDeadline": "2019-04-05T11:00:00.000Z"
 }
]
]
```

- 若要重新排程事件，請使用 [modify-instance-event-start-time](#) 命令。使用 `not-before` 參數指定新的事件開始時間。新的事件開始時間必須落在 `NotBeforeDeadline` 之前。

```
aws ec2 modify-instance-event-start-time \
 --instance-id i-1234567890abcdef0 \
 --instance-event-id instance-event-0d59937288b749b32 \
 --not-before 2019-03-25T10:00:00.000
```

可能需要 1-2 分鐘，[describe-instance-status](#) 命令才會傳回更新的 `not-before` 參數值。

## 限制

- 只能重新排程有事件截止日期的事件。事件最晚可以重新排程到事件截止日期。主控台中的「截止日期 `NotBeforeDeadline`」欄和中的欄位會 AWS CLI 指出事件是否有截止日期。
- 只能重新排程尚未開始的事件。主控台中的「開始時間 `NotBefore`」欄和中的欄位 AWS CLI 表示事件開始時間。已排程在接下來 5 分鐘內開始的事件不能重新排程。
- 新的事件開始時間與目前時間至少必須相隔 60 分鐘。
- 如果您使用主控台重新排程多個事件，則由具有最早事件截止日期的事件決定事件截止日期。

## 定義排程事件的事件時段

您可以為重新啟動、停止或終止 Amazon EC2 執行個體的排程事件定義自訂每週重複的事件時段。您可以將一或多個執行個體關聯到事件時段。如果已計劃這些執行個體的排程事件，AWS 會在相關聯的事件時段中排程事件。

您可以使用事件時段，透過指定在工作負載離峰期間出現的事件時段，將工作負載可用性最大化。您也可以調整事件時段與內部維護排程。

可以透過指定一組時間範圍來定義事件時段。最短時間範圍為 2 小時。合併的時間範圍總計必須至少 4 小時。

您可以使用執行個體 ID 或執行個體標籤，將一或多個執行個體與事件時段建立關聯。您也可以使用主機 ID，將專用執行個體與事件時段建立關聯。

### Warning

事件時段僅適用於停止、重新啟動或終止執行個體的排程事件。

事件時段不適用於：

- 快速排程事件和網路維護事件。
- 不定期的維護，例如意外 AutoRecovery 重新啟動。

### 使用事件時段

- [考量事項](#)
- [檢視事件時段](#)
- [建立事件時段](#)
- [修改事件時段](#)
- [刪除事件時段](#)
- [標記事件時段](#)

### 考量事項

- 所有事件時段的時間皆為 UTC 格式。
- 最短每週事件時段為 4 小時。
- 事件時段內的時間範圍必須至少為 2 小時。

- 只有一個目標類型 (執行個體 ID、專用執行個體 ID 或執行個體標籤) 可以與事件時段關聯。
- 一個目標類型 (執行個體 ID、專用執行個體 ID 或執行個體標籤) 只能與一個事件時段關聯。
- 事件時段最多可與 100 個執行個體 ID 或 50 個專用執行個體 ID 或 50 個執行個體標籤相關聯。執行個體標籤可與任意數量的執行個體關聯。
- 每個 AWS 區域最多可建立 200 個事件視窗。
- 與事件時段相關聯的多個執行個體可能會同時發生排程事件。
- 如果 AWS 已排程事件，修改事件視窗不會變更已排程事件的時間。如果事件有截止日期，您可以[重新排程事件](#)。
- 您可以在排程事件之前停止並啟動執行個體，這樣會將執行個體遷移到新主機，而排程的事件將不再發生。

## 檢視事件時段

您可以使用下列其中一種方法來檢視事件時段。

### Console

#### 檢視事件時段

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Events (事件)。
3. 選擇 Actions (動作)、Manage event windows (管理事件時段)。
4. 選取事件時段以檢視其詳細資訊。

### AWS CLI

#### 描述所有事件時段

使用 [describe-instance-event-windows](#) 命令。

```
aws ec2 describe-instance-event-windows \
 --region us-east-1
```

#### 預期的輸出結果

```
{
```

```

 "InstanceEventWindows": [
 {
 "InstanceEventWindowId": "iew-0abcdef1234567890",
 "Name": "myEventWindowName",
 "CronExpression": "* 21-23 * * 2,3",
 "AssociationTarget": {
 "InstanceIds": [
 "i-1234567890abcdef0",
 "i-0598c7d356eba48d7"
],
 "Tags": [],
 "DedicatedHostIds": []
 },
 "State": "active",
 "Tags": []
 }
 ...
],
 "NextToken": "9d624e0c-388b-4862-a31e-a85c64fc1d4a"
 }

```

## 描述所有特定事件時段

使用 [describe-instance-event-windows](#) 命令與 `--instance-event-window-id` 參數來描述特定事件時段。

```

aws ec2 describe-instance-event-windows \
 --region us-east-1 \
 --instance-event-window-id iew-0abcdef1234567890

```

## 描述符合一個或多個篩選條件的事件時段

使用 [describe-instance-event-windows](#) 命令與 `--filters` 參數。在以下範例中，`instance-id` 篩選條件用於描述與指定執行個體相關聯的所有事件時段。

使用篩選條件時，它會執行直接比對。不過，`instance-id` 篩選條件是不同的。如果執行個體 ID 沒有直接匹配項，則會返回到與事件時段的間接關聯，例如執行個體的標籤或專用執行個體 ID (如果執行個體位於專用執行個體上)。

如需支援的篩選條件清單，請參閱 AWS CLI 參考中的 [describe-instance-event-windows](#)。

```
aws ec2 describe-instance-event-windows \
 --region us-east-1 \
 --filters Name=instance-id,Values=i-1234567890abcdef0 \
 --max-results 100 \
 --next-token <next-token-value>
```

## 預期的輸出結果

在下列範例中，執行個體位於與事件時段相關聯的專用執行個體上。

```
{
 "InstanceEventWindows": [
 {
 "InstanceEventWindowId": "iew-0dbc0adb66f235982",
 "TimeRanges": [
 {
 "StartWeekDay": "sunday",
 "StartHour": 2,
 "EndWeekDay": "sunday",
 "EndHour": 8
 }
],
 "Name": "myEventWindowName",
 "AssociationTarget": {
 "InstanceIds": [],
 "Tags": [],
 "DedicatedHostIds": [
 "h-0140d9a7ecbd102dd"
]
 },
 "State": "active",
 "Tags": []
 }
]
}
```

## 建立事件時段

您可以建立一或多個事件時段。對於每個事件時段，您可以指定一個或多個時間段。例如，您可以建立一個事件時段，其時間段為每天凌晨 4 點開始，持續 2 小時。或者，您可以建立一個事件時段，其時間段為星期日凌晨 2 點到凌晨 4 點和星期三凌晨 3 點到凌晨 5 點。

如需事件時段限制，請參閱本主題前面的[考量事項](#)。

事件時段會每週重複出現，直到您刪除為止。

使用下列其中一種方法來建立事件時段。

## Console

### 建立事件時段

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
  2. 在導覽窗格中，選擇 Events (事件)。
  3. 選擇 Actions (動作)、Manage event windows (管理事件時段)。
  4. 選擇 Create instance event window (建立執行個體事件時段)。
  5. 對於 Event window name (事件時段名稱)，輸入事件時段的描述性名稱。
  6. 對於 Event window schedule (事件時段排程)，透過使用 cron 排程建置器或指定時間範圍，進行選擇以在事件時段中指定時間段。
    - 如果選擇 Cron schedule builder (Cron 排程建置器)，請指定以下內容：
      1. 對於 Days (UTC) (天 (UTC))，指定事件時段發生的星期日。
      2. 對於 Start time (UTC) (開始時間 (UTC))，指定事件時段開始的時間。
      3. 對於 Duration (持續時間)，在事件時段中指定時間段的持續時間。每個時間段的最短持續時間為 2 小時。事件時段的最短持續時間總計必須等於或超過 4 小時。所有時間均以 UTC 表示。
    - 如果選擇 Time ranges (時間範圍)，請選擇 Add new time range (新增時間範圍)，然後指定開始日期和時間以及結束日期和時間。對每個時間範圍重複此步驟。每個時間範圍的最短持續時間為 2 小時。所有時間範圍合併的最小持續時間總計必須等於或超過 4 小時。
  7. (選用) 對於 Target details (目標詳細資訊)，請將一個或多個執行個體與事件時段關聯，如果執行個體已排程進行維護，則排程的事件會在關聯的事件時段中發生。您可以使用執行個體 ID 或執行個體標籤，將一個或多個執行個體與事件時段關聯。您可以使用主機 ID，將專用執行個體與事件時段關聯。
- 請注意，您可以建立事件時段，而無需將目標與時段建立關聯。稍後，您可以修改時段以關聯一個或多個目標。
8. (選用) 對於 Event window tags (事件時段標籤)，選擇 Add tag (新增標籤)，然後輸入標籤的金鑰和值。針對每個標籤，重複執行上述動作。

## 9. 選擇 Create event window (建立事件時段)。

### AWS CLI

若要使用建立事件視窗 AWS CLI，請先建立事件視窗，然後將一或多個目標與事件視窗相關聯。

#### 建立事件時段

您可以在建立事件時段時定義一組時間範圍或 cron 表達式，但不能同時定義兩者。

#### 建立具有時間範圍的事件時段

使用 [create-instance-event-window](#) 命令並指定 `--time-range` 參數。您不能指定 `--cron-expression` 參數。

```
aws ec2 create-instance-event-window \
 --region us-east-1 \
 --time-range StartWeekDay=monday,StartHour=2,EndWeekDay=wednesday,EndHour=8 \
 --tag-specifications "ResourceType=instance-event-
window,Tags=[{Key=K1,Value=V1}]" \
 --name myEventWindowName
```

#### 預期的輸出結果

```
{
 "InstanceEventWindow": {
 "InstanceEventWindowId": "iew-0abcdef1234567890",
 "TimeRanges": [
 {
 "StartWeekDay": "monday",
 "StartHour": 2,
 "EndWeekDay": "wednesday",
 "EndHour": 8
 }
],
 "Name": "myEventWindowName",
 "State": "creating",
 "Tags": [
 {
 "Key": "K1",
 "Value": "V1"
 }
]
 }
}
```

```

]
 }
}

```

### 透過 cron 表達式建立事件時段

使用 [create-instance-event-window](#) 命令並指定 `--cron-expression` 參數。您不能指定 `--time-range` 參數。

```

aws ec2 create-instance-event-window \
 --region us-east-1 \
 --cron-expression "* 21-23 * * 2,3" \
 --tag-specifications "ResourceType=instance-event-
window,Tags=[{Key=K1,Value=V1}]" \
 --name myEventWindowName

```

### 預期的輸出結果

```

{
 "InstanceEventWindow": {
 "InstanceEventWindowId": "iew-0abcdef1234567890",
 "Name": "myEventWindowName",
 "CronExpression": "* 21-23 * * 2,3",
 "State": "creating",
 "Tags": [
 {
 "Key": "K1",
 "Value": "V1"
 }
]
 }
}

```

### 將目標與事件時段關聯

您只能將一種目標類型 (執行個體 ID、專用執行個體 ID 或執行個體標籤) 與事件時段關聯。

### 將執行個體標籤與事件時段關聯

使用 [associate-instance-event-window](#) 命令並指定 `instance-event-window-id` 參數來指定事件時段。若要關聯執行個體標籤，請指定 `--association-target` 參數，並針對參數值指定一個或多個標籤。



```
aws ec2 associate-instance-event-window \
 --region us-east-1 \
 --instance-event-window-id iew-0abcdef1234567890 \
 --association-target "InstanceTags=[{Key=k2,Value=v2},{Key=k1,Value=v1}]"
```

### 預期的輸出結果

```
{
 "InstanceEventWindow": {
 "InstanceEventWindowId": "iew-0abcdef1234567890",
 "Name": "myEventWindowName",
 "CronExpression": "* 21-23 * * 2,3",
 "AssociationTarget": {
 "InstanceIds": [],
 "Tags": [
 {
 "Key": "k2",
 "Value": "v2"
 },
 {
 "Key": "k1",
 "Value": "v1"
 }
],
 "DedicatedHostIds": []
 },
 "State": "creating"
 }
}
```

### 將一個或多個執行個體與事件時段關聯

使用 [associate-instance-event-window](#) 命令並指定 `instance-event-window-id` 參數來指定事件時段。若要關聯執行個體，請指定 `--association-target` 參數，並針對參數值指定一個或多個執行個體 ID。

```
aws ec2 associate-instance-event-window \
 --region us-east-1 \
 --instance-event-window-id iew-0abcdef1234567890 \
 --association-target "InstanceIds=i-1234567890abcdef0,i-0598c7d356eba48d7"
```

### 預期的輸出結果

```
{
 "InstanceEventWindow": {
 "InstanceEventWindowId": "iew-0abcdef1234567890",
 "Name": "myEventWindowName",
 "CronExpression": "* 21-23 * * 2,3",
 "AssociationTarget": {
 "InstanceIds": [
 "i-1234567890abcdef0",
 "i-0598c7d356eba48d7"
],
 "Tags": [],
 "DedicatedHostIds": []
 },
 "State": "creating"
 }
}
```

### 將專用執行個體與事件時段關聯

使用 [associate-instance-event-window](#) 命令並指定 `instance-event-window-id` 參數來指定事件時段。若要關聯專用執行個體，請指定 `--association-target` 參數，並針對參數值指定一個或多個專用執行個體 ID。

```
aws ec2 associate-instance-event-window \
 --region us-east-1 \
 --instance-event-window-id iew-0abcdef1234567890 \
 --association-target "DedicatedHostIds=h-029fa35a02b99801d"
```

### 預期的輸出結果

```
{
 "InstanceEventWindow": {
 "InstanceEventWindowId": "iew-0abcdef1234567890",
 "Name": "myEventWindowName",
 "CronExpression": "* 21-23 * * 2,3",
 "AssociationTarget": {
 "InstanceIds": [],
 "Tags": [],
 "DedicatedHostIds": [
 "h-029fa35a02b99801d"
]
 },
 },
}
```

```
 "State": "creating"
 }
}
```

## 修改事件時段

您可以修改事件時段的所有欄位，但其 ID 除外。例如，當日光節約時間開始時，您可能想要修改事件時段排程。對於現有事件時段，您可能會想要新增或移除目標。

使用下列其中一種方法來修改事件時段。

### Console

#### 修改事件時段

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Events (事件)。
3. 選擇 Actions (動作)、Manage event windows (管理事件時段)。
4. 選取要修改的事件時段，然後選擇 Actions (動作)、Modify instance event window (修改執行個體事件時段)。
5. 修改事件時段中的欄位，然後選擇 Modify event window (修改事件時段)。

### AWS CLI

若要使用修改事件視窗 AWS CLI，您可以修改時間範圍或 Cron 運算式，以及將一或多個目標與事件視窗關聯或取消關聯。

#### 修改事件時段時間

您可以在修改事件時段時修改時間範圍或 cron 表達式，但不能同時修改兩者。

#### 修改事件時段的時間範圍

使用 [modify-instance-event-window](#) 命令，然後指定要修改的事件時段。指定 `--time-range` 參數來修改時間範圍。您不能指定 `--cron-expression` 參數。

```
aws ec2 modify-instance-event-window \
 --region us-east-1 \
 --time-range
```

```
--instance-event-window-id iew-0abcdef1234567890 \
--time-range StartWeekDay=monday,StartHour=2,EndWeekDay=wednesday,EndHour=8
```

## 預期的輸出結果

```
{
 "InstanceEventWindow": {
 "InstanceEventWindowId": "iew-0abcdef1234567890",
 "TimeRanges": [
 {
 "StartWeekDay": "monday",
 "StartHour": 2,
 "EndWeekDay": "wednesday",
 "EndHour": 8
 }
],
 "Name": "myEventWindowName",
 "AssociationTarget": {
 "InstanceIds": [
 "i-0abcdef1234567890",
 "i-0be35f9acb8ba01f0"
],
 "Tags": [],
 "DedicatedHostIds": []
 },
 "State": "creating",
 "Tags": [
 {
 "Key": "K1",
 "Value": "V1"
 }
]
 }
}
```

## 修改事件時段的一組時間範圍

使用 [modify-instance-event-window](#) 命令，然後指定要修改的事件時段。指定 `--time-range` 參數來修改時間範圍。您不能指定相同呼叫中的 `--cron-expression` 參數。

```
aws ec2 modify-instance-event-window \
--region us-east-1 \

```

```
--instance-event-window-id iew-0abcdef1234567890 \
--time-range '[{"StartWeekDay": "monday", "StartHour": 2, "EndWeekDay":
wednesday", "EndHour": 8},
{"StartWeekDay": "thursday", "StartHour": 2, "EndWeekDay": "friday",
"EndHour": 8}]'
```

## 預期的輸出結果

```
{
 "InstanceEventWindow": {
 "InstanceEventWindowId": "iew-0abcdef1234567890",
 "TimeRanges": [
 {
 "StartWeekDay": "monday",
 "StartHour": 2,
 "EndWeekDay": "wednesday",
 "EndHour": 8
 },
 {
 "StartWeekDay": "thursday",
 "StartHour": 2,
 "EndWeekDay": "friday",
 "EndHour": 8
 }
],
 "Name": "myEventWindowName",
 "AssociationTarget": {
 "InstanceIds": [
 "i-0abcdef1234567890",
 "i-0be35f9acb8ba01f0"
],
 "Tags": [],
 "DedicatedHostIds": []
 },
 "State": "creating",
 "Tags": [
 {
 "Key": "K1",
 "Value": "V1"
 }
]
 }
}
```

## 修改事件時段的 cron 表達式

使用 [modify-instance-event-window](#) 命令，然後指定要修改的事件時段。指定 `--cron-expression` 參數來修改 cron 表達式。您不能指定 `--time-range` 參數。

```
aws ec2 modify-instance-event-window \
 --region us-east-1 \
 --instance-event-window-id iew-0abcdef1234567890 \
 --cron-expression "* 21-23 * * 2,3"
```

## 預期的輸出結果

```
{
 "InstanceEventWindow": {
 "InstanceEventWindowId": "iew-0abcdef1234567890",
 "Name": "myEventWindowName",
 "CronExpression": "* 21-23 * * 2,3",
 "AssociationTarget": {
 "InstanceIds": [
 "i-0abcdef1234567890",
 "i-0be35f9acb8ba01f0"
],
 "Tags": [],
 "DedicatedHostIds": []
 },
 "State": "creating",
 "Tags": [
 {
 "Key": "K1",
 "Value": "V1"
 }
]
 }
}
```

## 修改與事件時段相關聯的目標

您可以將其他目標與事件時段關聯。您也可以取消現有目標與事件時段的關聯。但是，只有一種目標類型 (執行個體 ID、專用執行個體 ID 或執行個體標籤) 可以與事件時段相關聯。

## 將其他目標與事件時段關聯

如需如何將目標與事件時段產生關聯的指示，請參閱 [Associate a target with an event window](#)。

## 解除執行個體標籤與事件時段的關聯

使用 [disassociate-instance-event-window](#) 命令並指定 `instance-event-window-id` 參數來指定事件時段。若要解除執行個體標籤的關聯，請指定 `--association-target` 參數，並針對參數值指定一個或多個標籤。

```
aws ec2 disassociate-instance-event-window \
 --region us-east-1 \
 --instance-event-window-id iew-0abcdef1234567890 \
 --association-target "InstanceTags=[{Key=k2,Value=v2},{Key=k1,Value=v1}]"
```

## 預期的輸出結果

```
{
 "InstanceEventWindow": {
 "InstanceEventWindowId": "iew-0abcdef1234567890",
 "Name": "myEventWindowName",
 "CronExpression": "* 21-23 * * 2,3",
 "AssociationTarget": {
 "InstanceIds": [],
 "Tags": [],
 "DedicatedHostIds": []
 },
 "State": "creating"
 }
}
```

## 解除一個或多個執行個體與事件時段的關聯

使用 [disassociate-instance-event-window](#) 命令並指定 `instance-event-window-id` 參數來指定事件時段。若要解除執行個體的關聯，請指定 `--association-target` 參數，並針對參數值指定一個或多個執行個體 ID。

```
aws ec2 disassociate-instance-event-window \
 --region us-east-1 \
 --instance-event-window-id iew-0abcdef1234567890 \
 --association-target "InstanceIds=i-1234567890abcdef0,i-0598c7d356eba48d7"
```

## 預期的輸出結果

```
{
```

```

 "InstanceEventWindow": {
 "InstanceEventWindowId": "iew-0abcdef1234567890",
 "Name": "myEventWindowName",
 "CronExpression": "* 21-23 * * 2,3",
 "AssociationTarget": {
 "InstanceIds": [],
 "Tags": [],
 "DedicatedHostIds": []
 },
 "State": "creating"
 }
 }
}

```

### 解除專用執行個體與事件時段的關聯

使用 [disassociate-instance-event-window](#) 命令並指定 `instance-event-window-id` 參數來指定事件時段。若要解除專用執行個體的關聯，請指定 `--association-target` 參數，並針對參數值指定一個或多個專用執行個體 ID。

```

aws ec2 disassociate-instance-event-window \
 --region us-east-1 \
 --instance-event-window-id iew-0abcdef1234567890 \
 --association-target DedicatedHostIds=h-029fa35a02b99801d

```

### 預期的輸出結果

```

{
 "InstanceEventWindow": {
 "InstanceEventWindowId": "iew-0abcdef1234567890",
 "Name": "myEventWindowName",
 "CronExpression": "* 21-23 * * 2,3",
 "AssociationTarget": {
 "InstanceIds": [],
 "Tags": [],
 "DedicatedHostIds": []
 },
 "State": "creating"
 }
}

```



## 刪除事件時段

您可以使用下列其中一種方法，一次刪除一個事件時段。

### Console

#### 刪除事件時段

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Events (事件)。
3. 選擇 Actions (動作)、Manage event windows (管理事件時段)。
4. 選取要刪除的事件時段，然後選擇 Actions (動作)、Delete instance event window (刪除執行個體事件時段)。
5. 當出現提示時，輸入 **delete**，然後選擇 Delete (刪除)。

### AWS CLI

#### 刪除事件時段

使用 [delete-instance-event-window](#) 命令，然後指定要刪除的事件時段。

```
aws ec2 delete-instance-event-window \
 --region us-east-1 \
 --instance-event-window-id iew-0abcdef1234567890
```

#### 強制刪除事件時段

使用 `--force-delete` 參數 (如果事件時段目前與目標相關聯)。

```
aws ec2 delete-instance-event-window \
 --region us-east-1 \
 --instance-event-window-id iew-0abcdef1234567890 \
 --force-delete
```

#### 預期的輸出結果

```
{
 "InstanceEventWindowState": {
```

```
 "InstanceEventWindowId": "iew-0abcdef1234567890",
 "State": "deleting"
 }
}
```

## 標記事件時段

您可以在建立事件時段時或之後對其進行標記。

若要在建立事件時段時進行標記，請參閱[建立事件時段](#)。

使用下列其中一種方法來標記事件時段。

### Console

#### 標記現有事件時段

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Events (事件)。
3. 選擇 Actions (動作)、Manage event windows (管理事件時段)。
4. 選取要標記的事件時段，然後選擇 Actions (動作)、Manage instance event window tags (管理執行個體事件時段標籤)。
5. 選擇 Add tag (新增標籤) 以新增標籤。針對每個標籤，重複執行上述動作。
6. 選擇 Save (儲存)。

### AWS CLI

#### 標記現有事件時段

使用[建立標籤](#)命令來標記現有資源。在下面的例子中，現有事件時段標記為 Key=purpose 和 Value=test。

```
aws ec2 create-tags \
 --resources iew-0abcdef1234567890 \
 --tags Key=purpose,Value=test
```

## 使用監控執行個體 CloudWatch

您可以使用 Amazon 監控執行個體 CloudWatch，Amazon 會從 Amazon EC2 收集原始資料並將其處理為可讀的近即時指標。這些統計資料會記錄 15 個月的時間，以便您存取歷史資訊，並更清楚 Web 應用程式或服務的執行效能。

根據預設，Amazon EC2 會 CloudWatch 在 5 分鐘內將指標資料傳送到。若要 CloudWatch 在 1 分鐘內將執行個體的指標資料傳送到，您可以啟用執行個體的詳細監控。如需詳細資訊，請參閱 [啟用或關閉執行個體的詳細監控](#)。

Amazon EC2 控制台根據來自 Amazon 的原始數據顯示一系列圖形 CloudWatch。根據您的需求，您可能希望從 Amazon 取得執行個體的資料，CloudWatch 而不是主控台中的圖形。

如需 Amazon CloudWatch 帳單和費用資訊，請參閱 Amazon CloudWatch 使用者指南中的 [CloudWatch 帳單和費用](#)。

### 目錄

- [Amazon EC2 實例警報](#)
- [啟用或關閉執行個體的詳細監控](#)
- [列出執行個體的可用 CloudWatch 指標](#)
- [使用 Amazon EC2 主控台安裝和設定 CloudWatch 代理程式，以新增其他指標](#)
- [取得您的執行個體指標的統計資料](#)
- [您的執行個體的圖形指標](#)
- [建立執行個體的 CloudWatch 警示](#)
- [建立警示以停止、終止、重新啟動或復原執行個體](#)

## Amazon EC2 實例警報

您可以從 Amazon EC2 主控台的執行個體畫面檢視和建立執行個體的 Amazon CloudWatch 警示。

下列螢幕擷取畫面顯示編號為 1 和 2 的主控制台控制項，用於從「執行個體」畫面檢視和建立警報。

**Instances (7) Info**

Find Instance by attribute or tag (case-sensitive) All states ▾

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status
<input type="checkbox"/>	My-1-Spot-Ins...	I-01aeed690c9fb5322	Running	t3.nano	2/2 checks p...	<b>1</b> View alarms
<input type="checkbox"/>	My-2-Spot-Ins...	I-0ba5e5bbc9d634fa6	Stopped	t3.nano	-	View ala <b>2</b>

## 從執行個體畫面檢視警示

您可以從「執行個體」畫面檢視每個執行個體的警報。

從執行個體畫面檢視執行個體的警示

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 在「執行個體」表格中，針對您選擇的執行個體選擇 [檢視鬧鐘] (在前面螢幕擷取畫面中編號為 1)
4. 在 **i-0123456789 ##**視窗的警示詳細資料中，選擇要在主控台中檢視警示的警示名稱。  
CloudWatch

## 從執行個體畫面建立警示

您可以從「執行個體」畫面為每個執行個體建立警示。

從執行個體畫面建立執行個體的警示

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 在「執行個體」表格中，為您選擇的執行個體選擇加號 (前面螢幕擷取畫面中的編號為 2)。
4. 在「管理 CloudWatch 鬧鐘」畫面中，建立鬧鐘。如需詳細資訊，請參閱 [建立執行個體的 CloudWatch 警示](#)。

## 啟用或關閉執行個體的詳細監控

依預設，啟用執行個體進行基本監控。可選擇啟用詳細監控。

下表重點說明執行個體的基本監控和詳細監控之間的差異。

監控類型	描述	費用
基本監控	只有狀態檢查指標在 1 分鐘內可用。  所有其他指標均在 5 分鐘內可用。	免費。
詳細監控	所有指標 (包括狀態檢查指標) 在 1 分鐘內可用。若要取得此層級的資	系統會依傳送目標的量度向您收費 CloudWatch。您不需為資料儲存付

監控類型	描述	費用
	料，您必須特別啟用執行個體的這項功能。對於已啟用詳細監控的執行個體，您亦可取得類似執行個體的跨群組彙總資料。	費。如需詳細資訊，請參閱 <a href="#">Amazon CloudWatch 定價頁面</a> 上的付費方案和範例 1-EC2 詳細監控。

## 主題

- [所需的 IAM 許可](#)
- [啟用詳細監控](#)
- [關閉詳細監控](#)

## 所需的 IAM 許可

若要啟用執行個體的詳細監控，您的使用者必須具有使用 [MonitorInstances](#) API 動作的許可。若要關閉執行個體的詳細監控，您的使用者必須具有使用 [UnmonitorInstances](#) API 動作的許可。

## 啟用詳細監控

執行個體啟動時，或在執行個體運作或停止後，您都可以啟用執行個體的詳細監控。啟用對執行個體的詳細監控時，不會影響對連接至執行個體的 EBS 磁碟區監控。如需詳細資訊，請參閱 [Amazon EBS 的 Amazon CloudWatch 指標](#)。

## Console

### 啟用現有執行個體的詳細監控

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體，然後選擇 [動作]、[監視和疑難排解]、[管理詳細]
4. 在 Detailed monitoring (詳細監控) 詳細資訊頁面上，針對 Detailed monitoring (詳細監控)，選取 Enable (啟用) 核取方塊。
5. 選擇 Save (儲存)。

### 在執行個體啟動時啟用詳細監控

使用 Amazon EC2 主控台啟動執行個體時，在 [進階詳細資料] 下，選取 [詳細 CloudWatch 監控] 核取方塊。

## AWS CLI

啟用現有執行個體的詳細監控

使用下列 [monitor-instances](#) 命令來啟用特定執行個體的詳細監控。

```
aws ec2 monitor-instances --instance-ids i-1234567890abcdef0
```

在執行個體啟動時啟用詳細監控

使用 [run-instances](#) 命令搭配 `--monitoring` 旗標來啟用特定執行個體的詳細監控。

```
aws ec2 run-instances --image-id ami-09092360 --monitoring Enabled=true...
```

## 關閉詳細監控

執行個體啟動時，或在執行個體運作或停止後，您都可以關閉執行個體的詳細監控。

## Console

關閉詳細監控

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體，然後選擇 [動作]、[監視和疑難排解]、[管理詳細]
4. 在 Detailed monitoring (詳細監控) 詳細資訊頁面上，針對 Detailed monitoring (詳細監控)，清除 Enable (啟用) 核取方塊。
5. 選擇 Save (儲存)。

## AWS CLI

關閉詳細監控

使用下列 [unmonitor-instances](#) 命令來關閉特定執行個體的詳細監控。

```
aws ec2 unmonitor-instances --instance-ids i-1234567890abcdef0
```

## 列出執行個體的可用 CloudWatch 指標

Amazon EC2 將指標發送到 Amazon CloudWatch。您可以使用 AWS Management Console、AWS CLI、或 API 列出 Amazon EC2 傳送到哪些指標 CloudWatch。每個資料點預設會涵蓋執行個體從啟動時間後 5 分鐘內的活動。若您已啟用詳細監控功能，則每個資料點會涵蓋從啟動時間後下一分鐘的活動。請注意，針對 Minimum (最小值)、Maximum (最大值) 和 Average (平均值) 等統計數字，EC2 提供的指標最精細的程度是 1 分鐘。

如需取得這些指標統計資料的資訊，請參閱[取得您的執行個體指標的統計資料](#)。

### 內容

- [執行個體指標](#)
- [CPU 額度指標](#)
- [專用執行個體指標](#)
- [Nitro 型執行個體的 Amazon EBS 指標](#)
- [狀態檢查指標](#)
- [流量鏡像指標](#)
- [Auto Scaling 群組指標](#)
- [Amazon EC2 指標維度](#)
- [Amazon EC2 用量指標](#)
- [使用主控台列出指標](#)
- [使用列出量度 AWS CLI](#)

### 執行個體指標

AWS/EC2 命名空間包含下列執行個體指標。

指標	描述	單位	有意義的統計資料
CPUUtilization	Amazon EC2 用於執行 EC2 執行個體的實體 CPU 時間百分比，其中包括執行使用者程式碼和 Amazon EC2 程式碼所花費的時間。	百分比	<ul style="list-style-type: none"><li>• 平均數</li><li>• 下限</li><li>• 最大</li></ul>

指標	描述	單位	有意義的統計資料
	<p>整體而言，CPUUtilization 是訪客 CPUUtilization 和 Hypervisor CPUUtilization 的總和。</p> <p>CloudWatch 由於傳統裝置模擬、非舊式裝置的設定、大量中斷的工作負載、即時移轉和即時更新等因素，作業系統中的工具顯示的百分比可能會有所不同。</p>		
DiskReadOps	<p>在指定期間，執行個體可用之所有執行個體存放區磁碟區的已完成讀取操作。</p> <p>若要計算該期間的每秒平均 I/O 操作數 (IOPS)，請將該期間的總操作數除以該期間的秒數。</p> <p>如果沒有執行個體存放區磁碟區，數值可能為 0 或指標未回報。</p>	計數	<ul style="list-style-type: none"> <li>• 總和</li> <li>• 平均數</li> <li>• 下限</li> <li>• 最大</li> </ul>
DiskWriteOps	<p>在指定期間，執行個體可用之所有執行個體存放區磁碟區的已完成寫入操作。</p> <p>若要計算該期間的每秒平均 I/O 操作數 (IOPS)，請將該期間的總操作數除以該期間的秒數。</p> <p>如果沒有執行個體存放區磁碟區，數值可能為 0 或指標未回報。</p>	計數	<ul style="list-style-type: none"> <li>• 總和</li> <li>• 平均數</li> <li>• 下限</li> <li>• 最大</li> </ul>



指標	描述	單位	有意義的統計資料
DiskReadBytes	<p>從執行個體可用之所有執行個體存放區磁碟區讀取的位元組。</p> <p>此指標用來判斷應用程式從執行個體硬碟中讀取的資料磁碟區。這可以用來判斷應用程式的速度。</p> <p>所報告的數目是在期間內收到的位元組總數。如果您要使用基本 (5 分鐘) 監控，則可以將此數字除以 300，以找到位元組數/秒。如果您具有詳細 (1 分鐘) 監控，請將它除以 60。您也可以使用度 CloudWatch 量數學函數DIFF_TIME 來尋找每秒的位元組數。例如，如果您已繪製 CloudWatch 為圖形m1，度量數學公式會m1/(DIFF_TIME(m1)) 傳回以DiskReadBytes 位元組/秒為單位的量度。如需有關以DIFF_TIME 及其他度量數學函數的詳細資訊，請參閱 Amazon 使用 CloudWatch 者指南中的<a href="#">使用指標數學運算</a>。</p> <p>如果沒有執行個體存放區磁碟區，數值可能為 0 或指標未回報。</p>	位元組	<ul style="list-style-type: none"> <li>• 總和</li> <li>• 平均數</li> <li>• 下限</li> <li>• 最大</li> </ul>

指標	描述	單位	有意義的統計資料
DiskWrite Bytes	<p>寫入至執行個體可用之所有執行個體存放區磁碟區的位元組。</p> <p>此指標用來判斷應用程式寫入至執行個體硬碟中的資料磁碟區。這可以用來判斷應用程式的速度。</p> <p>所報告的數目是在期間內收到的位元組總數。如果您要使用基本 (5 分鐘) 監控，則可以將此數字除以 300，以找到位元組數/秒。如果您具有詳細 (1 分鐘) 監控，請將它除以 60。您也可以使用度 CloudWatch 量數學函數DIFF_TIME 來尋找每秒的位元組數。例如，如果您已繪製 CloudWatch 為圖形m1，度量數學公式會m1/(DIFF_TIME(m1)) 傳回以DiskWrite Bytes 位元組/秒為單位的量度。如需有關以DIFF_TIME 及其他度量數學函數的詳細資訊，請參閱 Amazon 使用 CloudWatch 者指南中的<a href="#">使用指標數學運算</a>。</p> <p>如果沒有執行個體存放區磁碟區，數值可能為 0 或指標未回報。</p>	位元組	<ul style="list-style-type: none"> <li>• 總和</li> <li>• 平均數</li> <li>• 下限</li> <li>• 最大</li> </ul>
MetadataNoToken	<p>使用不使用 Token 的方法成功存取執行個體中繼資料服務 (IMDS) 的次數。</p> <p>此測量結果可用來判斷是否有任何處理程序正在使用「執行處理描述資料服務第 1 版 (IMDSv1)」(未使用 Token) 的執行處理中繼資料。如果所有要求都使用憑證支援的工作階段，也就是執行個體中繼資料服務版本 2 (IMDSv2)，則值為 0。如需詳細資訊，請參閱<a href="#">轉換為使用執行個體中繼資料服務第 2 版</a>。</p>	計數	<ul style="list-style-type: none"> <li>• 總和</li> <li>• 百分位數</li> </ul>

指標	描述	單位	有意義的統計資料
MetadataNoTokenRejected	<p>在停用 IMDSv1 之後，嘗試進行 IMDSv1 呼叫的次數。</p> <p>如果出現此量度，表示已嘗試進行 IMDSv1 呼叫並遭到拒絕。您可以重新啟用 IMDSv1，或確定所有通話都使用 IMDSv2。如需詳細資訊，請參閱 <a href="#">轉換為使用執行個體中繼資料服務第 2 版</a>。</p>	計數	<ul style="list-style-type: none"> <li>• 總和</li> <li>• 百分位數</li> </ul>
NetworkIn	<p>執行個體在所有網路介面上收到的位元組數目。此指標識別流向單一執行個體之傳入網路流量的磁碟區。</p> <p>所報告的數目是在期間內收到的位元組總數。如果您要使用基本 (5 分鐘) 監控並且統計數字為總和，則可以將此數字除以 300，以找出每秒的位元組數。如果您具有詳細 (1 分鐘) 監控並且統計數字為總和，請將它除以 60。您也可以使用度 CloudWatch 量數學函數 DIFF_TIME 來尋找每秒的位元組數。例如，如果您已繪製 CloudWatch 為圖形 m1，度量數學公式會 <math>m1 / (\text{DIFF\_TIME}(m1))</math> 傳回以 NetworkIn 位元組/秒為單位的量度。如需有關以 DIFF_TIME 及其他度量數學函數的詳細資訊，請參閱 <a href="#">Amazon 使用 CloudWatch 者指南中的使用指標數學運算</a>。</p>	位元組	<ul style="list-style-type: none"> <li>• 總和</li> <li>• 平均數</li> <li>• 下限</li> <li>• 最大</li> </ul>

指標	描述	單位	有意義的統計資料
NetworkOut	<p>執行個體在所有網路介面上送出的位元組數目。此指標識別來自單一執行個體之傳出網路流量的磁碟區。</p> <p>所報告的數目是在期間內送出的位元組總數。如果您要使用基本 (5 分鐘) 監控並且統計數字為總和，則可以將此數字除以 300，以找出每秒的位元組數。如果您具有詳細 (1 分鐘) 監控並且統計數字為總和，請將它除以 60。您也可以使用度 CloudWatch 量數學函數DIFF_TIME 來尋找每秒的位元組數。例如，如果您已繪製 CloudWatch 為圖形m1，度量數學公式會m1/(DIFF_TIME(m1)) 傳回以NetworkOut 位元組/秒為單位的量度。如需有關以DIFF_TIME 及其他度量數學函數的詳細資訊，請參閱 Amazon 使用 CloudWatch 者指南<a href="#">中的使用指標數學運算</a>。</p>	位元組	<ul style="list-style-type: none"> <li>• 總和</li> <li>• 平均數</li> <li>• 下限</li> <li>• 最大</li> </ul>
NetworkPacketsIn	<p>執行個體在所有網路介面上收到的封包數目。此指標識別單一執行個體上的傳入流量 (封包數目)。</p> <p>此指標僅適用於基本監控 (每隔 5 分鐘)。若要計算執行個體在 5 秒時間每秒接收的封包數 (PPS)，請將 Sum 統計數字值除以 300。您也可以使用 CloudWatch 公制數學函數DIFF_TIME 來尋找每秒封包數。例如，如果您已繪製 CloudWatch 為圖形 NetworkPacketsIn m1，度量數學公式會m1/(DIFF_TIME(m1)) 傳回以封包/秒為單位的量度。如需有關以DIFF_TIME 及其他度量數學函數的詳細資訊，請參閱 Amazon 使用 CloudWatch 者指南<a href="#">中的使用指標數學運算</a>。</p>	計數	<ul style="list-style-type: none"> <li>• 總和</li> <li>• 平均數</li> <li>• 下限</li> <li>• 最大</li> </ul>

指標	描述	單位	有意義的統計資料
NetworkPacketsOut	<p>執行個體在所有網路介面上送出的封包數目。此指標識別單一執行個體上的傳出流量 (封包數目)。</p> <p>此指標僅適用於基本監控 (每隔 5 分鐘)。若要計算執行個體在 5 分鐘時間每秒傳送的封包數 (PPS)，請將 Sum 統計數字值除以 300。您也可以使用 CloudWatch 公制數學函數 DIFF_TIME 來尋找每秒封包數。例如，如果您已繪製 CloudWatch 為圖形 NetworkPacketsOut m1，度量數學公式會 <math>m1 / (\text{DIFF\_TIME}(m1))</math> 傳回以封包/秒為單位的量度。如需有關以 DIFF_TIME 及其他度量數學函數的詳細資訊，請參閱 Amazon 使用 CloudWatch 者指南中的 <a href="#">使用指標數學運算</a>。</p>	計數	<ul style="list-style-type: none"> <li>• 總和</li> <li>• 平均數</li> <li>• 下限</li> <li>• 最大</li> </ul>

## CPU 額度指標

AWS/EC2 命名空間包含 [高載效能執行個體](#) 的下列 CPU 額度指標。

指標	描述	單位	有意義的統計資料
CPUCreditUsage	<p>執行個體為 CPU 使用率花費的 CPU 額度數量。一個 CPU 額度等於一個 vCPU 以 100% 使用率執行 1 分鐘，或同等的 vCPU、使用率與時間的組合 (例如，一個 vCPU 以 50% 使用率執行 2 分鐘，或兩個 vCPU 以 25% 使用率執行 2 分鐘)。</p> <p>CPU 額度指標僅提供 5 分鐘頻率。如果您要指定大於 5 分鐘的期間，請使用 Sum 統計資訊代替 Average 統計資訊。</p>	額度 (vCPU-分鐘)	<ul style="list-style-type: none"> <li>• 總和</li> <li>• 平均數</li> <li>• 下限</li> <li>• 最大</li> </ul>

指標	描述	單位	有意義的統計資料
CPUCreditBalance	<p>自執行個體啟動或開始後，累積獲得的 CPU 額度數量。如果是 T2 Standard，CPUCreditBalance 也包含已產生的啟動額度。</p> <p>獲得額度後，額度會在額度餘額中累積，並在支付額度時，從額度餘額中移出。額度餘額有最大值限制，它取決於執行個體大小。到達限制之後，任何獲得的新額度都會遭到捨棄。如果是 T2 Standard，啟動額度不會計入此限制。</p> <p>CPUCreditBalance 中的額度可供執行個體支付以大幅提升並超越基準 CPU 使用率。</p> <p>當執行個體執行時，CPUCreditBalance 中的額度不會過期。當 T3 或 T3a 執行個體停止時，CPUCreditBalance 值會持續保存七天。因此，所有累積的額度都會消失。當 T2 執行個體停止時，CPUCreditBalance 值不會持續保存，而且所有累積的額度都將消失。</p> <p>CPU 額度指標僅提供 5 分鐘頻率。</p>	額度 (vCPU-分鐘)	<ul style="list-style-type: none"> <li>• 總和</li> <li>• 平均數</li> <li>• 下限</li> <li>• 最大</li> </ul>
CPU SurplusCreditBalance	<p>當 unlimited 執行個體的 CPUCreditBalance 值為 0 時，該執行個體已支出的剩餘額度數量。</p> <p>CPU SurplusCreditBalance 值由獲得的 CPU 額度支付。如果剩餘額度超過執行個體在 24 小時期間可獲得的最大額度數量，超過最大值的支出剩餘額度將必須負擔額外的費用。</p> <p>CPU 額度指標僅提供 5 分鐘頻率。</p>	額度 (vCPU-分鐘)	<ul style="list-style-type: none"> <li>• 總和</li> <li>• 平均數</li> <li>• 下限</li> <li>• 最大</li> </ul>

指標	描述	單位	有意義的統計資料
CPUSurplusCreditsCharged	<p>若支出剩餘額度數量未由獲得的 CPU 額度付清，會產生額外的費用。</p> <p>發生以下任何情況時，將收取支出剩餘額度的費用。</p> <ul style="list-style-type: none"> <li>支出剩餘額度超過執行個體在 24 小時期間可獲得的最大額度數量。在小時結束時，將收取超過最大值的支出剩餘額度的費用。</li> <li>執行個體已停止或終止。</li> <li>執行個體從 unlimited 切換至 standard。</li> </ul> <p>CPU 額度指標僅提供 5 分鐘頻率。</p>	額度 (vCPU-分鐘)	<ul style="list-style-type: none"> <li>總和</li> <li>平均數</li> <li>下限</li> <li>最大</li> </ul>

## 專用執行個體指標

AWS/EC2 命名空間包含下列 T3 專用執行個體的指標。

指標	描述	單位	有意義的統計資料
DedicatedHostCPUUtilization	專用執行個體上執行的執行個體目前正在使用的已配置運算容量百分比。	百分比	<ul style="list-style-type: none"> <li>總和</li> <li>平均數</li> <li>下限</li> <li>最大</li> </ul>

## Nitro 型執行個體的 Amazon EBS 指標

AWS/EC2 命名空間包含連接至 Nitro 型執行個體 (不屬於裸機執行個體) 的磁碟區的其它 Amazon EBS 指標。

指標	描述	單位	有意義的統計資料
EBSReadOps	<p>在指定期間，從連接至執行個體的所有 Amazon EBS 磁碟區完成讀取的操作數。</p> <p>若要計算該期間的每秒平均讀取 I/O 操作數 (讀取 IOPS)，請將該期間的總操作數除以該期間的秒數。如果您正使用基本 (5 分鐘) 監控，則可以將此數字除以 300，以計算讀取 IOPS。如果您具有詳細 (1 分鐘) 監控，請將它除以 60。您也可以使用 CloudWatch 公制數學函數DIFF_TIME 來尋找每秒運算數。例如，如果您已繪製 CloudWatch 為圖形 EBSReadOps m1，則度量數學公式會 <math>m1 / (DIFF\_TIME(m1))</math> 傳回運算/秒的量度。如需有關以DIFF_TIME 及其他度量數學函數的詳細資訊，請參閱 Amazon 使用 CloudWatch 者指南中的<a href="#">使用指標數學運算</a>。</p>	計數	<ul style="list-style-type: none"> <li>• 總和</li> <li>• 平均數</li> <li>• 下限</li> <li>• 最大</li> </ul>
EBSWriteOps	<p>在指定期間，從連接至執行個體的所有 EBS 磁碟區完成寫入的操作數。</p> <p>若要計算該期間的每秒平均寫入 I/O 操作數 (寫入 IOPS)，請將該期間的總操作數除以該期間的秒數。如果您正使用基本 (5 分鐘) 監控，則可以將此數字除以 300，以計算寫入 IOPS。如果您具有詳細 (1 分鐘) 監控，請將它除以 60。您也可以使用 CloudWatch 公制數學函數DIFF_TIME 來尋找每秒運算數。例如，如果您已繪製 CloudWatch 為圖形 EBSWriteOps m1，則度量數學公式會 <math>m1 / (DIFF\_TIME(m1))</math> 傳回運算/秒的量度。如需有關以DIFF_TIME 及其他度量數學函數的詳細資訊，請參閱 Amazon 使用 CloudWatch 者指南中的<a href="#">使用指標數學運算</a>。</p>	計數	<ul style="list-style-type: none"> <li>• 總和</li> <li>• 平均數</li> <li>• 下限</li> <li>• 最大</li> </ul>



指標	描述	單位	有意義的統計資料
EBSReadBytes	<p>在指定期間內，從連接至執行個體的所有 EBS 磁碟區所讀取的位元組。</p> <p>所報告的數目是在該期間內讀取的位元組總數。如果您正使用基本 (5 分鐘) 監控，則可以將此數字除以 300，以得到所讀取的位元組數/秒。如果您具有詳細 (1 分鐘) 監控，請將它除以 60。您也可以使用度 CloudWatch 量數學函數DIFF_TIME 來尋找每秒的位元組數。例如，如果您已繪製 CloudWatch 為圖形m1，度量數學公式會m1/(DIFF_TIME(m1)) 傳回以EBSReadBytes 位元組/秒為單位的量度。如需有關以DIFF_TIME 及其他度量數學函數的詳細資訊，請參閱 Amazon 使用 CloudWatch 者指南<a href="#">中的使用指標數學運算</a>。</p>	位元組	<ul style="list-style-type: none"> <li>• 總和</li> <li>• 平均數</li> <li>• 下限</li> <li>• 最大</li> </ul>
EBSWriteBytes	<p>在指定期間內，從所有連接至執行個體的 EBS 磁碟區所寫入的位元組。</p> <p>所報告的數目是在該期間內寫入的位元組總數。如果您正使用基本 (5 分鐘) 監控，則可以將此數字除以 300，得到所寫入的位元組數/秒。如果您具有詳細 (1 分鐘) 監控，請將它除以 60。您也可以使用度 CloudWatch 量數學函數DIFF_TIME 來尋找每秒的位元組數。例如，如果您已繪製 CloudWatch 為圖形m1，度量數學公式會m1/(DIFF_TIME(m1)) 傳回以EBSWriteBytes 位元組/秒為單位的量度。如需有關以DIFF_TIME 及其他度量數學函數的詳細資訊，請參閱 Amazon 使用 CloudWatch 者指南<a href="#">中的使用指標數學運算</a>。</p>	位元組	<ul style="list-style-type: none"> <li>• 總和</li> <li>• 平均數</li> <li>• 下限</li> <li>• 最大</li> </ul>

指標	描述	單位	有意義的統計資料
EBSIOBalance%	<p>提供叢發儲存貯體中剩餘 I/O 額度百分比資訊。只有基本監控才提供此指標。</p> <p>此指標僅適用於一些大小為 *.4xlarge 及更小的執行個體，至少每 24 小時維持最佳效能 30 分鐘。</p> <p>Sum 統計資料不適用於此指標。</p>	百分比	<ul style="list-style-type: none"> <li>• 下限</li> <li>• 最大</li> </ul>
EBSByteBalance%	<p>提供叢發儲存貯體中剩餘傳輸量額度百分比的資訊。只有基本監控才提供此指標。</p> <p>此指標僅適用於一些大小為 *.4xlarge 及更小的執行個體，至少每 24 小時維持最佳效能 30 分鐘。</p> <p>Sum 統計資料不適用於此指標。</p>	百分比	<ul style="list-style-type: none"> <li>• 下限</li> <li>• 最大</li> </ul>

如需針對 EBS 磁碟區提供的指標的相關資訊，請參閱 [Amazon EBS 使用者指南中的 Amazon EBS 磁碟區](#) 指標。如需用於 Spot Fleet 的指標資訊，請參閱 [CloudWatch 競價型艦隊的指標](#)。

## 狀態檢查指標

根據預設，狀態檢查指標的頻率為 1 分鐘，而且免費。對於新啟動的執行個體，只有在執行個體完成初始化狀態 (在執行個體進入 running 狀態的幾分鐘內) 之後，才能使用狀態檢查指標資料。如需有關 EC2 狀態檢查的詳細資訊，請參閱 [您的執行個體的状态檢查](#)。

AWS/EC2 命名空間包含下列狀態檢查指標。

指標	描述	單位	有意義的統計資料
StatusCheckFailed	<p>報告執行個體是否在最後一分鐘通過執行個體狀態檢查與系統狀態檢查。</p> <p>此指標可以是 0 (通過) 或 1 (失敗)。</p>	計數	<ul style="list-style-type: none"> <li>• 總和</li> <li>• 平均數</li> </ul>

指標	描述	單位	有意義的統計資料
	根據預設，此指標的頻率為 1 分鐘，而且免費。		
StatusCheckFailed_Instance	報告執行個體是否在最後一分鐘通過執行個體狀態檢查。 此指標可以是 0 (通過) 或 1 (失敗)。 根據預設，此指標的頻率為 1 分鐘，而且免費。	計數	<ul style="list-style-type: none"> <li>總和</li> <li>平均數</li> </ul>
StatusCheckFailed_System	報告執行個體是否在最後一分鐘通過系統狀態檢查。 此指標可以是 0 (通過) 或 1 (失敗)。 根據預設，此指標的頻率為 1 分鐘，而且免費。	計數	<ul style="list-style-type: none"> <li>總和</li> <li>平均數</li> </ul>
StatusCheckFailed_AttachedEBS	報告執行個體是否在最後一分鐘通過附接的 EBS 狀態檢查。 此指標可以是 0 (通過) 或 1 (失敗)。 根據預設，此指標的頻率為 1 分鐘，而且免費。	計數	<ul style="list-style-type: none"> <li>總和</li> <li>平均數</li> </ul>

AWS/EBS命名空間包含下列狀態檢查測量結果。

指標	描述	單位	有意義的統計資料
VolumeStalledIOCheck	注意：僅適用於 Nitro 執行個體。未針對連接到 Amazon ECS 和 AWS Fargate 任務的磁碟區發佈。 在最後一分鐘報告磁碟區停止的 IO 檢查是通過還是失敗。此指標可以是 0 (通過) 或 1 (失敗)。	計數	<ul style="list-style-type: none"> <li>總和</li> <li>平均數</li> <li>下限</li> <li>最大</li> </ul>

## 流量鏡像指標

AWS/EC2 命名空間包含鏡像流量的指標。如需詳細資訊，請參閱 [Amazon VPC 流量鏡像指南](#) [CloudWatch](#) 中的 [使用 Amazon 監控鏡像流量](#)。

## Auto Scaling 群組指標

AWS/AutoScaling 命名空間包含 Auto Scaling 組的指標。如需詳細資訊，請參閱 [Amazon EC2 Auto Scaling 使用者指南](#) 中的 [監控 Auto Scaling 群組和執行個體的指 CloudWatch 標](#)。

## Amazon EC2 指標維度

您可以使用下列維度來強化上表中列出的指標。

維度	描述
AutoScalingGroupName	此維度可篩選您為指定容量群組中所有執行個體請求的資料。如果您使用 Auto Scaling，則「Auto Scaling 群組」是您定義的執行個體集合。如果執行個體位於 Auto Scaling 群組中，則此維度僅適用於 Amazon EC2 指標。適用於已啟用詳細或基本監控的執行個體。
ImageId	此維度可篩選您為執行此 Amazon EC2 Amazon Machine Image (AMI) 的所有執行個體請求的資料。適用於已啟用詳細監控的執行個體。
InstanceId	此維度會篩選僅針對已識別執行個體所請求的資料。如此可協助您找出要監控資料的確切執行個體。
InstanceType	此維度可篩選您為以此指定執行個體類型執行的所有執行個體請求的資料。如此可協助您依據執行的執行個體類型將您的資料進行分類。例如，您可以比較來自 m1.small 執行個體與 m1.large 執行個體的資料，判斷哪個執行個體對您的應用程式而言具有較高的商業價值。適用於已啟用詳細監控的執行個體。

## Amazon EC2 用量指標

您可以使用 CloudWatch 使用量度來提供您帳戶資源使用情況的可見度。使用這些指標，在 CloudWatch 圖形和儀表板上視覺化您目前的服務使用情況。

Amazon EC2 使用量指標對應於 AWS 服務配額。您可以設定警示，在您的用量接近服務配額時發出警示。如需與服務配額 CloudWatch 整合的詳細資訊，請參閱 Amazon [AWS 使用 CloudWatch 者指南中的使用量指標](#)。

Amazon EC2 在 AWS/Usage 命名空間中發佈下列指標。

指標	描述
ResourceCount	<p>您的帳戶中正在執行的特定資源數量。資源由與指標相關聯的維度定義。</p> <p>此指標最有用的統計資料是 MAXIMUM，它表示 1 分鐘內使用的最大資源數量。</p>

以下維度用於強化 Amazon EC2 發佈的用量指標。

維度	描述
Service	包含資源的 AWS 服務名稱。對於 Amazon EC2 用量指標，此維度的值為 EC2。
Type	正在報告的實體類型。目前，Amazon EC2 用量指標的唯一有效值為 Resource。
Resource	正在執行的資源類型。目前，Amazon EC2 用量指標的唯一有效值為 vCPU，它會傳回正在執行之執行個體的相關資訊。
Class	<p>正在追蹤的資源類別。對於以 vCPU 做為 Resource 維度值的 Amazon EC2 用量指標，有效值為 Standard/OnDemand、F/OnDemand、G/OnDemand、Inf/OnDemand、P/OnDemand 和 X/OnDemand。</p> <p>該維度的值定義指標報告之執行個體類型的第一個字母。例如，Standard/OnDemand 傳回開頭為 A、C、D、H、I、M、R、T 和 Z 之所有執行中執行個體的相關資訊，而 G/OnDemand 傳回開頭為 G 之所有執行中執行個體的相關資訊。</p>

## 使用主控台列出指標

指標會先依據命名空間進行分組，再依據各命名空間內不同的維度組合進行分組。例如，您可以檢視所有 Amazon EC2 提供的指標，或以執行個體 ID、執行個體類型、映像 (AMI) ID 或 Auto Scaling 群組分組的指標。

依類別檢視可用的指標 (主控台)

1. 開啟主 CloudWatch 控制台，網址為 <https://console.aws.amazon.com/cloudwatch/>。
2. 在導覽窗格中，展開 [量度]，然後選擇 [所有量度]。
3. 選擇 EC2 指標命名空間。

The screenshot shows the AWS CloudWatch console interface. At the top, there are tabs for 'Browse', 'Multi source query', 'Graphed metrics', 'Options', and 'Source'. Below these are buttons for 'Add math' and 'Add query'. The main section is titled 'Metrics (1,153) Info'. There are options for 'Alarm recommendations', 'Download alarm code', 'Create alarm', 'Graph with SQL', and 'Graph search'. A search bar is present with the text 'Search for any metric, dimension, resource id or account id'. The region is set to 'Ireland'. Below the search bar is a grid of metrics:

Backup	16	Directory Service	62	EBS	47
EC2	93	EC2/API	152	EC2 Capacity Reservations	8
EC2 Spot	618	EFS	36	Events	1
Logs	3	NATGateway	15	S3	12
SSM Run Command	3	Usage	87		

4. 選取指標維度 (例如 Per-Instance Metrics (每個執行個體指標))。

5. 若要排序指標，請使用直欄標題。若要將指標圖形化，請選取指標旁的核取方塊。若要依資源篩選，請選擇資源 ID，然後選擇 Add to search (新增至搜尋)。若要依指標篩選，請選擇指標名稱，然後選擇 Add to search (新增至搜尋)。

<input type="checkbox"/>	Instance name 92/92	InstanceId	Metric name	Alarms
<input type="checkbox"/>	fingerprint	i-047470286...		No alarms
<input type="checkbox"/>	fingerprint	i-04747028607e6		No alarms
<input type="checkbox"/>	fingerprint	i-04747028607e6		No alarms
<input type="checkbox"/>	fingerprint	i-04747028607e6		No alarms
<input type="checkbox"/>	fingerprint	i-04747028607e6		No alarms
<input type="checkbox"/>	fingerprint	i-04747028607e6		No alarms
<input type="checkbox"/>	fingerprint	i-04747028607e6		No alarms
<input type="checkbox"/>	fingerprint	i-04747028607e6		No alarms
<input type="checkbox"/>	fingerprint	i-04747028607e63eaa	StatusCheckFailed	No alarms

## 使用列出量度 AWS CLI

使用 [list-metrics](#) 命令列出執行個體的 CloudWatch 指標。

列出 Amazon EC2 (AWS CLI) 的所有可用指標

以下範例指定 AWS/EC2 命名空間以檢視 Amazon EC2 的所有指標。

```
aws cloudwatch list-metrics --namespace AWS/EC2
```

下列為範例輸出：

```
{
 "Metrics": [
 {
 "Namespace": "AWS/EC2",
 "Dimensions": [
 {
 "Name": "InstanceId",
 "Value": "i-1234567890abcdef0"
 }
],
 "MetricName": "NetworkOut"
 },
 {
 "Namespace": "AWS/EC2",
 "Dimensions": [
 {
 "Name": "InstanceId",
 "Value": "i-1234567890abcdef0"
 }
],
 "MetricName": "CPUUtilization"
 },
 {
 "Namespace": "AWS/EC2",
 "Dimensions": [
 {
 "Name": "InstanceId",
 "Value": "i-1234567890abcdef0"
 }
],
 "MetricName": "NetworkIn"
 },
 ...
]
}
```

列出單一執行個體的所有可用指標 (AWS CLI)

以下範例指定 AWS/EC2 命名空間和 InstanceId 維度，以僅檢視指定執行個體的結果。



```
aws cloudwatch list-metrics --namespace AWS/EC2 --dimensions
Name=InstanceId,Value=i-1234567890abcdef0
```

列出所有執行個體的指標 (AWS CLI)

以下範例指定 AWS/EC2 命名空間和指標名稱，以僅檢視指定指標的結果。

```
aws cloudwatch list-metrics --namespace AWS/EC2 --metric-name CPUUtilization
```

## 使用 Amazon EC2 主控台安裝和設定 CloudWatch 代理程式，以新增其他指標

使用 Amazon EC2 主控台安裝和設定 CloudWatch 代理程式已處於測試階段，且可能會有所變更。

根據預設，亞馬遜 CloudWatch 提供基本指標，例如 CPUUtilization 和 NetworkIn，用於監控您的 Amazon EC2 執行個體。若要收集其他指標，您可以在 EC2 執行個體上安裝 CloudWatch 代理程式，然後將代理程式設定為發出選取的指標。您可以使用 Amazon EC2 主控台為您執行此操作，而不是在每個 EC2 執行個體上手動安裝和設定 CloudWatch 代理程式。

本主題說明如何使用 Amazon EC2 主控台在執行個體上安裝 CloudWatch 代理程式，並將代理程式設定為發出選取的指標。

如需此程序的手動步驟，請參閱 Amazon 使用 CloudWatch 者指南 AWS Systems Manager 中的 [使用安裝 CloudWatch 代理程式](#)。如需 CloudWatch 代理程式的詳細資訊，請參閱 [使用代理程式收集指標、記錄檔和追蹤](#)。

### 主題

- [必要條件](#)
- [運作方式](#)
- [成本](#)
- [安裝和設定 CloudWatch 代理程式](#)

### 必要條件

若要使用 Amazon EC2 安裝和設定 CloudWatch 代理程式，您必須符合本節所述的使用者和執行個體先決條件。

## 使用者必要

若要使用此功能，您的 IAM 主控台使用者或角色必須具有使用 Amazon EC2 所需的許可和下列 IAM 許可：

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ssm:GetParameter",
 "ssm:PutParameter"
],
 "Resource": "arn:aws:ssm:*:*:parameter/EC2-Custom-Metrics-*"
 },
 {
 "Effect": "Allow",
 "Action": [
 "ssm:SendCommand",
 "ssm:ListCommandInvocations",
 "ssm:DescribeInstanceInformation"
],
 "Resource": [
 "*"
]
 },
 {
 "Effect": "Allow",
 "Action": [
 "iam:GetInstanceProfile",
 "iam:SimulatePrincipalPolicy"
],
 "Resource": "*"
 }
]
}
```

## 實例先決條件

- 執行個體狀態：running
- 支援的作業系統：

- AWS Systems Manager 代理程式 (SSM 代理程式)：已安裝。SSM 代理程式的兩個注意事項：
  - SSM 代理程式已預先安裝在由 AWS 受信任的協力廠商提供的某些 Amazon 機器映像 (AMI) 上。如需支援 AMI 的相關資訊以及安裝 SSM 代理程式的指示，請參閱 [《使用者指南》中預先安裝 SSM 代理程式的 Amazon 機器映像 \(AMI\)](#)。AWS Systems Manager
  - 如果 SSM 代理程式發生問題，請參閱《AWS Systems Manager 使用者指南》中的 [《SSM 代理程式疑難排解》](#)。
- 執行個體的 IAM 許可：必須將下列 AWS 受管政策新增至連接至執行個體的 IAM 角色：
  - [AmazonSSM ManagedInstance 核心](#) — 讓執行個體使用 Systems Manager 來安裝和設定代理程式。CloudWatch
  - [CloudWatchAgentServer原則](#) — 可讓執行個體使用 CloudWatch代理程式將資料寫入 CloudWatch。

如需如何將 IAM 許可新增至執行個體的詳細資訊，請參閱 IAM 使用者指南 [中的使用執行個體設定檔](#)。

## 運作方式

在使用 Amazon EC2 主控台安裝和設定 CloudWatch 代理程式之前，必須確保 IAM 使用者或角色以及要新增指標的執行個體符合特定先決條件。然後，您可以使用 Amazon EC2 主控台在選取的執行個體上安裝和設定 CloudWatch 代理程式。

### 首先符合 [先決條件](#)

- 您需要必要的 IAM 許可 — 開始之前，請確保您的控制台使用者或角色具有使用此功能所需的 IAM 許可。
- 執行個體 — 若要使用此功能，您的 EC2 執行個體必須是 Linux 執行個體、已安裝 SSM 代理程式、擁有必要的 IAM 許可並正在執行。

### 然後，您可以 [使用該功能](#)

1. 選取執行個體 — 在 Amazon EC2 主控台中，選取要在其上安裝和設定 CloudWatch 代理程式的執行個體。然後，您可以選擇設定 CloudWatch 代理程式來啟動程序。
2. 驗證 SSM 代理程式 — Amazon EC2 會檢查 SSM 代理程式是否已在每個執行個體上安裝並啟動。任何未通過此檢查的執行個體都會從程序中排除。SSM 代理程式用於在此程序期間對執行個體執行動作。

3. 驗證 IAM 許可 — Amazon EC2 會檢查每個執行個體是否具有此程序所需的 IAM 許可。任何未通過此檢查的執行個體都會從程序中排除。IAM 權限可讓 CloudWatch 代理程式從執行個體收集指標，並與之整合 AWS Systems Manager 以使用 SSM 代理程式。
4. 驗證 CloudWatch 代理程式 — Amazon EC2 會檢查 CloudWatch 代理程式是否已在每個執行個體上安裝並執行。如果有任何執行個體未通過此檢查，Amazon EC2 會為您安裝和啟動 CloudWatch 代理程式。此程序完成後，CloudWatch 代理程式會在每個執行個體上收集選取的指標。
5. 選取指標組態 — 您可以選取 CloudWatch 代理程式要從執行個體發出的指標。選取之後，Amazon EC2 會在參數存放區中儲存一個組態檔案，直到程序完成為止。除非程序中斷，否則 Amazon EC2 會從參數存放區刪除組態檔案。請注意，如果您未選取指標，但先前已將其新增至執行個體，則在完成此程序後，系統會將該指標從您的執行個體中移除。
6. 更新 CloudWatch 代理程式組態 — Amazon EC2 會將指標組態傳送給 CloudWatch 代理程式。這是該過程的最後一步。如果成功，您的執行個體可以發出所選指標的資料，而 Amazon EC2 會從參數存放區刪除組態檔案。

## 成本

您在此程序期間新增的其他指標會以自訂指標計費。如需有關 CloudWatch 指標定價的詳細資訊，請參閱 [Amazon CloudWatch 定價](#)。

## 安裝和設定 CloudWatch 代理程式

您可以使用 Amazon EC2 主控台安裝和設定 CloudWatch 代理程式，以新增其他指標。

### Note

每次執行此程序時，都會覆寫現有的 CloudWatch 代理程式組態。如果您沒有選取先前選取的指標，它會從執行個體中移除。

## 使用 Amazon EC2 主控台安裝和設定 CloudWatch 代理程式

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取要安裝的執行個體，並設定 CloudWatch 代理程式。
4. 選擇 [動作]、[監視和疑難排解]、[設定 CloudWatch 代

**i** Tip

此功能並非所有可用 AWS 區域。如果無法使用 [設定 CloudWatch 代理程式]，請嘗試其他區域。

5. 對於程序中的每個步驟，請閱讀主控台文字，然後選擇 [下一步]。
6. 若要完成此程序，請在最後一個步驟中選擇 [完成]。

## 取得您的執行個體指標的統計資料

您可以取得執行個體 CloudWatch 指標的統計資料。

### 目錄

- [統計資料概觀](#)
- [取得特定執行個體的統計資料](#)
- [彙總所有執行個體的統計資料](#)
- [依據 Auto Scaling 群組彙總統計資料](#)
- [依據 AMI 的彙總統計資料](#)

### 統計資料概觀

統計資料是指定期間內的測量結果資料彙總。CloudWatch 根據您的自訂資料提供或在中由其他服務提供的指標資料點 AWS 來提供統計資料 CloudWatch。彙總會使用在您指定期間的命名空間、公制名稱、維度和量測資料點單位來進行。下表說明可用的統計資料。

統計數字	描述
Minimum	在指定期間內觀察到的最低值。您可以使用這個值來判斷適用於您應用程式的低活動磁碟區。
Maximum	在指定期間內觀察到的最高值。您可以使用這個值來判斷適用於您應用程式的高活動磁碟區。
Sum	為符合指標總和提交的所有值。這個統計資料在決定指標的總磁碟區時非常有用。

統計數字	描述
Average	指定期間內的 Sum/SampleCount 值。透過比較這個統計資料與 Minimum 和 Maximum，您可以確定指標的完整範圍，以及 Minimum 和 Maximum 與平均使用的接近程度。這個比較可協助您知道何時需要增加或減少您的資源。
SampleCount	用於統計資料計算的資料點計數 (數值)。
pNN.NN	指定百分位數的值。您可以指定任何百分位數，最多使用兩位小數 (例如，p95.45)。

## 取得特定執行個體的統計資料

下列範例說明如何使用 AWS Management Console 或判斷特定 EC2 執行個體的最大 CPU 使用率。  
AWS CLI

### 要求

- 您必須擁有該執行個體的 ID。您可以使用 AWS Management Console 或 [describe-instances](#) 命令以取得執行個體 ID。
- 預設會啟用基本監控，但您可以啟用詳細監控。如需詳細資訊，請參閱 [啟用或關閉執行個體的詳細監控](#)。

### 顯示特定執行個體的 CPU 使用率 (主控台)

1. 開啟主 CloudWatch 控制台，網址為 <https://console.aws.amazon.com/cloudwatch/>。
2. 在導覽窗格中，選擇 指標。
3. 選擇 EC2 指標命名空間。

**Metrics (1,153) Info**

Alarm recommendations  [Download alarm code](#) [Create alarm](#) [Graph with SQL](#) [Graph search](#)

Ireland   Search iGraph

Backup	16	Directory Service	62	EBS	47
EC2	93	EC2/API	152	EC2 Capacity Reservations	8
EC2 Spot	618	EFS	36	Events	1
Logs	3	NATGateway	15	S3	12
SSM Run Command	3	Usage	87		

4. 選擇 Per-Instance Metrics (每個執行個體指標) 維度。

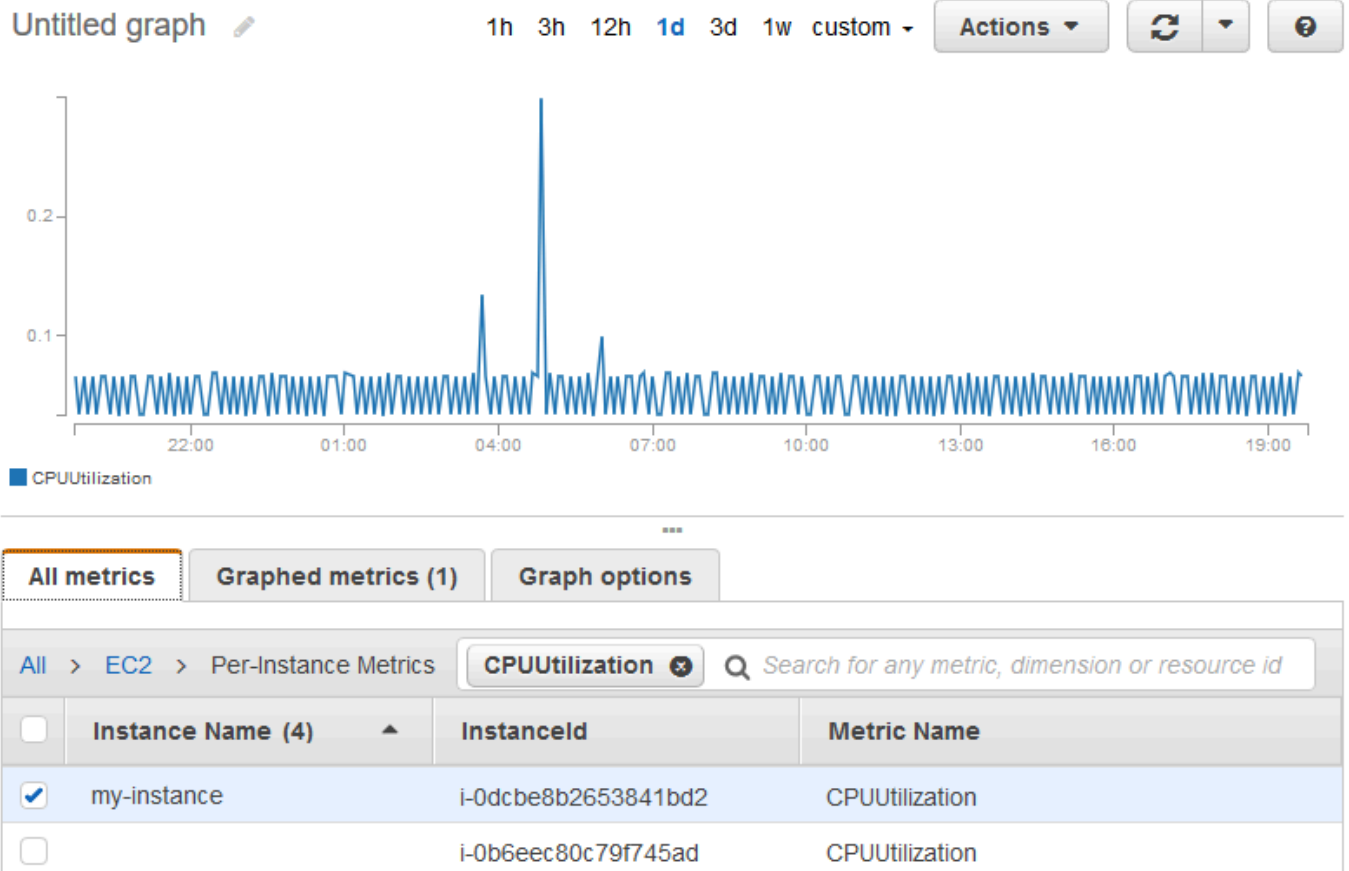
**Metrics (93) Info**

Alarm recommendations  [Download alarm code \(14\)](#) [Create alarm](#) [Graph with SQL](#) [Graph search](#)

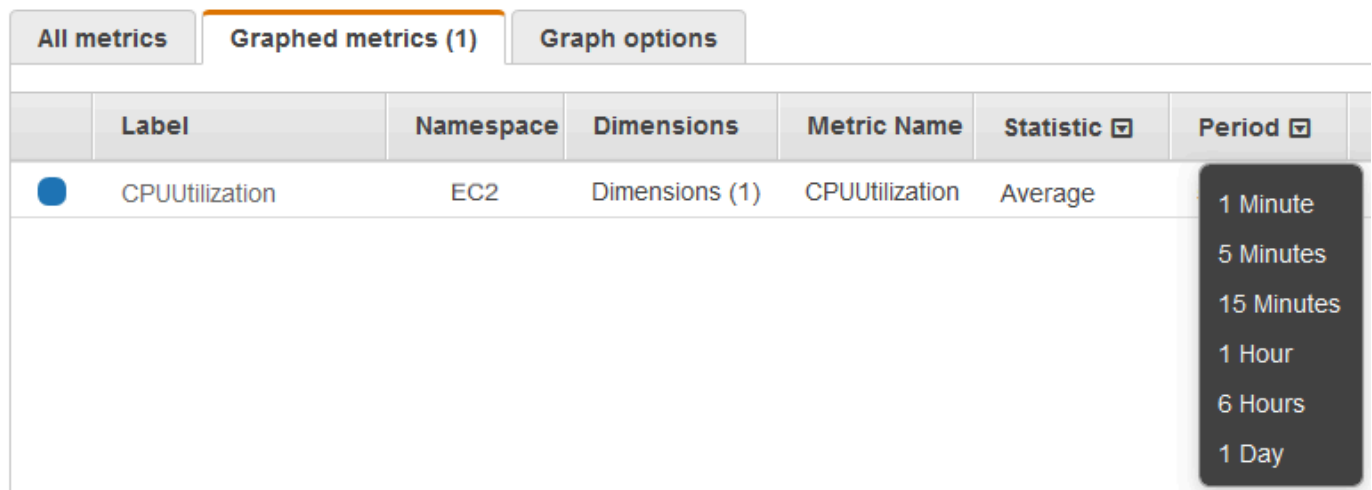
Ireland [All](#) > EC2

HostId	1	Per-Instance Metrics	92
--------	---	----------------------	----

5. 在搜尋欄位中，輸入 **CPUUtilization**，然後按 Enter 鍵。選擇特定執行個體的資料列，這會顯示該執行個體 CPUUtilization 指標的圖形。若要為圖形命名，請選擇鉛筆圖示。若要變更時間範圍，請選取一個預先定義的值，或選擇 custom (自訂)。



6. 若要變更指標的統計資料或期間，請選擇 Graphed metrics (圖表化指標) 索引標籤。選擇欄位標題或個別的值，然後選擇不同的值。



取得特定執行個體的 CPU 使用率 (AWS CLI)



透過下列 [get-metric-statistics](#) 命令，使用指定的期間和時間間隔來取得指定執行個體的 CPUUtilization 指標：

```
aws cloudwatch get-metric-statistics --namespace AWS/EC2 --metric-name CPUUtilization
--period 3600 \
--statistics Maximum --dimensions Name=InstanceId,Value=i-1234567890abcdef0 \
--start-time 2022-10-18T23:18:00 --end-time 2022-10-19T23:18:00
```

下列為範例輸出。每個值代表單一 EC2 執行個體的最大 CPU 使用率百分比。

```
{
 "Datapoints": [
 {
 "Timestamp": "2022-10-19T00:18:00Z",
 "Maximum": 0.33000000000000002,
 "Unit": "Percent"
 },
 {
 "Timestamp": "2022-10-19T03:18:00Z",
 "Maximum": 99.670000000000002,
 "Unit": "Percent"
 },
 {
 "Timestamp": "2022-10-19T07:18:00Z",
 "Maximum": 0.34000000000000002,
 "Unit": "Percent"
 },
 {
 "Timestamp": "2022-10-19T12:18:00Z",
 "Maximum": 0.34000000000000002,
 "Unit": "Percent"
 },
 ...
],
 "Label": "CPUUtilization"
}
```

## 彙總所有執行個體的統計資料

已啟用詳細監控的執行個體可使用彙總統計資料。彙總資料不含使用基本監控的執行個體。在取得所有執行個體的彙總統計資料前，您必須[啟用詳細監控](#) (額外付費)，它以 1 分鐘的間隔時間提供資料。

請注意，Amazon CloudWatch 無法跨 AWS 區域彙總資料。區域之間的指標是完全獨立的。

此範例顯示如何使用詳細監控取得 EC2 執行個體的平均 CPU 使用量。因為未指定維度，所以會 CloudWatch 傳回 AWS/EC2 命名空間中所有維度的統計資料。

**⚠ Important**

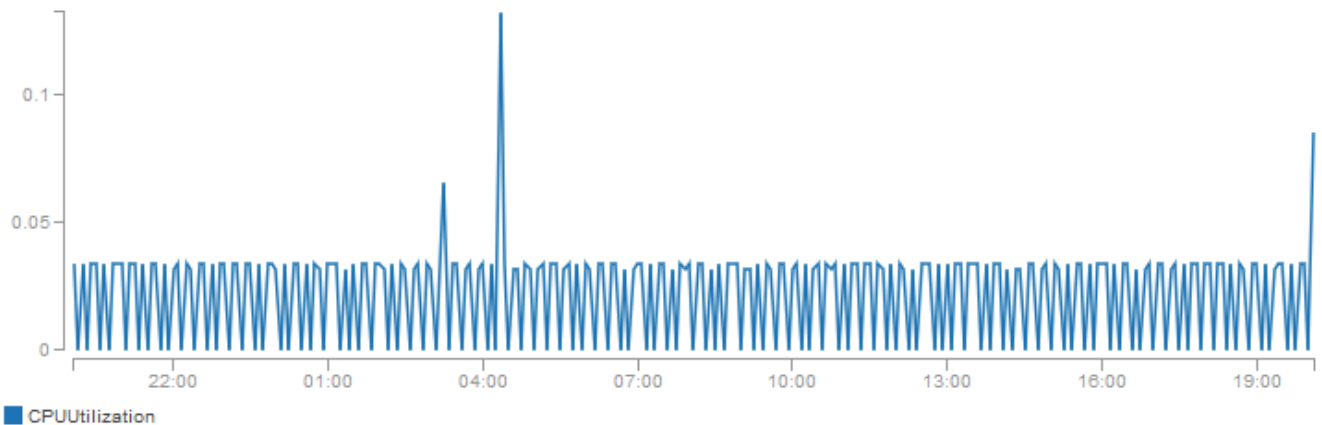
這種跨 AWS 命名空間擷取所有維度的技術不適用於您發佈到 Amazon CloudWatch 的自訂命名空間。使用自訂命名空間，您必須指定與任何特定資料點建立關聯的一組完整維度，以擷取包含該資料點的統計資料。

顯示您所有執行個體的平均 CPU 使用率 (主控台)

1. 開啟主 CloudWatch 控制台，網址為 <https://console.aws.amazon.com/cloudwatch/>。
2. 在導覽窗格中，選擇 指標。
3. 選擇 EC2 命名空間，然後選擇 Across All Instances (在所有執行個體中)。
4. 選擇包含 CPUUtilization 的資料列，這會顯示您所有 EC2 執行個體指標的圖形。若要為圖形命名，請選擇鉛筆圖示。若要變更時間範圍，請選取一個預先定義的值，或選擇 custom (自訂)。

Untitled graph 1h 3h 12h **1d** 3d 1w custom ▾

Actions ▾



■ CPUUtilization

All metrics

Graphed metrics (1)

Graph options

All > EC2 > Across All Instances

<input type="checkbox"/>	Metric Name (7)
<input checked="" type="checkbox"/>	CPUUtilization
<input type="checkbox"/>	DiskReadBytes

- 若要變更指標的統計資料或期間，請選擇 Graphed metrics (圖表化指標) 索引標籤。選擇欄位標題或個別的值，然後選擇不同的值。

取得您所有執行個體的平均 CPU 使用率 (AWS CLI)

如下所示，使用 [get-metric-statistics](#) 命令以取得所有執行個體 CPUUtilization 指標的平均值。

```
aws cloudwatch get-metric-statistics \
 --namespace AWS/EC2 \
 --metric-name CPUUtilization \
 --period 3600 --statistics "Average" "SampleCount" \
 --start-time 2022-10-11T23:18:00 \
 --end-time 2022-10-12T23:18:00
```

下列為範例輸出：

```
{
 "Datapoints": [
```

```
{
 "SampleCount": 238.0,
 "Timestamp": "2022-10-12T07:18:00Z",
 "Average": 0.038235294117647062,
 "Unit": "Percent"
},
{
 "SampleCount": 240.0,
 "Timestamp": "2022-10-12T09:18:00Z",
 "Average": 0.16670833333333332,
 "Unit": "Percent"
},
{
 "SampleCount": 238.0,
 "Timestamp": "2022-10-11T23:18:00Z",
 "Average": 0.041596638655462197,
 "Unit": "Percent"
},
...
],
"Label": "CPUUtilization"
}
```

## 依據 Auto Scaling 群組彙總統計資料

您可以彙總 Auto Scaling 群組中 EC2 執行個體的統計資料。請注意，Amazon CloudWatch 無法跨 AWS 區域彙總資料。區域之間的指標是完全獨立的。

此範例顯示如何擷取一個 Auto Scaling 群組寫入磁碟的總位元組。總計是計算指定 Auto Scaling 群組中的所有 EC2 執行個體，以 24 小時為間隔取得每隔 1 分鐘的數量。

若要顯示 DiskWriteBytes Auto Scaling 群組中的執行個體 (主控台)

1. 開啟主 CloudWatch 控制台，網址為 <https://console.aws.amazon.com/cloudwatch/>。
2. 在導覽窗格中，選擇 指標。
3. 選擇 EC2 命名空間，然後選擇 By Auto Scaling Group (依據 Auto Scaling 群組)。
4. 選擇「DiskWrite位元組」測量結果和特定「Auto Scaling 例」群組的資料列，此群組會顯示「Auto Scaling 例」群組中執行個體的度量圖表。若要為圖形命名，請選擇鉛筆圖示。若要變更時間範圍，請選取一個預先定義的值，或選擇 custom (自訂)。
5. 若要變更指標的統計資料或期間，請選擇 Graphed metrics (圖表化指標) 索引標籤。選擇欄位標題或個別的值，然後選擇不同的值。

## 顯示 DiskWriteBytes Auto Scaling 群組中的例證 (AWS CLI)

使用 [get-metric-statistics](#) 命令，如下所示。

```
aws cloudwatch get-metric-statistics --namespace AWS/EC2 --metric-name DiskWriteBytes
--period 360 \
--statistics "Sum" "SampleCount" --dimensions Name=AutoScalingGroupName,Value=my-asg --
start-time 2022-10-16T23:18:00 --end-time 2022-10-18T23:18:00
```

下列為範例輸出：

```
{
 "Datapoints": [
 {
 "SampleCount": 18.0,
 "Timestamp": "2022-10-19T21:36:00Z",
 "Sum": 0.0,
 "Unit": "Bytes"
 },
 {
 "SampleCount": 5.0,
 "Timestamp": "2022-10-19T21:42:00Z",
 "Sum": 0.0,
 "Unit": "Bytes"
 }
],
 "Label": "DiskWriteBytes"
}
```

## 依據 AMI 的彙總統計資料

您可以為已啟用詳細監控的執行個體彙總統計資料。彙總資料不含使用基本監控的執行個體。在取得所有執行個體的彙總統計資料前，您必須[啟用詳細監控](#) (額外付費)，它以 1 分鐘的間隔時間提供資料。

請注意，Amazon CloudWatch 無法跨 AWS 區域彙總資料。區域之間的指標是完全獨立的。

此範例顯示如何判斷使用特定 Amazon Machine Image (AMI) 之所有執行個體的平均 CPU 使用率。此平均是以一天期間內每 60 秒的時間間隔計算。

依據 AMI 顯示平均 CPU 使用率 (主控台)

1. 開啟主 CloudWatch 控制台，網址為 <https://console.aws.amazon.com/cloudwatch/>。

2. 在導覽窗格中，選擇 指標。
3. 選擇 EC2 命名空間，然後選擇 By Image (AMI) Id (依據映像 (AMI) ID)。
4. 選擇 CPUUtilization 指標與特定 AMI，這會顯示指定 AMI 的指標圖形。若要為圖形命名，請選擇鉛筆圖示。若要變更時間範圍，請選取一個預先定義的值，或選擇 custom (自訂)。
5. 若要變更指標的統計資料或期間，請選擇 Graphed metrics (圖表化指標) 索引標籤。選擇欄位標題或個別的值，然後選擇不同的值。

取得某映像 ID 的平均 CPU 使用率 (AWS CLI)

使用 [get-metric-statistics](#) 命令，如下所示。

```
aws cloudwatch get-metric-statistics --namespace AWS/EC2 --metric-name CPUUtilization
--period 3600 \
--statistics Average --dimensions Name=ImageId,Value=ami-3c47a355 --start-
time 2022-10-10T00:00:00 --end-time 2022-10-11T00:00:00
```

下列為範例輸出。每個值代表執行指定 AMI 之 EC2 執行個體的平均 CPU 使用率百分比。

```
{
 "Datapoints": [
 {
 "Timestamp": "2022-10-10T07:00:00Z",
 "Average": 0.041000000000000009,
 "Unit": "Percent"
 },
 {
 "Timestamp": "2022-10-10T14:00:00Z",
 "Average": 0.079579831932773085,
 "Unit": "Percent"
 },
 {
 "Timestamp": "2022-10-10T06:00:00Z",
 "Average": 0.0360000000000000011,
 "Unit": "Percent"
 },
 ...
],
 "Label": "CPUUtilization"
}
```

## 您的執行個體的圖形指標

啟動執行個體後，您可開啟 Amazon EC2 主控台，在 Monitoring (監控) 索引標籤檢視執行個體的監控圖表。每個圖表都以一個可用的 Amazon EC2 指標為基礎。

下列圖表可供使用：

- 平均 CPU 使用率 (百分比)
- 平均磁碟讀取 (位元組)
- 平均磁碟寫入 (位元組)
- 最大網路輸入 (位元組)
- 最大網路輸出 (位元組)
- 磁碟讀取操作摘要 (計數)
- 磁碟寫入操作摘要 (計數)
- 狀態摘要 (任意)
- 執行個體狀態摘要 (計數)
- 系統狀態摘要 (計數)

如需指標及其向圖表提供資料的詳細資訊，請參閱 [列出執行個體的可用 CloudWatch 指標](#)。

使用 CloudWatch 主控台繪製指標圖形

您也可以使用 CloudWatch 主控台繪製 Amazon EC2 和其他 AWS 服務產生的指標資料圖形。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的繪製指標圖形。

## 建立執行個體的 CloudWatch 警示

您可以建立 CloudWatch 警示來監控其中一個執行個體的 CloudWatch 指標。CloudWatch 當測量結果達到您指定的臨界值時，會自動傳送通知給您。您可以使用 Amazon EC2 主控台或使用主控台提供的更進階選項建立 CloudWatch 警示。CloudWatch

使用 CloudWatch 主控台建立鬧鐘

如需範例，請參閱 [Amazon CloudWatch 使用者指南中的建立 Amazon CloudWatch 警示](#)。

使用 Amazon EC2 主控台建立警示

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。

2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體，然後選擇 [動作]、[監控及疑難排解]、CloudWatch [管理]
4. 在 [管理 CloudWatch 鬧鐘詳細資料] 頁面的 [新增或編輯鬧鐘] 底下，選取 [建立鬧鐘]。
5. 針對警示通知，選擇是否設定 Amazon Simple Notification Service (Amazon SNS) 通知。輸入現有 Amazon SNS 主題或輸入名稱以建立新主題。
6. 在警示動作選擇是否要指定觸發警示時採取的動作。從清單中選擇一個動作。
7. 針對 Alarm thresholds (警示閾值)，選取警示的指標和準則。例如，若要建立會在 CPU 使用率達到 80% 並維持 5 分鐘時觸發的警示，請執行下列操作：
  - a. 保留範例分群依據 (平均) 和要取樣的資料類型 (CPU 使用率) 的預設設定。
  - b. 在警示時機選擇  $\geq$  並在百分比輸入 **0.80**。
  - c. 在連續期間輸入 **1**，並在期間選取 5 分鐘。
8. (選用) 針對 Sample metric data (範例指標資料)，選擇 Add to dashboard (新增至儀表板)。
9. 選擇建立。

您可以從 Amazon EC2 主控台或主控台編輯 CloudWatch 警示設定。CloudWatch 如果您要刪除鬧鐘，可以從 CloudWatch 主控台刪除鬧鐘。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的[編輯或刪除 CloudWatch 警示](#)。

## 建立警示以停止、終止、重新啟動或復原執行個體

使用 Amazon CloudWatch 警示動作，您可以建立自動停止、終止、重新開機或復原執行個體的警示。當執行個體不再需要執行，您可以使用停止或終止動作以協助您節省成本。如果發生系統受損，您可以使用重新啟動和復原動作，自動重新啟動這些執行個體或將它們復原到新的硬體。

### Note

如需 Amazon CloudWatch 警示帳單和定價資訊，請參閱 Amazon CloudWatch 使用者指南中的[CloudWatch 帳單和費用](#)。

AWSServiceRoleForCloudWatchEvents 服務連結角色可 AWS 讓您代表執行警示動作。當您第一次在 AWS Management Console、或 IAM API 中建 CloudWatch 立警示時 AWS CLI，會為您建立服務連結角色。



在許多情況下，您可能想要自動停止或終止您的執行個體。例如，您可能有專門批次處理薪資作業或科學運算任務的執行個體，它們在執行一段時間後完成工作。您不必讓這些執行個體閒置 (及累積費用)，您可以停止或終止它們以協助您節省成本。使用停止和終止警示動作的主要差別是，如果執行個體之後需要重新執行，您將可以輕鬆地啟動已停止的執行個體，而且您可以保持相同的執行個體 ID 和根磁碟區。不過，您無法啟動已終止的執行個體。相反地，您必須啟動新的執行個體。在停止或終止執行個體時，執行個體儲存體磁碟區上的資料會遺失。

您可以將停止、終止、重新啟動或復原動作新增至 Amazon EC2 每個執行個體指標上設定的任何警示，包括 Amazon CloudWatch (在 AWS/EC2 命名空間中) 提供的基本和詳細監控指標，以及包含 InstanceId 維度的任何自訂指標，只要其值參考有效執行中的 Amazon EC2 執行個體即可。

### Important

如果缺少指標資料點，狀態檢查警示可以暫時進入狀態。INSUFFICIENT\_DATA 雖然很少見，但是當指標報告系統發生中斷時，即使執行個體健康狀態良好，也可能發生這種情況。我們建議您將 INSUFFICIENT\_DATA 狀態視為遺失資料，而非警示洩漏，尤其是將警示設定為停止、終止、重新啟動或復原執行個體時。

## 主控台支援

您可以使用 Amazon EC2 主控台或主控台建立警示。CloudWatch 本文件中的程序使用 Amazon EC2 主控台。如需使用 CloudWatch 主控台的程序，請參閱 [Amazon 使用 CloudWatch 者指南中的建立停止、終止、重新開機或復原執行個體的警示](#)。

## 許可

您必須擁有 `iam:CreateServiceLinkedRole` 才能建立或修改執行 EC2 警示動作的警示。服務角色是服務擔任的 [IAM 角色](#)，可代您執行動作。IAM 管理員可以從 IAM 內建立、修改和刪除服務角色。如需更多資訊，請參閱 IAM 使用者指南中的 [建立角色以委派許可給 AWS 服務](#)。

## 目錄

- [向 Amazon CloudWatch 警報添加停止操作](#)
- [將終止動作新增至 Amazon CloudWatch 警示](#)
- [將重新開機動作新增至 Amazon CloudWatch 警示](#)
- [將恢復動作添加到 Amazon CloudWatch 警報](#)
- [使用 Amazon 主 CloudWatch 控制台檢視警示和動作歷史記錄](#)

- [Amazon CloudWatch 警報動作案例](#)

## 向 Amazon CloudWatch 警報添加停止操作

您可以建立警示，在符合特定閾值時停止 Amazon EC2 執行個體。例如，您可以執行開發或測試執行個體，並偶爾忘記關閉它們。您可以建立警示，在平均 CPU 使用率百分比已低於 10% 達 24 小時時觸發，以通知其為閒置且不再使用。您可以調整閾值、持續時間和期間以符合您的需求，而且您可以新增 Amazon Simple Notification Service (Amazon SNS) 通知，當觸發提醒時，您將會收到電子郵件。

使用 Amazon EBS 磁碟區做為根裝置的執行個體可以停止或終止，而使用執行個體存放區做為根裝置的執行個體只能終止。在終止或停止執行個體時，執行個體儲存體磁碟區上的資料會遺失。

### 建立警示以停止閒置的執行個體 (Amazon EC2 主控台)

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體，然後選擇 [動作]、[監控及疑難排解]、CloudWatch [管理


或者，您可以在 Alarm status (警示狀態) 資料欄中選擇加號



)。

4. 在「管理 CloudWatch 警示」頁面上，執行下列動作：
  - a. 選擇 Create an alarm (建立警示)。
  - b. 若要在警示觸發時接收電子郵件，請在 Alarm notification (警示通知) 選擇現有的 Amazon SNS 主題。您首先需要使用 Amazon SNS 主控台建立 Amazon SNS 主題。如需詳細資訊，請參閱 [Amazon 簡單通知服務開發人員指南中的使用 Amazon SNS 進行 application-to-person \(A2P\) 簡訊](#)。
  - c. 切換 Alarm action (警示動作)，然後選擇 Stop (停止)。
  - d. 若要將 Group samples by (分組範例條件) 和 Type of data to sample (要作為範例的資料類型)，請選擇統計資料和指標。在此範例中，選擇 Average (平均) 和 CPU Utilization (CPU 使用率)。
  - e. 針對 Alarm When (警示時間) 和 Percent (百分比)，指定指標閾值。在此範例中，指定 <= 和 10%。
  - f. 針對 Consecutive period (連續期間) 和 Period (期間)，指定警示的評估期間。在此範例中，請指定 1 個每隔 5 Minute (5 分鐘) 的連續期間。

- g. Amazon CloudWatch 會自動為您創建一個警報名稱。若要變更名稱，請在 Alarm name (警示名稱) 中輸入新名稱。警示名稱只能包含 ASCII 字元。

 Note

您可以在建立警示之前根據自己的需求調整警示組態，也可以稍後再進行編輯。這包括指標、閾值、持續時間、動作和通知設定。不過，在您建立警示之後，以後就無法編輯其名稱。

- h. 選擇建立。

## 將終止動作新增至 Amazon CloudWatch 警示

您可以建立警示，在符合特定閾值時自動終止 EC2 執行個體 (前提是執行個體未啟用終止保護)。例如，您可能想要在執行個體完成作業時予以終止，而且不再需要該執行個體。如果您之後還要使用該執行個體，您應該停止而非終止執行個體。在終止執行個體時，執行個體儲存體磁碟區上的資料會遺失。如需為執行個體啟用和停用終止保護的詳細資訊，請參閱 [啟用終止保護](#)。

建立警示以終止閒置的執行個體 (Amazon EC2 主控台)

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體，然後選擇 [動作]、[監控及疑難排解]、CloudWatch [管理


或者，您可以在 Alarm status (警示狀態) 資料欄中選擇加號



)。

4. 在「管理 CloudWatch 警示」頁面上，執行下列動作：
  - a. 選擇 Create an alarm (建立警示)。
  - b. 若要在警示觸發時接收電子郵件，請在 Alarm notification (警示通知) 選擇現有的 Amazon SNS 主題。您首先需要使用 Amazon SNS 主控台建立 Amazon SNS 主題。如需詳細資訊，請參閱 [Amazon 簡單通知服務開發人員指南中的使用 Amazon SNS 進行 application-to-person \(A2P\) 簡訊](#)。
  - c. 切換 Alarm action (警示動作)，然後選擇 Terminate (終止)。

- d. 若要將 Group samples by (分組範例條件) 和 Type of data to sample (要作為範例的資料類型)，請選擇統計資料和指標。在此範例中，選擇 Average (平均) 和 CPU Utilization (CPU 使用率)。
- e. 針對 Alarm When (警示時間) 和 Percent (百分比)，指定指標閾值。在此範例中，指定  $\geq$  和 10%。
- f. 針對 Consecutive period (連續期間) 和 Period (期間)，指定警示的評估期間。在此範例中，請指定 24 個每隔 1 Hour (1 小時) 的連續期間。
- g. Amazon CloudWatch 會自動為您創建一個警報名稱。若要變更名稱，請在 Alarm name (警示名稱) 中輸入新名稱。警示名稱只能包含 ASCII 字元。

 Note


您可以在建立警示之前根據自己的需求調整警示組態，也可以稍後再進行編輯。這包括指標、閾值、持續時間、動作和通知設定。不過，在您建立警示之後，以後就無法編輯其名稱。

- h. 選擇建立。

## 將重新開機動作新增至 Amazon CloudWatch 警示

您可以建立 Amazon CloudWatch 警示來監控 Amazon EC2 執行個體並自動重新啟動執行個體。重新啟動警示動作建議用於執行個體運作狀態檢查失敗 (相對的，復原警示動作則適用於系統運作狀態檢查失敗)。重新啟動執行個體等同於重新啟動作業系統。在大多數情況下，將執行個體重新開機只需要幾分鐘的時間。當您重新啟動執行個體時，它會維持在相同的實體主機上，所以您的執行個體會保有其公有 DNS 名稱、私有 IP 地址和執行個體存放區磁碟區上的任何資料。

有別於停用和重新開始執行個體，重新啟動執行個體不會啟動新的執行個體計費時段 (最少收費一分鐘)。在重新啟動執行個體時，會保留執行個體儲存體磁碟區上的資料。執行個體儲存體磁碟區必須在重新啟動後重新掛載到檔案系統中。如需詳細資訊，請參閱 [重新啟動您的執行個體](#)。

 Important

為了避免重新開機和復原動作之間的競爭情況，請避免為重新開機警示和復原警示設定相同的評估期間數。我們建議您將重新開機警示設定為三個各一分鐘的評估期間。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的 [評估警示](#)。

## 建立警示以重新啟動執行個體 (Amazon EC2 主控台)

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體，然後選擇 [動作]、[監控及疑難排解]、CloudWatch [管理  
或者，您可以在 Alarm status (警示狀態) 資料欄中選擇加號  
(+) )。
4. 在「管理 CloudWatch 警示」頁面上，執行下列動作：
  - a. 選擇 Create an alarm (建立警示)。
  - b. 若要在警示觸發時接收電子郵件，請在 Alarm notification (警示通知) 選擇現有的 Amazon SNS 主題。您首先需要使用 Amazon SNS 主控台建立 Amazon SNS 主題。如需詳細資訊，請參閱 [Amazon 簡單通知服務開發人員指南中的使用 Amazon SNS 進行 application-to-person \(A2P\) 簡訊](#)。
  - c. 切換 Alarm action (警示動作)，然後選擇 Reboot (重新開機)。
  - d. 若要將 Group samples by (分組範例條件) 和 Type of data to sample (要作為範例的資料類型)，請選擇統計資料和指標。在此範例中，選擇 Average (平均) 和 Status check failed: instance (狀態檢查失敗：執行個體)。
  - e. 針對 Consecutive period (連續期間) 和 Period (期間)，指定警示的評估期間。在此範例中，請輸入 3 個每隔 5 Minutes (5 分鐘) 的連續期間。
  - f. Amazon CloudWatch 會自動為您創建一個警報名稱。若要變更名稱，請在 Alarm name (警示名稱) 中輸入新名稱。警示名稱只能包含 ASCII 字元。
  - g. 選擇建立。

## 將恢復動作添加到 Amazon CloudWatch 警報

您可以建立監控 Amazon EC2 執行個體的亞馬遜 CloudWatch 警示。如果執行個體因為基礎硬體故障或需要 AWS 參與修復的問題而受損，您可以自動復原執行個體。已終止的執行個體無法復原。復原後的執行個體與原始執行個體相同，包括執行個體 ID、私有 IP 地址、彈性 IP 地址及所有執行個體中繼資料。

CloudWatch 防止您將復原動作新增至不支援復原動作的執行個體上的警示。

當 StatusCheckFailed\_System 提醒觸發且復原動作啟動時，您會收到 Amazon SNS 主題通知，這是您在建立提醒時選擇並與復原動作關聯的通知。在執行個體復原期間，執行個體會重新啟動期間

遷移，記憶體內的任何資料都將遺失。當程序完成時，會將資訊發佈到您為此警示設定的 SNS 主題。訂閱此 SNS 主題的所有使用者都會收到電子郵件通知，其中包含復原嘗試的狀態和進一步的說明。您會發現執行個體在已復原的執行個體上重新啟動。

#### Note

復原動作只能用於 `StatusCheckFailed_System`，而非 `StatusCheckFailed_Instance`。

下列為可能導致系統狀態檢查失敗的問題：

- 網路連線中斷
- 系統電力中斷
- 實體主機的軟體問題
- 實體主機上會影響網路連線的硬體問題

復原動作僅在符合某些特性的執行個體上受到支援：如需詳細資訊，請參閱 [執行個體恢復](#)。

如果您的執行個體具有公有 IP 地址，它將在復原後保留公有 IP 地址。

#### Important

為了避免重新開機和復原動作之間的競爭情況，請避免為重新開機警示和復原警示設定相同的評估期間數。我們建議您將復原警示設定為兩個各一分鐘的評估期間。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的 [評估警示](#)。

建立警示以復原執行個體 (Amazon EC2 主控台)


1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體，然後選擇 [動作]、[監控及疑難排解]、CloudWatch [管理

或者，您可以在 Alarm status (警示狀態) 資料欄中選擇加號

(+)

)。

4. 在「管理 CloudWatch 警示」頁面上，執行下列動作：
  - a. 選擇 Create an alarm (建立警示)。
  - b. 若要在警示觸發時接收電子郵件，請在 Alarm notification (警示通知) 選擇現有的 Amazon SNS 主題。您首先需要使用 Amazon SNS 主控台建立 Amazon SNS 主題。如需詳細資訊，請參閱 [Amazon 簡單通知服務開發人員指南中的使用 Amazon SNS 進行 application-to-person \(A2P\) 簡訊](#)。

 Note

使用者必須訂閱指定的 SNS 主題，以在警示觸發時收到電子郵件通知。即使未指定 SNS 主題或根使用者未訂閱指定的 SNS 主題，也—AWS 帳戶根使用者 律會在執行個體復原動作發生時收到電子郵件通知。

- c. 切換 Alarm action (警示動作)，然後選擇 Recover (復原)。
- d. 若要將 Group samples by (分組範例條件) 和 Type of data to sample (要作為範例的資料類型)，請選擇統計資料和指標。在此範例中，選擇 Average (平均) 和 Status check failed: system (狀態檢查失敗：系統)。
- e. 針對 Consecutive period (連續期間) 和 Period (期間)，指定警示的評估期間。在此範例中，請輸入 2 個每隔 5 Minutes (5 分鐘) 的連續期間。
- f. Amazon CloudWatch 會自動為您創建一個警報名稱。若要變更名稱，請在 Alarm name (警示名稱) 中輸入新名稱。警示名稱只能包含 ASCII 字元。
- g. 選擇建立。

## 使用 Amazon 主 CloudWatch 控制台檢視警示和動作歷史記錄

您可以在 Amazon CloudWatch 主控台中檢視警示和動作歷史記錄。Amazon CloudWatch 保留了最近兩週的警報和行動歷史。

若要檢視觸發警示和動作的歷史記錄 (CloudWatch 主控台)

1. 開啟主 CloudWatch 控制台，[網址為 https://console.aws.amazon.com/cloudwatch/](https://console.aws.amazon.com/cloudwatch/)。
2. 在導覽窗格中，選擇警示。
3. 選擇警示。
4. Details (詳細資訊) 標籤會顯示隨著時間轉換的最新狀態及指標值。
5. 選擇 History (歷史記錄) 標籤可檢視最新的歷史記錄項目。

## Amazon CloudWatch 警報動作案例

您可以使用 Amazon EC2 主控台來建立提醒動作，在滿足特定條件時停止或終止 Amazon EC2 執行個體。在下列用來設定警示動作的主控台頁面螢幕截圖中，我們為設定加上編號。我們也在隨後的案例中為設定加上編號，協助您建立適當的動作。

### New console

The screenshot shows the Amazon CloudWatch Alarm configuration console. It is divided into three main sections: Alarm notification, Alarm action, and Alarm thresholds. Each section has a title, a subtitle, and a toggle switch. The Alarm notification section has a search input field. The Alarm action section has a dropdown menu. The Alarm thresholds section has several input fields and dropdown menus. Numbered callouts (1-7) are placed over various elements to highlight them.

**Alarm notification** [Info](#)

Configure the alarm to send notifications to an Amazon SNS topic when it is triggered.

**1**

**Alarm action** [Info](#)

Specify the action to take when the alarm is triggered.

**Alarm thresholds**

Specify the metric thresholds for the alarm.

Group samples by  age

Type of data to sample

Alarm When

Consecutive Period  Period  nutes

Alarm name



## Old console

### Create Alarm ✕

You can use CloudWatch alarms to be notified automatically whenever metric data reaches a level you define.  
To edit an alarm, first choose whom to notify and then define when the notification should be sent.

**1**  Send a notification to:  [create topic](#)

Take the action:

- Recover this instance (i)
- Stop this instance (i)
- Terminate this instance (i)
- Reboot this instance (i)

---

Whenever: **2**  of **3**

Is: **4**  **5**  Percent

For at least: **6**  consecutive period(s) of **7**

Name of alarm:

Cancel
Create Alarm

**CPU Utilization Percent**

## 案例 1：停止閒置的開發和測試執行個體

建立警示，在用於軟體開發或測試的執行個體已閒置至少一小時後，將其停止。

設定	Value
1	停止
2	最大
3	CPU 使用率
4	<=
5	10%
6	1
7	1 小時

## 案例 2：停止閒置的執行個體

建立警示，在執行個體已閒置 24 小時後，停止執行個體並傳送電子郵件。

設定	Value
1	停止和寄送電子郵件
2	平均數
3	CPU 使用率
4	<=
5	5%
6	24
7	1 小時

## 案例 3：針對發生異常高流量的 Web 伺服器傳送電子郵件

建立警示，在執行個體每天超過 10 GB 傳出網路流量後，傳送電子郵件。

設定	Value
1	電子郵件
2	總和
3	網路輸出
4	>
5	10 GB
6	24
7	1 小時

## 案例 4：停止發生異常高流量的 Web 伺服器

建立警示，在傳出流量每小時超過 1 GB 後，停止執行個體並傳送簡訊 (SMS)。

設定	Value
1	停止並傳送簡訊
2	總和
3	網路輸出
4	>
5	1 GB
6	1
7	1 小時

## 案例 5：停止受損的執行個體

建立警示，停止未通過三次連續運作狀態檢查的執行個體 (以 5 分鐘間隔執行)。

設定	Value
1	停止
2	平均數
3	狀態檢查失敗：系統
4	-
5	-
6	1
7	15 分鐘

## 案例 6：批次處理任務完成時終止執行個體

建立警示，當執行批次任務的執行個體不再傳送結果資料時，終止該執行個體。

設定	Value
1	終止
2	最大
3	網路輸出
4	<=
5	100,000 位元組
6	1
7	5 分鐘

## 使用自動化 Amazon EC2 EventBridge

您可以使用 Amazon EventBridge 自動化您的系統事件 AWS 服務 並自動回應，例如應用程式可用性問題或資源變更。來自 AWS 服務的事件會以近乎即時 EventBridge 的方式傳送到。您可建立規則來指示您在意的事件，以及當事件符合規則時執行的動作。可以自動觸發的動作如下：

- 調用一個 AWS Lambda 函數
- 叫用 Amazon EC2 執行命令
- 將事件轉送至 Amazon Kinesis Data Streams
- 啟動 AWS Step Functions 狀態機
- 通知 Amazon SNS 主題
- 通知 Amazon SQS 佇列

以下是如何 EventBridge 與 Amazon EC2 搭配使用的範例：

- 每當執行個體進入執行狀態時，啟用 Lambda 函數。
- 建立或修改 Amazon EBS 磁碟區時通知 Amazon SNS 主題。

- 每當另一個 AWS 服務中發生特定事件時，都可以使用 Amazon EC2 執行命令將命令傳送至一或多個 Amazon EC2 執行個體。

如需詳細資訊，請參閱 [Amazon EventBridge 使用者指南](#)。

## Amazon EC2 事件類型

Amazon EC2 支援以下事件類型：

- [EC2 AMI 狀態變更](#)
- [EC2 快速啟動狀態變更通知](#)
- [EC2 機群錯誤](#)
- [EC2 機群資訊](#)
- [EC2 機群執行個體變更](#)
- [EC2 機群 Spot 執行個體請求變更](#)
- [EC2 機群狀態變更](#)
- [EC2 執行個體重新平衡推薦](#)
- [EC2 執行個體狀態 - 變更通知](#)
- [EC2 Spot 機群錯誤](#)
- [EC2 Spot 機群資訊](#)
- [EC2 Spot 機群執行個體變更](#)
- [EC2 Spot 機群 Spot 執行個體請求變更](#)
- [EC2 Spot 機群狀態變更](#)
- [EC2 Spot 執行個體中斷警告](#)
- [EC2 Spot 執行個體請求履行](#)
- [EC2 ODCR 使用率不足通知](#)

如需 Amazon EBS 支援之事件類型的相關資訊，請參閱 [Amazon EBS 相EventBridge 關](#)資訊。

## 使用記錄 Amazon EC2 API 呼叫 AWS CloudTrail

Amazon EC2 API 已與服務整合 AWS CloudTrail，該服務可提供使用者、角色或角色所採取的動作記錄 AWS 服務。CloudTrail 以事件形式擷取 Amazon EC2 的所有 API 呼叫，包括從主控台呼叫以及從

程式碼呼叫到 API 作業的呼叫。使用收集的資訊 CloudTrail，您可以判斷向 Amazon EC2 API 發出的請求、提出請求的來源 IP 地址、發出請求的時間等。

若要進一步了解 CloudTrail，請參閱使[AWS CloudTrail 用者指南](#)。

## Amazon EC2 API 信息 CloudTrail

CloudTrail 在您創建帳戶 AWS 帳戶時啟用。當 Amazon EC2 和 Amazon EBS 中發生活動時，該活動會與事件歷史記錄中的其他 CloudTrail 事件一起記錄在 AWS 服務事件中。您可以檢視、搜尋和下載 AWS 帳戶的最新事件。如需詳細資訊，請參閱[使用 CloudTrail 事件歷程記錄檢視事件](#)。

若要不持續記錄 AWS 帳戶中的事件 (包括 Amazon EC2 和 Amazon EBS 事件)，請建立追蹤。追蹤可 CloudTrail 將日誌檔交付到 Amazon S3 儲存貯體。依預設，當您在主控台中建立追蹤時，該追蹤會套用至所有的 AWS 區域。追蹤記錄來自 AWS 分區中所有區域的事件，並將日誌檔傳送到您指定的 Amazon S3 儲存貯體。此外，您還可以設定其他 AWS 服務，以進一步分析 CloudTrail 記錄中收集的事件資料並採取行動。如需詳細資訊，請參閱：

- [為您的建立追蹤 AWS 帳戶](#)
- [AWS 服務與 CloudTrail 記錄檔整合](#)
- [設定 Amazon SNS 通知 CloudTrail](#)
- [從多個區域接收 CloudTrail 日誌文件和從多個帳戶接收 CloudTrail 日誌文件](#)

所有 Amazon EC2 動作和亞馬遜 EBS 管理動作都由記錄下來，CloudTrail 並記錄在 [Amazon EC2 API 參考](#)中。例如，呼叫[RunInstancesDescribeInstances](#)、或[CreateImage](#)動作會在 CloudTrail 記錄檔中產生項目。

每一筆事件或日誌專案都會包含產生請求者的資訊。身分資訊可協助您判斷下列事項：

- 是否使用根使用者或 IAM 使用者憑證提出該請求。
- 提出該請求時，是否使用了特定角色或聯合身分使用者的暫時安全憑證。
- 該請求是否由另一項 AWS 服務服務提出。

如需詳細資訊，請參閱[CloudTrail user identity 元素](#)。

## 瞭解 Amazon EC2 API 日誌檔項目

追蹤是一種組態，可讓事件以日誌檔的形式傳遞到您指定的 Amazon S3 儲存貯體。CloudTrail 記錄檔包含一或多個記錄項目。事件代表來自任何來源的單一請求，包括有關請求的操作，動作的日期和時

間，請求參數等信息。CloudTrail 日誌文件不是公共 API 調用的有序堆棧跟踪，因此它們不會以任何特定順序顯示。

以下日誌檔記錄顯示某位使用者終止了執行個體。

```
{
 "Records": [
 {
 "eventVersion": "1.03",
 "userIdentity": {
 "type": "Root",
 "principalId": "123456789012",
 "arn": "arn:aws:iam::123456789012:root",
 "accountId": "123456789012",
 "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
 "userName": "user"
 },
 "eventTime": "2016-05-20T08:27:45Z",
 "eventSource": "ec2.amazonaws.com",
 "eventName": "TerminateInstances",
 "awsRegion": "us-west-2",
 "sourceIPAddress": "198.51.100.1",
 "userAgent": "aws-cli/1.10.10 Python/2.7.9 Windows/7botocore/1.4.1",
 "requestParameters": {
 "instancesSet": {
 "items": [
 {
 "instanceId": "i-1a2b3c4d"
 }
]
 }
 },
 "responseElements": {
 "instancesSet": {
 "items": [
 {
 "instanceId": "i-1a2b3c4d",
 "currentState": {
 "code": 32,
 "name": "shutting-down"
 },
 "previousState": {
 "code": 16,
 "name": "running"
 }
 }
]
 }
 }
 }
]
}
```

```
 },
 "requestID": "be112233-1ba5-4ae0-8e2b-1c302EXAMPLE",
 "eventID": "6e12345-2a4e-417c-aa78-7594fEXAMPLE",
 "eventType": "AwsApiCall",
 "recipientAccountId": "123456789012"
 }
]
}
```

## 用 AWS CloudTrail 於稽核使用 EC2 執行個體連線建立的 Connect 線

用 AWS CloudTrail 於稽核透過 EC2 執行個體 Connect 至執行個體的使用者。

使用 AWS CloudTrail 主控台透過 EC2 執行個體 Connect 稽核 SSH 活動

1. 開啟主 CloudTrail 控台，網址為 <https://console.aws.amazon.com/cloudtrail/>。
2. 確認您位於正確的區域。
3. 在導覽窗格中，選擇 Event history (事件歷史記錄)。
4. 對於 Filter (篩選條件)，請選擇 Event source (事件來源)、ec2-instance-connect.amazonaws.com。
5. (選用) 針對 Time range (時間範圍)，選取時間範圍。
6. 選擇 Refresh events (重新整理事件) 圖示。
7. 頁面會顯示與 [SendSSHPublicKey](#) API 呼叫對應的事件。使用箭頭展開事件以檢視其他詳細資料，例如用來建立 SSH 連線的使用者名稱和 AWS 存取金鑰，以及來源 IP 位址。
8. 若要以 JSON 格式顯示完整事件資訊，請選擇 View event (檢視事件)。requestParameters 欄位包含用來進行 SSH 連線的目的地執行個體 ID、OS 使用者名稱和公開金鑰。

```
{
 "eventVersion": "1.05",
 "userIdentity": {
 "type": "IAMUser",
 "principalId": "ABCDEFGONGNOM00CB6XYTQEXAMPLE",
 "arn": "arn:aws:iam::1234567890120:user/IAM-friendly-name",
 "accountId": "123456789012",
 "accessKeyId": "ABCDEFGUKZHNAW40SN2AEXAMPLE",
 "userName": "IAM-friendly-name",
 "sessionContext": {
 "attributes": {
 "mfaAuthenticated": "false",
```



```
 "creationDate": "2018-09-21T21:37:58Z"}
 },
 "eventTime": "2018-09-21T21:38:00Z",
 "eventSource": "ec2-instance-connect.amazonaws.com",
 "eventName": "SendSSHPublicKey ",
 "awsRegion": "us-west-2",
 "sourceIPAddress": "123.456.789.012",
 "userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
 "requestParameters": {
 "instanceId": "i-0123456789EXAMPLE",
 "osUser": "ec2-user",
 "SSHKey": {
 "publicKey": "ssh-rsa ABCDEFGHIJKLMN001234567890EXAMPLE"
 }
 },
 "responseElements": null,
 "requestID": "1a2s3d4f-bde6-11e8-a892-f7ec64543add",
 "eventID": "1a2w3d4r5-a88f-4e28-b3bf-30161f75be34",
 "eventType": "AwsApiCall",
 "recipientAccountId": "0987654321"
}
```

如果您已將 AWS 帳戶設定為收集 S3 儲存貯體中的 CloudTrail 事件，則可以透過程式設計方式下載和稽核資訊。若要[取得更多資訊](#)，請參閱《[AWS CloudTrail 使用指南](#)》中的〈[取得和檢視 CloudTrail 記錄檔](#)〉。

## 使用應用程式洞察來監視您的 .NET 和 SQL Server CloudWatch 應用程式

CloudWatch 應用程式洞見可協助您監控使用 Amazon EC2 執行個體以及其他應用程式[資源的 .NET 和 SQL Server 應用AWS 程式](#)。它會識別和設定金鑰指標日誌，並在您所有的應用程式資源和技術堆疊 (例如，Microsoft SQL Server 資料庫、Web (IIS) 和應用程式伺服器、作業系統、負載平衡器和佇列) 中發出警示。並持續監控指標和日誌，以偵測和建立異常及錯誤的關聯。偵測到錯誤和異常時，「應用程式深入解析」會產生[CloudWatch 事件](#)，讓您用來設定通知或採取動作。為協助故障診斷，它會建立已偵測到問題的自動化儀表板，包括關聯的指標異常和日誌錯誤以及其他洞見，指出可能的根本原因。自動化儀表板可協助您採用靈活的補救動作，讓應用程式保持良好的運作狀態，防止應用程式的最終使用者受到影響。

若要檢視支援的日誌和指標的完整清單，請參閱 [Amazon CloudWatch 應用程式洞察支援的日誌和指標](#)。

有關偵測到的問題資訊：

- 問題的簡短摘要
- 問題的開始時間和日期
- 問題嚴重性：高/中/低
- 偵測到的問題狀態：正在進行/已解決
- 洞見：對所偵測問題及可能根本原因自動產生的洞見
- 深入解析的意見反應：您所提供的意見反應，瞭解由 .NET 和 SQL Server 的 CloudWatch 應用程式深入解析產生的實用性
- 相關觀察：詳細檢視與跨各種應用程式元件問題有關的相關日誌指標異常和錯誤程式碼片段

### 意見回饋

您可以指出針對所偵測問題自動產生的洞見是否有用，提供意見回饋。您對洞見的意見回饋以及您的應用程式診斷 (指標異常和日誌例外狀況)，都會用來提升未來對類似問題的偵測。

如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的應用 CloudWatch [程式洞察](#) 文件。

## 追蹤您的 Amazon EC2 免費方案使用情況

如果您是 AWS 客戶少於 12 個月且仍在用 AWS 免費方案 量限制範圍內，則可以使用 Amazon EC2 而不會產生任何費用。請務必要追蹤免費方案的用量，以避免出現意外的帳單費用。如果您超過免費方案限制，就會產生標準 pay-as-go 費用。

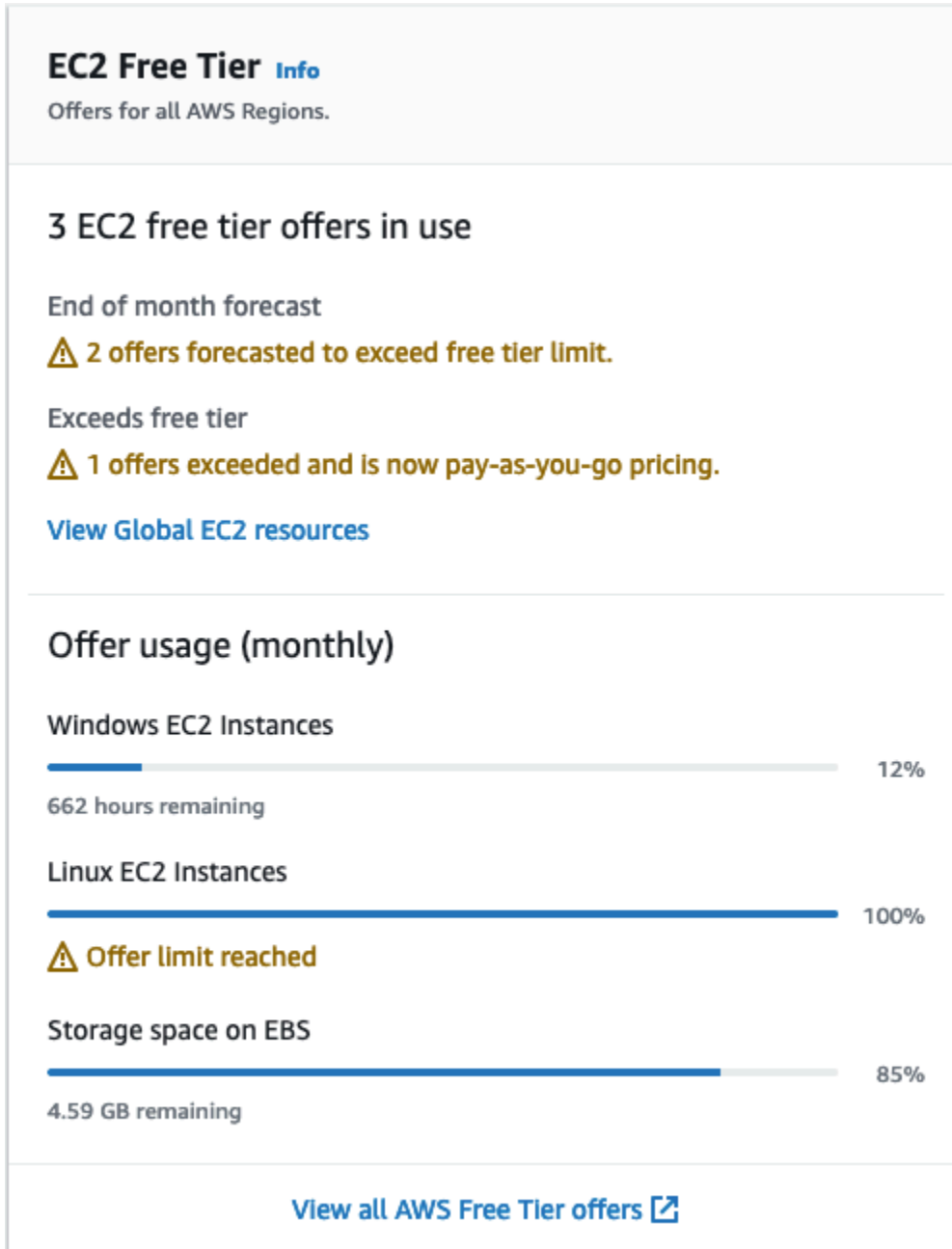
### Note

如果您成為 AWS 客戶超過 12 個月，就不再符合免費方案使用資格，而且不會看到下列程序中所述的 EC2 免費方案方塊。

### 追蹤您的免費方案用量

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 EC2 Dashboard (EC2 儀表板)。


### 3. 尋找 EC2 免費方案方塊 (位於右上角)。



### 4. 在 EC2 免費方案方塊中，檢查您的免費方案用量，如下所示：

- 在使用中的 EC2 免費方案優惠下，請注意以下警告：
  - 月底預測 – 這項警告會提醒您，若您繼續目前的用量模式，則本月將會產生費用。
  - 超過免費方案 – 這項警告會提醒您，已超過了免費方案限制且已產生費用。

- 請根據優惠用量 (每月) 注意您對 Linux 執行個體、Windows 執行個體和 EBS 儲存體的用量。該百分比表示您本月使用了多少免費方案限制。如果您達到 100%，則進一步使用將會產生費用。

 Note

本資訊僅限於在您建立了執行個體之後才會顯示。不過，用量資訊並非即時更新；其更新頻率為一天三次。

5. 為了避免產生進一步的費用，請刪除任何現在產生費用的資源，否則若超過免費方案限制用量，將會產生費用。
  - 如需有關刪除執行個體的指示，請移至本教學課程中的下一個步驟。
  - 若要檢查其他區域中是否有可能產生費用的資源，請在 EC2 免費方案方塊中選擇檢視全域 EC2 資源以開啟 EC2 全域檢視。如需詳細資訊，請參閱 [Amazon EC2 全域檢視](#)。
6. 若要在  AWS 服務 下方檢視所有人的資源使用量 AWS 免費方案，請在 EC2 免費方案方塊底部選擇 [檢視所有 AWS 免費方案 優惠]。如需詳細資訊，請參閱《AWS 帳單使用者指南》中的 [使用 AWS 免費方案](#)。

# 在 Amazon EC2 中聯網

Amazon VPC 可讓您將 AWS 資源 (例如 Amazon EC2 執行個體) 啟動到專屬於您 AWS 帳戶的虛擬網路 (稱為虛擬私有雲 (VPC))。啟動執行個體時，您可以從 VPC 中選取子網。執行個體設定為主要網路介面，這是邏輯虛擬網路卡。執行個體會從子網的 IPv4 地址接收主要私人 IP 地址，並將其指派給主要網路介面。

您可以控制執行個體是否從 Amazon 的公有 IP 地址集區接收公有 IP 地址。執行個體的公有 IP 地址只會與您的執行個體關聯，直到執行個體停止或終止為止。如果您需要永久的公用 IP 位址，您可以為您的 AWS 帳戶配置彈性 IP 位址，並將其與執行個體或網路介面建立關聯。彈性 IP 地址會與您的 AWS 帳戶保持關聯，直到您釋放它為止，您可以根據需要將其從一個實例移動到另一個實例。您可以將自己的 IP 地址範圍帶到您的 AWS 帳戶中，它會顯示為地址集區，然後從您的地址集區配置彈性 IP 地址。

若要提高網路效能並減少延遲，您可以在配置群組中啟動執行個體。使用增強型聯網，您可以獲得大幅提高的每秒封包數 (PPS) 效能。您可以使用 Elastic Fabric Adapter (EFA)，即彈性架構轉接器來加速高效能運算和機器學習應用程式，這是您可以連接至支援的執行個體類型的網路裝置。

## 功能

- [區域 \(Region\) 和區域 \(Zone\)](#)
- [Amazon EC2 執行個體 IP 定址](#)
- [Amazon EC2 執行個體主機名稱類型](#)
- [在 Amazon EC2 中使用自有 IP 地址 \(BYOIP\)](#)
- [彈性 IP 地址](#)
- [彈性網路介面](#)
- [Amazon EC2 執行個體網路頻寬](#)
- [Amazon EC2 上的增強型聯網](#)
- [Elastic Fabric Adapter](#)
- [Amazon EC2 執行個體拓撲](#)
- [置放群組](#)
- [EC2 執行個體的網路最大傳輸單位 \(MTU\)](#)
- [適用於 EC2 執行個體的虛擬私有雲](#)

## 區域 (Region) 和區域 (Zone)

Amazon EC2 託管於全球的多個地點。這些位置由可用區域 AWS 區域、Local Zones 和 Wavelength 區組成。AWS Outposts

- 各個區域為獨立的地理區域。
- 可用區域是每個區域內的多個隔離位置。
- Local Zones 可讓您將資源 (例如運算和儲存) 放置在靠近最終使用者的多個位置。
- AWS Outposts 將原生 AWS 服務、基礎架構和作業模型帶到幾乎任何資料中心、主機代管空間或內部部署設施。
- Wavelength 區域可讓開發人員為 5G 裝置與最終使用者建立提供極低延遲的應用程式。Wavelength 將標準 AWS 運算和儲存服務部署到電信業者的 5G 網路邊緣。

AWS 運作 state-of-the-art 的高可用性資料中心。儘管故障極為少見，但仍可能影響相同位置內執行個體的可用性。若您將所有執行個體都託管於單一位置，一旦該位置受故障影響，所有執行個體都將無法使用。

為了協助您判斷哪一個部署最適合您，請參閱 [AWS Wavelength 常見問答集](#)。

### 目錄

- [區域](#)
- [可用區域](#)
- [本機區域](#)
- [Wavelength 區域](#)
- [AWS Outposts](#)

## 區域

每個區域皆設計為與其他區域隔離。如此可達到最高的容錯能力與穩定性。

當檢視資源時，您只會看到與您指定的區域繫結的資源。這是因為區域彼此隔離，而且我們不會自動跨區域複製資源。

啟動執行個體時，您必須選取在相同區域內的 AMI。如果 AMI 位在其他區域，您可將 AMI 複製到正在使用的區域。如需詳細資訊，請參閱 [複製 AMI](#)。

請注意，跨區域傳輸資料會產生費用。如需詳細資訊，請參閱 [Amazon EC2 定價 - 資料傳輸](#)。

## 內容

- [可用的區域](#)
- [區域與端點](#)
- [描述您的地區](#)
- [取得區域顯示名稱](#)
- [指定資源的區域](#)

## 可用的區域

您的帳戶決定您可用的區域。

- AWS 帳戶 提供多個區域，因此您可以在符合需求的位置啟動 Amazon EC2 執行個體。例如，您可能想在歐洲啟動執行個體，以拉近與歐洲客戶的距離或符合法規要求。
- AWS GovCloud (美國西部) 帳戶可存取 AWS GovCloud (美國西部) 區域和 AWS GovCloud (美國東部) 區域。如需詳細資訊，請參閱 [AWS GovCloud \(US\)](#)。
- Amazon AWS (中國) 帳戶僅提供北京和寧夏地區的存取權。如需詳細資訊，請參閱 [Amazon Web Services in China](#) (Amazon Web Services (中國))。

下表列出由提供的區域 AWS 帳戶。您無法描述或存取其他地區 AWS 帳戶，例如 AWS GovCloud (US) Regions 或中國區域。若要使用 2019 年 3 月 20 日之後引進的區域，您必須啟用該區域。如需詳細資訊，請參閱 AWS Account Management 參考指南 [中的指定帳戶可以使用的 AWS 區域](#)。

Code	名稱	選擇參加狀態
us-east-2	美國東部 (俄亥俄)	非必要
us-east-1	美國東部 (維吉尼亞)	非必要
us-west-1	美國西部 (加州北部)	非必要
us-west-2	美國西部 (奧勒岡)	非必要
af-south-1	非洲 (開普敦)	必要
ap-east-1	亞太區域 (香港)	必要
ap-south-2	亞太區域 (海德拉巴)	必要

Code	名稱	選擇參加狀態
ap-southeast-3	亞太區域 (雅加達)	必要
ap-southeast-4	亞太區域 (墨爾本)	必要
ap-south-1	亞太區域 (孟買)	非必要
ap-northeast-3	亞太區域 (大阪)	非必要
ap-northeast-2	亞太區域 (首爾)	非必要
ap-southeast-1	亞太區域 (新加坡)	非必要
ap-southeast-2	亞太區域 (悉尼)	非必要
ap-northeast-1	亞太區域 (東京)	非必要
ca-central-1	加拿大 (中部)	非必要
ca-west-1	加拿大西部 (卡加利)	必要
eu-central-1	歐洲 (法蘭克福)	非必要
eu-west-1	歐洲 (愛爾蘭)	非必要
eu-west-2	歐洲 (倫敦)	非必要
eu-south-1	歐洲 (米蘭)	必要
eu-west-3	Europe (Paris)	非必要
eu-south-2	歐洲 (西班牙)	必要
eu-north-1	歐洲 (斯德哥爾摩)	非必要
eu-central-2	歐洲 (蘇黎世)	必要
il-central-1	以色列 (特拉維夫)	必要
me-south-1	Middle East (Bahrain)	必要



Code	名稱	選擇參加狀態
me-central-1	中東 (阿拉伯聯合大公國)	必要
sa-east-1	南美洲 (聖保羅)	非必要

如需詳細資訊，請參閱 [AWS 全球基礎設施](#)。

每區域的可用區域數量和映射可能因 AWS 帳戶 而不同。若要列出帳戶可用區域的清單，可用 Amazon EC2 主控台或命令列介面。如需詳細資訊，請參閱 [描述您的地區](#)。

## 區域與端點

使用命令列介面或 API 動作操作執行個體時，您必須指定其區域端點。如需有關 Amazon EC2 的區域與端點的詳細資訊，請參閱《Amazon Web Services 一般參考》中的 [Amazon EC2 端點與配額](#)。

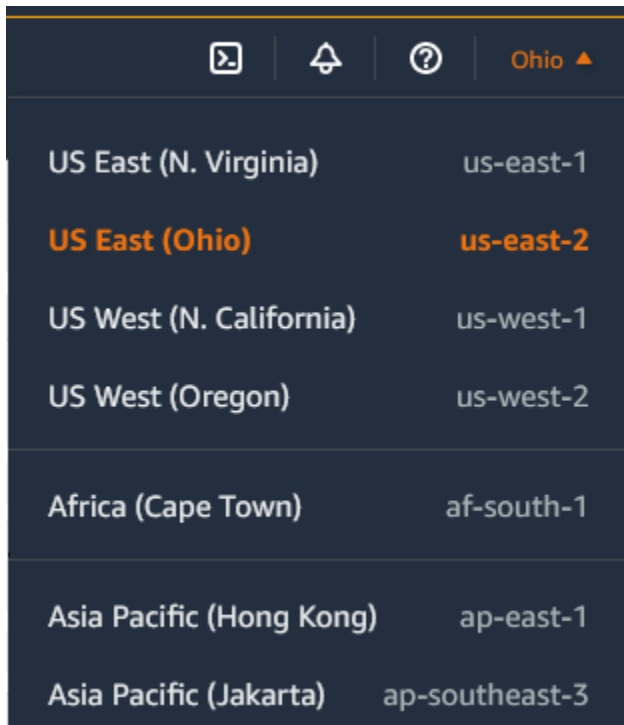
如需有關 AWS GovCloud (美國西部) 中端點和通訊協定的詳細資訊，請參閱 AWS GovCloud (US) 使用手冊中的「[服務端點](#)」。

## 描述您的地區

您可用 Amazon EC2 主控台或命令列介面決定帳戶適用的區域。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

### 使用主控台尋找您的區域

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 從導覽列中，選取 Regions (區域) 選取器。



Region Name	Region Code
US East (N. Virginia)	us-east-1
<b>US East (Ohio)</b>	<b>us-east-2</b>
US West (N. California)	us-west-1
US West (Oregon)	us-west-2
Africa (Cape Town)	af-south-1
Asia Pacific (Hong Kong)	ap-east-1
Asia Pacific (Jakarta)	ap-southeast-3

3. 所選區域的 EC2 資源會顯示在資源區段中的 EC2 儀表板。

若要尋找您的區域 AWS CLI

使用 [describe-regions](#) 命令描述針對您帳戶啟用的區域，如下所示。

```
aws ec2 describe-regions
```

若要描述所有區域，包括針對您帳戶停用的區域，請新增 `--all-regions` 選項，如下所示。

```
aws ec2 describe-regions --all-regions
```

## 取得區域顯示名稱

您可以使用「AWS Systems Manager 參數存放區」來檢視「區域」的顯示名稱。每個區域在以下路徑中包含公有參數。

```
/aws/service/global-infrastructure/regions/region-code
```

區域的公有參數包括以下內容：

- `/aws/service/global-infrastructure/regions/region-code/domain`

- /aws/service/global-infrastructure/regions/*region-code*/geolocationCountry
- /aws/service/global-infrastructure/regions/*region-code*/geolocationRegion
- /aws/service/global-infrastructure/regions/*region-code*/longName
- /aws/service/global-infrastructure/regions/*region-code*/partition

longName 參數包含區域顯示名稱。以下 [get-parameters-by-path](#) 命令傳回 af-south-1 區域的顯示名稱。該命令使用 --query 選項將輸出範圍定為區域的名稱。您必須在 Linux 上用單引號括住查詢字串。若要使用 Windows 命令提示執行此命令，請省略單引號或將它們變更為雙引號。

### AWS CLI on Linux

```
aws ssm get-parameters-by-path \
 --path /aws/service/global-infrastructure/regions/af-south-1 \
 --query 'Parameters[?Name.contains(@,`longName`)].Value' \
 --output text
```

### AWS CLI on Windows

```
aws ssm get-parameters-by-path ^
 --path /aws/service/global-infrastructure/regions/af-south-1 ^
 --query "Parameters[?Name.contains(@,`longName`)].Value" ^
 --output text
```

### Tools for PowerShell

如果未安裝，請安裝 AWS.Tools。SimpleSystems管理模塊到工具，以 PowerShell 通過運行 `Install-AWSToolsModule AWS.Tools.SimpleSystemsManagement -CleanUp`。

```
$parameterPath = "/aws/service/global-infrastructure/regions/af-south-1"
$substringToMatch = "longName"
$filteredParameters = Get-SSMParametersByPath -Path $parameterPath \
| Where-Object { $_.Name -like "$substringToMatch*" } \
| ForEach-Object { Write-Output $_.Value }
$filteredParameters
```

下列為範例輸出。

```
Africa (Cape Town)
```

如需詳細資訊，請參閱 AWS Systems Manager 使用者指南中的[使用公有參數](#)。

## 指定資源的區域

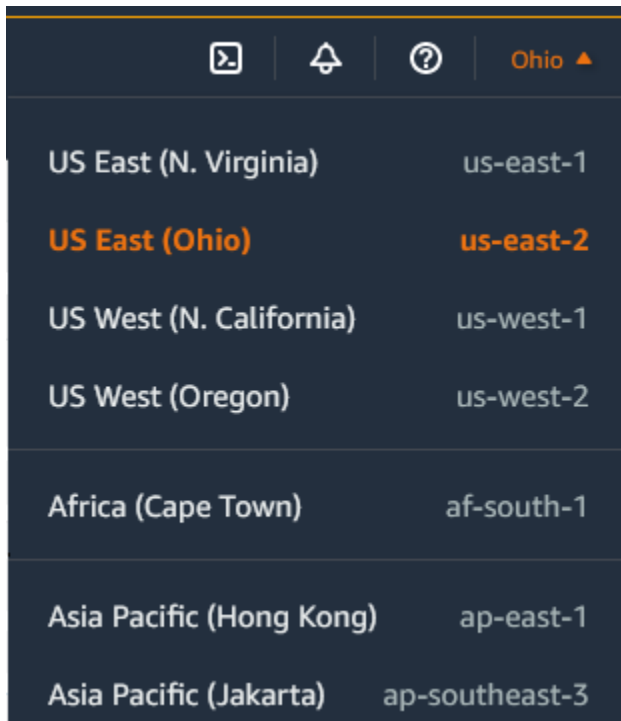
每次建立 Amazon EC2 資源時，您可以指定資源的區域。您可以使用 AWS Management Console 或命令行指定資源的 [區域]。

### 考量事項

部分 AWS 資源可能無法在所有區域使用。請在啟動執行個體之前，先確定您可以在所需區域中建立需要的資源。

### 使用主控台指定資源的區域

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 從導覽列中，選取 Regions (區域) 選取器，然後選取區域。



### 使用命令列指定預設區域

您可將環境變數的數值設為所要的區域端點 (例如 `https://ec2.us-east-2.amazonaws.com`) :

- `AWS_DEFAULT_REGION` (AWS CLI)
- `Set-AWSDefaultRegion` (AWS Tools for Windows PowerShell)

或者，您可在每個個別命令中使用 `--region` (AWS CLI) 或 `-Region` (AWS Tools for Windows PowerShell) 命令列選項。例如：`--region us-east-2`。

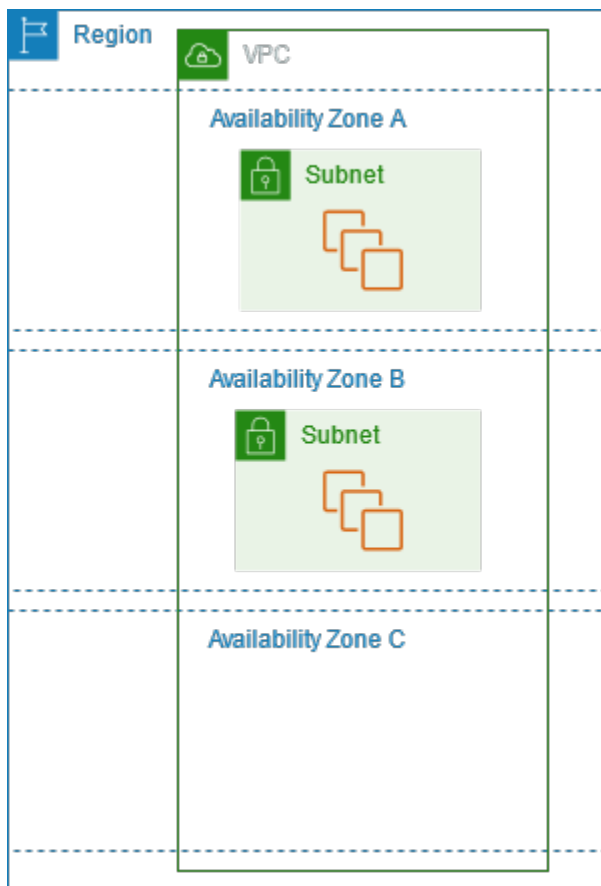
如需有關 Amazon EC2 端點的詳細資訊，[請參閱 AWS 一般參考](#)。

## 可用區域

每個區域擁有多個隔離位置，稱為可用區域。可用區域的代碼為其區域代碼，後續跟著一個字母識別符。例如 `us-east-1a`。

啟動執行個體時，您可選取區域和 virtual private cloud (VPC)，然後您可從其中一個可用區域選取子網，或讓我們為您選取一個子網。如果您要將執行個體分散至多個可用區域，但其中一個執行個體故障，則您可設計應用程式，讓其他可用區域內的執行個體來處理要求。您也可用彈性 IP 地址快速將地址重新映射到另一個可用區域中的執行個體，以遮罩某一可用區域內執行個體的故障。

下圖說明區域中的多個可用區域。AWS 可用區域 A 和可用區域 B 各有一個子網，每個子網都有執行個體。可用區域 C 沒有子網，因此您無法在此可用區域中啟動執行個體。



可用區域隨時間擴大時，我們擴展可用區域的能力將受到限制。假如出現此狀況，我們可能禁止您從受限的可用區域啟動執行個體，除非您在該可用區域已有執行個體。我們甚至可能需要為新帳戶將受限的

可用區域從可用區域清單中移除。因此您帳戶在某區域能取得的可用區域數量可能與其他帳戶的數量不同。

## 目錄

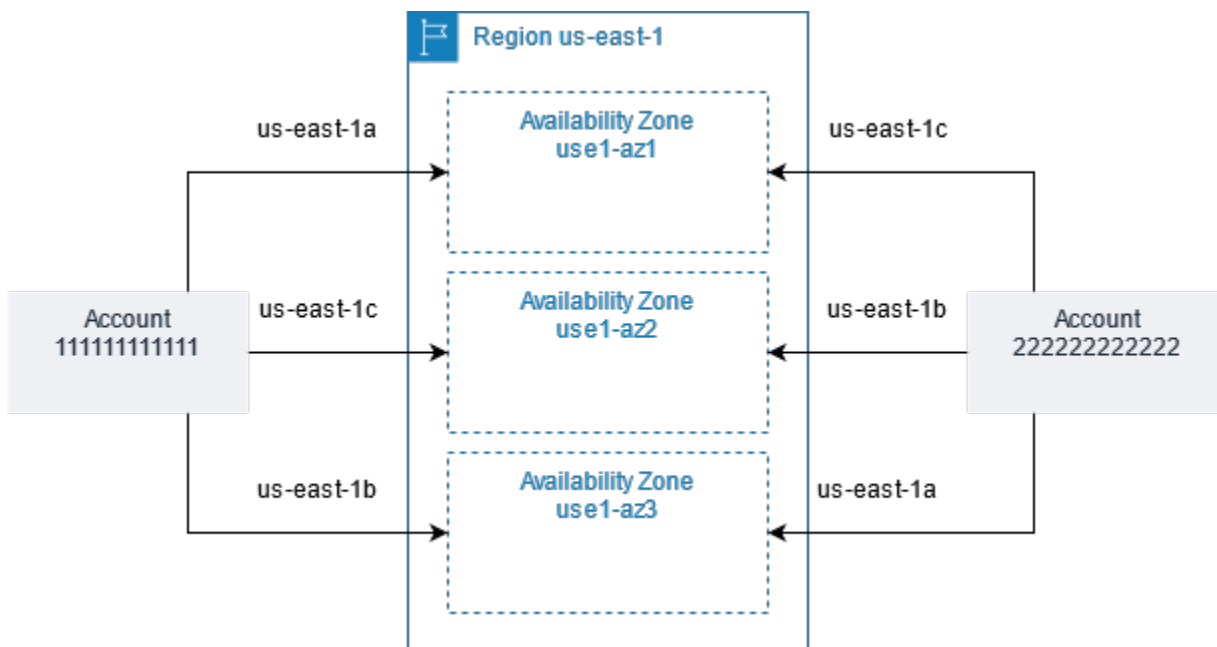
- [AZ ID](#)
- [描述您的可用區域](#)
- [在可用區域中啟動執行個體](#)
- [將執行個體遷移到其他的可用區域](#)

## AZ ID

為了確保資源分散到某個區域的可用區域，我們會將可用區域獨立對應至最舊區域 AWS 帳戶中每個區域的代碼。例如，您的 `us-east-1a` 的實際位置 AWS 帳戶可能與其他位置不同 AWS 帳戶。`us-east-1a`

若要協調所有區域中帳戶 (甚至對應可用區域的帳戶) 的可用區域，請使用 AZ ID，這是可用區域的唯一且一致的識別碼。例如，`use1-az1` 是區域的 AZ ID，並且在每個 `us-east-1` 區域中都有相同的實體位置 AWS 帳戶。您可檢視您帳戶的 AZ ID 來判斷您的資源在另一個帳戶中資源的相對實體位置。例如，如果您與另一個帳戶共享 AZ ID 為 `use1-az2` 的可用區域子網，則 AZ ID 也是 `use1-az2` 之可用區域中的該帳戶就可以使用此子網。

下圖說明兩個具有不同可用區域代碼至 AZ ID 映射的帳戶。



## 描述您的可用區域

您可用 Amazon EC2 主控台或命令列介面決定帳戶適用的可用區域。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

### 使用主控台尋找可用區域

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 從導覽列中，選取 Regions (區域) 選取器，然後選取區域。
3. 在導覽窗格中，選擇 EC2 Dashboard (EC2 儀表板)。
4. 可用區域列於 Service health (服務運作狀態) 窗格中。

### 若要尋找您的可用區域 AWS CLI

- 使用 [describe-availability-zones](#) 命令描述為您的帳戶啟用的指定區域內的可用區域，如下所示。

```
aws ec2 describe-availability-zones --region region-name
```

- 使用 [describe-availability-zones](#) 命令描述可用區域，而不論選擇加入狀態為何，如下所示。

```
aws ec2 describe-availability-zones --all-availability-zones
```

## 在可用區域中啟動執行個體

啟動執行個體時，選取可讓執行個體更接近特定客戶，或符合法律或您其他要求的區域。只要在個別的可有區域中啟動執行個體，您就可以保護應用程式免於發生單點故障。

您可在啟動執行個體時在您使用的區域中選擇性指定可用區域。如果您沒有指定可用區域，我們會為您選取可用區域。啟動初始執行個體時，建議您接受預設的可用區域，因為這可讓我們依照系統運作狀態和可用容量為您選取最適合的可用區域。如果您啟動其他的執行個體，則只有當您的新執行個體必須接近或隔離於執行中的執行個體時，您才能指定可用區域。

## 將執行個體遷移到其他的可用區域

如果需要，您可將執行個體從某一可用區域遷移到其他的可用區域。例如，若您嘗試修改您執行個體的執行個體類型，而我們無法在目前的可用區域中啟動新執行個體類型的執行個體，您可將執行個體遷移至具有新執行個體類型之容量的可用區域。

遷移程序包括：

- 從原始執行個體建立 AMI
- 在新的可用區域中啟動執行個體
- 更新新執行個體的組態，如下列程序所示

將執行個體遷移到其他的可用區域

1. 從執行個體建立 AMI。程序取決於執行個體的根裝置磁碟區類型。如需詳細資訊，請參閱與根裝置磁碟區對應的文件：
  - [創建一個 Amazon EBS-backed AMI](#)
  - [建立執行個體存放區後端 Linux AMI](#)
2. 如果您需要保留執行個體的私有 IPv4 地址，您必須在目前可用區域中刪除子網，接著在新的可用區域中建立子網並使用與原始子網相同的 IPv4 地址範圍。請注意，您必須先終止子網中所有的執行個體，接著才能刪除子網。因此，您應該從子網中的所有執行個體建立 AMI，接著才能將所有執行個體從目前子網移到新的子網。
3. 從剛建立的 AMI 啟動執行個體，指定新的可用區域或子網。您可用與原始執行個體相同的執行個體類型，或選取新的執行個體類型。如需詳細資訊，請參閱 [在可用區域中啟動執行個體](#)。
4. 如果原始執行個體有關聯的彈性 IP 地址，將其與新執行個體建立關聯。如需詳細資訊，請參閱 [取消彈性 IP 地址的關聯](#)。
5. 如果原始執行個體為預留執行個體，請變更保留的可用區域。(如果您同時變更了執行個體類型，也可以變更保留的執行個體類型。) 如需詳細資訊，請參閱 [提交修改請求](#)。
6. (選擇性) 終止原始的執行個體。如需詳細資訊，請參閱 [終止執行個體](#)。

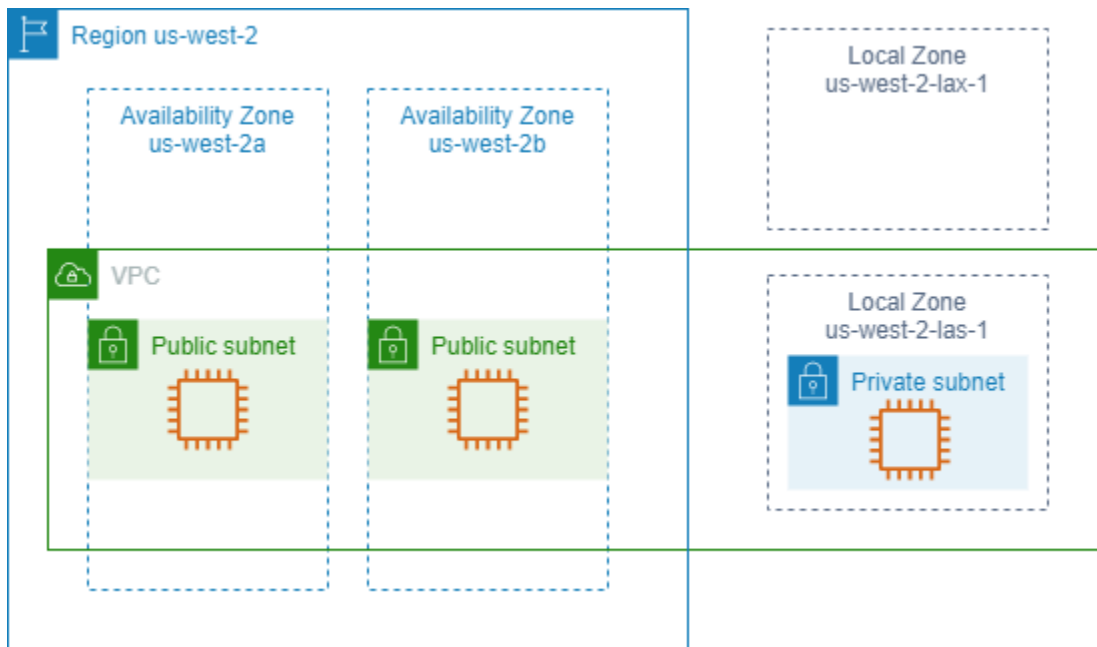
## 本機區域

「本地區域」是您使用者所在地理位置附近的「AWS 區域」的延伸。Local Zones 具有自己的網際網路連線和支援 AWS Direct Connect，因此在本機區域中建立的資源可以透過低延遲通訊為本機使用者提供服務。如需詳細資訊，請參閱 [什麼是 Local Zones?](#) 在「Local Zones 使用者指南」中。

本機區域的代碼為其區域代碼，後續跟著一個字母識別符，代表其實體位置。例如：us-west-2-lax-1 位於洛杉磯。

下圖說明「區 AWS 域」us-west-2、其兩個可用區域及其兩個「Local Zones」。VPC 跨越可用區域和其中一個本機區域。VPC 中的每個區域都有一個子網，而每個子網都有一個執行個體。





若要使用 Local Zone，首先您必須啟用它。如需詳細資訊，請參閱 [the section called “選擇加入 Local Zones”](#)。在 Local Zone 中建立子網。最後，啟動 Local Zone 子網路中的資源 (例如，執行個體)，讓您的應用程式更接近使用者。

## 目錄

- [可用 Local Zones](#)
- [選擇加入 Local Zones](#)
- [若要在 Local Zone 中啟動執行個體：](#)

## 可用 Local Zones

您可以使用 Amazon EC2 主控台或命令列介面，決定您的帳戶適用的 Local Zone。如需完整清單，請參閱 [AWS Local Zones 位置](#)。

### 使用主控台尋找您的 Local Zones

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 從導覽列中，選取 Regions (區域) 選取器，然後選取父系區域。
3. 在導覽窗格中，選擇 EC2 Dashboard (EC2 儀表板)。
4. 在頁面右上角選擇 Account Attributes (帳戶屬性) 及 Zones (區域)。

## 若要尋找您的 Local Zones AWS CLI

使用 [describe-availability-zones](#) 命令描述指定區域中的所有 Local Zone，即使未啟用，如下所示。若只要描述您已啟用的 Local Zone，請忽略 `--all-availability-zones` 選項。

```
aws ec2 describe-availability-zones --region region-name --filters Name=zone-type,Values=local-zone --all-availability-zones
```

## 選擇加入 Local Zones

在可以為資源或服務指定 Local Zone 之前，您必須選擇加入 Local Zones。

### 考量事項

部分 AWS 資源可能無法在所有區域使用。請確定在特定 Local Zone 中啟動執行個體之前，您可以在所需區域或 Local Zones 中建立需要的資源。如需每個 Local Zone 支援的服務清單，請參閱 [AWS Local Zones 功能](#)。

### 使用主控台選擇加入 Local Zones

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在頁面左上角，選取 New EC2 Experience (新增 EC2 體驗)。您無法使用舊的主控台體驗來完成這項任務。
3. 從導覽列中，選取 Regions (區域) 選取器，然後選取父系區域。
4. 在導覽窗格中，選擇 EC2 Dashboard (EC2 儀表板)。
5. 在頁面右上角選擇 Account Attributes (帳戶屬性) 及 Zones (區域)。
6. 選擇本機區域，然後選擇「動作」>「管理區域」群組。
7. 在選擇加入狀態下，選擇啟用。
8. 選擇更新。

若要選擇加入 Local Zones，請使用 AWS CLI

使用 [modify-availability-zone-group](#) 命令。

若要在 Local Zone 中啟動執行個體：

當您啟動執行個體時，可以指定位於 Local Zone 的子網。您可以從網路邊界群組中配置 IP 地址。網路邊界群組是一組唯一的可用區域、Local Zones 或 Wavelength 區域，AWS 會從中公告 IP 地址，例如 `us-west-2-lax-1a`。

您可以從網路邊界群組配置下列 IP 地址：

- Amazon 提供的彈性 IPv4 地址
- Amazon 提供的 IPv6 VPC 地址 (僅適用於洛杉磯地區)

如需如何在本機區域中啟動執行個體的詳細資訊，請參閱 [AWS Local Zones 使用者指南中的使用 AWS Local Zones 入門](#)。

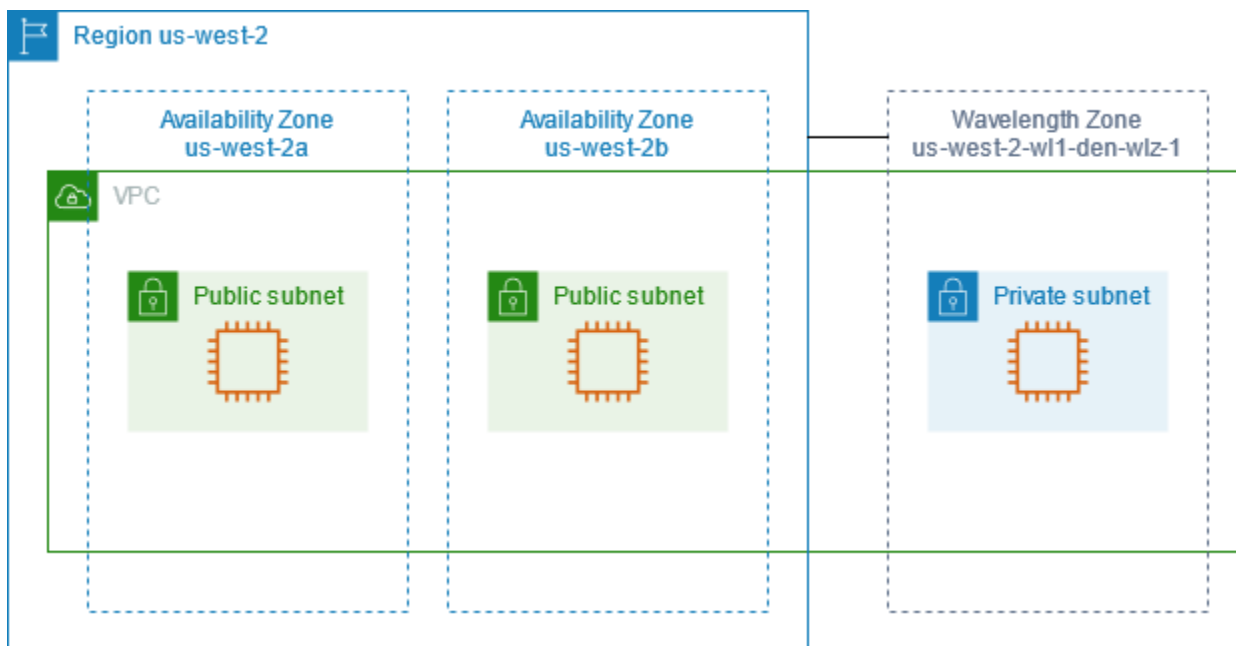
## Wavelength 區域

AWS Wavelength 讓開發人員能夠建置可為行動裝置和使用者提供超低延遲的應用程式。Wavelength 將標準 AWS 運算和儲存服務部署到電信業者的 5G 網路邊緣。開發人員可以將虛擬私有雲 (VPC) 延伸到一個或多個 Wavelength 區域，然後使用 Amazon EC2 執行個體等 AWS 資源執行需要超低延遲和區域中 AWS 服務連線的應用程式。

Wavelength 區域是部署 Wavelength 基礎設施之電信業者位置的隔離區域。Wavelength 區域繫結至區域。Wavelength 區域是區域的邏輯延伸，並且由區域中的控制平面管理。

Wavelength 區域的代碼為其區域代碼，後續跟著一個字母識別符，代表其實體位置。例如：us-east-1-wl1-bos-wlz-1 位於波士頓。

下圖說明了區域 AWS 域 us-west-2，其兩個可用區域和一個 Wavelength 區。VPC 跨越可用區域和該 Wavelength 區域。VPC 中的每個區域都有一個子網，而每個子網都有一個執行個體。



若要使用 Wavelength 區域，您必須先選擇加入「Zone (區域)」。如需詳細資訊，請參閱 [the section called “啟用 Wavelength 區域”](#)。接著，在 Wavelength 區域中建立一個子網。最後，在 Wavelength 區域子網中啟動您的資源，讓您的應用程式更接近最終使用者。

並非每個區域都有 Wavelength 區域。如需支援 Wavelength 區域的區域相關資訊，請參閱 AWS Wavelength 開發人員指南中的 [可用 Wavelength 區域](#)。

## 內容

- [描述您的 Wavelength 區域](#)
- [啟用 Wavelength 區域](#)
- [在 Wavelength 區域中啟動執行個體](#)

## 描述您的 Wavelength 區域

您可用 Amazon EC2 主控台或命令列介面決定帳戶適用的 Wavelength 區域。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

使用主控台尋找您的 Wavelength 區域

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 從導覽列中，選取 Regions (區域) 選取器，然後選取區域。
3. 在導覽窗格中，選擇 EC2 Dashboard (EC2 儀表板)。
4. 在頁面右上角選擇 Account Attributes (帳戶屬性) 及 Zones (區域)。

若要尋找您的 Wavelength 區，請使用 AWS CLI

- 使用 [describe-availability-zones](#) 命令描述為您的帳戶啟用的指定區域內的 Wavelength 區域，如下所示。

```
aws ec2 describe-availability-zones --region region-name
```

- 使用 [describe-availability-zones](#) 命令描述 Wavelength 區域，不論選擇加入狀態為何，如下所示。

```
aws ec2 describe-availability-zones --all-availability-zones
```

## 啟用 Wavelength 區域

在為資源或服務指定 Wavelength 區域之前，必須選擇加入 Wavelength 區域。

### 考量事項

- 部分 AWS 資源並不適用於所有區域。請確定在特定 Wavelength 區域中啟動執行個體之前，您可以在所需區域或 Wavelength 區域中建立需要的資源。

### 使用主控台選擇加入 Wavelength 區域

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在頁面左上角，選取 New EC2 Experience (新增 EC2 體驗)。您無法使用舊的主控台體驗來完成這項任務。
3. 從導覽列中，選取 Regions (區域) 選取器，然後選取區域。
4. 在導覽窗格中，選擇 EC2 Dashboard (EC2 儀表板)。
5. 在頁面右上角選擇 Account Attributes (帳戶屬性) 及 Zones (區域)。
6. 選擇「Wavelength 區域」，然後選擇「動作」>「管理區域」群組。
7. 在選擇加入狀態下，選擇啟用。
8. 選擇更新。

### 若要使用啟用 Wavelength 區 AWS CLI

使用 [modify-availability-zone-group](#) 命令。

### 在 Wavelength 區域中啟動執行個體

當您啟動執行個體時，您可以指定位於 Wavelength 區域的子網。您也可以從網路邊界群組配置電信業者 IP 地址，該網路邊界群組是一組唯一的可用區域、Local Zones 或 Wavelength 區域，AWS 會從中公告 IP 地址，例如，`us-east-1-wl1-bos-wlz-1`。

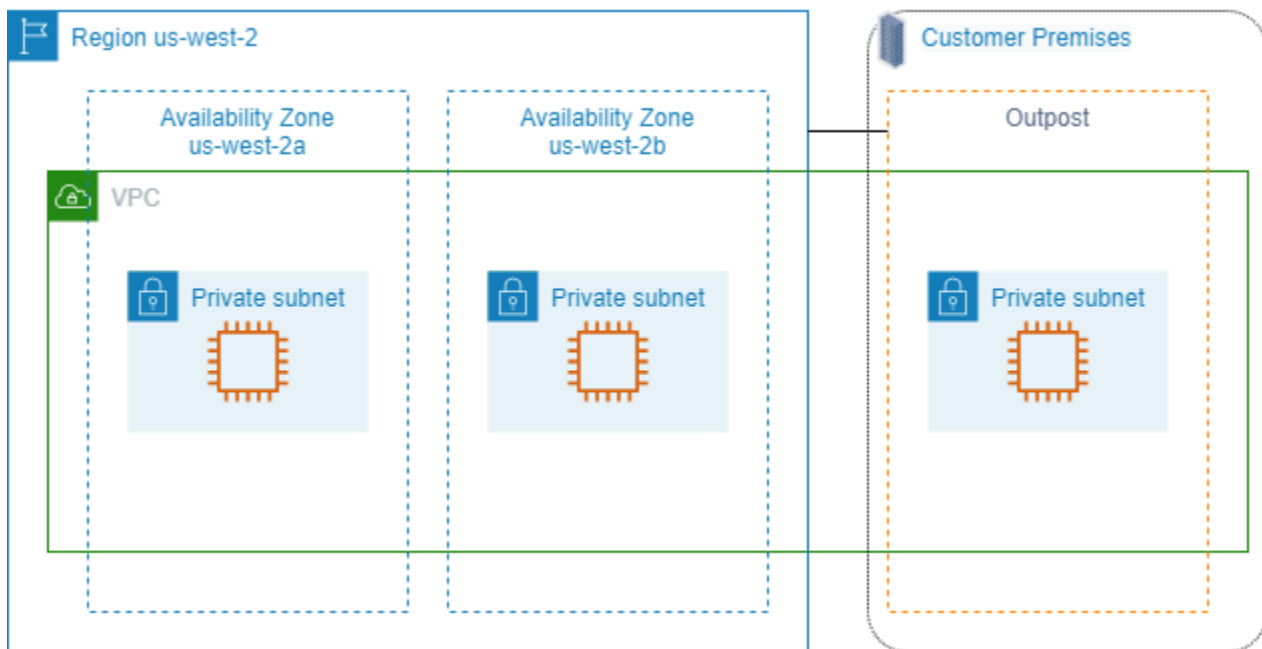
如需有關如何在 Wavelength 區域中啟動執行個體的資訊，請參閱 AWS Wavelength 開發人員指南中的 [AWS Wavelength 入門](#)。

## AWS Outposts

AWS Outposts 是一項全受管服務，可將 AWS 基礎架構、服務、API 和工具延伸至客戶場所。透過提供 AWS 受管理基礎架構的本機存取權，AWS Outposts 讓客戶能夠使用與 AWS 區域相同的程式設計介面在內部部署建置和執行應用程式，同時使用本機運算和儲存資源來降低延遲和本機資料處理需求。

Outpost 是部署在客戶站點的 AWS 計算和儲存容量集區。AWS 作為 AWS 區域的一部分來操作、監控和管理此容量。您可以在 Outpost 上建立子網路，並在建立 AWS 資源時指定子網路。Outpost 子網路中的執行個體會使用私有 IP 位址與 AWS 區域中的其他執行個體進行通訊，全部位於相同的 VPC 內。

下圖說明「區 AWS 域」us-west-2、其兩個可用區域及一個「前哨」。VPC 跨越可用區域和 Outpost。Outpost 位於內部部署客戶資料中心。VPC 中的每個區域都有一個子網，而每個子網都有一個執行個體。



要開始使用 AWS Outposts，您必須創建一個前哨並訂購前哨容量。如需 Outposts 組態的詳細資訊，請參閱[我們的目錄](#)。安裝 Outpost 設備之後，當您在 Outpost 上啟動 Amazon EC2 執行個體時，即可使用運算和儲存容量。

### 在 Outpost 上啟動執行個體

您可以在您建立的 Outpost 子網中啟動 EC2 執行個體。安全群組可控制 Outpost 子網中具有彈性網絡介面的執行個體的傳入與傳出流量，方法與可用區域子網中的執行個體相同。若要連線到 Outpost 子網中的 EC2 執行個體，當您啟動執行個體時，可以指定金鑰對，方法與可用區域子網中的執行個體相同。

建議您將 Outpost 機架上執行個體的根磁碟區限制為 30 GiB 或更小。您可以在 AMI 或執行個體的區塊型設備映射中指定資料磁碟區，以提供額外的儲存空間。若要從開機磁碟區修剪未使用的區塊，請參閱 AWS 合作夥伴網路部落格中的 [如何建立稀疏 EBS 磁碟區](#)。

我們建議您增加根磁碟區的 NVMe 逾時。如需詳細資訊，請參閱 [I/O 作業逾時](#)。

如需有關如何建立 Outpost 的資訊，請參閱《AWS Outposts 使用者指南》中的 [AWS Outposts 入門](#)。

## 在 Outpost 機架上建立磁碟區

AWS Outposts 提供機架和伺服器外型規格。如果您的容量位於 Outpost 機架上，您可以在您建立的 Outpost 子網中建立 EBS 磁碟區。建立磁碟區時，請指定 Outpost 的 Amazon Resource Name (ARN)。

以下 [create-volume](#) 命令會在指定的 Outpost 上建立空的 50 GB 磁碟區。

```
aws ec2 create-volume --availability-zone us-east-2a --outpost-arn arn:aws:outposts:us-east-2:123456789012:outpost/op-03e6fecad652a6138 --size 50
```

您可以動態地修改 Amazon EBS gp2 磁碟區的大小，而不將其分離。如需在不卸離磁碟區的情況下修改磁碟區的詳細資訊，請參閱 [要求對 EBS 磁碟區進行修改](#)。

## Amazon EC2 執行個體 IP 定址

Amazon EC2 和 Amazon VPC 同時支援 IPv4 和 IPv6 定址通訊協定。Amazon VPC 預設會使用 IPv4 定址通訊協定；您無法停用此行為。當您建立 VPC 時，必須指定 IPv4 CIDR 區塊 (私有 IPv4 地址的範圍)。您可以選擇將 IPv6 CIDR 區塊指派給 VPC，並將 IPv6 地址從該區塊指派給子網中的執行個體。

### 目錄

- [私有 IPv4 地址](#)
- [公有 IPv4 地址](#)
- [公用 IPv4 位址最佳化](#)
- [彈性 IP 地址 \(IPv4\)](#)
- [IPv6 地址](#)
- [使用執行個體的 IPv4 地址](#)

- [使用執行個體的 IPv6 地址](#)
- [適用於 EC2 執行個體的多個 IP 地址](#)
- [設定您的 Windows 執行個體的次要私有 IPv4 位址](#)
- [EC2 執行個體主機名稱](#)
- [Link-local 地址](#)

## 私有 IPv4 地址

私有 IPv4 地址是無法透過網際網路存取的 IP 地址。您可以在相同 VPC 的執行個體之間使用私有 IPv4 地址進行通訊。如需私有 IPv4 地址標準和規格的詳細資訊，請參閱 [RFC 1918](#)。我們會使用 DHCP 將私有 IPv4 地址配置給執行個體。

### Note

您可以建立具有可公開路由 CIDR 區塊的 VPC，而此 CIDR 區塊不在 RFC 1918 所指定的私有 IPv4 地址範圍內。不過，基於本文件的目的，我們會將私有 IPv4 地址 (或「私有 IP 地址」) 參照為 VPC 之 IPv4 CIDR 範圍內的 IP 地址。

VPC 子網可以是下列其中一個類型：

VPC 子網可以是下列其中一個類型：

- 僅限 IPv4 子網：您只能在已獲指派 IPv4 地址的這些子網中建立資源。
- 僅限 IPv6 子網：您只能在已獲指派 IPv6 地址的這些子網中建立資源。
- IPv4 和 IPv6 子網：您可以在已獲指派 IPv4 和 IPv6 地址的這些子網中建立資源。

當您將 EC2 執行個體啟動至僅限 IPv4 或雙堆疊 (IPv4 和 IPv6) 子網時，此執行個體會從子網的 IPv4 地址範圍接收主要私有 IP 地址。如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的 [IP 定址](#)。如果您未在啟動執行個體時指定主要私有 IP 地址，我們會在子網的 IPv4 範圍中選取可用 IP 地址。每個執行個體皆會有獲指派之主要私有 IPv4 地址的預設網路介面 (eth0)。您也可以指定其他私有 IPv4 地址 (稱為次要私有 IPv4 地址)。與主要私有 IP 地址不同，輔助私有 IP 地址可從原執行個體重新指派至別的執行個體。如需詳細資訊，請參閱 [適用於 EC2 執行個體的多個 IP 地址](#)。

私有 IPv4 地址，無論是主要或次要地址，在執行個體停止和啟動時或休眠和啟動時都會維持與網路介面的關聯，而會在執行個體終止時予以釋出。



## 公有 IPv4 地址

公有 IP 地址是可從網際網路存取的 IPv4 地址。您可以使用公有地址，在執行個體與網際網路之間通訊。

當您在預設 VPC 中啟動執行個體時，預設會將公有 IP 地址指派給它。當您將執行個體啟動到非預設 VPC 時，子網的屬性可判定啟動到該子網的執行個體是否接收公有 IPv4 地址集區中的公有 IP 地址。根據預設，我們不會將公有 IP 地址指派給非預設子網中啟動的執行個體。

您可以控制您的執行個體是否接收公有 IP 地址，如下所示：

- 修改子網的公有 IP 定址屬性。如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的[修改子網路的公有 IPv4 定址屬性](#)。
- 在啟動期間啟用或停用公有 IP 定址，其可覆寫子網的公有 IP 定址屬性。如需詳細資訊，請參閱[在啟動執行個體期間指派公有 IPv4 地址](#)。
- 啟動後，您可以[管理與網路介面關聯的 IP 位址，從執行個體取消指派公用 IP 位址](#)。

公有 IP 地址會從 Amazon 的公有 IPv4 地址集區指派給您的執行個體，而且與您的 AWS 帳戶不相關聯。取消公有 IP 地址與您執行個體的關聯後，會將之釋出回公有 IPv4 地址集區，且您將無法重複使用之。

在某些情況下，我們會從您的執行個體釋出公有 IP 位址，或為其指派新 IP 位址：

- 當您的執行個體停止、休眠或終止時，我們會釋出其公有 IP 地址。啟動您已停止或休眠的執行個體時，其會收到新的公有 IP 地址。
- 當您將彈性 IP 地址與執行個體建立關聯時，我們會釋出該執行個體的公有 IP 地址。當您取消彈性 IP 地址與執行個體的關聯時，執行個體收到新的公有 IP 地址。
- 如果釋放 VPC 中執行個體的公有 IP 地址，若有多個網路介面連接至執行個體，則不會收到新的公有 IP 地址。
- 當您的執行個體具有與彈性 IP 地址相關聯的次要私有 IP 地址時，如果釋放該執行個體的公有 IP 地址，則執行個體不會接收新的公有 IP 地址。

如果您需要可建立和取消與執行個體之關聯的持久性公有 IP 地址，請改用彈性 IP 地址。

如果您使用動態 DNS 將現有 DNS 名稱映射至新執行個體的公有 IP 地址，則最多可能需要 24 個小時，IP 地址才會傳播到整個網際網路。因此，已終止的執行個體還會持續收到請求，而新的執行個體

可能尚未開始接收流量。若要解決此問題，請使用彈性 IP 地址。您可以配置自己的彈性 IP 地址，並建立地址與您執行個體的關聯。如需更多詳細資訊，請參閱 [彈性 IP 地址](#)。

#### Note

- AWS 所有公用 IPv4 位址的費用，包括與執行中執行個體和彈性 IP 位址相關聯的公用 IPv4 位址。如需詳細資訊，請參閱 [Amazon VPC 定價頁面](#) 中的公有 IPv4 地址。
- 透過公有 NAT IP 地址存取其他執行個體的執行個體，會根據其是否位在相同的區域，而收取地區或網際網路資料傳輸的費用。

## 公用 IPv4 位址最佳化

AWS 所有公用 IPv4 位址的費用，包括與執行中執行個體和彈性 IP 位址相關聯的公用 IPv4 位址。如需詳細資訊，請參閱 [Amazon VPC 定價頁面](#) 中的公有 IPv4 地址。

下列清單包含您可以採取的動作來最佳化您使用的公用 IPv4 位址數目：

- 使用 [彈性負載平衡器](#) 負載平衡 EC2 執行個體的流量，並在 [指派給執行個體的主 ENI 上停用自動指派公有 IP](#)。負載平衡器使用單一公用 IPv4 位址，因此可減少公用 IPv4 位址計數。您可能還想要合併現有的負載平衡器，以進一步減少公用 IPv4 位址計數。
- 如果使用 NAT 閘道的唯一原因是將 SSH Connect 到私有子網路中的 EC2 執行個體進行維護或緊急情況，請考慮改用 [EC2 執行個體連接端點](#)。使用 EC2 執行個體 Connect 端點，您可以從網際網路連接到執行個體，而不需要執行個體擁有公有 IPv4 地址。
- 如果 EC2 執行個體位於已分配公有 IP 地址的公有子網路中，請考慮將執行個體移至私有子網路、移除公有 IP 地址，並使用 [公有 NAT 閘道](#) 允許從 EC2 執行個體進行存取。使用 NAT 閘道需要考量成本。使用此計算方法來決定 NAT 閘道是否符合成本效益。您可以 [建立「Number of public IPv4 addresses 帳 AWS 單成本與用量報表」](#) 來取得此計算所需的資料。

```
NAT gateway per hour + NAT gateway public IPs + NAT gateway transfer / Existing public IP cost
```

其中：

- NAT gateway per hour = \$0.045 \* 730 hours in a month \* Number of Availability Zones the NAT gateways are in
- NAT gateway public IPs = \$0.005 \* 730 hours in a month \* Number of IPs associated with your NAT gateways

- NAT gateway transfer = \$0.045 \* Number of GBs that will go through the NAT gateway in a month
- Existing public IP cost = \$0.005 \* 730 hours in a month \* Number of public IPv4 addresses

如果總數小於 1，則 NAT 閘道會比公用 IPv4 位址便宜。

- 用於私下連線[AWS PrivateLink](#)至 AWS 由其他 AWS 帳戶託管的服務或服務，而不是使用公用 IPv4 位址和網際網路閘道。
- 將[您自己的 IP 地址範圍 \(BYOIP\)](#)帶到公共 IPv4 地址 AWS 並使用該範圍，而不是使用亞馬遜擁有的公共 IPv4 地址。
- 針對啟[動至子網路的執行個體](#)，[關閉自動指派公用 IPv4 位址](#)。建立子網路時，VPC 通常預設會停用此選項，但您應該檢查現有的子網路以確保已停用。
- 如果您的 EC2 執行個體不需要公有 IPv4 地址，[請檢查連接到執行個體的網路界面是否已停用自動指派公用 IP](#)。
- 在私有子網路中[AWS Global Accelerator 為 EC2 執行個體設定加速器端點](#)，讓網際網路流量直接流向 VPC 中的端點，而不需要公有 IP 位址。您也可以將[自己的位址帶到](#)加速器的靜態 IP 位址，AWS Global Accelerator 並使用您自己的 IPv4 位址。

## 彈性 IP 地址 (IPv4)

彈性 IP 地址是您可以配置給您帳戶的公有 IPv4 地址。您可以根據需要將其關聯，並將其與執行個體取消關聯。在您選擇釋放之前，它會分配給您的帳戶。如需彈性 IP 地址和其使用方式的詳細資訊，請參閱[彈性 IP 地址](#)。

我們不支援 IPv6 的彈性 IP 地址。

## IPv6 地址

您可以選擇建立 IPv6 CIDR 區塊與 VPC 的關聯，並建立 IPv6 CIDR 區塊與子網的關聯。VPC 的 IPv6 CIDR 區塊會自動從 Amazon 的 IPv6 地址集區指派；您無法自行選擇範圍。如需詳細資訊，請參閱 Amazon VPC 使用者指南 中的下列主題：

- [您 VPC 和子網路的 IP 定址](#)
- [將 IPv6 CIDR 區塊新增至您的 VPC](#)
- [將 IPv6 CIDR 區塊新增至您的子網路](#)

IPv6 地址是全域不重複的地址，可設定為維持在私有不公開的狀態，或可透過網際網路來連線。如果建立 IPv6 CIDR 區塊與 VPC 和子網的關聯，以及符合下列其中一項，則您的執行個體會收到 IPv6 地址：

- 在啟動期間，您的子網會設定為自動將 IPv6 地址指派給執行個體。如需詳細資訊，請參閱[修改子網的 IPv6 定址屬性](#)。
- 在啟動期間，您將 IPv6 地址指派給您的執行個體。
- 在啟動之後，您將 IPv6 地址指派給執行個體的主要網路介面。
- 在啟動之後，您將 IPv6 地址指派給相同子網中的網路介面，並將網路介面連接至執行個體。

當您的執行個體在啟動期間收到 IPv6 地址時，會建立該地址與執行個體之主要網路介面 (eth0) 的關聯。您可以透過下列方式，管理執行個體主要網路介面 (eth0) 的 IPv6 位址：

- 從網路介面指派和取消指派 IPv6 位址 您可指派給網路介面的 IPv6 地址數目以及您可連接至執行個體的網路介面數目，會根據執行個體類型而不同。如需詳細資訊，請參閱 [每個執行個體類型每網路介面的 IP 地址](#)。
- 啟用主要 IPv6 位址。主要 IPv6 位址可讓您避免中斷傳送至執行個體或 ENI 的流量。如需詳細資訊，請參閱 [建立網路介面](#) 或 [管理 IP 地址](#)。

IPv6 地址會在您停止和啟動或休眠和啟動執行個體時持續保留，並在您終止執行個體時予以釋放。若 IPv6 地址已經指派給另一個網路介面，您無法重新指派該 IPv6 地址 (您必須先將之取消指派)。

您可以藉由控制您子網路的路由，或是使用安全群組和網路 ACL 規則，控制是否可透過執行個體的 IPv6 位址連線至執行個體。如需更多資訊，請參閱《Amazon VPC 使用者指南》中的「[網路間流量隱私權](#)」。

如需預留 IPv6 地址範圍的詳細資訊，請參閱 [IANA IPv6 Special-Purpose Address Registry](#) 和 [RFC4291](#)。

## 使用執行個體的 IPv4 地址

您可以在啟動執行個體時，將公有 IPv4 地址指派給該執行個體。您可以在主控台中透過 Instances (執行個體) 頁面或 Network Interfaces (網路介面) 頁面檢視您執行個體的 IPv4 地址。

### 目錄

- [檢視 IPv4 地址](#)
- [在啟動執行個體期間指派公有 IPv4 地址](#)

## 檢視 IPv4 地址

您可以使用 Amazon EC2 主控台檢視執行個體的公有和私有 IPv4 地址。您也可以使用執行個體中繼資料，來從執行個體內判定執行個體的公有 IPv4 和私有 IPv4 地址。如需詳細資訊，請參閱 [使用執行個體中繼資料](#)。

公有 IPv4 地址會顯示為主控台中網路介面的屬性，但會透過 NAT 映射至主要私有 IPv4 地址。因此，如果您檢查執行個體上網路介面的屬性 (例如，透過 `ifconfig` (Linux) 或 `ipconfig` (Windows))，則不會顯示公有 IPv4 地址。若要從執行個體判定執行個體的公有 IPv4 地址，請使用執行個體中繼資料。

### 使用命令行檢視執行個體的 IPv4 地址

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [describe-instances](#) (AWS CLI)
- [Get-EC2Instance](#) (AWS Tools for Windows PowerShell).

### 使用執行個體中繼資料判定執行個體的 IPv4 地址

1. 連線到您的執行個體。如需詳細資訊，請參閱 [Connect 至您的 EC2 執行個體](#)。
2. 使用下列命令來存取私人 IP 位址。

#### IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/local-ipv4
```

#### IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/local-ipv4
```

#### Tools for Windows PowerShell

```
PS C:\> Invoke-RestMethod http://169.254.169.254/latest/meta-data/local-ipv4
```

3. 使用下列命令存取公用 IP 位址。如果建立彈性 IP 地址與執行個體的關聯，傳回的值即為彈性 IP 地址的值。

## IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H
"X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
meta-data/public-ipv4
```

## IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/public-ipv4
```

## Tools for Windows PowerShell

```
PS C:\> Invoke-RestMethod http://169.254.169.254/latest/meta-data/public-ipv4
```

## 在啟動執行個體期間指派公有 IPv4 地址

每一個子網都會有一個屬性，以判定是否將公有 IP 地址指派至啟動到該子網的執行個體。非預設子網預設會將此屬性設定為 false，而預設子網會將此屬性設定為 true。當您啟動執行個體時，也可以使用公有 IPv4 定址功能控制是否將公有 IPv4 地址指派至執行個體；您可以覆寫子網之 IP 定址屬性的預設行為。公有 IPv4 地址會從 Amazon 的公有 IPv4 地址集區指派，並會指派給具有裝置索引 eth0 的網路介面。此功能取決於啟動執行個體時的特定條件。

### 考量事項

- 啟動後，您可以[管理與網路介面關聯的 IP 位址](#)，[從執行個體取消指派公用 IP 位址](#)。如需公有 IPv4 地址的詳細資訊，請參閱[公有 IPv4 地址](#)。
- 如果您指定多個網路介面，則無法自動指派公有 IP 地址。此外，如果您為 eth0 指定現有網路介面，則無法使用自動指派公有 IP 功能覆寫子網設定。
- 無論您是否在啟動期間為執行個體指派公用 IP 位址，都可以在啟動後將彈性 IP 位址與執行個體建立關聯。如需詳細資訊，請參閱[彈性 IP 地址](#)。您也可以修改子網的公有 IPv4 定址行為。如需詳細資訊，請參閱[修改子網路的公有 IPv4 定址屬性](#)。

## 若要在使用主控台啟動執行個體期間指派公有 IPv4 地址

遵循程序[啟動執行個體](#)，當您設定 [Network Settings](#) (網路設定) 時，選擇選項以 Auto-assign Public IP (自動指派公有 IP)。

使用命令列啟用或停用公有 IP 定址功能

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- 使用 `--associate-public-ip-address` 或 `--no-associate-public-ip-address` 選項並搭配 [run-instances](#) 命令 (AWS CLI)
- 透過 [New-EC2Instance](#) 指令使用 `-AssociatePublicIp` 參數 (AWS Tools for Windows PowerShell)

## 使用執行個體的 IPv6 地址

您可以檢視指派給執行個體的 IPv6 地址、將公有 IPv6 地址指派給執行個體，或從執行個體取消指派 IPv6 地址。您可以在主控台中透過 [Instances \(執行個體\)](#) 頁面或 [Network Interfaces \(網路介面\)](#) 頁面檢視這些地址。

內容

- [檢視 IPv6 地址](#)
- [將 IPv6 地址指派給執行個體](#)
- [從執行個體取消指派 IPv6 地址](#)

## 檢視 IPv6 地址

您可以使用 Amazon EC2 主控台和執行個體中繼資料來檢視執行個體的 IPv6 地址。AWS CLI

使用主控台檢視執行個體的 IPv6 地址

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 [Instances \(執行個體\)](#)。
3. 選取實例。
4. 在 [Networking \(聯網\)](#) 標籤上，找出 IPv6 addresses (IPv6 地址)。

使用命令行檢視執行個體的 IPv6 地址

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [describe-instances](#) (AWS CLI)

- [Get-EC2Instance](#) (AWS Tools for Windows PowerShell).

使用執行個體中繼資料檢視執行個體的 IPv6 地址

1. 連線到您的執行個體。如需詳細資訊，請參閱 [Connect 至您的 EC2 執行個體](#)。
2. 從中取得執行個體的 MAC 位址 `http://169.254.169.254/latest/meta-data/network/interfaces/macs/`。
3. 使用下列命令來檢視 IPv6 位址。

### IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/network/interfaces/macs/mac-address/ipv6s
```

### IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/network/interfaces/macs/mac-address/ipv6s
```

### Tools for Windows PowerShell

```
PS C:\> Invoke-RestMethod http://169.254.169.254/latest/meta-data/network/interfaces/macs/mac-address/ipv6s
```

## 將 IPv6 地址指派給執行個體

如果已建立 VPC 和子網與 IPv6 CIDR 區塊的關聯，您可以在啟動期間或之後將 IPv6 地址指派給執行個體。IPv6 地址會從子網的 IPv6 地址範圍指派，並會指派給具有裝置索引 eth0 的網路介面。

若要在啟動執行個體期間指派公有 IPv6 地址

遵循程序 [啟動執行個體](#)，當您設定 [Network Settings](#) (網路設定) 時，選擇選項以 Auto-assign IPv6 IP (自動指派 IPv6 IP)。

在啟動之後指派 IPv6 地址

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。



2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選取 Actions (動作)、Networking (聯網)、Manage IP addresses (管理 IP 地址)。
4. 展開網路介面。在 IPv6 addresses (IPv6 地址) 下選擇 Assign new IP address (指派新 IP 地址)。輸入子網範圍中的 IPv6 地址，或將欄位保留空白，讓 Amazon 為您選擇 IPv6 地址。
5. 選擇 Save (儲存)。

### 使用命令列指派 IPv6 地址

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- 使用 `--ipv6-addresses` 選項並搭配 [run-instances](#) 命令 (AWS CLI)
- 在 [New-EC2Instance](#) 指令 (AWS Tools for Windows PowerShell) `-NetworkInterface` 中使用的 `Ipv6Addresses` 屬性
- [assign-ipv6-addresses](#) (AWS CLI)
- [Register-EC2Ipv6 AddressList](#) (AWS Tools for Windows PowerShell)

### 從執行個體取消指派 IPv6 地址

您可以隨時從執行個體取消指派 IPv6 地址。

#### 使用主控台從執行個體取消指派 IPv6 地址

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選取 Actions (動作)、Networking (聯網)、Manage IP addresses (管理 IP 地址)。
4. 展開網路介面。在 IPv6 addresses (IPv6 地址) 下，選擇 IPv6 地址旁的 Unassign (取消指派)。
5. 選擇 Save (儲存)。

#### 使用命令列從執行個體取消指派 IPv6 地址。

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [unassign-ipv6-addresses](#) (AWS CLI)

- [Unregister-EC2Ipv6 AddressList](#) (AWS Tools for Windows PowerShell)。

## 適用於 EC2 執行個體的多個 IP 地址

您可以指定執行個體的多個私有 IPv4 和 IPv6 地址。您可指定給執行個體的網路介面數目以及私有 IPv4 和 IPv6 地址數目取決於執行個體類型。如需詳細資訊，請參閱 [每個執行個體類型每網路介面的 IP 地址](#)。

這可用於在 VPC 中將多個 IP 地址指派給執行個體來執行下列操作：

- 在單一伺服器上使用多個 SSL 憑證，以及建立每個憑證與特定 IP 地址的關聯，以在單一伺服器上託管多個網站。
- 操作網路應用裝置 (例如防火牆或負載平衡器)，其中的每個網路介面有多個 IP 地址。
- 將輔助 IP 地址重新指派給待命執行個體，以在執行個體失敗時將內部流量重新導向至備用執行個體。

### 內容

- [多個 IP 地址運作方式](#)
- [使用多個 IPv4 地址](#)
- [使用多個 IPv6 地址](#)

## 多個 IP 地址運作方式

下列清單說明多個 IP 地址如何與網路介面運作：

- 您可以將輔助私有 IPv4 地址指派給任何網路介面。
- 您可以將多個 IPv6 地址指派給網路介面，而網路介面位於具有相關聯 IPv6 CIDR 區塊的子網中。
- 您必須從網路介面之子網的 IPv4 CIDR 區塊範圍選擇輔助 IPv4 位址。
- 您必須從網路介面之子網的 IPv6 CIDR 區塊範圍選擇 IPv6 地址。
- 您可以將安全群組與網路介面 (而非個別的 IP 地址) 建立關聯。因此，您在網路介面中指定的每個 IP 地址會受其網路介面安全群組的規範。
- 多個 IP 地址可以指派和取消指派至連接至執行中或已停止執行個體的網路介面。
- 已指派給網路介面的輔助私有 IPv4 地址，可以經您的明確允許而重新指派給另一個網路介面。
- 無法將 IPv6 地址重新指派給另一個網路介面；您必須先從現有網路介面取消指派 IPv6 地址。

- 使用命令列工具或 API 將多個 IP 地址指派給網路介面時，如果無法指派其中一個 IP 地址，則整個操作會失敗。
- 輔助網路介面與執行個體分離或連接至執行個體時，網路介面會保有主要私有 IPv4 地址、輔助私有 IPv4 地址、彈性 IP 地址和 IPv6 地址。
- 雖然您無法從執行個體分離主要網路介面，但是您可以將主要網路介面的輔助私有 IPv4 地址重新指派給另一個網路介面。

下列清單說明多個 IP 地址如何與彈性 IP 地址運作 (僅限 IPv4)：

- 每個私有 IPv4 地址都可以建立與單一彈性 IP 地址的關聯，反之亦然。
- 將輔助私有 IPv4 地址重新指派給另一個介面時，輔助私有 IPv4 地址會保留其與彈性 IP 地址的關聯。
- 從介面取消指派輔助私有 IPv4 地址時，會自動從輔助私有 IPv4 地址取消相關聯彈性 IP 地址的關聯。

## 使用多個 IPv4 地址

您可以將輔助私有 IPv4 地址指派給執行個體、建立彈性 IPv4 地址與輔助私有 IPv4 地址的關聯，以及取消指派輔助私有 IPv4 地址。

### 任務

- [指派次要私有 IPv4 地址](#)
- [設定作業系統以辨識次要私人 IPv4 位址](#)
- [建立彈性 IP 地址與次要私有 IPv4 地址的關聯](#)
- [檢視您的次要私有 IPv4 地址](#)
- [取消指派次要私有 IPv4 地址](#)

### 指派次要私有 IPv4 地址

您可以在啟動執行個體時或在執行個體執行之後，將輔助私有 IPv4 地址指派給執行個體的網路介面。

### 啟動執行個體時指派次要私有 IPv4 地址

1. 請遵循該程序來[啟動執行個體](#)。針對[網路設定](#)，選擇編輯。
2. 選取 VPC 和子網路。

3. 展開 進階網路組態。
4. 對於次要 IP，請選擇「自動指派」並輸入 IP 位址數目 (Amazon 會自動指派次要 IPv4 位址)，或選擇「手動指派」並輸入 IPv4 位址。
5. 完成其餘步驟以[啟動執行個體](#)。

使用命令列在啟動期間指派輔助 IPv4 地址

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- `--secondary-private-ip-addresses` 選項搭配 [run-instances](#) 命令 (AWS CLI)
- 使用指 [New-EC2Instance](#) 命令定義 `-NetworkInterface` 並指定 `PrivateIpAddresses` 參數 (AWS Tools for Windows PowerShell)。

將輔助私有 IPv4 地址指派給網路介面

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在瀏覽窗格中，選擇 [網路介面]，然後選取執行個體的網路介面。
3. 選擇 Actions (動作)、Manage IP Addresses (管理 IP 地址)。
4. 展開網路介面。在 [IPv4 位址] 下方，選擇 [指派新 IP 位址]。
5. 輸入位於執行個體子網路範圍內的特定 IPv4 位址，或將欄位保留空白，讓 Amazon 為您選取 IPv4 位址。
6. (選擇性) 選取允許在次要私人 IP 位址已指派給其他網路介面時重新指定該 IP 位址。
7. 選擇儲存。

或者，您可以將輔助私有 IPv4 地址指派給執行個體。在導覽窗格中選取執行個體，並選取執行個體，然後選取動作、聯網、管理 IP 地址。您可以設定如同上述步驟中所設定的資訊。IP 地址會指派給執行個體的主要網路介面 (eth0)。

使用命令列將次要私人 IPv4 位址指派給現有執行個體

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [assign-private-ip-addresses](#) (AWS CLI)
- [Register-EC2PrivateIpAddress](#) (AWS Tools for Windows PowerShell)

## 設定作業系統以辨識次要私人 IPv4 位址

在您將輔助私有 IPv4 地址指派給執行個體之後，需要設定執行個體的作業系統，以辨識輔助私有 IP 地址。

### Linux 執行個體

- 如果您使用的是 Amazon Linux，ec2-net-utils 套件可以為您處理此步驟。套件會設定您在執行個體執行時所連接的額外網路介面、在 DHCP 租賃續約期間重新整理輔助 IPv4 地址，以及更新相關路由規則。您可以使用指令立即重新整理介面清單，`sudo service network restart` 然後使用檢視 up-to-date 清單 `ip addr li`。如果您需要手動控制網路組態，則可以移除 ec2-net-utils 套件。如需詳細資訊，請參閱 [使用適用於 Amazon Linux 2 的 ec2-net-utils 設定您的網路介面](#)。
- 如果您使用的是其他 Linux 發行版本，請參閱 Linux 發行版本的文件。搜尋有關設定額外網路介面和輔助 IPv4 地址的資訊。如果執行個體在相同的子網上有兩個或以上的介面，請搜尋有關使用路由規則暫時處理非對稱路由的資訊。

### Windows 執行個體

如需詳細資訊，請參閱 [設定您的 Windows 執行個體的次要私有 IPv4 位址](#)。

### 建立彈性 IP 地址與次要私有 IPv4 地址的關聯

### 建立彈性 IP 地址與次要私有 IPv4 地址的關聯

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Elastic IPs (彈性 IP)。
3. 選取彈性 IP 位址的核取方塊
4. 選擇 [動作]、[關聯彈性 IP 位址]。
5. 針對 [資源類型]，選擇 [網路介面]。選取網路介面，然後從 [私人 IP 位址] 清單中選取次要 IP 位址。
6. 在 [網路介面] 中，選取網路介面。從 [私人 IP 位址] 清單中選取次要 IP 位址。
7. 對於私人 IP 位址，請選取次要 IP 位址。
8. 選擇 Associate (關聯)。

### 使用命令列建立彈性 IP 地址與輔助私有 IPv4 地址的關聯

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [associate-address](#) (AWS CLI)
- [Register-EC2Address](#) (AWS Tools for Windows PowerShell)

檢視您的次要私有 IPv4 地址

檢視指派給網路介面的私有 IPv4 地址

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊。
4. 在 [詳細資料] 索引標籤的 [IP 位址] 下，找出 [私人 IPv4 位址] 和 [次要私人 IPv4 位址]。

檢視指派給執行個體的私有 IPv4 地址

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取該執行個體的核取方塊。
4. 在 [網路] 索引標籤的 [網路詳細資料] 下，找出 [私人 IPv4 位址] 和 [次要私人 IPv4 位址]。

取消指派次要私有 IPv4 地址

如果您不再需要輔助私有 IPv4 地址，您可以從執行個體或網路介面取消指派。從網路介面取消指派輔助私有 IPv4 地址時，也會取消關聯彈性 IP 地址 (若其存在)。

從執行個體取消指派輔助私有 IPv4 地址

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選取動作、聯網、管理 IP 地址。
4. 展開網路介面。對於 IPv4 位址，請針對要取消指派的 IPv4 位址選擇「取消指派」。
5. 選擇儲存。

從網路介面取消指派輔助私有 IPv4 地址

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。

2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面，選擇 [動作]、[管理 IP 位址]。
4. 展開網路介面。對於 IPv4 位址，請針對要取消指派的 IPv4 位址選擇「取消指派」。
5. 選擇儲存。

使用命令列取消指派輔助私有 IPv4 地址

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [unassign-private-ip-addresses](#) (AWS CLI)
- [Unregister-EC2PrivateIpAddress](#) (AWS Tools for Windows PowerShell)

## 使用多個 IPv6 地址

您可以將多個 IPv6 地址指派給執行個體、檢視指派給執行個體的 IPv6 地址，以及從執行個體取消指派 IPv6 地址。

內容

- [指定多個 IPv6 地址](#)
- [檢視您的 IPv6 地址](#)
- [取消指派 IPv6 地址](#)

### 指定多個 IPv6 地址

您可以在啟動期間或之後將一或多個 IPv6 地址指派給執行個體。若要將 IPv6 地址指派給執行個體，在其中啟動執行個體的 VPC 和子網必須有相關聯的 IPv6 CIDR 區塊。

在啟動期間指派多個 IPv6 地址

1. 請遵循該程序來[啟動執行個體](#)。針對[網路設定](#)，選擇編輯。
2. 選取 VPC 和子網路。
3. 展開 進階網路組態。
4. 對於 IPv6 IP，請選擇「自動指派」和 IP 位址的數量 (Amazon 會自動指派 IPv6 位址)，或選擇「手動指派」並輸入 IPv6 位址。
5. 完成其餘步驟以[啟動執行個體](#)。

您可以使用 Instances (執行個體) 畫面 Amazon EC2 主控台，將多個 IPv6 地址指派給現有執行個體。這會將 IPv6 地址指派給執行個體的主要網路介面(eth0)。若要將特定 IPv6 地址指派給執行個體，請確定尚未將 IPv6 地址指派給另一個執行個體或網路介面。

#### 將多個 IPv6 地址指派給現有執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，選擇 [動作]、[網路]、[管理 IP 位址]。
4. 展開網路介面。對於 IPv6 位址，請選擇為每個要新增的 IPv6 位址指派新 IP 位址。您可以從子網路範圍指定 IPv6 位址，或將欄位保留空白，讓 Amazon 為您選擇 IPv6 位址。
5. 選擇儲存。

或者，您可以將多個 IPv6 地址指派給現有網路介面。必須已在具有相關聯 IPv6 CIDR 區塊的子網中建立網路介面。若要將特定 IPv6 地址指派給網路介面，請確定尚未將 IPv6 地址指派給另一個網路介面。

#### 將多個 IPv6 地址指派給網路介面

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取您的網路介面，選擇 [動作]、[管理 IP 位址]。
4. 展開網路介面。對於 IPv6 位址，請選擇為每個要新增的 IPv6 位址指派新 IP 位址。您可以從子網路範圍指定 IPv6 位址，或將欄位保留空白，讓 Amazon 為您選擇 IPv6 位址。
5. 選擇 Save (儲存)。

#### CLI 概觀

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- 在啟動期間指派 IPv6 地址：
  - 使用 `--ipv6-addresses` 或 `--ipv6-address-count` 選項並搭配 [run-instances](#) 命令 (AWS CLI)
  - 使用指 [New-EC2Instance](#) 命令定義 `-NetworkInterface` 並指定 `Ipv6Addresses` 或 `Ipv6AddressCount` 參數 (AWS Tools for Windows PowerShell)。
- 將 IPv6 地址指派給網路介面：



- [assign-ipv6-addresses](#) (AWS CLI)
- [Register-EC2Ipv6 AddressList](#) (AWS Tools for Windows PowerShell)

## 檢視您的 IPv6 地址

您可以檢視執行個體或網路介面的 IPv6 地址。

### 檢視指派給執行個體的 IPv6 地址

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體的核取方塊。
4. 在 [網路功能] 索引標籤上，找出 IPv6 位址欄位。

### 檢視指派給網路介面的 IPv6 地址

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊。
4. 在 [詳細資料] 索引標籤的 [IP 位址] 下，找出 IPv6 位址欄位。

## CLI 概觀

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- 檢視執行個體的 IPv6 地址：
  - [describe-instances](#) (AWS CLI)
  - [Get-EC2Instance](#) (AWS Tools for Windows PowerShell).
- 檢視網路介面的 IPv6 地址：
  - [describe-network-interfaces](#) (AWS CLI)
  - [Get-EC2NetworkInterface](#) (AWS Tools for Windows PowerShell)

## 取消指派 IPv6 地址

您可以從執行個體的主要網路介面取消指派 IPv6 地址，也可以從網路介面取消指派 IPv6 地址。

## 從執行個體取消指派 IPv6 地址

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體的核取方塊，然後選擇 [動作]、[網路]、[管理 IP 位址]。
4. 展開網路介面。在 IPv6 addresses (IPv6 地址) 下，選擇 IPv6 地址旁的 Unassign (取消指派)。
5. 選擇 Save (儲存)。

## 從網路介面取消指派 IPv6 地址

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊，然後選擇 [動作] > [管理 IP 位址]。
4. 展開網路介面。在 IPv6 addresses (IPv6 地址) 下，選擇 IPv6 地址旁的 Unassign (取消指派)。
5. 選擇 Save (儲存)。

## CLI 概觀

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [unassign-ipv6-addresses](#) (AWS CLI)
- [Unregister-EC2Ipv6 AddressList](#) (AWS Tools for Windows PowerShell)

## 設定您的 Windows 執行個體的次要私有 IPv4 位址

您可以指定執行個體的多個私有 IPv4 地址。將輔助私有 IPv4 地址指派給執行個體之後，必須設定執行個體的作業系統，以辨識輔助私有 IPv4 地址。

### Note

這些說明是基於視窗服務器 2022。這些步驟的實作可能會因 Windows 執行個體的作業系統而有所不同。

## 任務

- [必要條件](#)

- [步驟 1：在執行個體中設定靜態 IP 位址](#)
- [步驟 2：設定您的執行個體的次要私有 IP 地址](#)
- [步驟 3：設定應用程式使用次要私有 IP 地址](#)

## 必要條件

1. 將次要私有 IPv4 地址指派給執行個體的網路介面。您可以在啟動執行個體時或在執行個體執行之後，指派次要私有 IPv4 地址。如需詳細資訊，請參閱 [指派次要私有 IPv4 地址](#)。
2. 分配彈性 IP 地址並將其與次要私有 IPv4 地址關聯。如需詳細資訊，請參閱 [配置彈性 IP 地址](#) 和 [建立彈性 IP 地址與次要私有 IPv4 地址的關聯](#)。

## 步驟 1：在執行個體中設定靜態 IP 位址

若要讓 Windows 執行個體使用多個 IP 地址，您必須設定執行個體使用靜態 IP 定址，而不是 DHCP 伺服器。

### Important

在執行個體中設定靜態 IP 位址時，IP 位址必須完全符合主控台、CLI 或 API 中顯示的 IP 位址。如果您不正確地輸入這些 IP 地址，執行個體可能會變成無法存取。

## 在 Windows 執行個體上設定靜態 IP 定址

1. 連線到您的執行個體。
2. 執行下列步驟，尋找執行個體的 IP 地址、子網遮罩和預設閘道地址：
  - 在中執行下列命令 PowerShell：

```
ipconfig /all
```

檢閱輸出並記下網路介面的 IPv4 位址、子網路遮罩、預設閘道和 DNS 伺服器值。您的輸出應該類似於下列範例：

```
...
```

```
Ethernet adapter Ethernet 4:
```

```

Connection-specific DNS Suffix . : us-west-2.compute.internal
Description : Amazon Elastic Network Adapter #2
Physical Address. : 02-9C-3B-FC-8E-67
DHCP Enabled. : Yes
Autoconfiguration Enabled : Yes
Link-local IPv6 Address : fe80::f4d1:a773:5afa:cd1%7(Preferred)
IPv4 Address. : 10.200.0.128(Preferred)
Subnet Mask : 255.255.255.0
Lease Obtained. : Monday, April 8, 2024 12:19:29 PM
Lease Expires : Monday, April 8, 2024 4:49:30 PM
Default Gateway : 10.200.0.1
DHCP Server : 10.200.0.1
DHCPv6 IAID : 151166011
DHCPv6 Client DUID. : 00-01-00-01-2D-67-AC-FC-12-34-9A-BE-A5-
E7
DNS Servers : 10.200.0.2
NetBIOS over Tcpi. : Enabled

```

3. 在中執行下列命令，以開啟「網路和共用中心」 PowerShell：

```
& $env:SystemRoot\system32\control.exe ncpa.cpl
```

4. 開啟網路介面（「區域連線」或「乙太網路」）的內容（按一下滑鼠右鍵）選單，然後選擇「內容」。
5. 選擇網際網路通訊協定第 4 版 (TCP/IPv4)、內容。
6. 在網際網路通訊協定第 4 版 (TCP/IPv4) - 內容對話方塊中，選擇使用下列 IP 地址，並輸入下列值，然後選擇確定。

欄位	Value
IP 地址	前述步驟 2 中取得的 IPv4 地址。
子網遮罩	前述步驟 2 中取得的子網遮罩。
預設閘道	前述步驟 2 中取得的預設閘道地址。
慣用 DNS 伺服器	前述步驟 2 中取得的 DNS 伺服器。

欄位	Value
其他 DNS 伺服器	前述步驟 2 中取得的其他 DNS 伺服器。如果未列出其他 DNS 伺服器，請將此欄位保留空白。

**⚠ Important**

如果您將 IP 地址設定為目前 IP 地址以外的任何值，則會中斷與執行個體的連線。

Internet Protocol Version 4 (TCP/IPv4) Properties

General

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

Obtain an IP address automatically

Use the following IP address:

IP address: 10 . 200 . 0 . 128

Subnet mask: 255 . 255 . 255 . 0

Default gateway: 10 . 200 . 0 . 1

Obtain DNS server address automatically

Use the following DNS server addresses:

Preferred DNS server: 10 . 200 . 0 . 2

Alternate DNS server: . . . .

Validate settings upon exit

Advanced...

OK Cancel

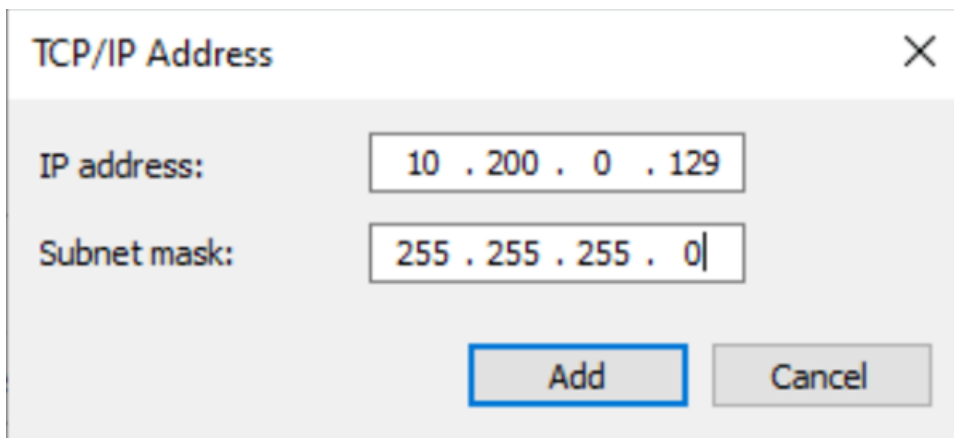
您將會中斷與 Windows 執行個體的 RDP 連線數秒，而執行個體會從使用 DHCP 轉換成靜態定址。執行個體會保留先前的 IP 地址資訊，但現在此資訊是靜態的，且並非透過 DHCP 管理。

## 步驟 2：設定您的執行個體的次要私有 IP 地址

在您設定 Windows 執行個體的靜態 IP 定址之後，即可準備第二個私有 IP 地址。

### 設定次要 IP 地址

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取您的執行個體。
3. 在 Networking (聯網) 標籤上，記下輔助 IP 地址。
4. 連線到您的執行個體。
5. 在 Windows 執行個體上，選擇開始、控制台。
6. 選擇網路和網際網路、網路和共用中心。
7. 選取網路介面 (「區域連線」或「乙太網路」)，然後選擇「內容」。
8. 在區域連線內容頁面上，選擇網際網路通訊協定第 4 版 (TCP/IPv4)、內容、進階。
9. 選擇 Add (新增)。
10. 在 TCP/IP 地址對話方塊中，針對 IP 地址輸入輔助私有 IP 地址。針對子網遮罩，輸入您在 [步驟 1：在執行個體中設定靜態 IP 位址](#) 中於主要私有 IP 地址輸入的相同子網遮罩，然後選擇新增。



The image shows a Windows dialog box titled "TCP/IP Address". It has a close button (X) in the top right corner. The dialog contains two input fields: "IP address:" with the value "10 . 200 . 0 . 129" and "Subnet mask:" with the value "255 . 255 . 255 . 0|". At the bottom, there are two buttons: "Add" and "Cancel".

11. 確認 IP 地址設定，然後選擇確定。

**Advanced TCP/IP Settings**

IP Settings | DNS | WINS

IP addresses

IP address	Subnet mask
10.200.0.128	255.255.255.0
10.200.0.129	255.255.255.0

Add... Edit... Remove

Default gateways:

Gateway	Metric
10.200.0.1	Automatic

Add... Edit... Remove

Automatic metric

Interface metric:

OK Cancel

12. 選擇確定、關閉。

13. 若要確認次要 IP 位址已新增至作業系統，請在中執行 `ipconfig /all` 指令 PowerShell。輸出應類似以下內容：

```
Ethernet adapter Ethernet 4:
```

```

Connection-specific DNS Suffix . . :
Description : Amazon Elastic Network Adapter #2
Physical Address. : 02-9C-3B-FC-8E-67
DHCP Enabled. : No
Autoconfiguration Enabled : Yes
Link-local IPv6 Address : fe80::f4d1:a773:5afa:cd1%7(Preferred)
IPv4 Address. : 10.200.0.128(Preferred)

```

```
Subnet Mask : 255.255.255.0
IPv4 Address. : 10.200.0.129(Preferred)
Subnet Mask : 255.255.255.0
Default Gateway : 10.200.0.1
DHCPv6 IAID : 151166011
DHCPv6 Client DUID. : 00-01-00-01-2D-67-AC-FC-12-34-9A-BE-A5-E7
DNS Servers : 10.200.0.2
NetBIOS over Tcpi. : Enabled
```

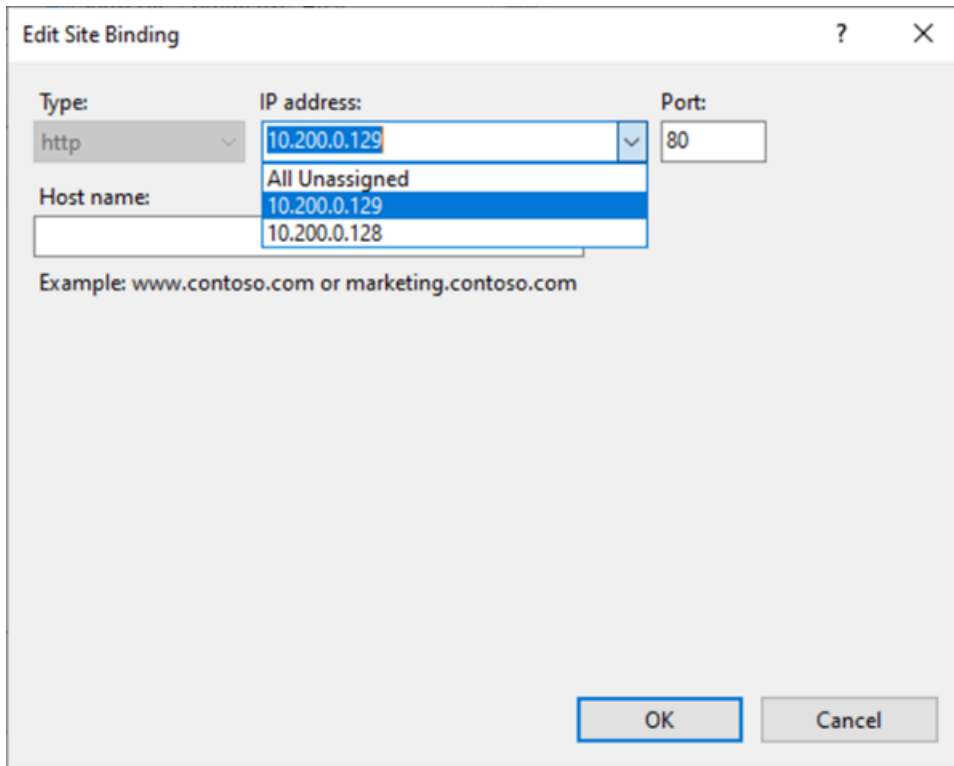
### 步驟 3：設定應用程式使用次要私有 IP 地址

您可以設定任何應用程式使用輔助私有 IP 地址。例如，如果您的執行個體在 IIS 上執行網站，即可設定 IIS 使用輔助私有 IP 地址。

#### 設定 IIS 使用輔助私有 IP 地址

1. 連線到您的執行個體。
2. 開啟 Internet Information Services (IIS) 管理員。
3. 在連線窗格中，展開網站。
4. 開啟您網站的內容 (右鍵) 選單，然後選擇編輯繫結。
5. 在網站繫結對話方塊中，針對類型，選擇 http、編輯。
6. 在編輯網站繫結對話方塊中，針對 IP 地址，選取輔助私有 IP 地址。(根據預設，每個網站都接受來自所有 IP 地址的 HTTP 請求)。





The screenshot shows a dialog box titled "Edit Site Binding". It contains the following fields and options:

- Type:** A dropdown menu with "http" selected.
- IP address:** A text input field containing "10.200.0.129".
- Port:** A text input field containing "80".
- Host name:** A dropdown menu with "All Unassigned" selected. A list of options is visible below it: "All Unassigned", "10.200.0.129", and "10.200.0.128".

Below the fields, there is an example: "Example: www.contoso.com or marketing.contoso.com". At the bottom right, there are "OK" and "Cancel" buttons.

7. 選擇確定、關閉。

## EC2 執行個體主機名稱

建立 EC2 執行個體時，AWS 會為該執行個體建立主機名稱。如需主機名稱類型及其佈建方式的詳細資訊 AWS，請參閱[Amazon EC2 執行個體主機名稱類型](#)。Amazon 所提供的 DNS 伺服器可將 Amazon 提供的主機名稱解析為 IPv4 和 IPv6 地址。Amazon DNS 伺服器位於 VPC 網路範圍基礎加 2。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的[VPC 的 DNS 屬性](#)。

## Link-local 地址

Link-local 地址是眾所周知不可路由的 IP 地址。Amazon EC2 使用來自 Link-local 地址空間的地址來提供只能從 EC2 執行個體存取的服務。這些服務不在執行個體上執行，而是在基礎主機上執行。當您存取這些服務的 Link-local 地址時，您正在與 Xen 虛擬化管理程序或 Nitro 控制器進行通訊。

### Link-local 地址範圍

- IPv4 – 169.254.0.0/16 (169.254.0.0 至 169.254.255.255)
- IPv6 – fe80::/10

## 使用 link-local 地址存取的服務

- [執行個體中繼資料服務](#)
- [Amazon Route 53 Resolver](#) (也稱為 Amazon DNS 伺服器)
- [Amazon Time Sync Service](#)

## Amazon EC2 執行個體主機名稱類型

本節說明將執行個體啟動至 VPC 子網時可用的 Amazon EC2 執行個體客體作業系統主機名稱類型。

主機名稱可區分您網路上的 EC2 執行個體。例如，如果您希望執行指令碼，與網路上的部分或全部執行個體通訊，則可以使用執行個體的主機名稱。

### 目錄

- [EC2 主機名稱的類型](#)
- [查看資源名稱與 IP 名稱的位置](#)
- [如何決定是選擇資源名稱還是 IP 名稱](#)
- [修改主機名稱類型和 DNS 主機名稱組態](#)

## EC2 主機名稱的類型

在 VPC 中啟動 EC2 執行個體時，客體作業系統主機名稱有兩種主機名稱類型：

- IP name (IP 名稱)：舊式命名結構描述，其中當您啟動執行個體時，執行個體的私有 IPv4 地址會包含在執行個體的主機名稱中。IP 名稱存在於 EC2 執行個體的生命週期中。用作私有 DNS 主機名稱時，它只會傳回私有 IPv4 地址 (A 記錄)。
- 資源名稱：當您啟動執行個體時，EC2 執行個體 ID 會包含在執行個體的主機名稱中。資源名稱存在於 EC2 執行個體的生命週期中。用作私有 DNS 主機名稱時，它可以同時傳回私有 IPv4 地址 (A 記錄) 和/或 IPv6 全域單點傳送地址 (AAAA 記錄)。

EC2 執行個體客體作業系統主機名稱類型取決於子網設定：

- 如果執行個體啟動至僅限 IPv4 子網，您可以選取 IP 名稱或資源名稱。
- 如果執行個體啟動至雙堆疊 (IPv4+IPv6) 子網，您可以選取 IP 名稱或資源名稱。
- 如果執行個體啟動至僅限 IPv6 子網，則會自動使用資源名稱。

## 目錄

- [IP 名稱](#)
- [資源名稱](#)
- [IP 名稱與資源名稱之間的區別](#)

## IP 名稱

當您使用 Hostname type (主機名稱類型) 的 IP name (IP 名稱) 啟動 EC2 執行個體時，客體作業系統主機名稱會設定為使用私有 IPv4 地址。

- us-east-1 中執行個體的格式：*private-ipv4-address.ec2.internal*
- 範例：*ip-10-24-34-0.ec2.internal*
- 任何其他 AWS 區域中執行個體的格式：*private-ipv4-address.region.compute.internal*
- 範例：*ip-10-24-34-0.us-west-2.compute.internal*

## 資源名稱

在僅限 IPv6 子網中啟動 EC2 執行個體時，預設會將 Hostname type (主機名稱類型) 選取為 Resource name (資源名稱)。在僅限 IPv4 或雙堆疊 (IPv4+IPv6) 子網中啟動執行個體時，Resource name (資源名稱) 是可選取的選項。在啟動執行個體之後，您可以管理主機名稱組態。如需詳細資訊，請參閱 [修改主機名稱類型和 DNS 主機名稱組態](#)。

當您使用 Hostname type (主機名稱類型) 的 Resource name (資源名稱) 啟動 EC2 執行個體時，客體作業系統主機名稱會設定為使用 EC2 執行個體 ID。

- us-east-1 中執行個體的格式：*ec2-instance-id.ec2.internal*
- 範例：*i-0123456789abcdef.ec2.internal*
- 任何其他 AWS 區域中執行個體的格式：*ec2-instance-id.region.compute.internal*
- 範例：*i-0123456789abcdef.us-west-2.compute.internal*

## IP 名稱與資源名稱之間的區別

IP 名稱與資源名稱的 DNS 查詢共存，以確保向後相容性並允許您從 IP 型主機命名遷移至資源型命名。若為以 IP 名稱為基礎的私有 DNS 主機名稱，您無法設定是否要回應執行個體的 DNS A 記錄查詢。DNS A 記錄查詢始終會回應，而不管客體作業系統主機名稱設定如何。相反地，若為以資源名稱

為基礎的私有 DNS 主機名稱，您可以設定是否要回應執行個體的 DNS A 和/或 DNS AAAA 查詢。您可以在啟動執行個體或修改子網時設定回應行為。如需詳細資訊，請參閱 [修改主機名稱類型和 DNS 主機名稱組態](#)。

## 查看資源名稱與 IP 名稱的位置

本節介紹您在 EC2 主控台中查看主機名稱類型資源名稱和 IP 名稱的位置。

### 目錄

- [建立 EC2 執行個體時](#)
- [檢視現有 EC2 執行個體的詳細資訊時](#)

### 建立 EC2 執行個體時

當您建立 EC2 執行個體時，根據您選取的子網類型，Hostname type (主機名稱類型) 的 Resource name (資源名稱) 可能是可用的，或者可能已選取且無法修改。本節介紹您查看主機名稱類型資源名稱和 IP 名稱的案例。

#### 案例 1

您可以在精靈中建立 EC2 執行個體 (請參閱 [使用新的啟動執行個體精靈啟動執行個體](#))，並在設定詳細資訊時，選擇已設定為僅 IPv6 的子網。

在此情況下，Hostname type (主機名稱類型) 的 Resource name (資源名稱) 會自動選取且無法修改。Enable IP name IPv4 (A record) DNS requests (啟用 IP 名稱 IPv4 (A 記錄) DNS 請求) 和 Enable resource-based IPv4 (A record) DNS requests (啟用以資源為基礎的 IPv4 (A 記錄) DNS 請求) 的 DNS Hostname (DNS 主機名稱) 選項將自動取消選取，並且不可修改。預設會選取 Enable resource-based IPv6 (AAAA record) DNS requests (啟用以資源為基礎的 IPv6 (AAAA 記錄) DNS 請求)，但可以修改。如果選取此項，對資源名稱的 DNS 請求將解析為此 EC2 執行個體的 IPv6 地址 (AAAA 記錄)。

#### 案例 2

您可以在精靈中建立 EC2 執行個體 (請參閱 [使用新的啟動執行個體精靈啟動執行個體](#))，並在設定詳細資訊時，選擇以 IPv4 CIDR 區塊設定的子網，或同時以 IPv4 和 IPv6 CIDR 區塊 (「雙堆疊」) 設定的子網。

在此情況下，會自動選取 Enable IP name IPv4 (A record) DNS requests (啟用 IP 名稱 IPv4 (A 記錄) DNS 請求) 且無法變更。這表示對 IP 名稱的請求將解析為此 EC2 執行個體的 IPv4 地址 (A 記錄)。

選項預設為子網的組態，但您可以修改此執行個體的選項，取決於子網設定：

- **Hostname type (主機名稱類型)**：決定要將 EC2 執行個體的客體作業系統主機名稱做為資源名稱還是 IP 名稱。預設值為 IP name (IP 名稱)。
- **Enable resource-based IPv4 (A record) DNS requests (啟用資源型 IPv4 (A 記錄) DNS 請求)**：決定對資源名稱的請求是否解析為此 EC2 執行個體的私有 IPv4 地址 (A 記錄)。預設不會選取此選項。
- **Enable resource-based IPv6 (AAAA record) DNS requests (啟用資源型 IPv6 (AAAA 記錄) DNS 請求)**：決定對資源名稱的請求是否解析為此 EC2 執行個體的 IPv6 GUA 地址 (AAAA 記錄)。預設不會選取此選項。

## 檢視現有 EC2 執行個體的詳細資訊時

您可以在 EC2 執行個體的 Details (詳細資訊) 標籤中查看現有 EC2 執行個體的主機名稱值：

- **Hostname type (主機名稱類型)**：IP 名稱或資源名稱格式的主機名稱。
- **Private IP DNS name (IPv4 only) (私有 IP DNS 名稱 (僅限 IPv4))**：IP 名稱，其一律會解析為執行個體的私有 IPv4 地址。
- **Private resource DNS name (私有資源 DNS 名稱)**：資源名稱，其會解析為針對此執行個體選取的 DNS 記錄。
- **Answer private resource DNS name (回答私有資源 DNS 名稱)**：資源名稱，其會解析為 IPv4 (A)、IPv6 (AAA) 或 IPv4 和 IPv6 (A 和 AAA) DNS 記錄。

此外，如果直接透過 SSH 連線至 EC2 執行個體，並輸入 `hostname` 命令，您會看到採用 IP 名稱或資源名稱格式。

## 如何決定是選擇資源名稱還是 IP 名稱

啟動 EC2 執行個體時 (請參閱 [使用新的啟動執行個體精靈啟動執行個體](#))，如果您將 Hostname type (主機名稱類型) 選擇為 Resource name (資源名稱)，EC2 執行個體將以資源名稱格式的主機名稱啟動。在這種情況下，此 EC2 執行個體的 DNS 記錄也可以指向資源名稱。這讓您能夠靈活地選擇該主機名稱是解析為執行個體的 IPv4 地址、IPv6 地址，還是同時解析為 IPv4 和 IPv6 地址。如果您計劃在將來使用 IPv6，或者如果您目前正在使用雙堆疊子網，則最好針對 Hostname type (主機名稱類型) 使用 Resource name (資源名稱)，以便您變更執行個體主機名稱的 DNS 解析，而無需對 DNS 記錄本身做出任何變更。資源名稱允許您在 EC2 執行個體上新增和移除 IPv4 和 IPv6 DNS 解析。

而如果您將 Hostname type (主機名稱類型) 選擇為 IP name (IP 名稱)，並將其用作 DNS 主機名稱，則只能解析為執行個體的 IPv4 地址。即使執行個體同時具有 IPv4 地址和與之關聯的 IPv6 地址，也不會解析為執行個體的 IPv6 地址。

## 修改主機名稱類型和 DNS 主機名稱組態

請依照本節中的步驟，在子網或 EC2 執行個體啟動後，為其修改主機名稱類型和 DNS 主機名稱組態。

### 目錄

- [子網](#)
- [EC2 執行個體](#)

### 子網

在 VPC 主控台中選取子網，然後選取 Actions (動作)、Edit subnet settings (編輯子網設定)，以修改子網的組態。

#### Note

變更子網設定不會變更已在子網中啟動的 EC2 執行個體的組態。

- Hostname type (主機名稱類型)：決定要將 EC2 執行個體 (已在子網中啟動) 之客體作業系統主機名稱的預設設定做為資源名稱還是 IP 名稱。
- Enable DNS hostname IPv4 (A record) requests (啟用 DNS 主機名稱 IPv4 (A 記錄) 請求)：決定對資源名稱的 DNS 請求/查詢是否解析為此 EC2 執行個體的私有 IPv4 地址 (A 記錄)。
- Enable DNS hostname IPv6 (AAAA record) requests (啟用 DNS 主機名稱 IPv6 (AAAA 記錄) 請求)：決定對資源名稱的 DSN 請求/查詢是否解析為此 EC2 執行個體的 IPv6 地址 (AAAA 記錄)。

### EC2 執行個體

請遵循本節中的步驟來修改 EC2 執行個體的主機名稱類型和 DNS 主機名稱組態。

#### Important

- 若要變更 Use resource based naming as guest OS hostname (使用資源型命名作為客體作業系統主機名稱) 設定，必須首先停止執行個體。若要變更 Answer DNS hostname IPv4 (A record) request (回答 DNS 主機名稱 IPv4 (A 記錄) 請求) 或 Answer DNS hostname IPv6 (AAAA record) requests (回答 DNS 主機名稱 IPv6 (AAAA 記錄) 請求) 設定，您不必停止執行個體。

- 若要修改非 EBS 支援之 EC2 執行個體類型的任何設定，您無法停止執行個體。您必須終止執行個體，並使用所需的主機名稱類型和 DNS 主機名稱組態來啟動新的執行個體。

## 修改 EC2 執行個體的主機名稱類型和 DNS 主機名稱組態

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 如果您即將變更 Use resource based naming as guest OS hostname (使用資源型命名作為客體作業系統主機名稱) 設定，首先停止 EC2 執行個體。否則，請跳過這個步驟。

若要停止執行個體，請選取執行個體，然後選取 Instance state (執行個體狀態)、Stop instance (停止執行個體)。

3. 選取執行個體，然後選取 Actions (動作)、Instance settings (執行個體設定)、Change resource based naming options (變更資源型命名選項)。
  - Use resource based naming as guest OS hostname (使用資源型命名作為客體作業系統主機名稱)：決定要將 EC2 執行個體的客體作業系統主機名稱做為資源名稱還是 IP 名稱。
  - Answer DNS hostname IPv4 (A record) requests (回答 DNS 主機名稱 IPV4 (A 記錄) 請求)：決定對資源名稱的 DNS 請求/查詢是否解析為此 EC2 執行個體的私有 IPv4 地址。
  - Answer DNS hostname IPv6 (AAAA record) requests (回答 DNS 主機名稱 IPV6 (AAAA 記錄) 請求)：決定對資源名稱的 DSN 請求/查詢是否解析為此 EC2 執行個體的 IPv6 地址 (AAAA 記錄)。
4. 選擇儲存。
5. 如果已停止執行個體，請重新啟動它。

## 在 Amazon EC2 中使用自有 IP 地址 (BYOIP)

您可以將部分或全部的公開路由 IPv4 或 IPv6 位址範圍從內部部署網路移至您的 AWS 帳戶。您可以繼續控制地址範圍，並且可以通過在 Internet 上通告地址範圍 AWS。將地址範圍帶到之後 AWS，它會以地址集區的形式出現在您的 AWS 帳戶中。

如需可使用 BYOIP 的區域清單，請參閱 [區域可用性](#)。

### Note

- 本頁中的步驟說明如何使用您自己的 IP 地址範圍僅供在 Amazon EC2 中使用。

- 若要使用您自己的 IP 位址範圍 AWS Global Accelerator，請參閱AWS Global Accelerator 開發人員指南中的[攜帶您自己的 IP 位址 \(BYOIP\)](#)。
- 若要攜帶您自己的 IP 位址範圍以供搭配使用 Amazon VPC IP Address Manager，請參閱 Amazon VPC [IPAM 使用者指南中的教學課程：將 IP 位址帶到 IPAM](#)。

## 目錄

- [BYOIP 定義](#)
- [需求與配額](#)
- [BYOIP 地址範圍加入先決條件](#)
- [加入您的 BYOIP](#)
- [使用您的位址範圍](#)
- [驗證您的 BYOIP](#)
- [區域可用性](#)
- [Local Zone 可用性](#)
- [進一步了解](#)

## BYOIP 定義

- X.509 自我簽署的憑證 - 最常用來加密和驗證網路中資料的憑證標準。它是用來驗證從 RDAP 記錄 IP 空間的控制權的證書。AWS 若要取得有關 X.509 憑證的更多資訊，請參閱 [RFC 3280](#)。
- 自治系統編號 (ASN) – 這是一個全域唯一識別符，可定義由一個或多個網路業者執行的 IP 字首群組，而這些網路業者維護單一且明確定義的路由政策。
- 區域網際網路註冊管理機構 (RIR) – 管理全球某個地區內 IP 地址和 ASN 之配置與註冊的組織。
- 註冊資料訪問協定 (RDAP)：用來查詢 RIR 目前註冊資料的唯讀協定。查詢的 RIR 資料庫中的項目稱為「RDAP 記錄」。客戶必須透過 RIR 提供的機制更新特定記錄類型。查詢這些記錄 AWS 以驗證 RIR 中地址空間的控制權。
- 路由來源授權 (ROA) - 由 RIR 建立的物件，可供客戶於特定自治系統中驗證 IP 公告。如需概觀，請參閱 ARIN 網站上的[路由來源授權 \(ROAS\)](#)。
- 本地網際網路註冊機構 (LIR) – 從 RIR 為其客戶分配 IP 地址區塊的網際網路服務供應商等組織。



## 需求與配額

- 地址範圍必須在您的區域互聯網註冊處 ( RIR ) 註冊。有關地理區域的任何政策，請參閱您的 RIR。BYOIP 目前支援在美洲網際網路號碼註冊管理機構 (ARIN) 或歐洲 IP 網路資源協調中心 (RIPE) 或亞太區域資訊中心 (APNIC) 進行註冊。它必須以商業或機構實體註冊，而無法以個人身分註冊。
- 您可以使用的最特定 IPv4 位址範圍是 /24。
- [您可以提供的最具體 IPv6 位址範圍為 /48 \(適用於可公開廣告的 CIDR\)，而不是可公開廣告的 CIDR 則為 /56。](#)
- 對於不可公開廣告的 CIDR 範圍，不需要 ROA，但仍需要更新 RDAP 記錄。
- 您一次可以將每個地址範圍帶到一個地 AWS 區。
- 您可以將每個 AWS 區域總共五個 BYOIP IPv4 和 IPv6 位址範圍帶入您的帳戶。AWS 您無法使用 Service Quotas 主控台調整 BYOIP CIDR 的配額，但您可以透過連絡 Sup AWS port 中心要求增加配額，如中的[AWS 服務配額](#)中所述。AWS 一般參考
- 除非您使用 Amazon VPC IP 位址管理員 (IPAM) 並將 IPAM 與 Organizations 整合，AWS RAM 否則您無法使用與其他帳戶共用您的 IP 位址範圍。AWS 如需詳細資訊，請參閱 [Amazon VPC IPAM 使用者指南中的整合 IPAM 與 Organizations AWS 織](#)。
- IP 地址範圍中的地址都必須有良好的歷史記錄。我們會調查 IP 地址範圍的評價，如果 IP 地址範圍包含的 IP 地址評價不佳或與惡意行為有關，我們保留拒絕該範圍的權利。
- 舊版地址空間 (即在區域網際網路註冊管理機構 (RIR) 系統形成之前，由網際網路號碼分配局 (IANA) 的中央登錄檔所分配的 IPv4 地址空間) 仍然需要對應的 ROA 物件。
- 對於 LIR，其通常使用手動程序來更新其記錄。這可能需要幾天時間才能部署，依 LIR 而定。
- 大型 CIDR 區塊需要單一的 ROA 物件和 RDAP 記錄。您可以使用單一物件和記錄 AWS，將多個較小的 CIDR AWS 區塊從該範圍引入，甚至跨多個區域。
- Wavelength 區域或開啟時不支援 BYOIP。AWS Outposts
- 不要在 RADb 或任何其他 LIR 中對 BYOIP 進行任何手動變更。BYOIP 會自動更新 RADb。任何包含 BYOIP ASN 的手動變更將會導致 BYOIP 佈建操作失敗。
- 將 IPv4 位址範圍帶到之後 AWS，您就可以使用範圍內的所有 IP 位址，包括第一個位址 (網路位址) 和最後一個位址 (廣播位址)。

## BYOIP 地址範圍加入先決條件

BYOIP 的加入過程分為兩個階段，您必須執行三個步驟。這些步驟與下圖中說明的步驟相對應。我們在本文件中包含手動步驟，但您的 RIR 可能會提供受管服務來協助您完成這些步驟。

## 準備階段

1. [請建立私密金鑰](#)，然後使用其來產生自我簽署的 X.509 憑證進行身分驗證。此憑證僅在佈建階段使用。

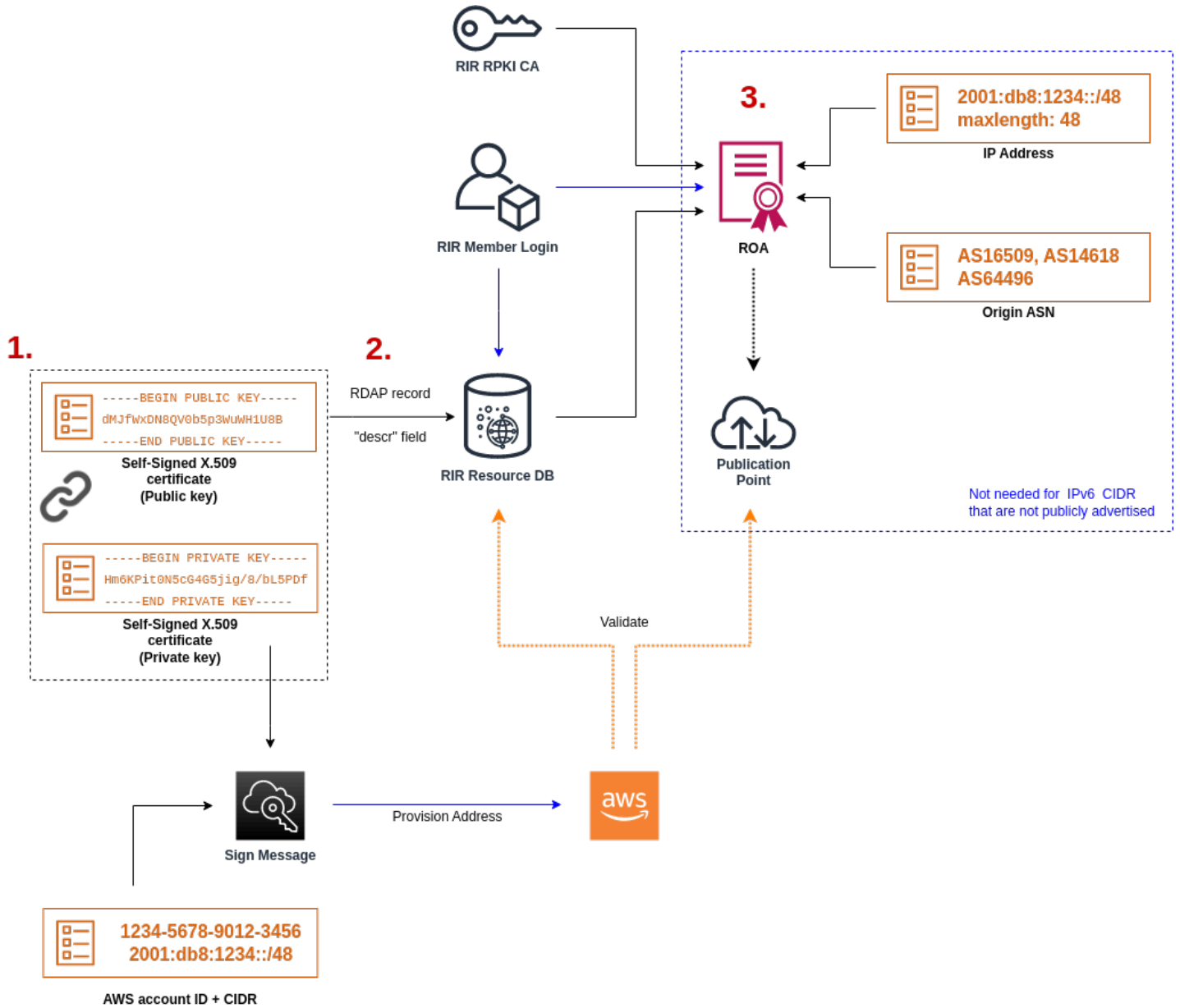
## RIR 組態階段

2. [將自我簽署的憑證上傳](#)至您 RDAP 記錄註解。

3. [在 RIR 中建立一個 ROA 物件](#)。ROA 定義了所需的地址範圍、允許公告地址範圍的自治系統編號 (ASN) 以及向您 RIR 的資源公有金鑰基礎設施 (RPKI) 註冊的到期日期。

### Note

不可公開公告的 IPv6 地址空間不需要 ROA。



如要引入多個不連續的地址範圍，您必須針對每個地址範圍重複此程序。但是，如果跨多個不同區域分割連續 AWS 區塊，則不需要重複準備和 RIR 設定步驟。

引入地址範圍不會影響您先前引入的任何地址範圍。

**⚠ Important**

加入您的地址範圍之前，請先完成下列先決條件。本節中的任務需要 Linux 終端，而且可以使用 Linux、[AWS CloudShell](#) 或 [Windows Subsystem for Linux](#) 執行。

## 1. 建立私密金鑰並產生 X.509 憑證

使用以下程序建立自簽 X.509 憑證，並將其加入到您 RIR 的 RDAP 記錄中。此金鑰對可用來向 RIR 驗證地址範圍。openssl 命令需要 OpenSSL 1.0.2 版或更新版本。

複製下列命令，並只取代預留位置值 (以彩色斜體文字表示)。

此程序會遵循加密私有 RSA 金鑰並需要密碼短語才能進行存取的最佳實務。

### 1. 產生 RSA 2048 位元私密金鑰，如下所示。

```
$ openssl genpkey -aes256 -algorithm RSA -pkeyopt rsa_keygen_bits:2048 -out
private-key.pem
```

-aes256 參數會指定用來加密私有金鑰的演算法。此命令會傳回下列輸出，包括設定密碼短語的提示：

```
.....+++
.+++
Enter PEM pass phrase: xxxxxxx
Verifying - Enter PEM pass phrase: xxxxxxx
```

您可以使用下列命令來檢查金鑰：

```
$ openssl pkey -in private-key.pem -text
```

這會傳回密碼短語提示和金鑰內容，應類似於以下內容：

```
Enter pass phrase for private-key.pem: xxxxxxx
-----BEGIN PRIVATE KEY-----
MIIEVgIBADANBgkqhkiG9w0BAQEFAASCbKgwggSkAgEAAoIBAQDFBXHRI4HVKAhh
3seiciooizCRTbJe1+YsxNTja4XyKypVGIFWDGhZs44FCH1P00SVJ+NqP74w96oM
7DPS3xo9kaQyZBFn2YEp2EBq5vf307KHNRmZZUmkn0zH0SEpNmY2fMxISBxewlxR
FAniwmSd/8TDvHJMY9FvAIVWuTsv510tJKk+a91K4+t03UdDR7Sno5WEXefsBrW3
g1ydo3TBsx8i5/YiV0cNApy7ge2/FiwY3aCXJB6r6nuF6H8mRgI4r4vkMRs01AhJ
DnZPNNeweboo+K3Q31wbgbmOKD/z9svk8N/+hUTBtIX0fRtbG+PLIw3xWRHGGrMSn2
BzsPVuDLAgMBAEACggEACiJUj2hfJkKv47Dc3es3Zex67A5uDVjXmxf0x2Xhdupn
fAcNqAptV6fXt0SPUNbhUxbBKNbshoJGufFwXPli1SXnpzvkdU4Hyc04zgbhXFsE
RNYjYf0GzTPwdBLpNMB6k3Tp4RHse6dNr1H0jDhpioL8cQEBdBJyVF5X0wymEbmV
mC0jgH/MxsBAPWW6ZKicg9ULM1WiAZ3MRAZPjHHgpYkAAsUWKAbCBwVQcVjG059W
jfZjzTX5pQtVvH68ruciH88DTZCwjCkjBhxg+0IkJBLE5wkh82jIHSivZ63flwLw
```

```

z+E0+HhELSZJrn2MY6Jxmik3qNNUOF/Z+3msdj2luQKBgQDjw1C/3jxp8zJy6P8o
JQKv7TdvMwUj4VSW0HZBHLv4evJaaia0uQjIo1UDa8AYitqhX1NmCCehGH8yuXj/
v6V3CzMKDkmRr1Nr0NnSz5QsndQ04Z6ihAQ1PmJ96g4wKtgoC7AYpyP0g1a+4/sj
b1+o3YQI4pD/F71c+qaztH7PRwKBgQDdc23yNmT3+Jyptf0fKjEv0NK+XwUKzi9c
L/0zBq5y0IC1Pz2T85g0e1i8kwZws+xlPG6uBT61mIJELd0k59FyupNu4dPvX5SD
6GGqdx4jk9KvI74usGe0BohmF0phTHkrWKBxXiyT0oS8zjnJlEn8ysIpGg028jJR
LpaHNZ/MXQKBgQDFLNcnS0LzpsS2aK0tzyZU8SMYqVH0GMxj7quhneBq2T6FbiLD
T9TV1YaGNZ0j71vQaLI19q0ubWymbautH00p5KV8owdf4+bf1/NJaPI0zhDUSIjD
Qo01WW31Z9XDSRhKFTnWzmCjBdeIcajyzf10YKsycAW91Itu8aBrMndnQKBgQDb
nNp/JyRwqj0rNljK7DHEs+SD39kHQzzCfQd+dnTPv2sc06+cpym3yu1QcbokULpy
fmRo3bin/pvJQ3aZX/Bdh9woTXqhXDdrrSwWInVYMQPyPk8f/D9mIOJp5FUWMwHD
U+whIZSxsEeE+jtixlWtheKRYkQmzQZXbWdIhYyI3QKBgD+F/6wcZ85QW8nAUyKA
3WrSIx/3cwDgm4NRGct8Z0ZjTHjiy9ojMOD1L7iMhRQ/3k3hUsin5LDMp/ryWGG
x4uIaLat40kiC7T4I66DM7P59euqdz3w0PD+VU+h7GSivvsFDdySut7bNK0AUVLh
dMJfWxDN8QV0b5p3WuWH1U8B
-----END PRIVATE KEY-----

```

-----END PRIVATE KEY-----

Private-Key: (2048 bit)

modulus:

```

00:c5:05:71:d1:23:81:d5:28:08:61:de:c7:a2:72:
2a:28:8b:30:91:4d:b2:5e:d7:e6:2c:c4:d4:e3:6b:
85:f2:2b:2a:55:18:81:56:0c:68:59:b3:8e:05:08:
79:4f:38:e4:95:27:e3:6a:3f:be:30:f7:aa:0c:ec:
33:d2:df:1a:3d:91:a4:32:64:11:67:d9:81:29:d8:
40:6a:e6:f7:f7:d3:b2:87:35:19:99:65:49:a4:9f:
4c:c7:39:21:29:36:66:36:7c:cc:48:48:1c:5e:c2:
5c:51:14:09:e2:c2:64:9d:ff:c4:c3:bc:72:4c:63:
d1:6f:00:8b:d6:b9:3b:2f:e6:5d:2d:24:a9:3e:6b:
dd:4a:e3:eb:4e:dd:47:43:47:b4:a7:a3:95:97:13:
17:ec:06:b5:b7:83:5c:9d:a3:74:c1:b3:1f:22:e7:
f6:22:54:e7:0d:02:9c:bb:81:ed:bf:16:2c:18:dd:
a0:97:24:1e:ab:ea:7b:85:e8:7f:26:46:02:38:af:
8b:e4:31:1b:0e:94:08:49:0e:76:4f:35:ec:1e:6e:
8a:3e:2b:74:37:97:06:e0:6e:63:8a:0f:fc:fd:b2:
f9:3c:37:ff:a1:51:30:6d:21:7d:1f:46:d6:c6:f8:
f2:c8:c3:7c:56:44:71:ab:31:29:f6:07:3b:0f:56:
e0:cb

```

publicExponent: 65537 (0x10001)

privateExponent:

```

0a:22:54:8f:68:5f:26:42:af:e3:b0:dc:dd:eb:37:
65:ec:7a:ec:0e:6e:0d:58:d7:9b:17:e8:c7:65:e1:
76:ea:67:7c:07:0d:a8:0a:6d:57:a7:d7:b7:44:8f:
50:d6:e1:53:16:c1:28:d6:ec:86:82:46:b9:f1:70:
5c:f9:62:d5:25:e7:a7:3b:e4:75:4e:07:c9:ca:38:
ce:06:e1:5c:5b:04:44:d6:23:61:f3:86:cd:33:f0:

```

```
74:12:e9:34:c0:7a:93:74:e9:e1:11:ec:7b:a7:4d:
ae:51:f4:8c:38:69:8a:82:fc:71:01:01:74:12:72:
54:5e:57:d3:0c:a6:11:b9:95:98:2d:23:80:7f:cc:
c6:c0:40:3d:65:ba:64:a8:9c:83:d5:0b:32:55:a2:
01:9d:cc:44:06:4f:8c:71:e0:a5:89:00:02:c5:16:
28:06:c2:07:05:50:71:58:c6:3b:9f:56:8d:f6:63:
cd:35:f9:a5:0b:55:54:7e:bc:ae:e7:22:1f:cf:03:
4d:90:b0:8c:29:23:06:1c:60:f8:e2:24:24:12:c4:
e7:09:21:f3:68:c8:1d:28:af:67:ad:df:97:02:f0:
cf:e1:34:f8:78:44:2d:26:49:ae:7d:8c:63:a2:71:
9a:29:37:a8:d3:54:38:5f:d9:fb:79:ac:76:3d:a5:
b9
```

prime1:

```
00:e3:c2:50:bf:de:3c:69:f3:32:72:e8:ff:28:25:
02:af:ed:37:6f:33:05:23:e1:54:96:38:76:41:1c:
bb:f8:7a:f2:5a:6a:26:b4:b9:08:c8:a3:55:03:6b:
c0:18:8a:da:a1:5f:53:66:08:27:a1:18:7f:32:b9:
78:ff:bf:a5:77:0b:33:0a:0e:49:91:af:53:6b:38:
d9:d2:cf:94:2c:9d:d4:34:e1:9e:a2:84:04:25:3e:
62:7d:ea:0e:30:2a:d8:28:0b:b0:18:a7:23:f4:83:
56:be:e3:fb:23:6f:5f:a8:dd:84:08:e2:90:ff:17:
bd:5c:fa:a6:b3:b4:7e:cf:47
```

prime2:

```
00:dd:73:6d:f2:36:64:f7:f8:9c:a9:b5:fd:1f:2a:
31:2f:38:d2:be:c7:05:0a:ce:2f:5c:2f:f3:b3:06:
ae:72:38:80:b5:3f:3d:93:f3:98:0e:7b:58:bc:93:
06:70:b3:ec:65:a4:6e:ae:05:3e:a5:98:82:44:2d:
dd:24:e7:d1:72:ba:93:6e:e1:d3:ef:5f:94:83:e8:
61:aa:77:1e:23:93:d2:af:23:be:2e:b0:67:8e:06:
88:66:17:4a:61:4c:79:2b:58:a0:71:5e:2c:93:d2:
84:bc:ce:39:c9:94:49:fc:ca:c2:29:1a:03:b6:f2:
38:eb:2e:96:87:35:9f:cc:5d
```

exponent1:

```
00:df:2c:d7:27:4b:42:f3:a6:c4:b6:68:ad:2d:cf:
26:54:f1:23:32:a9:51:ce:18:cc:63:ee:ab:a1:9d:
e0:6a:d9:3e:85:6e:22:c3:4f:d4:d5:95:86:86:35:
9d:23:ef:5b:d0:68:b2:35:f6:a3:ae:6d:6c:a6:6d:
ab:ad:1f:43:a9:e4:a5:7c:a3:07:5f:e3:e6:df:d7:
f3:49:68:f2:0e:ce:10:d4:48:88:c3:42:8d:35:59:
6d:f5:67:d5:c3:49:18:4a:15:39:d6:ce:60:a3:05:
d7:88:71:a8:f2:cd:fd:74:60:ab:32:71:a0:16:f6:
52:2d:bb:c6:81:ac:c9:dd:9d
```

exponent2:

```
00:db:9c:da:7f:27:24:70:aa:33:ab:36:58:e4:ec:
```

```

31:c4:b3:e4:83:df:d9:07:43:3c:c2:7e:a7:7e:76:
74:cf:bf:6b:1c:d3:af:9c:a7:29:b7:ca:e9:50:71:
ba:24:50:ba:72:7e:64:68:dd:b8:a7:fe:9b:c9:43:
76:99:5f:f0:5d:87:dc:28:4d:7a:a1:5c:37:6b:ad:
2c:16:22:75:58:31:03:f2:3e:4f:1f:fc:3f:66:20:
e2:69:e4:55:16:33:01:c3:53:ec:21:21:94:b1:b0:
47:84:fa:3b:62:c6:55:ad:85:e2:91:62:44:26:cd:
06:57:6d:67:48:85:8c:88:dd
coefficient:
3f:85:ff:ac:1c:67:ce:50:5b:c9:c0:53:29:00:dd:
6a:d2:23:1f:f7:73:00:c6:76:6e:0d:44:67:2d:f1:
93:99:8d:31:e3:8b:2f:68:8c:c3:83:d4:be:e2:32:
14:50:ff:79:37:85:4b:22:9f:92:c3:32:9f:eb:c9:
61:86:c7:8b:88:68:b6:ad:e3:49:22:0b:b4:f8:23:
ae:83:33:b3:f9:f5:eb:aa:77:3d:f0:d0:f0:fe:55:
4f:a1:ec:64:a2:be:fb:05:0d:dc:92:52:de:db:34:
ad:00:51:52:e1:74:c2:5f:5b:10:cd:f1:05:74:6f:
9a:77:5a:e5:87:d5:4f:01

```

不使用私有金鑰時，請將它保管在安全位置。

2. 使用在上一個步驟中建立的私密金鑰產生 X.509 憑證。在此範例中，憑證會在 365 天後過期；在這段時間之後，就無法再信任此憑證。務必設定適當的過期期限。憑證只能在佈建程序期間有效。佈建完成後，您可以從 RIR 記錄中移除憑證。`tr -d "\n"` 命令從輸出中刪除換行符號字元 (換行符)。出現提示時，您需要提供 Common Name (通用名稱)，但其他欄位可以保留空白。

```
$ openssl req -new -x509 -key private-key.pem -days 365 | tr -d "\n" >
certificate.pem
```

此輸出結果類似以下內容：

```

Enter pass phrase for private-key.pem: xxxxxxxx
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.

Country Name (2 letter code) []:
State or Province Name (full name) []:
Locality Name (eg, city) []:

```

```
Organization Name (eg, company) []:
Organizational Unit Name (eg, section) []:
Common Name (eg, fully qualified host name) []:example.com
Email Address []:
```

### Note

AWS 佈建時不需要「一般名稱」。其可為任何內部或公有網域名稱。

您可以使用下列命令來檢查憑證：

```
$ cat certificate.pem
```

輸出應該是一個 PEM 編碼的長字串，沒有換行符，前面為 -----BEGIN CERTIFICATE-----，後面為 -----END CERTIFICATE-----。

## 2. 將 X.509 憑證上傳至 RIR 中的 RDAP 記錄

將您先前建立的憑證新增到 RIR 的 RDAP 記錄。請務必包含 -----BEGIN CERTIFICATE----- 和 -----END CERTIFICATE----- 字串之前和之後的編碼部分。所有這些內容都必須在單獨的長線上。更新 RDAP 的程序取決於您的 RIR：

- 若是 ARIN，請使用 [客戶經理入口網站](#)，在代表您位址範圍的「網路資訊」物件的「Public Comments」區段中新增憑證。請勿將其新增至您組織的評論區段。
- 若是 RIPE，請將憑證作為新的「descr」欄位新增至代表您位址範圍的「inetnum」或「inet6num」物件。您通常可以在 [RIPE 資料庫入口網站](#) 的「我的資源」區段找到這些內容。請勿將其新增至組織的註解區段，或上述物件的「備註」欄位。
- 對於 APNIC，請透過電子郵件將憑證傳送至 [helpdesk@apnic.net](mailto:helpdesk@apnic.net)，以手動將其新增至位址範圍的 "remarks" 欄位。使用 IP 地址的 APNIC 授權聯絡人發送電子郵件。

您可以在下列佈建階段完成後，從 RIR 記錄中移除憑證。

## 3. 在 RIR 中建立一個 ROA 物件

建立 ROA 物件來授權 Amazon ASN 16509 和 14618 以宣傳您的地址範圍，以及目前獲授權宣傳地址範圍的 ASN。對於 AWS GovCloud (US) Regions，請授權出貨預先通知 8987，而不是 16509 及



14618。您必須將長度上限設定為您要使用的 CIDR 大小。您可以使用的最特定 IPv4 字首為 /24。針對可公開公告的 CIDR，您可以使用的最特定 IPv6 位址範圍是 /48，而針對不可公開公告的 CIDR，則是 /56。

### Important

如果您要為 Amazon VPC IP Address Manager (IPAM) 建立 ROA 物件，則當您建立 ROA 時，您必須針對 IPv4 CIDR 將 IP 地址字首的長度上限設定為 /24。若要將 IPv6 CIDR 新增至可公告集區，則 IP 地址字首的長度上限必須為 /48。這樣可確保您有充分的靈活性，可以跨 AWS 區域劃分您的公有 IP 地址。IPAM 會強制執行您設定的長度上限。如需有關 IPAM 的 BYOIP 地址的詳細資訊，請參閱《Amazon VPC IPAM 使用者指南》中的[教學課程：IPAM 的 BYOIP 地址 CIDR](#)。

可能最多需要 24 小時 Amazon 才能使用 ROA。如需詳細資訊，請參閱 RIR：

- ARIN - [ROA Requests](#)
- RIPE - [Managing ROAs](#)
- APNIC - [路由管理](#)

將廣告從現場部署工作負載移轉到時 AWS，必須先為現有 ASN 建立 ROA，然後才能為 Amazon 的 ASN 建立 ROA。否則，可能會對現有路由和廣告造成影響。

### Important

若要讓 Amazon 公告並繼續公告您的 IP 地址範圍，您使用 Amazon ASN 的 ROA 必須符合上述準則。如果您的 ROA 無效或不符合上述準則，Amazon 保留停止廣告您的 IP 位址範圍的權利。

### Note

不可公開公告的 IPv6 地址空間不需要此步驟。

## 加入您的 BYOIP

BYOIP 的上線程序會根據您的需求執行下列工作。

## 任務

- [在 AWS 中佈建一個可公開公告的地址範圍](#)
- [佈建不可公開公告的 IPv6 地址範圍](#)
- [通告位址範圍 AWS](#)
- [解除佈建地址範圍](#)

## 在 AWS 中佈建一個可公開公告的地址範圍

佈建要搭配使用的地址範圍時 AWS，您確認自己控制了地址範圍，並授權 Amazon 宣傳該地址範圍。我們也會透過簽署的授權訊息驗證您確實控制該地址範圍。此訊息會使用您在使用 X.509 憑證更新 RDAP 記錄時所使用的自我簽署 X.509 key pair 簽署。AWS 需要提供給 RIR 的密碼編譯簽名授權訊息。RIR 會根據新增至 RDAP 的憑證來驗證簽章，並根據 ROA 檢查授權詳細資訊。

### 佈建地址範圍

#### 1. 編寫訊息

撰寫純文字授權訊息。訊息格式如下所示，其中的日期是訊息的過期日期：

```
1|aws|account|cidr|YYYYMMDD|SHA256|RSAPSS
```

用您自己的值取代帳戶號碼、地址範圍和到期日，建立類似下列內容的訊息：

```
text_message="1|aws|0123456789AB|198.51.100.0/24|20211231|SHA256|RSAPSS"
```

這不能與具有類似特征的 ROA 訊息混淆。

#### 2. 簽署訊息

使用您先前建立的私有金鑰來簽署純文字訊息。此命令傳回的簽章是一個長字串，您需要在下一個步驟中使用它。

#### Important

建議您複製並貼上此命令。除訊息內容外，請勿修改或取代任何值。

```
signed_message=$(echo -n $text_message | openssl dgst -sha256 -sigopt
rsa_padding_mode:pss -sigopt rsa_pss_saltlen:-1 -sign private-key.pem -keyform PEM
| openssl base64 | tr -- '+=/' '-_~' | tr -d "\n")
```

### 3. 佈建地址

使用 AWS CLI [佈建自動設定-cidr 指令](#)來佈建位址範圍。--cidr-authorization-context 選項會使用您先前建立的訊息和簽章字串。

#### Important

如果 BYOIP 範圍與您的組態不同，您AWS CLI 必須指定應佈建 BYOIP 範圍的 AWS 區域。 Default region name

```
aws ec2 provision-byoip-cidr --cidr address-range --cidr-authorization-context
Message="$text_message",Signature="$signed_message" --region us-east-1
```

佈建地址範圍是一種非同步操作；系統會立即傳回呼叫，但地址範圍要等到其狀態從 pending-provision 變更為 provisioned 時，才能開始使用。

### 4. 監控進度

雖然大部分佈建都會在兩小時內完成，但最多可能需要一週的時間才能完成可公開廣告範圍的佈建程序。使用 [describe-byoip-cidrs](#) 命令監控進度，如下例所示：

```
aws ec2 describe-byoip-cidrs --max-results 5 --region us-east-1
```

如果在佈建期間發生問題且狀態變成 failed-provision，您必須在問題解決後再次執行 provision-byoip-cidr 命令。

## 佈建不可公開公告的 IPv6 地址範圍

根據預設，可向網際網路公開公告所佈建的地址範圍。您可以佈建不可公開公告的 IPv6 地址範圍。對於不可公開公告的路由，佈建程序通常會在幾分鐘內完成。當您將來自非公有地址範圍的 IPv6 CIDR 區塊與 VPC 產生關聯時，只能透過支援 IPv6 的混合連線選項存取該 IPv6，例如 [AWS Direct Connect](#)、[AWS Site-to-Site VPN](#)，或 [Amazon VPC 傳輸閘道](#)

佈建非公有位址範圍不需要 ROA。

### Important

- 您只能指定在佈建期間可否公開公告地址範圍。您稍後無法變更可公告狀態。
- Amazon VPC 不支援[唯一本機地址](#) (ULA) CIDR。所有 VPC 都必須具有唯一的 IPv6 CIDR。兩個 VPC 不能具有相同的 IPv6 CIDR 範圍。

若要佈建不可公開公告的 IPv6 地址範圍，請使用下列 [provision-byoip-cidr](#) 命令。

```
aws ec2 provision-byoip-cidr --cidr address-range --cidr-authorization-context
Message="$text_message",Signature="$signed_message" --no-publicly-advertisable --
region us-east-1
```

## 通告位址範圍 AWS

佈建好地址範圍之後，即可將其公告。佈建好地址範圍之後，您必須公布。對於已經佈建好的地址範圍，您不能必須只公布一部分。

如果您佈建了不會公開公告的 IPv6 位址範圍，則不需要完成此步驟。

我們建議您在通過廣告之前，停止在其他位置廣告該地址範圍或範圍的任何部分 AWS。如果您持續從其他位置宣傳您的 IP 位址範圍或其任何部分，我們將無法可靠地支援或疑難排解問題。具體而言，我們無法保證到達位址範圍或部分範圍的流量會進入我們的網路。

為了盡量減少停機時間，您可以將 AWS 資源設定為在廣告位址集區之前使用位址集區中的位址，然後同時停止從目前位置發佈廣告，並開始通過 AWS 廣告。如需從您的地址集區配置彈性 IP 地址的詳細資訊，請參閱[配置彈性 IP 地址](#)。

### 限制

- 即使您每次都指定不同的地址範圍，最多仍只能每 10 秒執行一次 `advertise-byoip-cidr` 命令。
- 即使您每次都指定不同的地址範圍，最多仍只能每 10 秒執行一次 `withdraw-byoip-cidr` 命令。

若要公告地址範圍，請使用下列 [advertise-byoip-cidr](#) 命令。

```
aws ec2 advertise-byoip-cidr --cidr address-range --region us-east-1
```

若要停止公告地址範圍，請使用下列 [withdraw-byoip-cidr](#) 命令。

```
aws ec2 withdraw-byoip-cidr --cidr address-range --region us-east-1
```

## 解除佈建地址範圍

若要停止使用您的位址範圍 AWS，請先釋出任何彈性 IP 位址，並取消仍從位址集區配置的任何 IPv6 CIDR 區塊的關聯。然後停止公告位址範圍，最後取消佈建位址範圍。

您無法取消佈建一部分的位址範圍。如果您想要使用更特定的位址範圍 AWS，請取消佈建整個位址範圍，並佈建更特定的位址範圍。

(IPv4) 若要釋出每個彈性 IP 地址，請使用下列 [release-address](#) 命令。

```
aws ec2 release-address --allocation-id eipalloc-12345678abcabcabc --region us-east-1
```

(IPv6) 若要解除 IPv6 CIDR 區塊的關聯，請使用下列 [disassociate-vpc-cidr-block](#) 命令。

```
aws ec2 disassociate-vpc-cidr-block --association-id vpc-cidr-assoc-12345abcd1234abc1
--region us-east-1
```

若要停止公告地址範圍，請使用下列 [withdraw-byoip-cidr](#) 命令。

```
aws ec2 withdraw-byoip-cidr --cidr address-range --region us-east-1
```

若要解除佈建地址範圍，請使用下列 [deprovision-byoip-cidr](#) 命令。

```
aws ec2 deprovision-byoip-cidr --cidr address-range --region us-east-1
```

取消佈建位址範圍最多可能需要一天的時間。

## 使用您的位址範圍

您可以檢視和使用已在帳戶中佈建的 IPv4 和 IPv6 地址範圍。

### IPv4 位址範圍

您可以從 IPv4 地址集區建立彈性 IP 地址，並將其與 EC2 執行個體、NAT 閘道和網路負載平衡器等 AWS 資源搭配使用。

若要檢視您在帳戶中佈建之 IPv4 位址集區的相關資訊，請使用下列 [describe-public-ipv4-pools](#) 命令。

```
aws ec2 describe-public-ipv4-pools --region us-east-1
```

若要從您的 IPv4 位址集區建立彈性 IP 地址，請使用 [allocate-address](#) 命令。您可以使用 `--public-ipv4-pool` 選項來指定 `describe-byoip-cidrs` 傳回的位址集區 ID。或者，您可以使用 `--address` 選項來指定您佈建的位址範圍。

## IPv6 位址範圍

若要檢視您在帳戶中佈建之 IPv6 位址集區的相關資訊，請使用下列 [describe-ipv6-pools](#) 命令。

```
aws ec2 describe-ipv6-pools --region us-east-1
```

若要建立 VPC 並指定來自 IPv6 位址集區的 IPv6 CIDR，請使用下列 [create-vpc](#) 命令。若要讓 Amazon 從您的 IPv6 位址集區中選擇 IPv6 CIDR，請省略 `--ipv6-cidr-block` 選項。

```
aws ec2 create-vpc --cidr-block 10.0.0.0/16 --ipv6-cidr-block ipv6-cidr --ipv6-pool pool-id --region us-east-1
```

若要將來自您的 IPv6 位址集區的 IPv6 CIDR 區塊與 VPC 建立關聯，請使用下列 [associate-vpc-cidr-block](#) 命令。若要讓 Amazon 從您的 IPv6 位址集區中選擇 IPv6 CIDR，請省略 `--ipv6-cidr-block` 選項。

```
aws ec2 associate-vpc-cidr-block --vpc-id vpc-123456789abc123ab --ipv6-cidr-block ipv6-cidr --ipv6-pool pool-id --region us-east-1
```

若要檢視您的 VPC 和相關聯的 IPv6 位址集區資訊，請使用 [describe-vpcs](#) 命令。若要檢視來自特定 IPv6 位址集區的關聯 IPv6 CIDR 區塊的相關資訊，請使用下列取得 [get-associated-ipv6-pool-cidrs](#) 命令。

```
aws ec2 get-associated-ipv6-pool-cidrs --pool-id pool-id --region us-east-1
```

如果您取消 IPv6 CIDR 區塊與 VPC 的關聯，即會將它釋出回您的 IPv6 位址集區中。

## 驗證您的 BYOIP

### 1. 驗證自我簽署的 x.509 金鑰對

透過 `whois` 命令，驗證憑證是否已上傳且有效。

若為 ARIN，請使用 `whois -h whois.arin.net r + 2001:0DB8:6172::/48` 來查詢您地址範圍的 RDAP 記錄。檢查命令輸出中 NetRange (網路範圍) 的 Public Comments 部分。該憑證應新增至地址範圍的 Public Comments 部分中。

您可使用下列命令來檢查包含憑證的 Public Comments：

```
whois -h whois.arin.net r + 2001:0DB8:6172::/48 | grep Comments | grep BEGIN
```

此會傳回含有金鑰內容的輸出，應類似於下列內容：

```
Public Comments:
-----BEGIN CERTIFICATE-----
MIID1zCCAr+gAwIBAgIUBkRPNLrPqbRAFP8RDAHSP+I1TowDQYJKoZIhvcNAQE
LBQAwezELMAkGA1UEBhMCTloETAPBgNVBAGMCEF1Y2tsYW5kMREwDwYDVQQHDA
hBdWNrbGFuZDEcMBoGA1UECgwTQW1hem9uIFdlYiBTZXJ2aWNlcjETMBEGA1UEC
wwKQ11PSVAgRGVtbzETMBEGA1UEAwwKQ11PSVAgRGVtbzAeFw0yMTEyMDcyMDI0
NTRaFw0yMjE2MDcyMDI0NTRaMHsxCzAJBgNVBAYTAk5aMREwDwYDVQQIDAhBdWN
rbGFuZDERMA8GA1UEBwwIQXVja2xhbmQxHDAaBgNVBAoME0FtYXpvaWwvYXNjaWZl
VydmljZXNlZXRARBgNVBAsMCKJZT0lQIERlbW8xEzARBgNVBAMMCKJZT0lQIERlb
W8wggeiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQCfmacvDp0wZ0ceiXXc
R/q27mHI/U5HKt7SST4X2eAqur9WxkfNanAEskgAseyFypwEEQr4CJijI/5hp9
prh+jsWHWwkFRoBRR9FBtwcU/45XDXLga7D3stsI5QeshVRw0aXUdprAnndaTug
mDPkD0vr1475JWDSIm+PUxGWLy+60aBqiaZq35wU/x+wXlAqBXg4MZK2KoUu27k
Yt2zhmy0S7Ky+oRfRj9QbAiSu/RwhQbh5Mkp1ZnVlc7NqnhdeIW48QaYjhM1UEf
xdaqYUinz8KpjfADZ4Hvqj9jWZ/eXo/9b2rG1HWkJsbhr0VEUyAGu1bwkgcdww
3A7Nj0xQbAgMBAAGjUzBRMB0GA1UdDgQWBBSstFyujN6SYBr2glHpGt0XGF7GbGT
AfBgNVHSMEGDAWGBStFyujN6SYBr2glHpGt0XGF7GbGTAPBgNVHRMBAf8EBTADA
QH/MA0GCSqGSIb3DQEBCwUAA4IBAQBx6nn6YLhz5211fyVfxY0t6o3410bQAeAF
08ud+ICtmQ4IO4A4B7zV3zIVYr0clr00aFyLxngwMYN0XY5tVhDQqk4/gmDNEKS
Zy2QkX4Eg0YUWVz0yt6fPzj0vJLcsqc1hcF9wySL507XQz76Uk5cFypB0zbnk35
UkWrzA9KK97cXckfIESgK/k1N4ecwxwG6VQ8mBGqVpPpey+dXpzzzv1iBKN/VY4
ydjgH/LBfdTsVarmmy2vtWBxwrqkFvpdhSGCvRD1/qd0/GIDJi77dmZWkh/ic90
MNk1f38gs1jrCj8lThoar17Uo9y/Q5qJlIsoNPyQrJRzqFU9F3FBjiPJF
-----END CERTIFICATE-----
```

若為 RIPE，請使用 `whois -r -h whois.ripe.net 2001:0DB8:7269::/48` 來查詢您地址範圍的 RDAP 記錄。檢查命令輸出中 inetnum 物件 (網路範圍) 的 descr 部分。該憑證應新增為地址範圍的新 descr 欄位。

您可使用下列命令來檢查包含憑證的 descr：

```
whois -r -h whois.ripe.net 2001:0DB8:7269::/48 | grep descr | grep BEGIN
```

此會傳回含有金鑰內容的輸出，應類似於下列內容：

```
descr:
-----BEGIN CERTIFICATE-----MIID1zCCA+r+gAwIBAgIUBkRPNSLrPqbRAFP8
RDAHSP+I1TowDQYJKoZIhvcNAQELBQAwezELMAkGA1UEBhMCT1oxETAPBgNVBAg
MCEF1Y2tsYW5kMREwDwYDVQQHDAhBdWNrbGFuZDEcMBoGA1UECgwTQW1hem9uIF
d1YiBTZXJ2aWNlczETMBEGA1UECwwKQ11PSVAgRGVtbzETMBEGA1UEAwwKQ11PS
VAgRGVtbzAeFw0yMTEyMDcyMDI0NTRaFw0yMjEyMDcyMDI0NTRaMHsxCzAJBgNV
BAYTAk5aMREwDwYDVQQIDAhhBdWNrbGFuZDERMA8GA1UEBwwIQXVja2xhbmQxHDA
aBgNVBAoME0FtYXpvcjEwYjEwYjEwYjEwYjEwYjEwYjEwYjEwYjEwYjEwYjEwYjEw
8xEzARBgNVBAMMCKJZT0lQIERlbW8wggeiMA0GCSqGSIb3DQEBAQUAA4IBDwAwg
gEKAoIBAQCfmacvDp0wZ0ceiXXcR/q27mHI/U5HKt7SST4X2eAqufR9wXkfNanA
EskgAseyFypwEEQr4CJijI/5hp9prh+jsWHWwkFRoBRR9FBtwcU/45XDxLga7D3
stsI5QeshVRw0aXUdprAnndaTugmDPkD0vr1475JWDSIm+PUxGWLy+60aBqiaZq
35wU/x+wX1AqBXg4MZK2KoUu27kYt2zhmy0S7Ky+oRfRJ9QbAiSu/RwhQbh5Mkp
1ZnVic7NqnhdEiW48QaYjhM1UEfxdaqYUinz8KpjfADZ4Hvqj9jWZ/eXo/9b2r
G1HWkJsbnhr0VEUYAGu1bwkgcdww3A7Nj0xQbAgMBAAGjUzBRMB0GA1UdDgQWBBS
tFyujN6SYBr2g1HpGt0XGF7GbGTAFBgNVHSMEGDAWgBStFyujN6SYBr2g1HpGt0
XGF7GbGTAPBgNVHRMBAf8EBTADAQH/MA0GCSqGSIb3DQEBCwJAA4IBAQB6nn6Y
Lhz5211fyVfxY0t6o3410bQAeAF08ud+ICtmQ4I04A4B7zV3zIVYr0clr00aFyL
xngwMYN0XY5tVhDQqk4/gmDNEKSZy2QkX4Eg0YUWVz0yt6fPzj0vJLcsqc1hcF9
wySL507XQz76Uk5cFypB0zbnk35UkWzA9KK97cXckfIESgK/k1N4ecwxwG6VQ8
mBGqVpPpey+dXpzzzv1iBKN/VY4ydjgH/LBfdTsVarmmy2vtWBxwrqkFvpdhSGC
vRD1/qd0/GIDJi77dmZWkh/ic90MNk1f38gs1jrCj8lThoar17Uo9y/Q5qJISon
PyQrJRzqFU9F3FBjiPJF
-----END CERTIFICATE-----
```

若為 APNIC，請使用 `whois -h whois.apnic.net 2001:0DB8:6170::/48` 來查詢您 BYOIP 地址範圍的 RDAP 記錄。檢查命令輸出中 `inetnum` 物件 (網路範圍) 的 `remarks` 部分。該憑證應新增為地址範圍的新 `remarks` 欄位。

您可使用下列命令來檢查包含憑證的 `remarks`：

```
whois -h whois.apnic.net 2001:0DB8:6170::/48 | grep remarks | grep BEGIN
```

此會傳回含有金鑰內容的輸出，應類似於下列內容：

```
remarks:
```



```

-----BEGIN CERTIFICATE-----
MIID1zCCAr+gAwIBAgIUBkRPNLrPqbRAFP8RDAHSP+I1TowDQYJKoZIhvcNAQE
LBQAwezELMAkGA1UEBhMCT1oxETAPBgNVBAGMCEF1Y2tsYW5kMREwDwYDVQQHDA
hBdWNrbGFuZDEcMBoGA1UECgwTQW1hem9uIFdlYiBTZXJ2aWN1czETMBEGA1UEC
wwKQ11PSVAgRGVtbzETMBEGA1UEAwwKQ11PSVAgRGVtbzAeFw0yMTEyMDcyMDI0
NTRaFw0yMjE5MDcyMDI0NTRaMHsxCzAJBgNVBAYTAk5aMREwDwYDVQQIDAhBdWN
rbGFuZDERMA8GA1UEBwwIQXVja2xhbmQxHDAaBgNVBAoME0FtYXpviBXZWIgU2
VydmIjZXMxEzARBgNVBAsMCkZJT01QIERlbW8xEzARBgNVBAMMckZJT01QIERlb
W8wggeEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQCfmacvDp0wZ0ceiXXc
R/q27mHI/U5HKt7SST4X2eAqurF9wXkfNanAEskgAseyFypwEEQr4CJijI/5hp9
prh+jsWHWwkFRoBRR9FBtwcU/45XDxLga7D3stsI5QeshVRw0aXUdprAnndaTug
mDPkD0vr1475JWDSIm+PUxGwLy+60aBqiaZq35wU/x+wXlAqBXg4MZK2KoUu27k
Yt2zhmy0S7Ky+oRfRj9QbAiSu/RwhQbh5Mkp1ZnVIc7NqnhdeIW48QaYjhM1UEf
xdaqYUinzz8KpjfADZ4Hvqj9jWZ/eXo/9b2rG1HwkJsbhr0VEUyAGu1bwkgcdww
3A7Nj0xQbAgMBAAGjUzBRMB0GA1UdDgQWBBSstFyujN6SYBr2glHpGt0XGF7GbGT
AfBgNVHSMEGDAWgBStFyujN6SYBr2glHpGt0XGF7GbGTAPBgNVHRMBAf8EBTADA
QH/MA0GCSqGSIb3DQEBCwUAA4IBAQBx6nn6YLhz5211fyVfxY0t6o3410bQAeAF
08ud+ICtmQ4I04A4B7zV3zIVYr0clr00aFyLxngwMYN0XY5tVhDQqk4/gmDNEKS
Zy2QkX4Eg0YUWVz0yt6fPzj0vJLcsqc1hcF9wySL507XQz76Uk5cFypB0zbnk35
UkWrzA9KK97cXckfIESgK/k1N4ecwxwG6VQ8mBGqVpPpey+dXpzzv1iBKN/VY4
ydjgH/LBfdTsVarmmy2vtWBxwrqkFvphSGCvRD1/qd0/GIDJi77dmZWkh/ic90
MNk1f38gs1jrCj8lThoar17Uo9y/Q5qJiSoNPYqrJRzqFU9F3FBjiPJF
-----END CERTIFICATE-----

```

## 2. 驗證 ROA 物件的建立

使用 RIPEstat Data API，驗證是否成功建立 ROA 物件。請務必依據 Amazon ASN 16509 和 14618，及目前授權公布地址範圍的 ASN 來測試您的地址範圍。

您可使用以下命令，檢查具有您地址範圍不同 Amazon ASN 的 ROA 物件：

```
curl --location --request GET "https://stat.ripe.net/data/rpki-validation/data.json?
resource=ASN&prefix=CIDR"
```

於此範例輸出中，Amazon ASN 16509 的回應結果為 "status": "valid"。這表示地址範圍的 ROA 物件已成功建立：

```
{
 "messages": [],
 "see_also": [],
 "version": "0.3",
 "data_call_name": "rpki-validation",
```

```
"data_call_status": "supported",
"cached": false,
"data": {
 "validating_roas": [
 {
 "origin": "16509",
 "prefix": "2001:0DB8::/32",
 "max_length": 48,
 "validity": "valid"
 },
 {
 "origin": "14618",
 "prefix": "2001:0DB8::/32",
 "max_length": 48,
 "validity": "invalid_asn"
 },
 {
 "origin": "64496",
 "prefix": "2001:0DB8::/32",
 "max_length": 48,
 "validity": "invalid_asn"
 }
],
 "status": "valid",
 "validator": "routinator",
 "resource": "16509",
 "prefix": "2001:0DB8::/32"
},
"query_id": "20230224152430-81e6384e-21ba-4a86-852a-31850787105f",
"process_time": 58,
"server_id": "app116",
"build_version": "live.2023.2.1.142",
"status": "ok",
"status_code": 200,
"time": "2023-02-24T15:24:30.773654"
}
```

“unknown”的狀態表示尚未建立地址範圍的 ROA 物件。“invalid\_asn”的狀態表示未成功建立地址範圍的 ROA 物件。

## 區域可用性

BYOIP 功能目前適用於所有商業 [AWS 區域](#) (中國地區除外)。

## Local Zone 可用性

「[本地區域](#)」是您使用者所在地理位置附近的「AWS 區域」的延伸。Local Zones 會分組為「網路邊界群組」。在中 AWS，網路邊界群組是可用區域 (AZ)、本機區域或 Wavelength 區域的集合，這些區域會從中 AWS 通告公用 IP 位址。Local Zones 的網路邊界群組可能與 AWS 區域中的 AZ 不同，以確保 AWS 網路與存取這些區域中資源的客戶之間的最小延遲或實際距離。

您可以使用 `--network-border-group` 選項，在下列 Local Zone 網路邊界群組中佈建 BYOIPv4 地址範圍，並公告這些地址範圍：

- us-east-1-dfw-2
- us-west-2-lax-1
- us-west-2-phx-2

如果您已啟用 Local Zones (請參閱[啟用 Local Zone](#))，您可以在佈建和公告 BYOIPv4 CIDR 時為 Local Zones 選擇網路邊界群組。請謹慎選擇網路邊界群組，因為 EIP 及其關聯的 AWS 資源必須位於相同的網路邊界群組中。

### Note

您目前無法在 Local Zones 中佈建或公告 BYOIPv6 地址範圍。

## 進一步了解

如需詳細資訊，請參閱 [AWS 線上技術講座深入探討自攜 IP](#)。

## 彈性 IP 地址

彈性 IP 地址是針對動態雲端運算設計的靜態 IPv4 地址。彈性 IP 位址會分配給您的 AWS 帳戶，在您釋放之前是您的。透過彈性 IP 地址，您可以快速地將地址重新映射至帳戶中的另一個執行個體，以遮罩執行個體或軟體的故障。或者，您也可以在您網域的 DNS 紀錄中指定彈性 IP 地址，以便您的網域指向您的執行個體。如需詳細資訊，請參閱您網域註冊商的說明文件。

彈性 IP 地址是可從網際網路存取的公有 IPv4 地址。如果您的執行個體沒有公有 IPv4 地址，則可將彈性 IP 地址與執行個體建立關聯，進而啟用與網際網路通訊的功能。例如，這可讓您從本機電腦連線到執行個體。

## 目錄

- [彈性 IP 地址定價](#)
- [彈性 IP 地址基本概念](#)
- [使用彈性 IP 地址](#)
- [彈性 IP 地址配額](#)

## 彈性 IP 地址定價

AWS 所有公用 IPv4 位址的費用，包括與執行中執行個體和彈性 IP 位址相關聯的公用 IPv4 位址。如需詳細資訊，請參閱 [Amazon VPC 定價頁面](#) 中的公有 IPv4 地址。

## 彈性 IP 地址基本概念

以下是彈性 IP 地址的基本特性：

- 彈性 IP 地址是靜態的地址；不會隨著時間而變更。
- 彈性 IP 地址僅適用於特定區域，無法移至不同區域。
- 彈性 IP 地址來自亞馬遜的 IPv4 地址池，或者來自您 AWS 帶到帳戶的自定義 IPv4 地址池。
- 若要使用彈性 IP 地址，您可以先將一個地址配置給帳戶，再將其與您的執行個體或網路介面建立關聯。
- 當您將彈性 IP 地址與執行個體建立關聯時，它也會與執行個體的主要網路介面相關聯。當您將彈性 IP 地址與連接到執行個體的網路介面建立關聯時，它也會與執行個體相關聯。
- 當您將彈性 IP 地址與執行個體或其主要網路界面建立關聯時，如果該執行個體已有與其相關聯的公用 IPv4 地址，則該公用 IPv4 地址會釋放回 Amazon 的公有 IPv4 地址集區，而彈性 IP 地址會改為與執行個體相關聯。您無法重複使用先前與執行個體相關聯的公用 IPv4 位址，也無法將該公用 IPv4 位址轉換為彈性 IP 位址。如需詳細資訊，請參閱 [公有 IPv4 地址](#)。
- 您可以取消彈性 IP 地址與資源的關聯，並將其與不同的資源重新關聯。若要避免非預期的行為，請確保在變更前，已關閉與現有關聯中命名之資源的所有啟用連線。將彈性 IP 地址關聯至不同資源之後，您可以重新開啟與新關聯資源的連線。

- 在您明確將其釋放之前，取消關聯的彈性 IP 地址會維持配置給您的帳戶。無論帳戶中的所有彈性 IP 地址是否與執行個體相關聯或取消關聯，都需要向您收取費用。如需詳細資訊，請參閱 [Amazon VPC 定價頁面](#) 中的公有 IPv4 地址。
- 當您將彈性 IP 地址與先前具有公有 IPv4 地址的執行個體建立關聯時，執行個體的公有 DNS 主機名稱會變更以符合彈性 IP 地址。
- 我們會將公有 DNS 主機名稱解析為執行個體網路外部之執行個體的公有 IPv4 地址或彈性 IP 地址，以及解析為執行個體網路內執行個體的私有 IPv4 地址。
- 當您從帶入 AWS 帳戶的 IP 位址集區配置彈性 IP 位址時，該位址不會計入您的彈性 IP 位址限制。如需詳細資訊，請參閱 [彈性 IP 地址配額](#)。
- 當您配置彈性 IP 地址時，可以將彈性 IP 地址與網路邊界群組建立關聯。這是我們公告 CIDR 區塊的位置。設定網路邊界群組會將 CIDR 區塊限制在此群組。如果您未指定網路邊界群組，我們會設定包含區域中所有可用區域的邊界群組 (例如，us-west-2)。
- 彈性 IP 地址只能用於特定網路邊界群組。

## 使用彈性 IP 地址

下列各節說明如何使用彈性 IP 地址。

### 任務

- [配置彈性 IP 地址](#)
- [描述彈性 IP 地址](#)
- [為彈性 IP 地址套用標籤](#)
- [建立彈性 IP 地址與執行個體或網路介面的關聯](#)
- [取消彈性 IP 地址的關聯](#)
- [轉移彈性 IP 地址](#)
- [釋出彈性 IP 地址](#)
- [復原彈性 IP 地址](#)
- [針對電子郵件應用程式使用反向 DNS](#)

## 配置彈性 IP 地址

您可以從 Amazon 的公有 IPv4 地址集區，或從您帶到 AWS 帳戶的自訂 IP 地址集區分配彈性 IP 地址。如需將自己的 IP 位址範圍帶入 AWS 帳戶的詳細資訊，請參閱 [在 Amazon EC2 中使用自有 IP 地址 \(BYOIP\)](#)。

您可以使用下列其中一種方法來配置彈性 IP 地址。

## Console

### 配置彈性 IP 地址

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，於 Network & Security (網路與安全) 中，選擇 Elastic IPs (彈性 IP)。
3. 選擇 Allocate Elastic IP address (配置彈性 IP 地址)。
4. (選用) 配置彈性 IP 地址 (EIP) 時，您可以選擇要配置 EIP 的網路邊界群組。網路邊界群組是可用區域 (AZ)、Local Zones 或 Wavelength 區域的集合，用於通 AWS 公用 IP 位址。Local Zones 和 Wavelength 區域可能具有與區域中的 AZ 不同的網路邊界組，以確保 AWS 網路與訪問這些區域中資源的客戶之間的最小延遲或物理距離。

#### Important

您必須在與 EIP 關聯的 AWS 資源相同的網路邊界群組中配置 EIP。一個網路邊界群組中的 EIP 只能在該網路邊界群組中的區域中進行公告，而不能在其他網路邊界群組所代表的任何其他區域中進行公告。

如果您已啟用 Local Zones 或 Wavelength Zones (如需詳細資訊，請參閱[啟用 Local Zone](#) 或[啟用 Wavelength Zones](#))，您可以為 AZ、Local Zones 或 Wavelength Zones 選擇網路邊界群組。請謹慎選擇網路邊界群組，因為 EIP 及其關聯的 AWS 資源必須位於相同的網路邊界群組中。您可以使用 EC2 主控台檢視可用區域、本機區域或 Wavelength 區域所在的網路邊界群組。一般而言，區域中的所有可用區域屬於相同的網路邊界群組，而 Local Zones 或 Wavelength Zones 則屬於其各自的網路邊界群組。

如果您沒有啟用 Local Zones 或 Wavelength Zones，則當您配置 EIP 時，代表該區域所有 AZ 的網路邊界群組 (例如 us-west-2) 會為您預先定義，且您無法予以變更。這表示您配置給此網路邊界群組的 EIP 會在您所在區域的所有 AZ 中進行公告。

5. 對於 公有 IPv4 地址集區，請選擇下列其中一項：
  - Amazon 的 IPv4 地址集區—若您要從 Amazon IPv4 地址集區配置一個 IPv4 地址。
  - 您帶到 AWS 帳戶的公用 IPv4 位址 — 如果您想要從您帶入帳戶的 IP 位址集區配置 IPv4 位址。AWS 如果您沒有任何 IP 地址集區，則會停用此選項。

- 客戶擁有的 IPv4 地址集區 - 若您要從內部部署網路建立的集區配置 IPv4 地址，以搭配使用 AWS Outpost。如果您沒有 AWS 前哨，則會停用此選項。

#### 6. (選用) 新增或移除標籤。

[新增標籤] 選擇新增標籤，並執行下列動作：

- 對於 Key (金鑰)，輸入金鑰名稱。
- 對於 Value (值)，進入金鑰值。

[移除標籤] 選擇標籤「金鑰」和「值」右側移除。

#### 7. 選擇 Allocate (配置)。

## AWS CLI

### 配置彈性 IP 地址

使用 [allocate-address](#) AWS CLI 命令。

## PowerShell

### 配置彈性 IP 地址

使用指 [New-EC2Address](#) AWS Tools for Windows PowerShell 令。

## 描述彈性 IP 地址

您可以使用下列其中一種方法來描述彈性 IP 地址。

## Console

### 描述彈性 IP 地址

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Elastic IPs (彈性 IP)。
3. 選取要檢視的彈性 IP 地址並依序選取 Actions (動作)、View details (檢視詳細資料)。

## AWS CLI

### 描述彈性 IP 地址

使用 [描述位址指令](#) AWS CLI。

## PowerShell

描述彈性 IP 地址

使用指 [Get-EC2Address](#) AWS Tools for Windows PowerShell 令。

## 為彈性 IP 地址套用標籤

您可以將自訂標籤指派給彈性 IP 地址，以不同的方式將其分類，例如：依用途、擁有者或環境。這可協助您根據所指派的自訂標籤，快速找到特定彈性 IP 地址。

不支援使用彈性 IP 地址標籤來追蹤成本分配。

您可以使用下列其中一種方法標記彈性 IP 地址。

## Console

為彈性 IP 地址套用標籤

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Elastic IPs (彈性 IP)。
3. 選取要標記的彈性 IP 地址並依序選取 Actions (動作)、View details (檢視詳細資料)。
4. 在 Tags (標籤) 區段中，選擇 Manage tags (管理標籤)。
5. 指定標籤金鑰和值對。
6. (選用) 選擇 (Add tag) 新增標籤以新增其他標籤。
7. 選擇 Save (儲存)。

## AWS CLI

為彈性 IP 地址套用標籤

使用「[建立標籤](#)」指令 AWS CLI。

```
aws ec2 create-tags --resources eipalloc-12345678 --tags Key=Owner,Value=TeamA
```

## PowerShell

為彈性 IP 地址套用標籤



使用指 [New-EC2Tag](#) AWS Tools for Windows PowerShell 令。

New-EC2Tag 命令需要 Tag 參數，這會指定要用於彈性 IP 地址標籤的索引鍵和值對。下列命令會建立 Tag 參數。

```
PS C:\> $tag = New-Object Amazon.EC2.Model.Tag
PS C:\> $tag.Key = "Owner"
PS C:\> $tag.Value = "TeamA"
```

```
PS C:\> New-EC2Tag -Resource eipalloc-12345678 -Tag $tag
```

## 建立彈性 IP 地址與執行個體或網路介面的關聯

如果您想要將彈性 IP 地址與執行個體建立關聯來啟用與網際網路通訊的功能，您也必須確定執行個體位於公有子網中。如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的 [網際網路閘道](#)。

您可以使用下列其中一種方法將彈性 IP 地址與執行個體或網路介面建立關聯。

### Console

#### 建立彈性 IP 地址與執行個體的關聯

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Elastic IPs (彈性 IP)。
3. 選取要關聯的彈性 IP 地址並依序選取 Actions (動作)、Associate Elastic IP address (關聯彈性 IP 地址)。
4. 對於 Resource type (資源類型)，選擇 Instance (執行個體)。
5. 例如，選擇要與彈性 IP 地址建立關聯的執行個體。您也可以輸入文字來搜尋特定執行個體。
6. (選用) 對於 Private IP address (私人 IP 地址)，指定要與彈性 IP 地址建立關聯的私人 IP 地址。
7. 選擇 Associate (關聯)。

#### 建立彈性 IP 地址與網路介面的關聯。

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Elastic IPs (彈性 IP)。

3. 選取要關聯的彈性 IP 地址並依序選取 Actions (動作)、Associate Elastic IP address (關聯彈性 IP 地址)。
4. 對於資源類型，選擇 Network interface (網路介面)。
5. 在 Network interface (網路介面) 中，選擇要與彈性 IP 地址建立關聯的網路介面。您也可以輸入文字來搜尋特定網路介面。
6. (選用) 對於 Private IP address (私人 IP 地址)，指定要與彈性 IP 地址建立關聯的私人 IP 地址。
7. 選擇 Associate (關聯)。

## AWS CLI

### 建立彈性 IP 地址的關聯

使用[關聯位址](#) AWS CLI 指令。

## PowerShell

### 建立彈性 IP 地址的關聯

使用指[Register-EC2Address](#) AWS Tools for Windows PowerShell 令。

## 取消彈性 IP 地址的關聯

您可以隨時解除彈性 IP 地址與執行個體或網路介面的關聯。解除與彈性 IP 地址的關聯後，您可以將它與其他資源重新建立關聯。

您可以使用下列其中一種方法解除彈性 IP 地址的關聯。

## Console

### 解除關聯並重新關聯彈性 IP 地址

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Elastic IPs (彈性 IP)。
3. 選取要解除彈性 IP 地址的關聯，請依序選取 Actions (動作)、Disassociate Elastic IP address (解除彈性 IP 地址的關聯)。
4. 選擇 Disassociate (取消關聯)。

## AWS CLI

取消與彈性 IP 地址的關聯

使用 [取消關聯](#) AWS CLI 地址命令。

## PowerShell

取消與彈性 IP 地址的關聯

使用 [Unregister-EC2Address](#) 命令。AWS Tools for Windows PowerShell

## 轉移彈性 IP 地址

本節旨在說明如何將彈性 IP 地址從一個 AWS 帳戶 轉移到另一個。轉移彈性 IP 地址適用於下列情況：

- 組織重組 — 使用彈性 IP 位址傳輸，快速將工作負載從一個移 AWS 帳戶 至另一個。您無須等待新的彈性 IP 地址列在安全群組和 NACL 的允許清單中。
- 集中式安全性管理 — 使用集中式 AWS 安全性帳戶來追蹤和傳輸已針對安全性合規審查的彈性 IP 位址。
- 災難復原 – 發生緊急事故時，使用彈性 IP 地址轉移可快速重新對應 IP，以處理公開網站的工作負載。

轉移彈性 IP 地址無需付費。

### 任務

- [啟用彈性 IP 地址轉移](#)
- [停用彈性 IP 地址轉移](#)
- [接受轉移後的彈性 IP 地址](#)

### 啟用彈性 IP 地址轉移

本節旨在說明如何接受轉移後的彈性 IP 地址。請注意下列與啟用彈性 IP 地址轉移的相關限制：

- 您可以將彈性 IP 地址從任何 AWS 帳戶（源帳戶）轉移到同一 AWS 區域中的任何其他 AWS 帳戶（轉移帳戶）。

- 當您轉移彈性 IP 地址時，AWS 帳戶之間會發生兩步驟交握。來源帳戶開始轉移時，轉移帳戶有七天時間可以接受彈性 IP 地址轉移。在這七天內，來源帳戶可以檢視擱置中的傳輸 (例如，在 AWS 主控台中或使用[描述-位址-傳輸](#) AWS CLI 令)。七天後轉移將到期，屆時彈性 IP 地址的擁有權會回到來源帳戶。
- 在接受傳輸後的三天內，來源帳戶可以看到接受的傳輸 (例如在 AWS 主控台中或使用[描述位址-傳輸](#) AWS CLI 命令)。
- AWS 不會通知轉移帳戶有關待處理的彈性 IP 地址轉移請求。來源帳戶的擁有者必須通知轉移帳戶的擁有者，有必須接受的彈性 IP 地址轉移的請求。
- 轉移完成時，會重設與要轉移之彈性 IP 地址相關的所有標籤。
- 您無法轉移從您 AWS 帳戶 帶入的公用 IPv4 位址集區配置的彈性 IP 位址，通常稱為自攜 IP (BYOIP) 位址集區。
- 如果您嘗試轉移的彈性 IP 地址，具有與其相關聯的反向 DNS 記錄，您可以開始轉移程序，但轉移帳戶無法接受轉移，除非移除相關聯的 DNS 記錄。
- 如果您已啟用並設定 AWS Outposts，則可能已從客戶擁有的 IP 位址集區 (CoIP) 配置彈性 IP 位址。您無法轉移從 CoIP 配置的彈性 IP 地址。但是，您可以使 AWS RAM 用與另一個帳戶共享 CoIP。如需詳細資訊，請參閱 AWS Outposts 使用者指南中的[客戶擁有的 IP 地址](#)。
- 您可使用 Amazon VPC IPAM 來追蹤彈性 IP 地址在 AWS Organizations 組織中的帳戶之間轉移的情況。如需詳細資訊，請參閱 [View IP address history](#) (檢視 IP 地址歷程記錄)。如果將彈性 IP 地址轉移至組織外部的 AWS 帳戶，就會失去彈性 IP 地址的 IPAM 稽核歷史記錄。

這些步驟必須由來源帳戶完成。

## Console

### 啟用彈性 IP 地址轉移

1. 確定您使用的是來源 AWS 帳戶。
2. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
3. 在導覽窗格中，選擇 Elastic IPs (彈性 IP)。
4. 選取要啟用轉移的一或多個彈性 IP 地址，然後選擇 Actions (動作)、Enable transfer (啟用轉移)。
5. 如果要轉移多個彈性 IP 地址，您會看到 Transfer type (轉移類型) 選項。請選擇下列其中一個選項：
  - 如果要將彈性 IP 地址傳輸到單個帳戶，請選擇單個 AWS 帳戶。

- 如果要將彈性 IP 地址傳輸到多個帳戶，請選擇「多個 AWS 帳戶」。
6. 在 Transfer account ID (轉移帳戶 ID) 底下，輸入您要將彈性 IP 地址轉移至哪個 AWS 帳戶 ID。
  7. 在文字方塊中輸入 **enable**，以便確認轉移。
  8. 選擇提交。
  9. 若要接受轉移，請參閱 [接受轉移後的彈性 IP 地址](#)。若要停用轉移，請參閱 [停用彈性 IP 地址轉移](#)。

## AWS CLI

若要啟用彈性 IP 地址轉移

使用 [enable-address-transfer](#) 命令。

## PowerShell

若要啟用彈性 IP 地址轉移

使用 [Enable-EC2AddressTransfer](#) 命令。

## 停用彈性 IP 地址轉移

本節旨在說明如何在啟用轉移後停用彈性 IP 轉移。

這些步驟必須由啟用轉移的來源帳戶完成。

## Console

停用彈性 IP 地址轉移

1. 確定您使用的是來源 AWS 帳戶。
2. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
3. 在導覽窗格中，選擇 Elastic IPs (彈性 IP)。
4. 在彈性 IP 的資源清單中，確保您已啟用顯示 Transfer status (轉移狀態) 欄位的屬性。
5. 選取 Transfer status (轉移狀態) 為 Pending (待定) 的一或多個彈性 IP 地址，然後選擇 Actions (動作)、Disable transfer (停用轉移)。
6. 在文字方塊中輸入 **disable**，以便進行確認。

## 7. 選擇提交。

### AWS CLI

若要停用彈性 IP 地址轉移

使用 [disable-address-transfer](#) 命令。

### PowerShell

若要停用彈性 IP 地址轉移

使用 [Disable-EC2AddressTransfer](#) 命令。

### 接受轉移後的彈性 IP 地址

本節旨在說明如何接受轉移後的彈性 IP 地址。

當您轉移彈性 IP 地址時，AWS 帳戶之間會發生兩步驟交握。來源帳戶開始轉移時，轉移帳戶有七天時間可以接受彈性 IP 地址轉移。在這七天內，來源帳戶可以檢視擱置中的傳輸 (例如，在 AWS 主控台中或使用[描述-位址-傳輸](#) AWS CLI 令)。七天後轉移將到期，屆時彈性 IP 地址的擁有權會回到來源帳戶。

接受轉移時，請注意下列可能發生的異常情況，以及如何解決這些異常問題：

- AddressLimit已超過：如果您的移轉帳戶已超過彈性 IP 位址配額，則來源帳戶可以啟用彈性 IP 位址傳輸，但是當移轉帳戶嘗試接受移轉時，就會發生此例外狀況。根據預設，每個區域的所有 AWS 帳戶限制為 5 個彈性 IP 位址。如需提高限制的指示，請參閱 [彈性 IP 地址配額](#)。
- InvalidTransfer。AddressCustomPtrSet: 如果您或組織中的某人已設定您嘗試轉移的彈性 IP 位址以使用反向 DNS 查閱，則來源帳戶可以啟用彈性 IP 位址的傳輸，但是當移轉帳戶嘗試接受轉移時，就會發生此例外狀況。若要解決這個問題，來源帳戶必須移除該彈性 IP 地址的 DNS 記錄。如需詳細資訊，請參閱 [針對電子郵件應用程式使用反向 DNS](#)。
- InvalidTransfer。AddressAssociated：如果彈性 IP 地址與 ENI 或 EC2 執行個體相關聯，則來源帳戶可以為彈性 IP 地址啟用傳輸，但是當轉移帳戶嘗試接受傳輸時，會發生此例外狀況。若要解決這個問題，來源帳戶必須取消與該彈性 IP 地址的關聯。如需詳細資訊，請參閱 [取消彈性 IP 地址的關聯](#)。

如有任何其他異常情況，[請聯絡 AWS Support](#)。

這些步驟必須由轉移帳戶完成。

## Console

### 接受彈性 IP 地址轉移

1. 確保您使用的是轉移帳戶。
2. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
3. 在導覽窗格中，選擇 Elastic IPs (彈性 IP)。
4. 選擇 Actions (動作)、Accept transfer (接受轉移)。
5. 當您接受轉移時，與該彈性 IP 地址有關的所有標籤都不會隨著該彈性 IP 地址轉移。如果您要為接受的彈性 IP 地址定義 Name (名稱) 標籤，請選取 Create a tag with a key of 'Name' and a value that you specify (建立索引鍵為 'Name' 並具有您指定值的標籤)。
6. 輸入您要轉移的彈性 IP 地址。
7. 如果您要接受多個轉移的彈性 IP 地址，請選擇 Add address (新增位址)，以輸入其他彈性 IP 地址。
8. 選擇提交。

## AWS CLI

### 若要接受彈性 IP 地址轉移

使用 [accept-address-transfer](#) 命令。

## PowerShell

### 若要接受彈性 IP 地址轉移

使用 [Approve-EC2AddressTransfer](#) 命令。

## 釋出彈性 IP 地址

如果您不再需要彈性 IP 地址，建議您使用下列其中一種方法來釋出此地址。要釋放的地址目前不得與 AWS 資源相關聯，例如 EC2 執行個體、NAT 閘道或 Network Load Balancer。

### Note

如果您聯絡 AWS 支援部門，為彈性 IP (EIP) 位址設定反向 DNS，您可以移除反向 DNS，但是您無法釋放彈性 IP 位址，因為它已被 AWS 支援鎖定。若要解除彈性 IP 地址的鎖定，請聯絡 [AWS Support](#)。解鎖彈性 IP 地址後，您可以釋放彈性 IP 地址。

## Console

### 釋出彈性 IP 地址

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Elastic IPs (彈性 IP)。
3. 選取要釋出的彈性 IP 地址，然後依序選取 Actions (動作)、Release Elastic IP addresses (釋出彈性 IP 地址)。
4. 選擇 Release (釋出)。

## AWS CLI

### 釋出彈性 IP 地址

使用發布地址命令 AWS CLI 。

## PowerShell

### 釋出彈性 IP 地址

使用指 [Remove-EC2Address](#) AWS Tools for Windows PowerShell 令。

## 復原彈性 IP 地址

如果您已釋放彈性 IP 地址，您可以將它復原。適用的規定如下：

- 如果彈性 IP 地址已配置給其他 AWS 帳戶，或復原之後會導致您超過彈性 IP 地址限制，則您無法復原此彈性 IP 地址。
- 您無法復原與彈性 IP 地址相關聯的標籤。
- 您只能使用 Amazon EC2 API 或命令列工具來復原彈性 IP 地址。

## AWS CLI

### 復原彈性 IP 地址

使用 [分配地址](#) AWS CLI 命令，並使用 `--address` 參數指定 IP 地址，如下所示。

```
aws ec2 allocate-address --domain vpc --address 203.0.113.3
```



## PowerShell

### 復原彈性 IP 地址

使用命 [New-EC2Address](#) AWS Tools for Windows PowerShell 令並使用 `-Address` 參數指定 IP 位址，如下所示。

```
PS C:\> New-EC2Address -Address 203.0.113.3 -Domain vpc -Region us-east-1
```

## 針對電子郵件應用程式使用反向 DNS

如果您打算從執行個體傳送電子郵件給第三方，我們建議您佈建一或多個彈性 IP 地址，並將靜態反向 DNS 紀錄指派給用於傳送電子郵件的彈性 IP 地址。這有助於避免某些反垃圾郵件組織將您的電子郵件標記為垃圾郵件。AWS 與 ISP 和互聯網反垃圾郵件組織合作，以減少從這些地址發送的電子郵件被標記為垃圾郵件的機會。

### 考量事項

- 建立反向 DNS 紀錄之前，您必須設定對應的轉寄 DNS 紀錄 (紀錄類型 A)，並指向您的彈性 IP 地址。
- 如果反向 DNS 紀錄與彈性 IP 地址相關聯，該彈性 IP 地址會鎖定到您的帳戶，而且在移除記錄之前都無法從您的帳戶釋放。
- AWS GovCloud (US) Region

您無法使用主控台或建立反向 DNS 紀錄 AWS CLI。AWS 必須為您指派靜態反向 DNS 紀錄。打開 [請求移除反向 DNS 及電子郵件傳送限制](#)，並將您的彈性 IP 地址和反向 DNS 紀錄提供給我們。

### 建立反向 DNS 記錄

若要建立反向 DNS 記錄，請選擇與慣用方法相符的索引標籤。

### Console

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Elastic IPs (彈性 IP)。
3. 選取彈性 IP 地址，然後選取 Actions (動作)、Update reverse DNS (更新反向 DNS)。
4. 針對 Reverse DNS domain name (反向 DNS 網域名稱)，輸入網域名稱。

5. 輸入 **update** 以確認。
6. 選擇 Update (更新)。

## AWS CLI

使用中的 [modify-address-attribute](#) 指令 AWS CLI，如下列範例所示：

```
aws ec2 modify-address-attribute --allocation-id eipalloc-abcdef01234567890 --
domain-name example.com
{
 "Addresses": [
 {
 "PublicIp": "192.0.2.0",
 "AllocationId": "eipalloc-abcdef01234567890",
 "PtrRecord": "example.net."
 "PtrRecordUpdate": {
 "Value": "example.com.",
 "Status": "PENDING"
 }
 }
]
}
```

## 移除反向 DNS 記錄

若要移除反向 DNS 記錄，請選擇與慣用方法相符的索引標籤。

### Console

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Elastic IPs (彈性 IP)。
3. 選取彈性 IP 地址，然後選取 Actions (動作)、Update reverse DNS (更新反向 DNS)。
4. 針對 Reverse DNS domain name (反向 DNS 網域名稱)，清除相應的網域名稱。
5. 輸入 **update** 以確認。
6. 選擇 Update (更新)。

## AWS CLI

使用中的 [reset-address-attribute](#) 指令 AWS CLI，如下列範例所示：

```
aws ec2 reset-address-attribute --allocation-id eipalloc-abcdef01234567890 --
attribute domain-name
{
 "Addresses": [
 {
 "PublicIp": "192.0.2.0",
 "AllocationId": "eipalloc-abcdef01234567890",
 "PtrRecord": "example.com."
 "PtrRecordUpdate": {
 "Value": "example.net.",
 "Status": "PENDING"
 }
 }
]
}
```

### Note

如果您在執行命令時收到下列錯誤，您可以提交[要求以移除電子郵件傳送限制](#)以取 AWS Support 得協助。

The address with allocation id cannot be released because it is locked to your account (帶分配 ID 的地址已鎖定至您的帳戶，無法釋出)。

## 彈性 IP 地址配額

根據預設，每個區域的所有 AWS 帳戶都有五 (5) 個彈性 IP 位址的配額，因為公用 (IPv4) 網際網路位址是稀缺的公用資源。我們強烈建議您將彈性 IP 地址主要用在發生執行個體故障時將地址重新映射至另一個執行個體，並針對所有其他節點間通訊使用 [DNS 主機名稱](#)。

### 確認使用中的彈性 IP 地址數目

開啟 Amazon EC2 主控台，網址為：<https://console.aws.amazon.com/ec2/>，然後從導覽窗格中選擇 Elastic IPs (彈性 IP)。

### 驗證彈性 IP 地址的目前帳戶配額

1. 開啟 Service Quotas 主控台，網址為 <https://console.aws.amazon.com/servicequotas/>。
2. 從導覽列 (位於畫面頂部) 中，選取一個區域。
3. 在儀表板上，選擇 Amazon Elastic Compute Cloud (Amazon EC2)。

如果儀表板上未列出 Amazon Elastic Compute Cloud (Amazon EC2)，請選擇 AWS services (AWS 服務)，在搜尋欄位中輸入 **EC2**，然後選擇 Amazon Elastic Compute Cloud (Amazon EC2)。

4. 在 Amazon EC2 服務配額頁面上，在搜尋欄位中輸入 **IP**。限制為 EC2-VPC 彈性 IP。如需詳細資訊，請選擇 limit (限制)。

如果您認為自己的架構需要更多的彈性 IP 地址，您可以直接從 Service Quotas 主控台請求提高配額。若要請求增加配額，請選擇在帳戶層級請求增加。如需詳細資訊，請參閱 [Amazon EC2 服務配額](#)。

## 彈性網絡介面

彈性網絡介面是代表虛擬網路卡之 VPC 中的邏輯網路元件。其可以包含下列屬性：

- 來自您 VPC 之 IPv4 地址範圍的主要私有 IPv4 地址
- 來自您 VPC 之 IPv6 位址範圍的主要 IPv6 位址
- 來自您 VPC 之 IPv4 地址範圍的一或多個輔助私有 IPv4 地址
- 每個私有 IP 地址一個彈性 IP 地址 (IPv4)
- 一個公有 IPv4 地址
- 一或多個 IPv6 地址
- 一或多個安全群組
- 一個 MAC 地址
- 一個來源/目標檢查標記
- 一項描述

您可建立並設定網路介面，然後將它們連接到同一可用區域中的執行個體。您的帳戶可能還具有由請求者管理的網路介面，這些介面由 AWS 服務建立和管理，讓您能夠使用其他資源和服務。您無法自行管理這些網路介面。如需詳細資訊，請參閱 [申請者受管的網路介面](#)。

此 AWS 資源在 AWS Management Console 和 Amazon EC2 API 中稱為網路界面。因此，我們在本文件中使用「網路介面」而非「彈性網路介面」。本文件中的「網路介面」一詞一律表示「彈性網路介面」。

### 內容

- [網路介面基本概念](#)

- [網路卡](#)
- [每個執行個體類型每網路介面的 IP 地址](#)
- [使用網路介面](#)
- [設定網路介面的最佳實務](#)
- [網路介面案例](#)
- [申請者受管的網路介面](#)
- [為 Amazon EC2 網路介面指派字首](#)

## 網路介面基本概念

您可以建立網路介面，將它連接到執行個體、與執行個體分離，以及將它連接到另一個執行個體。網路介面的屬性在它連接到執行個體或與執行個體分離，以及重新連接到另一個執行個體時，會一直跟著它。當您將網路介面從一個執行個體移至另一個執行個體時，網路流量會重新導向至新的執行個體。

### 主要網路介面

每個執行個體都有預設的網路介面，稱為主要網路介面。您無法分離主要網路介面和執行個體。您可建立和連接其他的網路介面。您可使用的網路介面數上限隨執行個體類型而異。如需詳細資訊，請參閱 [每個執行個體類型每網路介面的 IP 地址](#)。

### 網路介面的公有 IPv4 地址

在 VPC 中，所有子網都有可修改的屬性，決定在該子網中建立的網路介面 (因此執行個體在該子網中啟動) 是否獲派公有 IPv4 地址。如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的 [子網路設定](#)。公有 IPv4 地址是從 Amazon 的公有 IPv4 地址集區指派。當您啟動執行個體時，IP 地址是指派給已建立的主要網路介面。

當您建立網路介面時，它會繼承來自子網的公有 IPv4 定址屬性。以後若要修改子網的公有 IPv4 定址屬性，網路介面會保持建立時生效的設定。若您啟動執行個體並指定現有的網路介面做為主要網路介面，則此網路介面會決定公有 IPv4 位址屬性。

如需詳細資訊，請參閱 [公有 IPv4 地址](#)。

### 適用於網路介面的彈性 IP 地址

如有彈性 IP 地址，您可建立它與網路介面其中一個私有 IPv4 地址的關聯。您可建立一個彈性 IP 地址與每個私有 IPv4 地址的關聯。

如果您取消彈性 IP 地址與網路介面的關聯，您可以將其釋放回地址集區。這是建立彈性 IP 地址與不同子網或 VPC 中執行個體關聯的唯一方法，因為網路介面有專屬的子網。

## 網路介面的 IPv6 地址

如果您建立 IPv6 CIDR 區塊與您 VPC 及子網的關聯，您可以將子網範圍內的一或多個 IPv6 地址指派給網路介面。每個 IPv6 地址皆可指派給一個網路介面。

所有子網都有可修改的屬性，決定在該子網中建立的網路介面 (因此執行個體在該子網中啟動) 是否自動獲派子網範圍內 IPv6 地址。如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的[子網路設定](#)。當您啟動執行個體時，IPv6 位址是指派給已建立的主要網路介面。

如需詳細資訊，請參閱 [IPv6 地址](#)。

## 字首委派

Prefix Delegation 字首是保留的私有 IPv4 或 IPv6 CIDR 範圍，您可以將該範圍配置給與執行個體相關聯的網路介面的自動或手動指派。透過使用委派字首，您就可以將 IP 地址範圍指派為單一字首，以更快速地啟動服務。

## 終止行為

您可設定連接至執行個體之網路介面的終止行為。您可指定網路介面是否應該在您終止它所連接的執行個體時自動刪除。

## 來源/目標檢查

您可以啟用或停用來源/目的地檢查，這可確保執行個體是其接收之任何流量的來源或目的地。來源/目的地檢查預設為啟用。如果執行個體執行網路位置轉譯、路由或防火牆等服務，則必須停用來源/目標檢查。

## 監控 IP 流量

您可在您的網路介面上啟用 VPC 流量日誌，擷取出入網路介面的 IP 流量資訊。建立流量日誌後，您可以在 Amazon CloudWatch 日誌中檢視和擷取其資料。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的 [VPC 流量日誌](#)。

## 自動公用 IPv4 位址指派

您可以啟用和停用自動指派公用 IPv4 位址到網路介面。任何網路介面都可以啟用此選項，但僅適用於主要網路介面 (eth0)。如需詳細資訊，請參閱 [管理 IP 地址](#)。

## 網路卡

具有多張網路卡的執行個體可提供更佳的網路效能，包括超過 100 Gbps 的頻寬功能和經過改善的封包速率效能。每個網路介面都連接到網路卡。主要網路介面必須指派給網路卡索引 0。

如果您在啟動支援多個網路卡的執行個體時啟用 Elastic Fabric Adapter (EFA)，則所有網路卡都可用。對於每張網路卡您最多可以指派一個 EFA。EFA 視為網路介面。

下列執行個體支援多張網路卡。所有其他執行個體類型都支援一張網路卡。

執行個體類型	網路卡數量
c6in.32xlarge	2
c6in.metal	2
d11.24xlarge	4
hpc6id.32xlarge	2
hpc7a.12xlarge	2
hpc7a.24xlarge	2
hpc7a.48xlarge	2
hpc7a.96xlarge	2
m6idn.32xlarge	2
m6idn.metal	2
m6in.32xlarge	2
m6in.metal	2
p4d.24xlarge	4
p4de.24xlarge	4
p5.48xlarge	32

執行個體類型	網路卡數量
r6idn.32xlarge	2
r6idn.metal	2
r6in.32xlarge	2
r6in.metal	2
trn1.32xlarge	8
trn1n.32xlarge	16
u7in-16tb.224xlarge	2
u7in-24tb.224xlarge	2
u7in-32tb.224xlarge	2

## 每個執行個體類型每網路介面的 IP 地址

每個執行個體類型都支援網路介面的數目上限、每個網路介面的私有 IPv4 位址數目上限，以及每個網路介面的 IPv6 位址數目上限。每個網路介面的 IPv6 地址限制和私有 IPv4 地址限制是分開的。不是所有執行個體類型都支援 IPv6 定址。

### 可用的網路介面

Amazon EC2 執行個體類型指南提供每個執行個體類型可用網路界面的相關資訊。如需詳細資訊，請參閱下列內容：

- [網路規格 — 一般用途](#)
- [網路規格 — 運算最佳化](#)
- [網路規格 — 記憶體最佳化](#)
- [網路規格 — 儲存最佳化](#)
- [網路規格 — 加速運算](#)
- [網路規格 — 高效能運算](#)
- [網路規格 — 前一代](#)



## 若要使用擷取網路介面資訊 AWS CLI

您可以使用 [describe-Instance AWS CLI 命令](#) 來顯示執行個體類型的相關資訊，例如其支援的網路介面和每個介面的 IP 位址。下列範例會顯示所有 C5 執行個體的此資訊。

```
aws ec2 describe-instance-types --filters "Name=instance-type,Values=c5.*" --query
"InstanceTypes[].{Type: InstanceType, MaxENI: NetworkInfo.MaximumNetworkInterfaces,
IPv4addr: NetworkInfo.Ipv4AddressesPerInterface}" --output table
```

```

| DescribeInstanceTypes |
+-----+-----+-----+
| IPv4addr | MaxENI | Type |
+-----+-----+-----+
| 30 | 8 | c5.4xlarge |
| 50 | 15 | c5.24xlarge |
| 15 | 4 | c5.xlarge |
| 30 | 8 | c5.12xlarge |
| 10 | 3 | c5.large |
| 15 | 4 | c5.2xlarge |
| 50 | 15 | c5.metal |
| 30 | 8 | c5.9xlarge |
| 50 | 15 | c5.18xlarge |
+-----+-----+-----+
```

## 使用網路介面

您可以使用 Amazon EC2 主控台或命令列來使用網路介面。

### 內容

- [建立網路介面](#)
- [檢視網路介面的詳細資訊](#)
- [將網路介面連接到執行個體](#)
- [從執行個體分離網路介面](#)
- [管理 IP 地址](#)
- [修改網路介面屬性](#)
- [新增或編輯標籤](#)
- [刪除網路介面](#)

## 建立網路介面

您可在子網中建立網路介面。建立網路介面之後，無法將其移至另一個子網。必須將網路介面連接到同一可用區域中的執行個體。

### 使用主控台建立網路介面

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選擇 Create network interface (建立網路介面)。
4. (選用) 針對 Description (描述)，輸入描述性名稱。
5. 針對 Subnet (子網)，選取子網。後續步驟中可用的選項取決於您選取子網類型 (僅限 IPv4、僅限 IPv6 或雙堆疊 (IPv4 和 IPv6))。
6. 針對 Private IPv4 address (私有 IPv4 地址)，執行下列其中一項操作：
  - 選取 Auto-assign (自動指派)，以允許 Amazon EC2 從子網選取 IPv4 地址。
  - 選取 Custom (自訂)，然後輸入您從子網選取的 IPv4 地址。
7. (僅限使用 IPv6 地址的子網) 針對 IPv6 address (IPv6 地址)，執行以下其中一個動作：
  - 如果您不想將 IPv6 地址指派給網路介面，則選擇 None (無)。
  - 選取 Auto-assign (自動指派)，以允許 Amazon EC2 從子網選取 IPv6 地址。
  - 選取 Custom (自訂)，然後輸入您從子網選取的 IPv6 地址。
8. (選用) 如果您要在雙堆疊或僅限 IPv6 的子網路中建立網路介面，則可以選擇指派主要 IPv6 IP。這會將主要 IPv6 全域單播位址 (GUA) 指派給網路介面。指派主要 IPv6 地址可讓您避免中斷執行個體或 ENI 的流量。如果要連接此 ENI 的執行個體依賴其 IPv6 位址未變更，請選擇啟用。AWS 會自動將與連接到執行個體的 ENI 相關聯的 IPv6 位址指派為主要 IPv6 位址。在啟用 IPv6 GUA 地址作為主要 IPv6 地址後，您便無法停用該地址。當您啟用 IPv6 GUA 地址作為主要 IPv6 地址時，第一個 IPv6 GUA 地址會成為主要 IPv6 地址，直到執行個體終止或網路介面中斷連接為止。如果有多個與執行個體連接之 ENI 相關聯的 IPv6 地址，並且啟用主要 IPv6 地址，則與 ENI 相關聯的第一個 IPv6 GUA 地址會成為主要 IPv6 地址。
9. (選用) 若要建立 Elastic Fabric Adapter，選擇 Elastic Fabric Adapter、Enable (啟用)。
10. (選用) 在進階設定下的閒置連線追蹤逾時，修改預設閒置連線逾時。如需關於這些選項的詳細資訊，請參閱 [閒置連線追蹤逾時](#)。
  - 已建立的 TCP 逾時：處於已建立狀態的閒置 TCP 連線逾時 (以秒為單位)。最小值：60 秒。最大值：432000 秒 (5 天)。預設：432000 秒。建議使用：432000 秒以下。

- UDP 逾時：只出現單一方向或單一請求回應交易流量的閒置 UDP 流量逾時 (以秒為單位)。最小值：30 秒。最大值：60 秒。預設：30 秒。
- UDP 串流逾時：分類為出現多個請求回應交易串流的閒置 UDP 流量逾時 (以秒為單位)。最小值：60 秒。最大值：180 秒 (3 分鐘) 預設：180 秒。

11. 針對 Security groups (安全群組)，選取一個或多個安全群組。
12. (選用) 針對每個標籤，請選擇 Add new tag (新增標籤)，然後輸入標籤金鑰和選用標籤值。
13. 選擇 Create network interface (建立網路介面)。

### 使用命令列建立網路介面

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [create-network-interface](#) (AWS CLI)
- [New-EC2NetworkInterface](#) (AWS Tools for Windows PowerShell)

### 檢視網路介面的詳細資訊

您可以在自己的帳戶中檢視所有網路介面。

### 使用主控台描述網路介面

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 若要檢視網路介面的詳細資料頁面，請選取網路介面的 ID。或者，若要檢視資訊而不離開網路介面頁面，請選取網路介面的核取方塊。

### 使用命令列描述網路介面

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [describe-network-interfaces](#) (AWS CLI)
- [Get-EC2NetworkInterface](#) (AWS Tools for Windows PowerShell)

### 使用命令列描述網路介面屬性

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [describe-network-interface-attribute](#) (AWS CLI)
- [Get-EC2NetworkInterfaceAttribute](#) (AWS Tools for Windows PowerShell)

## 將網路介面連接到執行個體

使用 Amazon EC2 主控台的執行個體或網路介面頁面，可將網路介面連接到與該網路介面相同的可用區域內的任何執行個體。或者，可以在[啟動執行個體](#)時指定現有網路介面。

### Important

對於僅限 IPv6 子網中的 EC2 執行個體，如果您將次要網路介面連接到執行個體，則第二個網路介面的私有 DNS 主機名稱將解析為執行個體第一個網路介面上的第一個 IPv6 地址。如需 EC2 執行個體私有 DNS 主機名稱的詳細資訊，請參閱 [Amazon EC2 執行個體主機名稱類型](#)。

如已釋放您執行個體上的公有 IPv4 地址，如有多個網路介面連接到執行個體，就不會收到新的地址。如需公有 IPv4 地址行為的詳細資訊，請參閱[公有 IPv4 地址](#)。

## Instances page

使用 Instances (執行個體) 頁面將網路介面連接到執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體的核取方塊。
4. 選擇 Actions (動作)、Networking (網路)、Attach network interface (連接網路介面)。
5. 選擇 VPC。如果您要將次要網路介面連接至執行個體，則網路介面可以位於與執行個體相同的 VPC 中，或位於您擁有的不同 VPC 中 (只要網路介面位於與執行個體位於相同可用區域的子網路中)。如此一來，您就能在不同 VPC 建立多重主目錄執行個體，分別使用不同的網路和安全組態。
6. 選取網路介面。如果執行個體支援多張網路卡，您可以選擇網路卡。
7. 選擇 Attach (連接)。

## Network Interfaces page

使用 Network Interfaces (網路介面) 頁面將網路介面連接到執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊。
4. 選擇 Actions (動作)、Attach (連接)。
5. 選擇執行個體。如果執行個體支援多張網路卡，您可以選擇網路卡。
6. 選擇 Attach (連接)。

使用命令列將網路介面連接到執行個體

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱[存取 Amazon EC2](#)。

### Note

您可以使用附加網路介面指令，將位於另一個 VPC (但位於相同可用區域) 中的網路介面連接至執行個體。AWS CLI 您無法使用執行此操作 AWS Management Console。

- [attach-network-interface](#) (AWS CLI)
- [Add-EC2NetworkInterface](#) (AWS Tools for Windows PowerShell)

## 從執行個體分離網路介面

您可以使用 Amazon EC2 主控台的 Instances (執行個體) 或 Network Interfaces (網路介面) 頁面，隨時分離連接至 EC2 執行個體的輔助網路介面。

如果您嘗試從另一個服務 (例如 Elastic Load Balancing 負載平衡器、Lambda 函數、或 NAT 閘道) 中斷連接至資源的網路介面，您會收到錯誤訊息，指出您沒有存取資源的權限。WorkSpace 若要找出哪個服務建立了連接至網路介面的資源，請檢查網路介面的描述。如果您刪除資源，則會刪除其網路介面。

## Instances page

使用 Instances (執行個體) 頁面從執行個體分離網路介面

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體的核取方塊。檢查 Networking (網路) 標籤的 Network interfaces (網路介面) 區段，確認網路介面已連接至執行個體作為次要網路介面。
4. 選擇 Actions (動作)、Networking (網路)、Detach network interface (分離網路介面)。
5. 選取網路介面，然後選取 Detach (分離)。

## Network Interfaces page

使用 Network Interfaces (網路介面) 頁面從執行個體分離網路介面

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊。檢查 Networking (網路) 標籤的 Instance details (執行個體詳細資訊) 區段，確認網路介面已連接至執行個體作為次要網路介面。
4. 選擇 Actions (動作)、Detach (分離)。
5. 當出現確認提示時，選擇 [分離]。
6. 如果網路介面無法從執行個體分離，請選擇 Force detachment (強制分離)、Enable (啟用)，然後再試一次。我們建議強制分離只作為最後的手段。強制分離可以防止您在相同索引上連接不同的網路介面，直到重新啟動執行個體為止。它也可以防止執行個體中繼資料反映網路介面已分離，直到您重新啟動執行個體為止。

## 使用命令列分離網路介面

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [detach-network-interface](#) (AWS CLI)
- [Dismount-EC2NetworkInterface](#) (AWS Tools for Windows PowerShell)

## 管理 IP 地址

您可以管理以下網路介面的 IP 地址：

- 彈性 IP 地址 (每個私有 IPv4 地址各一個)
- IPv4 地址
- IPv6 地址
- 主要 IPv6 地址

### 管理主控台網路介面的彈性 IP 地址

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊。
4. 若要建立彈性 IP 地址的關聯，請執行下列操作：
  - a. 選擇 Actions (動作)、Associate address (建立與地址的關聯)。
  - b. 針對 Elastic IP address (彈性 IP 地址)，選取彈性 IP 地址。
  - c. 針對 Private IPv4 address (私有 IPv4 地址)，選取要與彈性 IP 地址建立關聯的私有 IPv4 地址。
  - d. (選用) 如果網路介面目前與另一個執行個體或網路介面建立關聯，選擇 Allow the Elastic IP address to be reassociated (允許與彈性 IP 地址重新建立關聯)。
  - e. 選擇 Associate (關聯)。
5. 若要取消彈性 IP 地址的關聯，請執行下列操作：
  - a. 選擇 Actions (動作)、Disassociate address (取消與地址的關聯)。
  - b. 針對 Public IP address (公有 IP 地址)，選取彈性 IP 地址。
  - c. 選擇 Disassociate (取消關聯)。

### 如何使用主控台管理網路介面的 IPv4 和 IPv6 地址

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面。
4. 選擇 Actions (動作)、Manage IP addresses (管理 IP 地址)。
5. 展開網路介面。

6. 針對 IPv4 addresses (IPv4 地址)，視需要修改 IP 地址。若要指派 IPv4 位址，請選擇 [指派新 IP 位址]，然後從子網路範圍指定 IPv4 位址，或讓您 AWS 選擇一個位址。若要取消指派 IPv4 地址，請選擇地址旁的取消指派。
7. 若要指派或取消指派公用 IPv4 位址給網路介面，請選擇「自動指派公用 IP」。您可以為任何網路介面啟用或停用此選項，但僅適用於主要網路介面 (eth0)。
8. 針對 IPv6 addresses (IPv6 地址)，視需要修改 IP 地址。若要指派 IPv6 位址，請選擇 [指派新 IP 位址]，然後從子網路範圍指定 IPv6 位址，或讓您 AWS 選擇一個位址。若要取消指派 IPv6 地址，請選擇地址旁的取消指派。
9. (選用) 如果您要在雙堆疊或僅限 IPv6 的子網路中修改網路介面，則可以選擇指派主要 IPv6 IP。指派主要 IPv6 地址可讓您避免中斷執行個體或 ENI 的流量。如果要連接此 ENI 的執行個體依賴其 IPv6 位址未變更，請選擇啟用。AWS 會自動將與連接到執行個體的 ENI 相關聯的 IPv6 位址指派為主要 IPv6 位址。在啟用 IPv6 GUA 地址作為主要 IPv6 地址後，您便無法停用該地址。當您啟用 IPv6 GUA 地址作為主要 IPv6 地址時，第一個 IPv6 GUA 地址會成為主要 IPv6 地址，直到執行個體終止或網路介面中斷連接為止。如果有多個與執行個體連接之 ENI 相關聯的 IPv6 地址，並且啟用主要 IPv6 地址，則與 ENI 相關聯的第一個 IPv6 GUA 地址會成為主要 IPv6 地址。
10. 選擇儲存。

### 管理網路介面的 IP 位址 AWS CLI

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [assign-ipv6-addresses](#)
- [associate-address](#)
- [disassociate-address](#)
- [unassign-ipv6-addresses](#)

### 使用 Windows 的工具來管理網路介面的 IP 位址 PowerShell

您可以使用下列其中一個命令。

- [Register-EC2Address](#)
- [Register-EC2Ipv6 AddressList](#)
- [Unregister-EC2Address](#)
- [Unregister-EC2Ipv6 AddressList](#)



## 修改網路介面屬性

您可以變更下列網路介面屬性：

- [Description](#)
- [安全群組](#)
- [在終止時刪除](#)
- [來源/目標檢查](#)

### 如何使用主控台變更網路介面的描述

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊。
4. 選擇 Actions (動作)、Change description (變更描述)。
5. 針對 Description (描述)，請輸入網路介面的描述。
6. 選擇 Save (儲存)。

### 如何使用主控台變更網路介面的安全群組

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊。
4. 選擇 Actions (動作)、Change security groups (變更安全群組)。
5. 針對 Associated security groups (關聯的安全群組)，選取要使用的安全群組，然後選取 Save (儲存)。

必須為同一個 VPC 建立安全群組和網路介面。若要變更 Elastic Load Balancing 等其他服務所擁有的執行個體的安全群組，請透過該服務執行操作。

### 如何使用主控台變更網路介面的終止行為

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。

2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊。
4. 選擇 Actions (動作)、Change termination behavior (變更終止行為)。
5. 視需選取或清除 Delete on termination (終止時刪除)、Enable (啟用)，然後選取 Save (儲存)。

#### 使用主控台變更網路介面的來源/目標檢查

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊。
4. 選擇 Actions (動作)、Change source/dest check (變更來源/目的地檢查)。
5. 視需選取或清除 Source/destination check (來源/目的地檢查)、Enable (啟用)，然後選取 Save (儲存)。

#### 變更閒置連線追蹤逾時的方法如下：

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊。
4. 選擇動作、修改連線逾時。
5. 修改閒置連線追蹤逾時。如需關於這些選項的詳細資訊，請參閱 [閒置連線追蹤逾時](#)。
  - 已建立的 TCP 逾時：處於已建立狀態的閒置 TCP 連線逾時 (以秒為單位)。最小值：60 秒。最大值：432000 秒 (5 天)。預設：432000 秒。建議使用：432000 秒以下。
  - UDP 逾時：只出現單一方向或單一請求回應交易流量的閒置 UDP 流量逾時 (以秒為單位)。最小值：30 秒。最大值：60 秒。預設：30 秒。
  - UDP 串流逾時：分類為出現多個請求回應交易串流的閒置 UDP 流量逾時 (以秒為單位)。最小值：60 秒。最大值：180 秒 (3 分鐘) 預設值：180 秒。
6. 選擇儲存。

#### 如何使用命令列修改網路介面屬性

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [modify-network-interface-attribute](#) (AWS CLI)
- [Edit-EC2NetworkInterfaceAttribute](#) (AWS Tools for Windows PowerShell)

## 新增或編輯標籤

標籤是您可新增至網路介面的中繼資料。標籤是私有的，只有您的帳戶看得見。每個標籤皆包含一個鍵和一個選用值。如需標籤的詳細資訊，請參閱[標記您的 Amazon EC2 資源](#)。

使用主控台新增或編輯網路介面的標籤

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊。
4. 在 Tags (標籤) 索引標籤中，選擇 Manage tags (管理標籤)。
5. 針對要建立的每個標籤，選擇 Add new tag (新增標籤)，然後輸入金鑰和選用值。完成後，選擇 Save (儲存)。

使用命令列新增或編輯網路介面的標籤

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱[存取 Amazon EC2](#)。

- [create-tags](#) (AWS CLI)
- [New-EC2Tag](#) (AWS Tools for Windows PowerShell)

## 刪除網路介面

刪除網路介面會釋放所有與介面關聯的屬性，並釋放其他執行個體要使用的所有私有 IP 地址或彈性 IP 地址。

您無法刪除正在使用的網路介面。首先，您必須[分開網路介面](#)。

使用主控台刪除網路介面

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊，然後選取 Actions (動作)、Delete (刪除)。
4. 出現確認提示時，請選擇 Delete (刪除)。

## 使用命令列刪除網路介面

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [delete-network-interface](#) (AWS CLI)
- [Remove-EC2NetworkInterface](#) (AWS Tools for Windows PowerShell)

## 設定網路介面的最佳實務

- 您可在執行個體執行時 (熱連接)、停止時 (暖連接) 或正在啟動時 (冷連接) 將網路介面連接到執行個體。
- 您可在執行個體執行時或停止時分離輔助網路介面。但是您不能分離主要網路介面。
- 如果執行個體位於相同的可用區域和 VPC，但位於不同的子網中，則您可將次要網路介面從原來執行個體移至另一個執行個體。
- 使用 CLI、API 或開發套件啟動執行個體時，您可以指定主要網路介面和其他網路介面。
- 以多個網路介面啟動 Amazon Linux 或 Windows Server 執行個體，會自動設定執行個體作業系統的介面、私有 IPv4 地址和路由表。
- 其他網路介面的暖或熱連接可能需要您手動帶出第二個介面、設定私有 IPv4 地址，以及相應修改路由表。執行 Amazon Linux 或 Windows Server 的執行個體會自動辨識暖或熱連接並自行設定。
- 您不能將另一個網路介面連接到執行個體 (例如，NIC 小組組態)，以增加或加倍雙目錄執行個體傳入或傳出的網路頻寬。
- 如果您將相同子網中兩個或以上的網路介面連接到執行個體，您可能會發生聯網問題，例如非對稱路由。如果可能，請改在主要網路介面上使用輔助私有 IPv4 地址。
- Windows 執行個體 — 如果您使用多個網路介面，則必須將網路介面配置為使用靜態路由。

## 使用適用於 Amazon Linux 2 的 ec2-net-utils 設定您的網路介面

### Note

對於 AL2023，amazon-ec2-net-utils 套件會在 /run/systemd/network 目錄中產生介面特定組態。如需詳細資訊，請參閱 Amazon Linux 2023 使用者指南中的 [聯網服務](#)。

Amazon Linux 2 AMI 可能包含由安裝的其他腳本 AWS，稱為 ec2 網絡實用程序。這些指令碼會選擇性自動化您網路介面的組態。這些指令碼僅供 Amazon Linux 2 使用。

使用下列命令在 Amazon Linux 2 上安裝套件 (如尚未安裝)，如已安裝且有其他更新可用，請更新套件：

```
$ yum install ec2-net-utils
```

下列元件屬於 ec2-net-utils：

udev 規則 (/etc/udev/rules.d)

當網路介面連接、分離或重新連接到執行中的執行個體時識別它們，確保 hotplug 指令碼執行 (53-ec2-network-interfaces.rules)。將 MAC 地址映射到裝置名稱 (75-persistent-net-generator.rules，這會產生 70-persistent-net.rules)。

hotplug 指令碼

產生適合搭配 DHCP 使用的介面組態檔 (/etc/sysconfig/network-scripts/ifcfg-ethN)。也產生路由組態檔案 (/etc/sysconfig/network-scripts/route-ethN)。

DHCP 指令碼

每當網路介面收到新的 DHCP 租賃，此指令碼就會查詢彈性 IP 地址的執行個體中繼資料。它會為每一個彈性 IP 地址在路由政策資料庫中新增規則，以確保該地址使用正確的網路介面外送流量。它還會將每個私有 IP 地址新增至網路介面做為輔助地址。

ec2ifup ethN (/usr/sbin/)

擴充標準 ifup 的功能。在此指令碼重新撰寫組態檔 ifcfg-ethN 和 route-ethN 之後，它會執行 ifup。

ec2ifdown ethN (/usr/sbin/)

擴充標準 ifdown 的功能。在此指令碼從路由政策資料庫中移除網路介面的所有規則後，它會執行 ifdown。

ec2ifscan (/usr/sbin/)

檢查是否有尚未設定的網路介面並加以設定。

ec2-net-utils 初始版本不提供此指令碼。

使用下列命令列出 ec2-net-utils 產生的所有組態檔：

```
$ ls -l /etc/sysconfig/network-scripts/*-eth?
```

若要停用自動化，您可將 EC2SYNC=no 新增至對應的 ifcfg-ethN 檔案。例如，使用以下命令停用 eth1 執行個體的自動化：

```
$ sed -i -e 's/^EC2SYNC=yes/EC2SYNC=no/' /etc/sysconfig/network-scripts/ifcfg-eth1
```

若要完全停用自動化，您可使用以下命令移除套件：

```
$ yum remove ec2-net-utils
```

## 網路介面案例

當您想要執行下列作業時，將多個網路介面連接到一個執行個體會很有用：

- 建立管理網路。
- 在您的虛擬私有雲端 (VPC) 中使用網路和安全設備。
- 在不同的子網上使用工作負載/角色建立雙主目錄的執行個體。
- 建立低預算、高可用性的解決方案。

### 建立管理網路

此案例說明如何根據下列準則和設定建立具有網路介面的管理網路 (影像如下)。

#### 條件

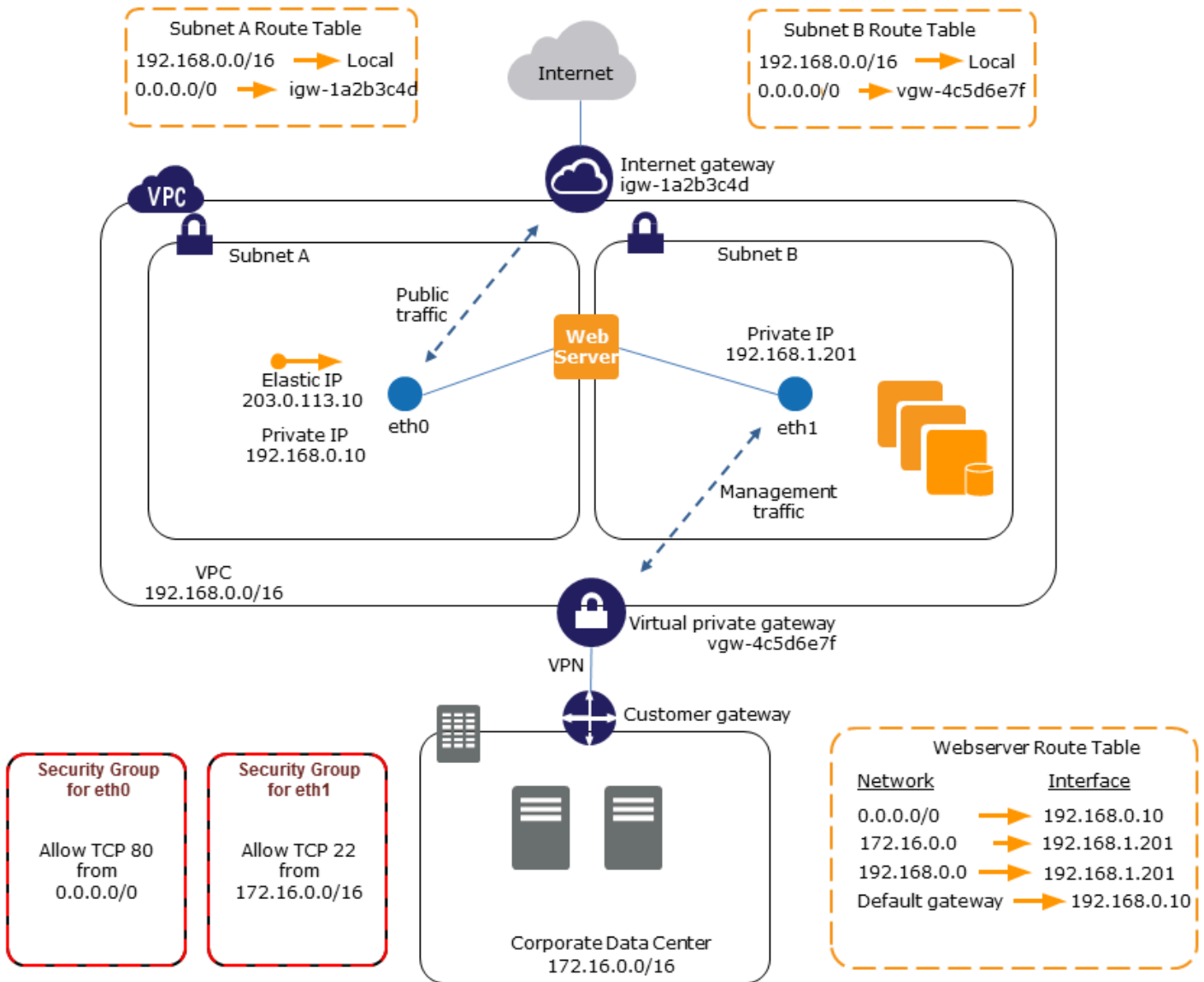
- 執行個體上的主要網路介面 (eth0) 會處理公有流量。
- 執行個體上的次要網路介面 (eth1) 會處理後端管理流量。它連線到另一個子網路，具有更嚴格的存取控制，並與主要網路界面位於同一個可用區域 (AZ)。

#### 設定

- 主要網路介面可能在負載平衡器後面，也可能不在負載平衡器後面，具有關聯的安全群組，可允許從網際網路存取伺服器。例如，允許從 0.0.0.0/0 或從負載平衡器存取 TCP 連接埠 80 和 443
- 次要網路介面有一個關聯的安全性群組，該群組僅允許從下列其中一個位置啟動 SSH 存取：
  - 允許的 IP 地址範圍 (VPC 內或來自網際網路)。
  - 與主要網路介面位於相同 AZ 中的私有子網路。
  - 虛擬私有閘道。

### Note

為確保容錯移轉功能，請考慮使用輔助私有 IPv4 處理網路介面的傳入流量。執行個體故障時，您可將執行個體及/或輔助私有 IPv4 地址移至待命的執行個體。



## 在您的 VPC 中使用網路和安全設備

有些網路和安全設備，如負載平衡器、網路位置轉譯 (NAT) 伺服器 and 代理伺服器，最好能設定多個網路介面。您可建立輔助網路介面，並將它們連接到正在執行這些應用程式類型的執行個體，然後使用它們自己的公有和私有 IP 地址、安全群組和來源/目標檢查來設定其他介面。

## 在不同的子網上使用工作負載/角色建立雙主目錄的執行個體

您可在連線至應用程式伺服器所在之中間層網路的每一個 Web 伺服器上放置網路介面。應用程式伺服器也可以是資料庫伺服器所在之後端網路 (子網) 的雙目錄。每個雙目錄執行個體都會在前端收到及處理請求、起始後端連線，然後將請求傳送至後端網路的伺服器，而非透過雙目錄執行個體路由網路套件。

## 在同一帳戶的不同 VPC 使用工作負載/角色建立雙主目錄的執行個體

您可以在一個 VPC 中啟動 EC2 執行個體，並將另一個 VPC (位於同一個可用區域) 的次要 ENI 連接到執行個體。如此一來，您就能在不同 VPC 建立多重主目錄執行個體，分別使用不同的網路和安全組態。您無法跨不同帳戶跨 VPC 建立多重主目錄執行個體。AWS

在下列使用案例中，您可以跨 VPC 使用雙主目錄執行個體：

- 克服兩個無法對等的 VPC 之間的 CIDR 重疊問題：您可以在 VPC 中運用次要 CIDR，並允許執行個體跨兩個非重疊的 IP 範圍進行通訊。
- 在單一帳戶內連接多個 VPC：啟用個別資源之間的通訊，這些資源通常會以 VPC 界限隔開。

## 建立低預算、高可用性的解決方案

如果提供特定功能的其中一個執行個體故障，其網路介面可連接到為相同角色預先設定之替代或熱待命的執行個體，以迅速復原服務。例如，您可使用網路介面做為重大服務的主要或輔助網路介面，例如資料庫執行個體或 NAT 執行個體。如果執行個體故障，您 (或更有可能是代您執行的程式碼) 可將網路介面連接到熱待命執行個體。因為執行個體維持其私有 IP 地址、彈性 IP 地址和 MAC 地址，所以您一旦將網路介面連接到替代執行個體，網路流量即開始流向待命的執行個體。使用者從執行個體故障到網路介面連接到待命執行個體這段時間內，會短暫遺失連線能力，但不需要變更路由表或您的 DNS 伺服器。

## 申請者受管的網路介面

請求者受管的網路介面是 AWS 服務代表您在您的 VPC 中建立的網路介面。網路介面與其他服務的資源相關聯，例如來自 Amazon RDS 的資料庫執行個體、NAT 閘道或來自 AWS PrivateLink 的介面 VPC 端點。

### 考量事項

- 您可以檢視您帳戶中請求者受管的網路介面。您可以新增或移除標籤，但不能變更請求者受管網路介面的其他屬性。



- 您無法分離請求者受管網路介面。
- 當您刪除與請求者管理的網路介面相關聯的資源時，會 AWS 服務 卸離網路介面並將其刪除。如果服務分離網路介面，但沒有將其刪除，則可以刪除分離的網路介面。

### 使用主控台檢視申請者受管的網路介面

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，依序選擇 Network & Security (網路與安全) 和 Network Interfaces (網路介面)。
3. 選取網路介面 ID，開啟其詳細資訊頁面。
4. 下列為可用來確定網路介面用途的重點欄位：
  - Description (說明)：由建立介面的 AWS 服務提供的說明。例如，"VPC Endpoint Interface vpce-089f2123488812123"。
  - 要求者管理：指出網路介面是否由管理。AWS
  - 請求者識別碼：建立網路介面之主體或服務的別名或 AWS 帳戶識別碼。如果您建立了網路介面，這就是您的 AWS 帳戶 ID。否則，其他主體或服務將會加以建立。

若要檢視請求者管理的網路介面，請使用 AWS CLI

使用 [describe-network-interfaces](#) 命令，如下所示。

```
aws ec2 describe-network-interfaces --filters Name=requester-managed,Values=true
```

下列為顯示可用來確定網路介面用途之重點欄位的範例輸出：Description 和 InterfaceType。

```
{
 ...
 "Description": "VPC Endpoint Interface vpce-089f2123488812123",
 ...
 "InterfaceType": "vpc_endpoint",
 ...
 "NetworkInterfaceId": "eni-0d11e3ccd2c0e6c57",
 ...
 "RequesterId": "727180483921",
 "RequesterManaged": true,
 ...
}
```

使用 Windows 的工具檢視請求者管理的網路介面 PowerShell

使用 [Get-EC2NetworkInterface](#) 指令程式，如下所示。

```
Get-EC2NetworkInterface -Filter @{ Name="requester-managed"; Values="true" }
```

下列為顯示可用來確定網路介面用途之重點欄位的範例輸出：Description 和 InterfaceType。

```
Description : VPC Endpoint Interface vpce-089f2123488812123
...
InterfaceType : vpc_endpoint
...
NetworkInterfaceId : eni-0d11e3ccd2c0e6c57
...
RequesterId : 727180483921
RequesterManaged : True
...
```

## 為 Amazon EC2 網路介面指派字首

您可以自動或手動將私有 IPv4 或 IPv6 CIDR 範圍指派給網路介面。透過指派字首，您就可以擴展和簡化應用程式的管理，包括需要在執行個體上有多個 IP 地址的容器和聯網應用程式。如需 IPv4 和 IPv6 地址的詳細資訊，請參閱 [Amazon EC2 執行個體 IP 定址](#)。

以下為可用的指派選項：

- 自動指派 — 從 VPC 子網路的 IPv4 或 IPv6 CIDR 區塊中 AWS 選擇首碼，並將其指定給您的網路介面。
- 手動指派 — 您可以從 VPC 子網路的 IPv4 或 IPv6 CIDR 區塊指定首碼，並在將前置詞指定給網路介面之前，先 AWS 確認該首碼尚未指定給其他資源。

指派字首有下列優點：

- 增加網路介面上的 IP 地址 – 當您使用字首時，會指派 IP 地址區塊，而非個別 IP 地址。這會增加網路介面的 IP 地址數量。
- 簡化容器的 VPC 管理 – 在容器應用程式中，每個容器都需要唯一的 IP 地址。為執行個體指派字首可簡化 VPC 的管理，因為您可以啟動和終止容器，而無需為個別 IP 指派呼叫 Amazon EC2 API。

## 目錄

- [指派字首的基本概念](#)
- [字首的考量與限制](#)
- [使用字首](#)

## 指派字首的基本概念

- 您可以為新的或現有的網路介面指派字首。
- 若要使用字首，請先為網路介面指派字首，然後將網路介面連接至執行個體，接著設定作業系統。
- 當您選擇指定字首的選項時，該字首必須符合下列需求：
  - 您可以指定的 IPv4 字首是 /28。
  - 您可以指定的 IPv6 字首是 /80。
  - 字首位於網路介面的子網 CIDR 中，且不會與指派給子網中現有資源的其他字首或 IP 地址重疊。
- 您可以將字首指派給主要或次要網路介面。
- 您可以將彈性 IP 地址指派給具有指派字首的網路介面。
- 您也可以將彈性 IP 地址指派給已指派字首的 IP 地址部分。
- 我們會將執行個體的私有 DNS 主機名稱解析為主要私有 IPv4 位址。
- 我們會使用以下格式，為網路介面指派每個私有 IPv4 地址，包括字首的私有 IPv4 地址：
  - us-east-1 區域

```
ip-private-ipv4-address.ec2.internal
```

- 所有其他區域

```
ip-private-ipv4-address.region.compute.internal
```

## 字首的考量與限制

使用字首時，請考慮下列事項：

- [AWS Nitro 系統上建置的執行個體支援具有前置字元的網路介面](#)。
- 網路介面的字首僅限於使用 IPv6 地址和私有 IPv4 地址。
- 您可以指派給網路介面的 IP 地址數量上限會因執行個體類型而異。指派給網路介面的每個字首都計為一個 IP 地址。例如，c5.large 執行個體每個網路介面的 IPv4 地址數量上限為 10 個。此執行個

體的每個網路介面都具備主要 IPv4 地址。如果網路介面沒有次要 IPv4 地址，則最多可以為網路介面指派 9 個字首。為網路介面多指派一個 IPv4 地址，可為網路介面指派的字首數量便少一個。如需詳細資訊，請參閱 [每個執行個體類型每網路介面的 IP 地址](#)。

- 來源/目標檢查中包含字首。

## 使用字首

您可以搭配使用字首與網路介面，如下所示。

### 任務

- [在建立網路介面期間指派字首](#)
- [為現有的網路介面指派字首](#)
- [使用字首設定網路介面的作業系統](#)
- [檢視被指派給網路介面的字首](#)
- [從網路介面移除字首](#)

### 在建立網路介面期間指派字首

如果您使用自動指派選項，您可以在子網路中保留 IP 位址區塊。AWS 從此塊中選擇前綴。如需詳細資訊，請參閱 [《Amazon VPC 使用者指南》](#) 中的子網 CIDR 保留。

建立網路介面後，請使用「[貼附網路介面](#)」[AWS CLI 指令將網路介面貼附至例證](#)。您必須使用字首設定使用網路介面的作業系統。如需詳細資訊，請參閱 [使用字首設定網路介面的作業系統](#)。

### 任務

- [在建立網路介面期間指派自動字首](#)
- [在建立網路介面期間指派特定的字首](#)

### 在建立網路介面期間指派自動字首


您可以使用下列方法之一在建立網路介面期間指派自動字首。

### Console

#### 在建立網路介面期間指派自動字首

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。

2. 在導覽窗格中，選擇 Network Interfaces (網路介面)，然後選擇 Create network interface (建立網路介面)。
3. 指定網路介面的描述，選取要在其中建立網路介面的子網，然後設定私有 IPv4 和 IPv6 地址。
4. 展開 Advanced settings (進階設定)，並執行下列動作：
  - a. 若要自動指派 IPv4 字首，對於 IPv4 prefix delegation (IPv4 字首委派)，選擇 Auto-assign (自動指派)。然後對於 Number of IPv4 prefixes (IPv4 字首的數目)，指定要指派的字首數目。
  - b. 若要自動指派 IPv4 字首，對於 IPv6 prefix delegation (IPv6 字首委派)，選擇 Auto-assign (自動指派)。然後對於 Number of IPv6 prefixes (IPv6 字首的數目)，指定要指派的字首數目。

 Note

IPv6 prefix delegation (IPv6 字首委派) 僅在為 IPv6 啟用選定的子網才會出現。

5. 選取要與網路介面建立關聯的安全群組，並視需要指派資源標籤。
6. 選擇 Create network interface (建立網路介面)。

## AWS CLI

在建立網路介面期間指派自動 IPv4 字首

使用指 [create-network-interface](#) 令並設定 `--ipv4-prefix-count` 為您要 AWS 指定的前置字元數目。在下列範例中，AWS 指定 1 前置字元。

```
$ C:\> aws ec2 create-network-interface \
--subnet-id subnet-047cfed18eEXAMPLE \
--description "IPv4 automatic example" \
--ipv4-prefix-count 1
```

## 範例輸出

```
{
 "NetworkInterface": {
 "AvailabilityZone": "us-west-2a",
 "Description": "IPv4 automatic example",
 "Groups": [
 {
```

```

 "GroupName": "default",
 "GroupId": "sg-044c2de2c4EXAMPLE"
 }
],
"InterfaceType": "interface",
"Ipv6Addresses": [],
"MacAddress": "02:98:65:dd:18:47",
"NetworkInterfaceId": "eni-02b80b4668EXAMPLE",
"OwnerId": "123456789012",
"PrivateIpAddress": "10.0.0.62",
"PrivateIpAddresses": [
 {
 "Primary": true,
 "PrivateIpAddress": "10.0.0.62"
 }
],
"Ipv4Prefixes": [
 {
 "Ipv4Prefix": "10.0.0.208/28"
 }
],
"RequesterId": "AIDAIV5AJI5LXF5XXDPC0",
"RequesterManaged": false,
"SourceDestCheck": true,
"Status": "pending",
"SubnetId": "subnet-047cfed18eEXAMPLE",
"TagSet": [],
"VpcId": "vpc-0e12f52b21EXAMPLE"
}
}

```

在建立網路介面期間指派自動 IPv6 字首

使用指 [create-network-interface](#) 令並設定 `--ipv6-prefix-count` 為您要 AWS 指定的前置字元數目。在下列範例中，AWS 指定1前置字元。

```

$ C:\> aws ec2 create-network-interface \
--subnet-id subnet-047cfed18eEXAMPLE \
--description "IPv6 automatic example" \
--ipv6-prefix-count 1

```

範例輸出

```
{
 "NetworkInterface": {
 "AvailabilityZone": "us-west-2a",
 "Description": "IPv6 automatic example",
 "Groups": [
 {
 "GroupName": "default",
 "GroupId": "sg-044c2de2c4EXAMPLE"
 }
],
 "InterfaceType": "interface",
 "Ipv6Addresses": [],
 "MacAddress": "02:bb:e4:31:fe:09",
 "NetworkInterfaceId": "eni-006edbcfa4EXAMPLE",
 "OwnerId": "123456789012",
 "PrivateIpAddress": "10.0.0.73",
 "PrivateIpAddresses": [
 {
 "Primary": true,
 "PrivateIpAddress": "10.0.0.73"
 }
],
 "Ipv6Prefixes": [
 {
 "Ipv6Prefix": "2600:1f13:fc2:a700:1768::/80"
 }
],
 "RequesterId": "AIDAIV5AJI5LXF5XXDPC0",
 "RequesterManaged": false,
 "SourceDestCheck": true,
 "Status": "pending",
 "SubnetId": "subnet-047cfed18eEXAMPLE",
 "TagSet": [],
 "VpcId": "vpc-0e12f52b21EXAMPLE"
 }
}
```

在建立網路介面期間指派特定的字首

您可以使用下列方法之一在建立網路介面期間指派特定字首。

## Console

在建立網路介面期間指派特定的字首

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)，然後選擇 Create network interface (建立網路介面)。
3. 指定網路介面的描述，選取要在其中建立網路介面的子網，然後設定私有 IPv4 和 IPv6 地址。
4. 展開 Advanced settings (進階設定)，並執行下列動作：
  - a. 若要指派特定的 IPv4 字首，對於 IPv4 prefix delegation (IPv4 字首委派)，選擇 Custom (自訂)。然後選擇 Add new prefix (新增字首)，並輸入要使用的字首。
  - b. 若要指派特定的 IPv6 字首，對於 IPv6 prefix delegation (IPv6 字首委派)，選擇 Custom (自訂)。然後選擇 Add new prefix (新增字首)，並輸入要使用的字首。

### Note

IPv6 prefix delegation (IPv6 字首委派) 僅在為 IPv6 啟用選定的子網才會出現。

5. 選取要與網路介面建立關聯的安全群組，並視需要指派資源標籤。
6. 選擇 Create network interface (建立網路介面)。

## AWS CLI

在建立網路介面期間指派特定的 IPv4 字首

使用指 [create-network-interface](#) 令並設定 `--ipv4-prefixes` 為前置字元。AWS 選取此範圍內的 IP 位址。在下列範例中，字首 CIDR 是 `10.0.0.208/28`。

```
$ C:\> aws ec2 create-network-interface \
 --subnet-id subnet-047cfed18eEXAMPLE \
 --description "IPv4 manual example" \
 --ipv4-prefixes Ipv4Prefix=10.0.0.208/28
```

範例輸出

```
{
 "NetworkInterface": {
 "AvailabilityZone": "us-west-2a",
```



```

 "Description": "IPv4 manual example",
 "Groups": [
 {
 "GroupName": "default",
 "GroupId": "sg-044c2de2c4EXAMPLE"
 }
],
 "InterfaceType": "interface",
 "Ipv6Addresses": [],
 "MacAddress": "02:98:65:dd:18:47",
 "NetworkInterfaceId": "eni-02b80b4668EXAMPLE",
 "OwnerId": "123456789012",
 "PrivateIpAddress": "10.0.0.62",
 "PrivateIpAddresses": [
 {
 "Primary": true,
 "PrivateIpAddress": "10.0.0.62"
 }
],
 "Ipv4Prefixes": [
 {
 "Ipv4Prefix": "10.0.0.208/28"
 }
],
 "RequesterId": "AIDAIV5AJI5LXF5XXDPC0",
 "RequesterManaged": false,
 "SourceDestCheck": true,
 "Status": "pending",
 "SubnetId": "subnet-047cfed18eEXAMPLE",
 "TagSet": [],
 "VpcId": "vpc-0e12f52b21EXAMPLE"
 }
}

```

在建立網路介面期間指派特定的 IPv6 字首

使用指 [create-network-interface](#) 令並設定 `--ipv6-prefixes` 為前置字元。AWS 選取此範圍內的 IP 位址。在下列範例中，字首 CIDR 是 `2600:1f13:fc2:a700:1768::/80`。

```

$ C:\> aws ec2 create-network-interface \
 --subnet-id subnet-047cfed18eEXAMPLE \
 --description "IPv6 manual example" \

```

```
--ipv6-prefixes Ipv6Prefix=2600:1f13:fc2:a700:1768::/80
```

## 範例輸出

```
{
 "NetworkInterface": {
 "AvailabilityZone": "us-west-2a",
 "Description": "IPv6 automatic example",
 "Groups": [
 {
 "GroupName": "default",
 "GroupId": "sg-044c2de2c4EXAMPLE"
 }
],
 "InterfaceType": "interface",
 "Ipv6Addresses": [],
 "MacAddress": "02:bb:e4:31:fe:09",
 "NetworkInterfaceId": "eni-006edbcfa4EXAMPLE",
 "OwnerId": "123456789012",
 "PrivateIpAddress": "10.0.0.73",
 "PrivateIpAddresses": [
 {
 "Primary": true,
 "PrivateIpAddress": "10.0.0.73"
 }
],
 "Ipv6Prefixes": [
 {
 "Ipv6Prefix": "2600:1f13:fc2:a700:1768::/80"
 }
],
 "RequesterId": "AIDAIV5AJI5LXF5XXDPC0",
 "RequesterManaged": false,
 "SourceDestCheck": true,
 "Status": "pending",
 "SubnetId": "subnet-047cfed18eEXAMPLE",
 "TagSet": [],
 "VpcId": "vpc-0e12f52b21EXAMPLE"
 }
}
```

## 為現有的網路介面指派字首

指定首碼後，請使用指 [attach-network-interface](#) AWS CLI 令將網路介面附加至例證。您必須使用字首設定使用網路介面的作業系統。如需詳細資訊，請參閱 [使用字首設定網路介面的作業系統](#)。

### 任務

- [將自動字首指派給現有的網路介面](#)
- [將特定字首指派給現有的網路介面](#)

### 將自動字首指派給現有的網路介面

您可以使用下列方法之一向現有網路介面指派自動字首。

#### Console

##### 將自動字首指派給現有網路介面

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取要指派字首的網路介面，然後選取 Actions (動作)、Manage prefixes (管理字首)。
4. 若要自動指派 IPv4 字首，對於 IPv4 prefix delegation (IPv4 字首委派)，選擇 Auto-assign (自動指派)。然後對於 Number of IPv4 prefixes (IPv4 字首的數目)，指定要指派的字首數目。
5. 若要自動指派 IPv6 字首，對於 IPv6 prefix delegation (IPv6 字首委派)，選擇 Auto-assign (自動指派)。然後對於 Number of IPv6 prefixes (IPv6 字首的數目)，指定要指派的字首數目。

#### Note

IPv6 prefix delegation (IPv6 字首委派) 僅在為 IPv6 啟用選定的子網才會出現。

6. 選擇儲存。

#### AWS CLI

您可以使用 [assign-ipv6-addresses](#) 命令來指派 IPv6 字首，並使用 [assign-private-ip-addresses](#) 命令將 IPv4 字首指派給現有的網路介面。

##### 將自動 IPv4 字首指派給現有網路介面

使用指 [assign-private-ip-addresses](#) 令並設定 `--ipv4-prefix-count` 為您要 AWS 指定的前置字元數目。在下列範例中，AWS 指定 1 IPv4 前置詞。

```
$ C:\> aws ec2 assign-private-ip-addresses \
--network-interface-id eni-081fbb4095EXAMPLE \
--ipv4-prefix-count 1
```

#### 範例輸出

```
{
 "NetworkInterfaceId": "eni-081fbb4095EXAMPLE",
 "AssignedIpv4Prefixes": [
 {
 "Ipv4Prefix": "10.0.0.176/28"
 }
]
}
```

#### 將自動 IPv6 字首指派給現有網路介面

使用指定 [-ipv6 位址指令](#)，並設定 `--ipv6-prefix-count` 為您要指派的前置字元數目。AWS 在下列範例中，AWS 指定 1 IPv6 前置詞。

```
$ C:\> aws ec2 assign-ipv6-addresses \
--network-interface-id eni-00d577338cEXAMPLE \
--ipv6-prefix-count 1
```

#### 範例輸出

```
{
 "AssignedIpv6Prefixes": [
 "2600:1f13:fc2:a700:18bb::/80"
],
 "NetworkInterfaceId": "eni-00d577338cEXAMPLE"
}
```

#### 將特定字首指派給現有的網路介面

您可以使用下列方法之一向現有網路介面指派特定字首。

## Console

將特定字首指派給現有的網路介面

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取要指派字首的網路介面，然後選取 Actions (動作)、Manage prefixes (管理字首)。
4. 若要指派特定的 IPv4 字首，對於 IPv4 prefix delegation (IPv4 字首委派)，選擇 Custom (自訂)。然後選擇 Add new prefix (新增字首)，並輸入要使用的字首。
5. 若要指派特定的 IPv6 字首，對於 IPv6 prefix delegation (IPv6 字首委派)，選擇 Custom (自訂)。然後選擇 Add new prefix (新增字首)，並輸入要使用的字首。

### Note

IPv6 prefix delegation (IPv6 字首委派) 僅在為 IPv6 啟用選定的子網才會出現。

6. 選擇儲存。

## AWS CLI

將特定 IPv4 字首指派給現有的網路介面

使用指 [assign-private-ip-addresses](#) 令並設定 `--ipv4-prefixes` 為前置字元。AWS 從此範圍中選取 IPv4 位址。在下列範例中，字首 CIDR 是 `10.0.0.208/28`。

```
$ C:\> aws ec2 assign-private-ip-addresses \
--network-interface-id eni-081fbb4095EXAMPLE \
--ipv4-prefixes 10.0.0.208/28
```

範例輸出

```
{
 "NetworkInterfaceId": "eni-081fbb4095EXAMPLE",
 "AssignedIpv4Prefixes": [
 {
 "Ipv4Prefix": "10.0.0.208/28"
 }
]
}
```

## 將特定 IPv6 字首指派給現有的網路介面

使用 [分配 ipv6 地址](#) 命令並設置為前綴。--ipv6-prefixes AWS 從此範圍中選取 IPv6 位址。在下列範例中，字首 CIDR 是 2600:1f13:fc2:a700:18bb::/80。

```
$ C:\> aws ec2 assign-ipv6-addresses \
--network-interface-id eni-00d577338cEXAMPLE \
--ipv6-prefixes 2600:1f13:fc2:a700:18bb::/80
```

## 範例輸出

```
{
 "NetworkInterfaceId": "eni-00d577338cEXAMPLE",
 "AssignedIpv6Prefixes": [
 {
 "Ipv6Prefix": "2600:1f13:fc2:a700:18bb::/80"
 }
]
}
```

## 使用字首設定網路介面的作業系統

Amazon Linux AMI 可能包含由安裝的其他指令碼 AWS，稱為 ec2-net-utils。這些指令碼會選擇性自動化您網路介面的組態。這些指令碼僅供 Amazon Linux 使用。

如果您不是使用 Amazon Linux，則可以使用容器網路介面 (CNI) 作為 Kubernetes 外掛程式，或 dockerd (如果您使用 Docker 來管理容器)。

## 檢視被指派給網路介面的字首

您可以使用下列方法之一檢視被指派給網路介面的字首。

### Console

#### 檢視被指派給現有網路介面的自動字首

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取要檢視字首的網路介面，並選取 Details (詳細資訊) 索引標籤。

4. IPv4 Prefix Delegation (IPv4 字首委派) 欄位會列出指派的 IPv4 字首，而 IPv6 Prefix Delegation (IPv6 字首委派) 欄位會列出指派的 IPv6 字首。

## AWS CLI

您可以使用指 [describe-network-interfaces](#) AWS CLI 令來檢視指派給網路介面的首碼。

```
$ C:\> aws ec2 describe-network-interfaces
```

### 範例輸出

```
{
 "NetworkInterfaces": [
 {
 "AvailabilityZone": "us-west-2a",
 "Description": "IPv4 automatic example",
 "Groups": [
 {
 "GroupName": "default",
 "GroupId": "sg-044c2de2c4EXAMPLE"
 }
],
 "InterfaceType": "interface",
 "Ipv6Addresses": [],
 "MacAddress": "02:98:65:dd:18:47",
 "NetworkInterfaceId": "eni-02b80b4668EXAMPLE",
 "OwnerId": "123456789012",
 "PrivateIpAddress": "10.0.0.62",
 "PrivateIpAddresses": [
 {
 "Primary": true,
 "PrivateIpAddress": "10.0.0.62"
 }
],
 "Ipv4Prefixes": [
 {
 "Ipv4Prefix": "10.0.0.208/28"
 }
],
 "Ipv6Prefixes": [],
 "RequesterId": "AIDAIV5AJI5LXF5XXDPC0",
 "RequesterManaged": false,
 }
]
}
```

```
 "SourceDestCheck": true,
 "Status": "available",
 "SubnetId": "subnet-05eef9fb78EXAMPLE",
 "TagSet": [],
 "VpcId": "vpc-0e12f52b2146bf252"
 },
 {
 "AvailabilityZone": "us-west-2a",
 "Description": "IPv6 automatic example",
 "Groups": [
 {
 "GroupName": "default",
 "GroupId": "sg-044c2de2c411c91b5"
 }
],
 "InterfaceType": "interface",
 "Ipv6Addresses": [],
 "MacAddress": "02:bb:e4:31:fe:09",
 "NetworkInterfaceId": "eni-006edbcfa4EXAMPLE",
 "OwnerId": "123456789012",
 "PrivateIpAddress": "10.0.0.73",
 "PrivateIpAddresses": [
 {
 "Primary": true,
 "PrivateIpAddress": "10.0.0.73"
 }
],
 "Ipv4Prefixes": [],
 "Ipv6Prefixes": [
 {
 "Ipv6Prefix": "2600:1f13:fc2:a700:1768::/80"
 }
],
 "RequesterId": "AIDAIV5AJI5LXF5XXDPC0",
 "RequesterManaged": false,
 "SourceDestCheck": true,
 "Status": "available",
 "SubnetId": "subnet-05eef9fb78EXAMPLE",
 "TagSet": [],
 "VpcId": "vpc-0e12f52b21EXAMPLE"
 }
]
}
```



## 從網路介面移除字首

您可以使用下列方法之一從您的網路介面移除字首。

### Console

#### 從網路介面移除字首

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取要從中移除字首的網路介面，然後選取 Actions (動作)、Manage prefixes (管理字首)。
4. 執行以下任意一項：
  - 若要移除所有指定的字首，對於 IPv4 prefix delegation (IPv4 字首委派) 和 IPv6 prefix delegation (IPv6 字首委派)，選擇 Do not assign (不指派)。
  - 如要移除特定指定的字首，對於 IPv4 prefix delegation (IPv4 字首委派) 或 IPv6 prefix delegation (IPv6 字首委派)，選擇 Custom (自訂)，然後選擇要移除的字首旁邊的 Unassign (取消指派)。

#### Note

IPv6 prefix delegation (IPv6 字首委派) 僅在為 IPv6 啟用選定的子網才會出現。

5. 選擇儲存。

### AWS CLI

您可以使用 [unassign-ipv6-addresses](#) 命令來移除 IPv6 字首，並使用 [unassign-private-ip-address](#) 命令，從現有的網路介面移除 IPv4 字首。

#### 從網路介面移除 IPv4 字首

使用 [unassign-private-ip-address](#) 命令，並將 `--ipv4-prefix` 設定為您要移除的位址。

```
$ C:\> aws ec2 unassign-private-ip-addresses \
--network-interface-id eni-081fbb4095EXAMPLE \
--ipv4-prefixes 10.0.0.176/28
```

## 從網路介面移除 IPv6 字首

使用 [unassign-ipv6-addresses](#) 命令，並將 `--ipv6-prefix` 設定為您要移除的位址。

```
$ C:\> aws ec2 unassign-ipv6-addresses \
--network-interface-id eni-00d577338cEXAMPLE \
--ipv6-prefix 2600:1f13:fc2:a700:18bb::/80
```

## Amazon EC2 執行個體網路頻寬

執行個體頻寬規格適用於執行個體的傳入和傳出流量。舉例來說，如果某個執行個體指定的頻寬最多為 10 Gbps，這意味著傳入流量最高可達 10 Gbps 的頻寬，而傳出流量最高可達 10 Gbps 的頻寬。EC2 執行個體可用的網路頻寬取決於數項因素，如下所示。

### 多流流量

執行個體可用的彙總多流流量頻寬取決於流量的目標。

- 區域內 – 流量可以利用執行個體可用的完整網路頻寬。
- 到其他區域、網際網路閘道、Direct Connect 或本機閘道 (LGW) – 流量可以利用高達 50% 的最新一代執行個體可用的網路頻寬，最少可使用 32 個 vCPU。具有小於 32 個 vCPU 的最新一代執行個體的頻寬限制為 5 Gbps。

### 單一流量

當執行個體不在同一個[集群放置群組](#)中時，單一流量的流量基準頻寬限制為 5 Gbps。若要降低延遲並增加單一流量頻寬，請嘗試下列其中一項：

- 使用集群放置群組可為相同置放群組中的執行個體實現最高 10 Gbps 的頻寬。
- 在兩個端點之間設定多個路徑，以使用多重路徑 TCP (MPTCP) 達到更高的頻寬。
- 針對相同子網路中的合格執行個體設定 ENA Express，以在這些執行個體之間達到最高 25 Gbps 的頻寬。

## 可用執行個體頻寬

執行個體的可用網路頻寬取決於其擁有的 vCPU 數量。例如，m5.8xlarge 執行個體具有 32 個 vCPU 和 10 Gbps 的網路頻寬，而 m5.16xlarge 執行個體具有 64 個 vCPU 和 20 Gbps 的網路頻寬。然而，如果執行個體超過執行個體層級的網路限額 (例如每秒封包數或追蹤連線數)，則執行個體可能無法

達到此頻寬。流量可以利用的可用頻寬大小取決於 vCPU 的數目和目標。例如，m5.16xlarge 執行個體具有 64 個 vCPU，因此區域中到另一個執行個體的流量可以利用完整的可用頻寬 (20 Gbps)。但是，到不同區域中另一個執行個體的流量只能使用 50% 的可用頻寬 (10 Gbps)。

通常，具有 16 個或更少 vCPU 的執行個體 (大小為 4xlarge 和更小) 會被記錄為具有「高達」指定的頻寬。例如，「高達 10 Gbps」。這些執行個體具有基準頻寬。若要滿足其他需求，他們可使用網路輸入/輸出額度機制，以實現超越基準頻寬的高載頻寬。執行個體可以在有限的時間內使用高載頻寬 (通常是 5 到 60 分鐘)，具體取決於執行個體大小。

執行個體在啟動時會收到最大的網路輸入/輸出額度數。如果執行個體耗盡其網路輸入/輸出額度，則會返回其基準頻寬。每當執行中的執行個體使用的網路頻寬比其基準頻寬少，就會獲得網路輸入/輸出額度。已停止的執行個體不會獲得網路輸入/輸出額度。由於高載頻寬是共用資源，所以即使執行個體具有可用額度，執行個體高載仍只會是盡量實現。

傳入和傳出流量有單獨的網路 I/O 額度儲存貯體。

## 基本和高載網路效能

Amazon EC2 執行個體類型指南說明每種執行個體類型的網路效能，以及可使用高載頻寬的執行個體可用的基準網路頻寬。如需詳細資訊，請參閱下列內容：

- [網路規格 — 一般用途](#)
- [網路規格 — 運算最佳化](#)
- [網路規格 — 記憶體最佳化](#)
- [網路規格 — 儲存最佳化](#)
- [網路規格 — 加速運算](#)
- [網路規格 — 高效能運算](#)
- [網路規格 — 前一代](#)

若要檢視網路效能，請使用 AWS CLI

您可以使用[描述執行個體類型 AWS CLI 指令來顯示執行個體類型](#)的相關資訊。下列範例會顯示所有 C5 執行個體的網路效能資訊。

```
aws ec2 describe-instance-types --filters "Name=instance-type,Values=c5.*"
--query "InstanceTypes[][InstanceType, NetworkInfo.NetworkPerformance,
NetworkInfo.NetworkCards[0].BaselineBandwidthInGbps]" --output table

| DescribeInstanceTypes |
```

c5.4xlarge	Up to 10 Gigabit	5.0
c5.xlarge	Up to 10 Gigabit	1.25
c5.12xlarge	12 Gigabit	12.0
c5.24xlarge	25 Gigabit	25.0
c5.metal	25 Gigabit	25.0
c5.9xlarge	12 Gigabit	12.0
c5.2xlarge	Up to 10 Gigabit	2.5
c5.large	Up to 10 Gigabit	0.75
c5.18xlarge	25 Gigabit	25.0

## 監控執行個體頻寬

您可以使用 CloudWatch 指標來監控執行個體網路頻寬以及傳送和接收的封包。當流量超過 Amazon EC2 在執行個體層級定義的網路允許時，可以使用彈性網路轉接器 (ENA) 驅動程式提供的網路效能指標進行監控。

您可以設定 Amazon EC2 是 CloudWatch 使用一分鐘期間還是五分鐘期間將執行個體的指標資料傳送到。網路效能測量結果可能會顯示超過限額且封包已被捨棄，而 CloudWatch 執行個體指標則沒有。當執行個體對網路資源的需求 (稱為微突發) 短暫激增時，可能會發生這種情況，但 CloudWatch 指標的精細度不足以反映這些微秒尖峰。

進一步了解

- [執行個體指標](#)
- [網路效能指標](#)

## Amazon EC2 上的增強型聯網

增強型聯網使用單一根目錄 I/O 虛擬化 (SR-IOV) 在[支援的執行個體類型](#)上提供高效能聯網功能。SR-IOV 是一種相較於傳統虛擬網路介面可提高 I/O 效能及降低 CPU 使用率的裝置虛擬化方式。增強聯網提供更高的頻寬、更高的每秒封包數 (PPS) 效能，以及一致較低的執行個體間延遲。使用增強型聯網無需額外收費。

如需每種執行個體類型的支援網路速度資訊，請參閱 [Amazon EC2 執行個體類型](#)。

內容

- [增強的網路支援](#)
- [使用 EC2 執行個體上的彈性網路介面卡 \(ENA\) 啟用增強型聯網](#)

- [在 EC2 執行個體上使用 ENA 快速提升網路效能](#)
- [使用 EC2 執行個體上的英特爾 82599 VF 界面啟用增強型聯網](#)
- [監控 EC2 執行個體的網路效能](#)
- [疑難排解 Linux 上的彈性網路介面卡](#)
- [排解彈性網路介面卡 Windows 驅動程式](#)
- [改善基於 Linux 的 Amazon EC2 執行個體的網路延遲](#)
- [對於性能調整的硝基系統考量](#)
- [最佳化 Windows 執行個體的網路效能](#)

## 增強的網路支援

除了 T2 執行個體外，所有目前世代的執行個體類型都支援增強型網路功能。

您可以使用下列其中一種機制來啟用增強型網路：

### 彈性網路轉接器 (ENA)

彈性網路轉接器 (ENA) 在支援的執行個體類型上支援高達 100 Gbps 的網路速度。

在 [AWS Nitro 系統上建置的所有執行個體](#) 都使用 ENA 進行增強型聯網。此外，下列 Xen 執行個體類型支援 ENA：H1、G3、m4.16xlarge、P2、P3、P3dn 和 R4。

如需詳細資訊，請參閱 [使用 EC2 執行個體上的彈性網路介面卡 \(ENA\) 啟用增強型聯網](#)。

### Intel 82599 虛擬功能 (VF) 介面

Intel 82599 虛擬功能介面在支援的執行個體類型上支援高達 10 Gbps 的網路速度。

下列執行個體類型使用 Intel 82599 VF 介面來增強網路功能：C3、C4、D2、I2、M4 (不包括 m4.16xlarge) 和 R3。

如需詳細資訊，請參閱 [使用 EC2 執行個體上的英特爾 82599 VF 界面啟用增強型聯網](#)。

## 使用 EC2 執行個體上的彈性網路介面卡 (ENA) 啟用增強型聯網

Amazon EC2 會透過彈性網路轉接器 (ENA) 提供增強型聯網功能。若要使用增強型聯網，您必須安裝必要的 ENA 模組，並啟用 ENA 支援。

### 目錄

- [要求](#)
- [增強型聯網：效能](#)
- [具有所需模組的 Linux AMI](#)
- [測試是否已啟用增強型網路](#)
- [在執行個體上啟用增強型網路](#)
- [驅動程式版本備註](#)

## 要求

若要準備使用 ENA 介面的增強型聯網，請依下列方式設定您的執行個體：

- 啟動在[AWS 硝基系統上構建的實例](#)。
- 確定該執行個體具有網際網路連線能力。
- 如果您的執行個體上有想要保留的重要資料，您現在應從執行個體建立 AMI，以備份這些資料。更新核心與核心模組，以及啟用 enaSupport 屬性，可能使執行個體變得不相容或使作業系統變得無法連線。如果您有較新的備份，發生這些狀況時資料便能獲得保留。
- Linux 執行個體 — 使用支援的 Linux 核心版本和支援的發行版本啟動執行個體，以便自動為您的執行個體啟用 ENA 增強型聯網。如需詳細資訊，請參閱 [ENA Linux 核心驅動程式版本備註](#)。
- 視窗執行個體 — 如果執行個體執行的是 Windows 伺服器 2008 R2 SP1，請確定具有 [SHA-2 程式碼簽章支援](#)更新。
- 在您選擇的任何電腦 [AWS Tools for Windows PowerShell](#) 上使用 [AWS CloudShell](#) AWS Management Console，[AWS CLI](#) 或安裝並設定或，最好是您的本機桌上型電腦或筆記型電腦。如需詳細資訊，請參閱 [存取 Amazon EC2](#) 和 [AWS CloudShell 使用者指南](#)。增強型聯網無法從 Amazon EC2 主控台管理。

## 增強型聯網：效能

下列文件提供支援 ENA 增強型聯網之執行個體類型的網路效能摘要：

- [加速運算執行個體的網路規格](#)
- [運算優化執行個體的網路規格](#)
- [一般用途執行個體的網路規格](#)
- [高效能運算執行個體網路規格](#)
- [記憶體最佳化執行個體的網路](#)

- [儲存優化執行個體的網路規格](#)

## 具有所需模組的 Linux AMI

下列 AMI 包含必要的 ENA 模組，並啟用 ENA 支援：

- AL2023
- Amazon Linux 2
- Amazon Linux AMI 2018.03 及更新版本
- Ubuntu 14.04 或更新版本 (帶 linux-aws 核心)

### Note

AWS 以重力為基礎的執行個體類型需要 Ubuntu 18.04 或更新版本搭配核心 linux-aws

- Red Hat Enterprise Linux 7.4 或更新版本
- SUSE Linux Enterprise Server 12 SP2 或更新版本
- CentOS 7.4.1708 或更新版本
- FreeBSD 11.1 或更新版本
- Debian GNU/Linux 9 或更新版本

若要測試增強型聯網是否已啟用，請確認您的執行個體上是否已安裝 ena 模組，並且已設定 enaSupport 屬性。如果是這樣，則命令 `ethtool -i ethn` 應該顯示該模塊在網絡接口上正在使用中。

### 核心模組 (ena)

若要確認 ena 模組已安裝，請使用 `modinfo` 命令如下列範例中顯示。

```
[ec2-user ~]$ modinfo ena
filename: /lib/modules/4.14.33-59.37.amzn2.x86_64/kernel/drivers/amazon/net/ena/
ena.ko
version: 1.5.0g
license: GPL
description: Elastic Network Adapter (ENA)
author: Amazon.com, Inc. or its affiliates
srcversion: 692C7C68B8A9001CB3F31D0
alias: pci:v00001D0Fd0000EC21sv*sd*bc*sc*i*
alias: pci:v00001D0Fd0000EC20sv*sd*bc*sc*i*
```

```
alias: pci:v00001D0Fd00001EC2sv*sd*bc*sc*i*
alias: pci:v00001D0Fd00000EC2sv*sd*bc*sc*i*
depends:
retpoline: Y
intree: Y
name: ena
...
```

在 Amazon Linux 執行個體中，已安裝該ena模組。

```
ubuntu:~$ modinfo ena
ERROR: modinfo: could not find module ena
```

在 Ubuntu 實例中，該模塊未安裝，因此您必須先安裝它。如需詳細資訊，請參閱 [Ubuntu](#)。

## 測試是否已啟用增強型網路

您可以測試執行個體或 AMI 是否已啟用增強型聯網。

### 實例屬性

若要查看執行個體是否設定增強型聯網的 `enaSupport` 屬性，請使用下列其中一項命令。如果屬性已設定，回應為 `true`。

- [describe-instances](#) (AWS CLI/AWS CloudShell)

```
aws ec2 describe-instances --instance-ids instance_id --query
"Reservations[].Instances[].EnaSupport"
```

- [Get-EC2Instance](#)(視窗工具 PowerShell)

```
(Get-EC2Instance -InstanceId instance-id).Instances.EnaSupport
```

### 圖片屬性

若要查看 AMI 是否設定增強型聯網的 `enaSupport` 屬性，請使用下列其中一項命令。如果屬性已設定，回應為 `true`。

- [describe-images](#) (AWS CLI/AWS CloudShell)

```
aws ec2 describe-images --image-id ami_id --query "Images[].EnaSupport"
```



- [Get-EC2Image](#)(視窗工具 PowerShell)

```
(Get-EC2Image -ImageId ami_id).EnaSupport
```

## Linux 網路介面驅動程式

請用下列命令確認 ena 模組使用於特定介面，並將介面名稱換成您想要檢查的介面。如果您使用單一介面 (預設)，介面為 eth0。如果作業系統支援[可預測的網路名稱](#)，則名稱可能像是 ens5。

在下列範例中，不會載入 ena 模組，因為列出的驅動程式為 vif。

```
[ec2-user ~]$ ethtool -i eth0
driver: vif
version:
firmware-version:
bus-info: vif-0
supports-statistics: yes
supports-test: no
supports-eeprom-access: no
supports-register-dump: no
supports-priv-flags: no
```

在此範例中，ena 模組已載入且為最低建議版本。此執行個體已正確設定增強型聯網。

```
[ec2-user ~]$ ethtool -i eth0
driver: ena
version: 1.5.0g
firmware-version:
expansion-rom-version:
bus-info: 0000:00:05.0
supports-statistics: yes
supports-test: no
supports-eeprom-access: no
supports-register-dump: no
supports-priv-flags: no
```

## 在執行個體上啟用增強型網路

您使用的程序視執行個體的作業系統而定。

## Amazon Linux

Amazon Linux 2 和最新版的 Amazon Linux AMI 包含已安裝 ENA 之增強型聯網所需的模組，並且已啟用 ENA 支援。因此，如果您在支援的執行個體類型上使用 Amazon Linux HVM 版本啟動執行個體，則您的執行個體已啟用增強型聯網。如需詳細資訊，請參閱 [測試是否已啟用增強型網路](#)。

如果您使用較舊的 Amazon Linux AMI 啟動執行個體，且其尚未啟用增強型聯網，請用下列程序啟用增強型聯網。

在 Amazon Linux AMI 上啟用增強型聯網

1. 連線到您的執行個體。
2. 在執行個體上，執行下列命令，將您的執行個體更新為最新的核心與核心模組，包括 ena：

```
[ec2-user ~]$ sudo yum update
```

3. 從本機電腦中，使用 Amazon EC2 主控台或下列其中一個命令來重新啟動執行個體：[reboot-instances](#) (AWS CLI)、[Restart-EC2Instance](#) (AWS Tools for Windows PowerShell)。
4. 再次連接至執行個體，並使用 ena 中的 `modinfo ena` 命令確認 [測試是否已啟用增強型網路](#) 模組已安裝且為最低建議版本。
5. [EBS 支援的執行個體] 從本機電腦中，使用 Amazon EC2 主控台或下列其中一個命令來停止執行個體：[stop-instances](#) (AWS CLI)、[Stop-EC2Instance](#) (AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中停止執行個體，讓執行個體狀態保持同步。

[執行個體存放區後端執行個體] 您無法停止執行個體來修改屬性。請改為執行此程序：[在 Amazon Linux AMI 上啟用增強型聯網 \(執行個體存放區後端執行個體\)](#)。

6. 在本機電腦上使用下列其中一個命令啟用增強型聯網屬性。

- [modify-instance-attribute](#) (AWS CLI)

```
aws ec2 modify-instance-attribute --instance-id instance_id --ena-support
```

- [Edit-EC2InstanceAttribute](#)(視窗工具 PowerShell)

```
Edit-EC2InstanceAttribute -InstanceId instance-id -EnaSupport $true
```

7. (選擇性) 從執行個體建立 AMI，如 [創建一個 Amazon EBS-backed AMI](#) 所述。AMI 將從執行個體繼承增強型聯網的 `enaSupport` 屬性。因此，您預設可用此 AMI 啟動其他啟用增強型聯網的執行個體。
8. 從本機電腦中，使用 Amazon EC2 主控台或下列其中一個命令來啟動執行個體：[start-instances](#) (AWS CLI)、[Start-EC2Instance](#) (AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中啟動執行個體，讓執行個體狀態保持同步。
9. 連接至執行個體，並使用 `ena` 中的 `ethtool -i ethn` 命令確認 [測試是否已啟用增強型網路](#) 模組已在網路介面上安裝及載入。

如果啟用增強型聯網後無法連接至您的執行個體，請參閱 [疑難排解 Linux 上的彈性網路介面卡](#)。

在 Amazon Linux AMI 上啟用增強型聯網 (執行個體存放區後端執行個體)

遵循先前的程序，一直進行到您停止執行個體的那個步驟。如 [建立執行個體存放區後端 Linux AMI](#) 中所述建立新的 AMI，且務必在註冊 AMI 時啟用增強型聯網屬性。

- [register-image](#) (AWS CLI)

```
aws ec2 register-image --ena-support ...
```

- [Register-EC2Image](#) (AWS Tools for Windows PowerShell)

```
Register-EC2Image -EnaSupport $true ...
```

## Ubuntu

最新的 Ubuntu HVM AMI 包含已安裝 ENA 之增強型聯網所需的模組，並且已啟用 ENA 支援。因此，如果您在支援的執行個體類型上使用最新的 Ubuntu HVM AMI 來啟動執行個體，則您的執行個體已啟用增強型聯網。如需詳細資訊，請參閱 [測試是否已啟用增強型網路](#)。

如果您使用較舊的 AMI 啟動執行個體，且其尚未啟用增強型聯網，您可以安裝 `linux-aws` 核心套件以取得最新的增強型聯網驅動程式，並更新必要的屬性。

安裝 **linux-aws** 核心套件 (Ubuntu 16.04 或更新版本)

Ubuntu 16.04 和 18.04 隨附 Ubuntu 自訂核心 (`linux-aws` 核心套件)。若要使用不同的核心，請聯絡 [AWS Support](#)。

## 安裝 **linux-aws** 核心套件 (Ubuntu Trusty 14.04)

1. 連線到您的 執行個體。
2. 更新套件快取和套件。

```
ubuntu:~$ sudo apt-get update && sudo apt-get upgrade -y linux-aws
```

### Important

如果更新程序期間提示您安裝 grub，請用 `/dev/xvda` 安裝 grub，然後選擇保留目前版本的 `/boot/grub/menu.lst`。

3. [EBS 支援的執行個體] 從本機電腦中，使用 Amazon EC2 主控台或下列其中一個命令來停止執行個體：[stop-instances](#) (AWS CLI)、[Stop-EC2Instance](#) (AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中停止執行個體，讓執行個體狀態保持同步。

[執行個體存放區後端執行個體] 您無法停止執行個體來修改屬性。請改為執行此程序：[在 Ubuntu 上啟用增強型聯網 \(執行個體後端執行個體\)](#)。

4. 在本機電腦上使用下列其中一個命令啟用增強型聯網屬性。
  - [modify-instance-attribute](#) (AWS CLI)

```
aws ec2 modify-instance-attribute --instance-id instance_id --ena-support
```

- [Edit-EC2InstanceAttribute](#)(視窗工具 PowerShell)

```
Edit-EC2InstanceAttribute -InstanceId instance-id -EnaSupport $true
```

5. (選擇性) 從執行個體建立 AMI，如 [創建一個 Amazon EBS-backed AMI](#) 所述。AMI 將從執行個體繼承增強型聯網的 `enaSupport` 屬性。因此，您預設可用此 AMI 啟動其他啟用增強型聯網的執行個體。
6. 從本機電腦中，使用 Amazon EC2 主控台或下列其中一個命令來啟動執行個體：[start-instances](#) (AWS CLI)、[Start-EC2Instance](#) (AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中啟動執行個體，讓執行個體狀態保持同步。

## 在 Ubuntu 上啟用增強型聯網 (執行個體後端執行個體)

遵循先前的程序，一直進行到您停止執行個體的那個步驟。如 [建立執行個體存放區後端 Linux AMI](#) 中所述建立新的 AMI，且務必在註冊 AMI 時啟用增強型聯網屬性。

- [register-image](#) (AWS CLI)

```
aws ec2 register-image --ena-support ...
```

- [Register-EC2Image](#) (AWS Tools for Windows PowerShell)

```
Register-EC2Image -EnaSupport $true ...
```

## 瑞爾、蘇塞、CentOS

適用於 Red Hat Enterprise Linux、SUSE Linux Enterprise Server 以及 CentOS 最新 AMI 包含使用 ENA 之增強型聯網所需的模組，並且已啟用 ENA 支援。因此，如果您在支援的執行個體類型上使用最新的 AMI 來啟動執行個體，則您的執行個體已啟用增強型聯網。如需詳細資訊，請參閱 [測試是否已啟用增強型網路](#)。

以下程序提供在 Amazon Linux AMI 或 Ubuntu 以外的其他 Linux 發行版本上啟用增強型聯網的一般步驟。如需像是詳細命令語法、檔案位置或套件和工具支援等詳細資訊，請參閱 Linux 發行版本的文件。

## 在 Linux 上啟用增強型聯網

1. 連線到您的執行個體。
2. 從 GitHub at 複製執行個體上 ena 模組的原始程式碼 <https://github.com/amzn/amzn-drivers>。(SUSE Linux Enterprise Server 12 SP2 和更新版本預設包含 ENA 2.02，因此您不需要下載和編譯 ENA 驅動程式。若為 SUSE Linux Enterprise Server 12 SP2 和更新版本，您應該提出請求，將您想要的驅動程式版本新增至庫存核心)。

```
git clone https://github.com/amzn/amzn-drivers
```

3. 在您的執行個體上編譯及安裝 ena 模組。這些步驟視 Linux 發行版本而定。如需有關在 RHEL 上編譯模組的詳細資訊，請參閱 [如何在執行 RHEL 的 Amazon EC2 執行個體上安裝最新的 ENS 驅動程式以獲得增強的網路支援？](#)
4. 執行 `sudo depmod` 命令更新模組相依性。
5. 在執行個體上更新 `initramfs`，確定開機時載入新模組。例如，如果您的發行版本支援 `dracut`，您可以使用下列命令。

```
dracut -f -v
```

- 判斷系統是否依預設使用可預測的網路介面名稱。使用 systemd 或 udev 197 或以上版本的系統可重新命名乙太網路裝置，但不保證單一網路介面會命名為 eth0。該行為會造成連線至執行個體時發生問題。如需詳細資訊及查看其他的組態選項，請參閱 freedesktop.org 網站上的 [可預測的網路介面名稱](#)。

- 您可用下列命令在 RPM 系統上檢查 systemd 或 udev 的版本。

```
rpm -qa | grep -e '^systemd-[0-9]\+|\^udev-[0-9]\+'
systemd-208-11.el7_0.2.x86_64
```

在上述的 Red Hat Enterprise Linux 7 範例中，systemd 版本為 208，因此可預測的網路介面名稱已停用。

- 若要停用可預測網路介面名稱，請將 net.ifnames=0 選項新增至 GRUB\_CMDLINE\_LINUX 的 /etc/default/grub 行中。

```
sudo sed -i '/^GRUB_CMDLINE_LINUX/s/\ "$\ net.ifnames=0\ "/' /etc/default/
grub
```

- 重建 grub 組態檔案。

```
sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

- [EBS 支援的執行個體] 從本機電腦中，使用 Amazon EC2 主控台或下列其中一個命令來停止執行個體：[stop-instances](#) (AWS CLI)、[Stop-EC2Instance](#) (AWS Tools for Windows PowerShell)。如果執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中停止執行個體，讓執行個體狀態保持同步。

[執行個體存放區後端執行個體] 您無法停止執行個體來修改屬性。請改為執行此程序：[在 Linux 上啟用增強型聯網 \(執行個體存放區後端執行個體\)](#)。

- 在本機電腦上使用下列其中一個命令啟用增強型聯網的 enaSupport 屬性：

- [modify-instance-attribute](#) (AWS CLI)

```
aws ec2 modify-instance-attribute --instance-id instance_id --ena-support
```

- [Edit-EC2InstanceAttribute](#)(視窗工具 PowerShell)

```
Edit-EC2InstanceAttribute -InstanceId instance-id -EnaSupport $true
```

9. (選擇性) 從執行個體建立 AMI，如 [創建一個 Amazon EBS-backed AMI](#) 所述。AMI 將從執行個體繼承增強型聯網的 `enaSupport` 屬性。因此，您預設可用此 AMI 啟動其他啟用增強型聯網的執行個體。

如果您執行個體的作業系統包含 `/etc/udev/rules.d/70-persistent-net.rules` 檔案，您必須先刪除檔案，然後再建立 AMI。此檔案包含原始執行個體乙太網路卡的 MAC 地址。如果其他執行個體使用此檔案開機，作業系統將無法找到設備，且 `eth0` 可能會失敗，因而造成開機問題。此檔案會在下次開機週期時重新產生，且從 AMI 啟動的任何執行個體將會建立自己的檔案版本。

10. 從本機電腦中，使用 Amazon EC2 主控台或下列其中一個命令來啟動執行個體：[start-instances](#) (AWS CLI)、[Start-EC2Instance](#) (AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中啟動執行個體，讓執行個體狀態保持同步。
11. (選擇性) 連線到執行個體，並確認模組已安裝。

如果啟用增強型聯網後無法連接至您的執行個體，請參閱 [疑難排解 Linux 上的彈性網路介面卡](#)。

在 Linux 上啟用增強型聯網 (執行個體存放區後端執行個體)

遵循先前的程序，一直進行到您停止執行個體的那個步驟。如 [建立執行個體存放區後端 Linux AMI](#) 中所述建立新的 AMI，且務必在註冊 AMI 時啟用增強型聯網屬性。

- [register-image](#) (AWS CLI)

```
aws ec2 register-image --ena-support ...
```

- [Register-EC2Image](#) (AWS Tools for Windows PowerShell)

```
Register-EC2Image -EnaSupport ...
```

## Ubuntu 與 DKMS

此方法僅用於測試和意見回饋用途。不適用於生產部署。如需生產部署，請參閱 [Ubuntu](#)。

**⚠ Important**

使用 DKMS 會使訂閱的支持協議無效。它不應該用於生產部署。

在 Ubuntu 上使用 ENA 啟用增強型聯網 (EBS 後端執行個體)

1. 請遵循 [Ubuntu](#) 中的步驟 1 和 2。
2. 安裝 `build-essential` 套件，以編譯核心模組和 `dkms` 套件，使 `ena` 模組在每次核心更新時重建。

```
ubuntu:~$ sudo apt-get install -y build-essential dkms
```

3. 從 GitHub 位置複製執行個體上 `ena` 模組的原始碼 <https://github.com/amzn/amzn-drivers>。

```
ubuntu:~$ git clone https://github.com/amzn/amzn-drivers
```

4. 將 `amzn-drivers` 套件移至 `/usr/src/` 目錄，讓 DKMS 可以找到套件並在每次核心更新時建置。附加原始碼版本編號 (目前的版本編號可在版本備註中找到) 到目錄名稱。例如，下列範例顯示 1.0.0 版本。

```
ubuntu:~$ sudo mv amzn-drivers /usr/src/amzn-drivers-1.0.0
```

5. 建立包含下列值的 DKMS 組態檔案，並換成您的 `ena` 版本。

建立檔案。

```
ubuntu:~$ sudo touch /usr/src/amzn-drivers-1.0.0/dkms.conf
```

編輯檔案並新增下列值。

```
ubuntu:~$ sudo vim /usr/src/amzn-drivers-1.0.0/dkms.conf
PACKAGE_NAME="ena"
PACKAGE_VERSION="1.0.0"
CLEAN="make -C kernel/linux/ena clean"
MAKE="make -C kernel/linux/ena/ BUILD_KERNEL=${kernelver}"
BUILT_MODULE_NAME[0]="ena"
BUILT_MODULE_LOCATION="kernel/linux/ena"
DEST_MODULE_LOCATION[0]="/updates"
DEST_MODULE_NAME[0]="ena"
```



```
AUTOINSTALL="yes"
```

6. 使用 DKMS 在您的執行個體上新增、建置及安裝 ena 模組。

將模組新增至 DKMS。

```
ubuntu:~$ sudo dkms add -m amzn-drivers -v 1.0.0
```

使用 dkms 命令建置模組。

```
ubuntu:~$ sudo dkms build -m amzn-drivers -v 1.0.0
```

使用 dkms 安裝模組。

```
ubuntu:~$ sudo dkms install -m amzn-drivers -v 1.0.0
```

7. 重建 initramfs，以在開機時載入正確的模組。

```
ubuntu:~$ sudo update-initramfs -u -k all
```

8. 使用 ena 中的 modinfo ena 命令確認 [測試是否已啟用增強型網路](#) 模組已安裝。

```
ubuntu:~$ modinfo ena
filename: /lib/modules/3.13.0-74-generic/updates/dkms/ena.ko
version: 1.0.0
license: GPL
description: Elastic Network Adapter (ENA)
author: Amazon.com, Inc. or its affiliates
srcversion: 9693C876C54CA64AE48F0CA
alias: pci:v00001D0Fd0000EC21sv*sd*bc*sc*i*
alias: pci:v00001D0Fd0000EC20sv*sd*bc*sc*i*
alias: pci:v00001D0Fd00001EC2sv*sd*bc*sc*i*
alias: pci:v00001D0Fd00000EC2sv*sd*bc*sc*i*
depends:
vermagic: 3.13.0-74-generic SMP mod_unload modversions
parm: debug:Debug level (0=none,...,16=all) (int)
parm: push_mode:Descriptor / header push mode (0=automatic,1=disable,3=enable)
 0 - Automatically choose according to device capability (default)
 1 - Don't push anything to device memory
 3 - Push descriptors and header buffer to device memory (int)
parm: enable_wd:Enable keepalive watchdog (0=disable,1=enable,default=1) (int)
```

```
parm: enable_missing_tx_detection:Enable missing Tx completions. (default=1)
 (int)
parm: numa_node_override_array:Numa node override map
 (array of int)
parm: numa_node_override:Enable/Disable numa node override (0=disable)
 (int)
```

## 9. 繼續執行 [Ubuntu](#) 中的步驟 3。

### 啟用 Windows 上的增強型網路

如果您啟動執行個體，且其尚未啟用增強型聯網，您必須在執行個體上下載並安裝必要的網路轉接器驅動程式，然後設定 `enaSupport` 執行個體屬性，以啟動增強型聯網。您只能在支援的執行個體類型上啟用此屬性，且需先安裝 ENA 驅動程式。如需詳細資訊，請參閱 [增強的網路支援](#)。

### 啟用增強型聯網

1. 連線到您的執行個體，並以本機管理員登入。
2. [僅限 Windows 伺服器 2016 年和 2019 年] 安裝驅動程式後，執行下列 EC2Launch PowerShell 指令碼以設定執行個體。

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 -
Schedule
```

3. 在執行個體上，安裝驅動程式，如下所示：
  - a. 將最新的驅動程式 [下載](#) 到執行個體。
  - b. 將 zip 封存檔解壓縮。
  - c. 透過執行 `install.ps1` PowerShell 指令碼安裝驅動程式。

#### Note

如果您收到執行政策錯誤，請將政策設定為 `Unrestricted` (預設設定為 `Restricted` 或 `RemoteSigned`)。在命令列中執行 `Set-ExecutionPolicy -ExecutionPolicy Unrestricted`，然後再次執行指令 `install.ps1` PowerShell 碼。

4. 從本機電腦中，使用 Amazon EC2 主控台或下列其中一個命令來停止執行個體：[stop-instances](#) (AWS CLI/AWS CloudShell)、[Stop-EC2Instance](#) (AWS Tools for Windows PowerShell)。如果執

行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中停止執行個體，讓執行個體狀態保持同步。

5. 在執行個體上啟用 ENA 支援，如下所示：

a. 在本機電腦上，執行下列其中一個命令，查看執行個體上的 EC2 執行個體 ENA 支援屬性。如果此屬性未啟用，輸出會是「[]」或空白。EnaSupport 預設會設定為 false。

- [describe-instances](#) (AWS CLI/AWS CloudShell)

```
aws ec2 describe-instances --instance-ids instance_id --query
"Reservations[].Instances[].EnaSupport"
```

- [Get-EC2Instance](#)(視窗工具 PowerShell)

```
(Get-EC2Instance -InstanceId instance-id).Instances.EnaSupport
```

b. 若要啟用 ENA 支援，請執行下列其中一項命令：

- [modify-instance-attribute](#) (AWS CLI/AWS CloudShell)

```
aws ec2 modify-instance-attribute --instance-id instance_id --ena-support
```

- [Edit-EC2InstanceAttribute](#) (AWS Tools for Windows PowerShell)

```
Edit-EC2InstanceAttribute -InstanceId instance_id -EnaSupport $true
```

若重新啟動執行個體時發生問題，您也可使用下列其中一項命令來停用 ENA 支援。

- [modify-instance-attribute](#) (AWS CLI/AWS CloudShell)

```
aws ec2 modify-instance-attribute --instance-id instance_id --no-ena-support
```

- [Edit-EC2InstanceAttribute](#) (AWS Tools for Windows PowerShell)

```
Edit-EC2InstanceAttribute -InstanceId instance_id -EnaSupport $false
```

c. 如先前所示，使用 true 或 describe-instances，確認屬性已設定為 Get-EC2Instance。您現在應該會看到下列輸出：

```

 true
]

```

6. 從本機電腦中，使用 Amazon EC2 主控台或下列其中一個命令來啟動執行個體：[start-instances](#) (AWS CLI/AWS CloudShell)、[Start-EC2Instance](#) (AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該使用 AWS OpsWorks 主控台啟動執行個體，讓執行個體狀態保持同步。
7. 在執行個體上，確認 ENA 驅動程式已安裝並啟用，如下所示：
  - a. 用滑鼠右鍵按一下網路圖示，然後選擇 Open Network and Sharing Center (開啟網路和共用中心)。
  - b. 選擇乙太網路卡 (例如，Ethernet 2)。
  - c. 請選擇 Details (詳細資訊)。針對 Network Connection Details (網路連線詳細資料)，檢查 Description (描述) 是否為 Amazon Elastic Network Adapter (Amazon 彈性網路轉接器)。
8. (選擇性) 從執行個體建立 AMI。AMI 將從執行個體繼承 enaSupport 屬性。因此，您預設可用此 AMI 啟動其他啟用 ENA 的執行個體。

## 驅動程式版本備註

### 驅動程式

如需 Linux ENA 驅動程式版本的相關資訊，請參閱 [ENA Linux 核心驅動程式版本備註](#)。

### 視窗 ENA 驅動

Windows AMI 包含 Amazon ENA 驅動程式，用於啟用增強型網路。

下表說明每個 Windows Server 版本的相應下載 ENA 驅動程式版本。

Windows Server 版本	ENA 驅動程式版本
Windows Server 2022	2.4.0 和更新版本
Windows Server 2019	最新
Windows Server 2016	最新
Windows Server 2012 R2	2.6.0 和更舊版本

Windows Server 版本	ENA 驅動程式版本
Windows Server 2012	2.6.0 和更舊版本
Windows Server 2008 R2	2.2.3 和舊版

下表摘要說明各版本的變更。

驅動程式版本	詳細資訊	發行日期
<a href="#">2.7.0</a>	<p>新功能</p> <ul style="list-style-type: none"> <li>刪除了對視窗服務器 2012 ( 視窗 8 ) 和視窗服務器 2012 R2 ( 視窗 8.1 ) 的支持。這些作業系統版本已經到來的支援結束 AWS。在 Windows 伺服器 2012 及更早版本上，驅動程式安裝將失敗。</li> <li>增加了對將 IPv6 Tx 校驗和計算卸載到設備的支持。</li> <li>增加了廣泛的低延遲隊列 ( LLQ ) 支持。這是根據裝置建議動態啟用的。您可以使用新的「WidellQ」登錄機碼覆寫此設定。</li> <li>已新增 Rx 溢位造成封包丟棄的報告，表示 Rx 環中傳入封包的空間不足。</li> <li>增加了對來自設備的次優配置通知的支持。請參閱 Windows 事件檢視器 59000 中的事件識別碼。</li> </ul> <p>錯誤修正</p> <ul style="list-style-type: none"> <li>避免標頭超過最大低延遲佇列 (LLQ) 標頭大小的 Tx 封包造成不必要的裝置重設。</li> </ul>	2024年5月1日

驅動程式版本	詳細資訊	發行日期
<a href="#">2.6.0</a>	<p>新功能</p> <ul style="list-style-type: none"><li>• 為支援 ENA Express 的執行個體類型新增下列網路效能指標。<ul style="list-style-type: none"><li>• ena_srd_mode</li><li>• ena_srd_tx_pkts</li><li>• ena_srd_eligible_tx_pkts</li><li>• ena_srd_rx_pkts</li><li>• ena_srd_resource_utilization</li></ul></li><li>• 為 Nitro 型執行個體類型新增 <code>contrack_allowance_available</code> 網路效能指標。</li><li>• 新增由於偵測到 RX 資料損毀而重設新介面卡的原因。</li><li>• 更新驅動程式記錄的基礎設施。</li></ul> <p>錯誤修正</p> <ul style="list-style-type: none"><li>• 防止介面卡在 CPU 耗盡導致網路效能指標更新為失敗的情況下進行重設。</li><li>• 防止對裝置心跳的中斷狀況進行錯誤偵測。</li><li>• 修復驅動程式安裝指令碼以支援進行降級操作。</li><li>• 修復接收錯誤計數統計的問題。</li></ul>	2023 年 6 月 20 日

驅動程式版本	詳細資訊	發行日期
2.5.0	<p data-bbox="402 254 472 289">公告</p> <p data-bbox="402 338 1198 464">由於無法在 Windows 域控制器上初始化，ENA Windows 驅動程式 2.5.0 版已復原。Windows Client 和 Windows Server 不受影響。</p>	2023 年 2 月 17 日
<a href="#">2.4.0</a>	<p data-bbox="402 512 496 548">新功能</p> <ul data-bbox="402 596 1170 884" style="list-style-type: none"><li data-bbox="402 625 967 661">• 新增對 Windows Server 2022 的支援。</li><li data-bbox="402 709 1016 745">• 移除對 Windows Server 2008 R2 的支援。</li><li data-bbox="402 793 1170 884">• 將低延遲佇列 (LLQ) 設定為一律開啟，以改善第六代 Amazon EC2 執行個體的效能。</li></ul> <p data-bbox="402 995 529 1031">錯誤修正</p> <ul data-bbox="402 1079 1198 1409" style="list-style-type: none"><li data-bbox="402 1108 1122 1186">• 修復將網路效能指標發佈到 Windows 效能計數器 (PCW) 系統的故障問題。</li><li data-bbox="402 1234 1084 1270">• 修復登錄機碼讀取操作期間的記憶體流失問題。</li><li data-bbox="402 1318 1198 1409">• 防止在轉接器重設過程中發生無法復原的錯誤時，出現無限重設迴圈。</li></ul>	2022 年 4 月 28 日

驅動程式版本	詳細資訊	發行日期
2.2.4	<p data-bbox="407 254 472 289">公告</p> <p data-bbox="401 333 1192 464">因第六代 EC2 執行個體的潛在效能下降，ENA Windows 驅動程式 2.2.4 版已回復。我們建議您使用下列其中一個方法，降級您的驅動程式：</p> <ul data-bbox="401 520 1130 716" style="list-style-type: none"><li data-bbox="401 520 561 577">• 安裝舊版<ol data-bbox="433 621 1130 716" style="list-style-type: none"><li data-bbox="433 621 1130 657">1. 從此表格中的連結下載舊版套件 (版本 2.2.3)。</li><li data-bbox="433 678 1130 716">2. 執行 <code>install.ps1</code> PowerShell 裝指令碼。</li></ol></li></ul> <p data-bbox="433 821 1141 905">如需有關安裝前和安裝後步驟的詳情，請參閱 <a href="#">啟用 Windows 上的增強型網路</a>。</p> <p data-bbox="433 947 1157 982">使用 Amazon EC2 Systems Manager 進行大量更新</p> <ul data-bbox="433 1031 1157 1226" style="list-style-type: none"><li data-bbox="433 1031 1157 1226">• 透過 SSM 文件 <code>AWS-ConfigureAWSPackage</code> 執行大量更新，參數如下：<ul data-bbox="498 1136 951 1226" style="list-style-type: none"><li data-bbox="498 1136 951 1171">• 名稱：AwsEnaNetworkDriver</li><li data-bbox="498 1192 695 1226">• 版本：2.2.3</li></ul></li></ul>	2021 年 10 月 26 日



驅動程式版本	詳細資訊	發行日期
<a href="#">2.2.3</a>	<p>新功能</p> <ul style="list-style-type: none"><li>• 新增對具有最高 400 Gbps 執行個體聯網之新 Nitro 卡的支援。</li></ul> <p>錯誤修正</p> <ul style="list-style-type: none"><li>• 修正 ENA 驅動程式的系統時間變更與系統時間查詢之間的競爭條件，這會導致硬體無回應的誤判偵測。</li></ul> <p>Windows ENA 驅動程式 2.2.3 版是支援 Windows Server 2008 R2 的最後一個版本。Windows Server 2008 R2 會繼續支援使用 ENA 的目前可用執行個體類型，且驅動程式可供下載。未來執行個體類型不會支援 Windows Server 2008 R2，而且您無法將 Windows Server 2008 R2 啟動、匯入或遷移到未來執行個體類型。</p>	2021 年 3 月 25 日
<a href="#">2.2.2</a>	<p>新功能</p> <ul style="list-style-type: none"><li>• 添加對查詢網路介面卡效能度量的支援，以 CloudWatch 及適用於 Windows 消費者的效能計數器。</li></ul> <p>錯誤修正</p> <ul style="list-style-type: none"><li>• 修正裸機執行個體的效能問題。</li></ul>	2020 年 12 月 21 日
<a href="#">2.2.1</a>	<p>新功能</p> <ul style="list-style-type: none"><li>• 新增方法以允許主機查詢彈性網路轉接器來取得網路效能指標。</li></ul>	2020 年 10 月 1 日

驅動程式版本	詳細資訊	發行日期
<a href="#">2.2.0</a>	<p>新功能</p> <ul style="list-style-type: none"> <li>• 新增對下一代硬體類型的支援。</li> <li>• 改善從停止休眠恢復之後的執行個體啟動時間，並消除誤判的 ENA 錯誤訊息。</li> </ul> <p>效能最佳化</p> <ul style="list-style-type: none"> <li>• 最佳化傳入流量的處理。</li> <li>• 改善低資源環境中的共用記憶體管理。</li> </ul> <p>錯誤修正</p> <ul style="list-style-type: none"> <li>• 避免在極少數情況下，驅動程式無法重設的 ENA 裝置移除時系統當機。</li> </ul>	2020 年 8 月 12 日
<a href="#">2.1.5</a>	<p>錯誤修正</p> <ul style="list-style-type: none"> <li>• 修正裸機執行個體上偶發性的網路介面卡初始化失敗。</li> </ul>	2020 年 6 月 23 日
<a href="#">2.1.4</a>	<p>錯誤修正</p> <ul style="list-style-type: none"> <li>• 預防因為來自網路堆疊的 LSO 封包中繼資料毀損而造成的連線問題。</li> <li>• 預防因為存取已釋放封包記憶體造成罕見競爭條件情況，而導致系統當機。</li> </ul>	2019 年 11 月 25 日

驅動程式版本	詳細資訊	發行日期
<a href="#">2.1.2</a>	<p>新功能</p> <ul style="list-style-type: none"><li>將支援新增至開發商 ID 報告以允許作業系統產生以 MAC 為基礎的 UUIDs。</li></ul> <p>錯誤修正</p> <ul style="list-style-type: none"><li>在初始化期間改善 DHCP 網路組態效能。</li><li>當最大傳輸單元 (MTU) 超過 4K 時，正確計算輸入 IPv6 流量的 L4 檢查總和。</li><li>一般性改進增加穩定性和次要錯誤修正。</li></ul>	2019 年 11 月 4 日
<a href="#">2.1.1</a>	<p>錯誤修正</p> <ul style="list-style-type: none"><li>防止從作業系統抵達的高度分散式 TCP LSO 封包卸下。</li><li>適當處理 IPv6 網路中 IPSec 內的封裝安全承載 (ESP) 通訊協定。</li></ul>	2019 年 9 月 16 日

驅動程式版本	詳細資訊	發行日期
<a href="#">2.1.0</a>	<p>ENA Windows 驅動程式 2.1 版推出新的 ENA 裝置功能、提供效能加速、加入新功能，並包含多個穩定性改善。</p> <ul style="list-style-type: none"><li>• 新功能<ul style="list-style-type: none"><li>• 對巨型訊框組態使用標準化的 Windows 登錄機碼。</li><li>• 允許透過 ENA 驅動程式屬性 GUI 進行 VLAN ID 設定。</li></ul></li><li>• 改善復原流程<ul style="list-style-type: none"><li>• 改善失敗識別機制。</li><li>• 已新增對可調整復原參數的支援。</li></ul></li><li>• 對具有 8 個以上 vCPU 的新 EC2 執行個體支援最多 32 個 I/O 佇列。</li><li>• 驅動程式記憶體使用量減少最高達 90%。</li><li>• 效能最佳化<ul style="list-style-type: none"><li>• 減少傳輸路徑延遲。</li><li>• 支援接收檢查總和卸載。</li><li>• 對負載繁重系統的效能最佳化 (鎖定機制的最佳化使用量)。</li><li>• 進一步的增強可減少 CPU 使用量，並改善系統在負載下的回應能力。</li></ul></li><li>• 錯誤修正<ul style="list-style-type: none"><li>• 修正由於非接續 Tx 標頭無效的剖析造成的損毀。</li></ul></li></ul>	2019 年 7 月 1 日

驅動程式版本	詳細資訊	發行日期
	<ul style="list-style-type: none"> <li>• 修正裸機執行個體上彈性網絡介面分離期間的驅動程式 1.5 版損毀。</li> <li>• 修正透過 IPv6 的 LSO 虛擬標頭檢查總和計算錯誤。</li> <li>• 修正初始化失敗時可能的記憶體資源流失。</li> <li>• 為 IPv4 區段停用 TCP/UDP 檢查總和卸載。</li> <li>• VLAN 組態的修正。在只應停用 VLAN 優先順序時，不正確地停用了 VLAN。</li> <li>• 啟用事件檢視器對自訂驅動程式訊息的正確剖析。</li> <li>• 由於無效的時間戳記處理，修正無法初始化驅動程式。</li> <li>• 修正資料處理與 ENA 裝置停用之間的競爭條件。</li> </ul>	
<a href="#">1.5.0</a>	<ul style="list-style-type: none"> <li>• 已提升穩定性並修正效能。</li> <li>• 在 ENA NIC 的 Advanced Properties (進階屬性) 中，您現在可以將 Receive Buffers (接收緩衝區) 的最高值設定為 8192。</li> <li>• 預設 Receive Buffers (接收緩衝區) 為 1,000。</li> </ul>	2018 年 10 月 4 日
<a href="#">1.2.3</a>	包含從 Windows Server 2008 R2 到 Windows Server 2016 的可靠性修正並統一支援。	2018 年 2 月 13 日
<a href="#">1.0.8</a>	初始版本。包含適用於 Windows Server 2008 R2、Windows Server 2012 RTM、Windows Server 2012 R2 和 Windows Server 2016 的 AMI。	2016 年 7 月

Amazon SNS 會在 EC2 Windows 驅動程式發佈新版本時通知您。使用下列程序訂閱這些通知。

### 訂閱 EC2 通知

1. 在 <https://console.aws.amazon.com/sns/v3/home> 開啟 Amazon SNS 主控台。
2. 如有必要，請在導覽列中將「區域」變更為美國東部 (維吉尼亞北部)。您必須選取此區域，因為您要訂閱的 SNS 通知位於本區域中。
3. 在導覽窗格中，選擇 Subscriptions (訂閱)。
4. 選擇 Create subscription (建立訂閱)。
5. 在 Create subscription (建立訂閱) 對話方塊中，執行下列動作：
  - a. 針對 TopicARN，複製下列 Amazon Resource Name (ARN)：  
`arn:aws:sns:us-east-1:801119661308:ec2-windows-drivers`
  - b. 針對 protocol (通訊協定)，選擇 Email。
  - c. 針對 Endpoint (端點)，請輸入可用於接收通知的電子郵件地址。
  - d. 選擇 Create subscription (建立訂閱)。
6. 您將會收到確認電子郵件。開啟電子郵件並遵循指示完成訂閱。

只要有新的 EC2 Windows 驅動程式發行，我們就會向訂閱者傳送通知。如果您不想再接收這些通知，請使用下列程序來取消訂閱。

### 取消訂閱 Amazon EC2 Windows 驅動程式通知

1. 在 <https://console.aws.amazon.com/sns/v3/home> 開啟 Amazon SNS 主控台。
2. 在導覽窗格中，選擇 Subscriptions (訂閱)。
3. 選取訂閱的核取方塊，然後選取 Actions (動作)，Delete subscriptions (刪除訂閱)。出現確認提示時，請選擇 Delete (刪除)。

## 在 EC2 執行個體上使用 ENA 快速提升網路效能

ENA Express 由可 AWS 擴展的可靠數據報 (SRD) 技術提供支持。SRD 是一種高效能的網路傳輸通訊協定，其使用動態路由來增加輸送量並將網路流量尾延遲降至最低。使用 ENA Express，您可以在相同可用區域中的兩個 EC2 執行個體之間進行通訊。

## ENA Express 的優勢

- 將單一流程在子網路中可使用的最高頻寬從 5 Gbps 增加到 25 Gbps，最高可達彙總執行個體限制。
- 減少 EC2 執行個體之間網路流量的尾延遲，特別是在高網路負載期間。
- 偵測並避免擁塞的網路路徑。
- 直接在網路層中處理某些任務，例如接收端的封包重新排序，以及大多數需要的重新傳輸。這會釋放應用程式層進行其他工作。

### Note

如果應用程式每秒傳送或接收大量封包，而且大部分時間需要最佳化延遲，尤其是在網路上沒有擁塞的期間，則 [增強型網路](#) 可能更適合您的網路。

在網路流量較輕的期間，當封包使用 ENA Express 時，您可能會注意到封包延遲會略有增加 (數十微秒)。在這段時間內，優先考慮特定網路效能特性的應用程式可以從 ENA Express 中受益，如下所示：

- 在相同可用區域內，將單一流量頻寬從 5 Gbps 增加到 25 Gbps，最高可達彙總執行個體限制，程序可受益。例如，如果特定執行個體類型最高支援 12.5 Gbps，則單一流量頻寬也會限制為 12.5 Gbps。
- 在網路擁塞期間，執行時間較長的程序會減少尾延遲。
- 程序可以從更順暢且更標準的網路回應時間分佈中獲益。

## Linux 執行個體的先決條件

為確保 ENA Express 可以有效運作，請按照以下方式更新執行個體的設定。

- 如果您的執行個體使用巨型訊框，則請執行下列命令，將最大傳輸單位 (MTU) 設定為 8900。

```
[ec2-user ~]$ sudo ip link set dev eth0 mtu 8900
```

- 增加接收器 (Rx) 環的大小，如下所示：

```
[ec2-user ~]$ ethtool -G device rx 8192
```

- 若要最大化 ENA Express 頻寬，請依照下列方式設定 TCP 佇列限制：

1. 將 TCP 小型佇列限制設定為 1 MB 或更大值。這會讓在通訊端上排入佇列並等待傳輸的資料量增多。

```
sudo sh -c 'echo 1048576 > /proc/sys/net/ipv4/tcp_limit_output_bytes'
```

2. 如果 Linux 發行版本已啟用 eth 裝置上的位元組佇列限制，請將其停用。對裝置佇列來說，這也會讓排入佇列並等待傳輸的資料增多。

```
sudo sh -c 'for txq in /sys/class/net/eth0/queues/tx-*; do echo max > ${txq}/byte_queue_limits/limit_min; done'
```

### Note

Amazon Linux 發行版本的 ENA 驅動程式預設會停用位元組佇列限制。

## ENA Express 的工作原理

ENA Express 由可 AWS 擴展的可靠數據報 (SRD) 技術提供支持。它會將每個網路流量的封包分配到不同的 AWS 網路路徑，並在偵測到擁塞跡象時動態調整分佈。它還管理接收端的封包重新排序。

若要確保 ENA Express 可以按預期管理網路流量，傳送和接收執行個體及其之間的通訊必須符合下列所有要求：

- 支援傳送和接收執行個體類型。如需詳細資訊，請參閱 [ENA Express 支援的執行個體類型](#) 資料表。
- 傳送和接收執行個體都必須設定 ENA Express。如果組態有任何差異，您可能會遇到流量預設為標準 ENA 傳輸的情況。下列情境說明會發生的情況。

情況：組態的差異

執行個體	ENA Express 已啟用	UDP 使用 ENA Express
執行個體 1	是	是
執行個體 2	是	否



在這種情況下，兩個執行個體之間的 TCP 流量可以使用 ENA Express，因為這兩個執行個體皆已將其啟用。但是，由於其中一個執行個體並未將 ENA Express 用於 UDP 流量，因此這兩個執行個體透過 UDP 之間的通訊會使用標準 ENA 傳輸。

- 傳送和接收執行個體必須在相同的可用區域中執行。
- 執行個體之間的網路路徑不得包含中介軟體方塊。ENA Express 目前不支援中介軟體。
- (僅限 Linux 執行個體) 若要充分利用可能的頻寬，請使用 2.2.9 或更新版本的驅動程式。
- (僅限 Linux 執行個體) 若要產生指標，請使用 2.8 或更新版本的驅動程式。

如果未滿足任意要求，執行個體會使用標準 TCP/UDP 通訊協定但不帶 SRD 進行通訊。

若要確保您的執行個體網路驅動程式設定為最佳效能，請參閱適用於 ENA 驅動程式的建議最佳實務。這些最佳實務也適用於 ENA Express。如需詳細資訊，請參閱 GitHub 網站上的 [ENA Linux 驅動程式最佳作法和效能最佳化指南](#)。

#### Note

Amazon EC2 是指執行個體與以附件形式連接到執行個體的網路介面之間的關係。ENA Express 設定會套用至附件。如果網路介面從執行個體中斷連接，則附件將不再存在，而套用至該執行個體的 ENA Express 設定將不再有效。執行個體終止時也是如此，即使網路介面仍然存在。

## ENA Express 支援的執行個體類型

下列索引標籤顯示支援 ENA Express 的執行個體類型。

### General purpose

執行個體類型	架構
m6a.12xlarge	x86_64
m6a.16xlarge	x86_64
m6a.24xlarge	x86_64

執行個體類型	架構
m6a.32xlarge	x86_64
m6a.48xlarge	x86_64
m6a.metal	x86_64
m6i.8xlarge	x86_64
m6i.12xlarge	x86_64
m6i.16xlarge	x86_64
m6i.24xlarge	x86_64
m6i.32xlarge	x86_64
m6i.metal	x86_64
m6id.8xlarge	x86_64
m6id.12xlarge	x86_64
m6id.16xlarge	x86_64
m6id.24xlarge	x86_64
m6id.32xlarge	x86_64
m6id.metal	x86_64
m7g.12xlarge	arm64
m7g.16xlarge	arm64
m7g.metal	arm64
m7gd.12xlarge	arm64
m7gd.16xlarge	arm64

執行個體類型	架構
m7gd.metal	arm64
m7i.12xlarge	x86_64
m7i.16xlarge	x86_64
m7i.24xlarge	x86_64
m7i.48xlarge	x86_64
m7i.metal-24x1	x86_64
m7i.metal-48x1	x86_64

### Compute optimized

執行個體類型	架構
c6a.12xlarge	x86_64
c6a.16xlarge	x86_64
c6a.24xlarge	x86_64
c6a.32xlarge	x86_64
c6a.48xlarge	x86_64
c6a.metal	x86_64
c6gn.16xlarge	arm64
c6i.8xlarge	x86_64
c6i.12xlarge	x86_64
c6i.16xlarge	x86_64

執行個體類型	架構
c6i.24xlarge	x86_64
c6i.32xlarge	x86_64
c6i.metal	x86_64
c6id.8xlarge	x86_64
c6id.12xlarge	x86_64
c6id.16xlarge	x86_64
c6id.24xlarge	x86_64
c6id.32xlarge	x86_64
c6id.metal	x86_64
c7g.12xlarge	arm64
c7g.16xlarge	arm64
c7g.metal	arm64
c7gd.12xlarge	arm64
c7gd.16xlarge	arm64
c7gd.metal	arm64
c7i.12xlarge	x86_64
c7i.16xlarge	x86_64
c7i.24xlarge	x86_64
c7i.48xlarge	x86_64
c7i.metal-24x1	x86_64

執行個體類型	架構
c7i.metal-48xl	x86_64

## Memory optimized

執行個體類型	架構
r6a.12xlarge	x86_64
r6a.16xlarge	x86_64
r6a.24xlarge	x86_64
r6a.32xlarge	x86_64
r6a.48xlarge	x86_64
r6a.metal	x86_64
r6i.8xlarge	x86_64
r6i.12xlarge	x86_64
r6i.16xlarge	x86_64
r6i.24xlarge	x86_64
r6i.32xlarge	x86_64
r6i.metal	x86_64
r6id.8xlarge	x86_64
r6id.12xlarge	x86_64
r6id.16xlarge	x86_64
r6id.24xlarge	x86_64

執行個體類型	架構
r6id.32xlarge	x86_64
r6id.metal	x86_64
r7g.12xlarge	arm64
r7g.16xlarge	arm64
r7g.metal	arm64
r7gd.12xlarge	arm64
r7gd.16xlarge	arm64
r7gd.metal	arm64
r7i.12xlarge	x86_64
r7i.16xlarge	x86_64
r7i.24xlarge	x86_64
r7i.48xlarge	x86_64
r7i.metal-24x1	x86_64
r7i.metal-48x1	x86_64
u7i-12tb.224xlarge	x86_64
u7in-16tb.224xlarge	x86_64
u7in-24tb.224xlarge	x86_64
u7in-32tb.224xlarge	x86_64
x2idn.16xlarge	x86_64
x2idn.24xlarge	x86_64

執行個體類型	架構
x2idn.32xlarge	x86_64
x2idn.metal	x86_64
x2iedn.8xlarge	x86_64
x2iedn.16xlarge	x86_64
x2iedn.24xlarge	x86_64
x2iedn.32xlarge	x86_64
x2iedn.metal	x86_64

### Accelerated computing

執行個體類型	架構
g6.48xlarge	x86_64

### Storage optimized

執行個體類型	架構
i4g.4xlarge	arm64
i4g.8xlarge	arm64
i4g.16xlarge	arm64
i4i.8xlarge	x86_64
i4i.12xlarge	x86_64
i4i.16xlarge	x86_64
i4i.24xlarge	x86_64

執行個體類型	架構
i4i.32xlarge	x86_64
i4i.metal	x86_64
im4gn.4xlarge	arm64
im4gn.8xlarge	arm64
im4gn.16xlarge	arm64

## 列出並檢視 ENA Express 設定

本節涵蓋如何從 AWS Management Console 或 AWS CLI 中列出及檢視 ENA Express 資訊。如需詳細資訊，請選擇與您將使用的方法相符的標籤。

### Console

此標籤說明如何尋找目前 ENA Express 設定的相關資訊，以及如何在 AWS Management Console 中檢視執行個體類型支援。

#### 檢視執行個體類型支援

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽窗格中，選擇 Instance Types (執行個體類型)。
3. 選取執行個體類型以查看該執行個體的詳細資訊。您可以選擇 Instance type (執行個體類型) 連結，以開啟詳細資訊頁面；或選取清單左側的核取方塊，以在頁面底部的詳細資訊窗格中檢視詳細資訊。
4. 在 Networking (聯網) 標籤或詳細資訊頁面上的該區段中，ENA Express support (ENA Express 支援) 會顯示 true 或 false 值，表示執行個體類型是否支援此功能。

#### 從網路介面清單檢視設定

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽窗格中，選擇 Network Interfaces (網路介面)。



3. 選取網路介面以查看該執行個體的詳細資訊。您可以選擇 Network interface ID (網路介面 ID) 連結，以開啟詳細資訊頁面；或選取清單左側的核取方塊，以在頁面底部的詳細資訊窗格中檢視詳細資訊。
4. 在 Details (詳細資訊) 標籤或詳細資訊頁面的 Network interface attachment (網路介面附件) 區段中，檢視 ENA Express 和 ENA Express UDP 的設定。

### 檢視執行個體的設定

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽窗格中選擇 (執行個體)。
3. 選取執行個體以查看該執行個體的詳細資訊。您可以選擇 Instance ID (執行個體 ID) 連結，以開啟詳細資訊頁面；或選取清單左側的核取方塊，以在頁面底部的詳細資訊窗格中檢視詳細資訊。
4. 在 Networking (聯網) 標籤的 Network interfaces (網路介面) 區段中，向右捲動以檢視 ENA Express 和 ENA Express UDP 的設定。

## AWS CLI

此標籤說明如何尋找目前 ENA Express 設定的相關資訊，以及如何在 AWS CLI 中檢視執行個體類型支援。

### 描述執行個體類型

如需有關特定例證類型之例證類型設定的資訊，請執行中的 [describe-instance-types](#) 指令 AWS CLI，然後以下列方式取代例證類型：

```
[ec2-user ~]$ aws ec2 describe-instance-types --instance-types m6i.metal
{
 "InstanceTypes": [
 {
 "InstanceType": "m6i.metal",
 "CurrentGeneration": true,
 ...
 },
 "NetworkInfo": {
 ...
 "EnaSrdSupported": true
 },
],
}
```

```
...
}
]
}
```

## 描述執行個體

如需指定執行處理之 ENA Express 組態的相關資訊，請執行中的 [describe-instances](#) 命令 AWS CLI，如下所示。此命令範例會傳回連接至 `--instance-ids` 參數所指定之每個執行中執行處理之網路介面的 ENA Express 組態清單。

```
[ec2-user ~]$ aws ec2 describe-instances --instance-ids i-1234567890abcdef0 i-0598c7d356eba48d7 --query 'Reservations[*].Instances[*].[InstanceId, NetworkInterfaces[*].Attachment.EnaSrdSpecification]'
```

```
[
 [
 "i-1234567890abcdef0",
 [
 {
 "EnaSrdEnabled": true,
 "EnaSrdUdpSpecification": {
 "EnaSrdUdpEnabled": false
 }
 }
]
],
 [
 [
 "i-0598c7d356eba48d7",
 [
 {
 "EnaSrdEnabled": true,
 "EnaSrdUdpSpecification": {
 "EnaSrdUdpEnabled": false
 }
 }
]
]
]
]
```

## 描述網路介面

如需有關網路介面 ENA Express 設定的資訊，請執行下列[describe-network-interfaces](#)命令：AWS CLI

```
[ec2-user ~]$ aws ec2 describe-network-interfaces
{
 "NetworkInterfaces": [
 {
 "Association": {
 IPs, DNS...
 },
 "Attachment": {
 "AttachTime": "2022-11-17T09:04:28+00:00",
 "AttachmentId": "eni-attach-0ab1c23456d78e9f0",
 "DeleteOnTermination": true,
 "DeviceIndex": 0,
 "NetworkCardIndex": 0,
 "InstanceId": "i-1234567890abcdef0",
 "InstanceOwnerId": "111122223333",
 "Status": "attached",
 "EnaSrdSpecification": {
 "EnaSrdEnabled": true,
 "EnaSrdUdpSpecification": {
 "EnaSrdUdpEnabled": true
 }
 }
 },
 ...
 "NetworkInterfaceId": "eni-1234567890abcdef0",
 "OwnerId": "111122223333",
 ...
 }
]
}
```

## PowerShell

此索引標籤說明如何尋找目前 ENA Express 設定的相關資訊，以及如何使用檢視執行個體類型支援 PowerShell。

### 描述執行個體類型

如需特定例證類型之例證類型設定的相關資訊，請[Get-EC2InstanceType Cmdlet](#)使用「工具」(Tools for) 執行 PowerShell，然後以下列方式取代例證類型：

```
PS C:\> Get-EC2InstanceType -InstanceType m6i.metal | `
Select-Object `
 InstanceType,
 CurrentGeneration,
 @{Name = 'EnaSrdSupported'; Expression = { $_.NetworkInfo.EnaSrdSupported } } | `
Format-List

InstanceType : m6i.metal
CurrentGeneration : True
EnaSrdSupported : True
```

如果已啟用 ENA Express，則傳回 True 的值。

### 描述網路介面

如需網路介面的 ENA Express 設定的相關資訊，請[Get-EC2NetworkInterface Cmdlet](#)使用「工具」執行，PowerShell 如下所示：

```
PS C:\> Get-EC2NetworkInterface -NetworkInterfaceId eni-0d1234e5f6a78901b | `
Select-Object `
 Association,
 NetworkInterfaceId,
 OwnerId,
 @{Name = 'AttachTime'; Expression = { $_.Attachment.AttachTime } },
 @{Name = 'AttachmentId'; Expression = { $_.Attachment.AttachmentId } },
 @{Name = 'DeleteOnTermination'; Expression =
{ $_.Attachment.DeleteOnTermination } },
 @{Name = 'NetworkCardIndex'; Expression = { $_.Attachment.NetworkCardIndex } },
 @{Name = 'InstanceId'; Expression = { $_.Attachment.InstanceId } },
 @{Name = 'InstanceOwnerId'; Expression = { $_.Attachment.InstanceOwnerId } },
 @{Name = 'Status'; Expression = { $_.Attachment.Status } },
 @{Name = 'EnaSrdEnabled'; Expression =
{ $_.Attachment.EnaSrdSpecification.EnaSrdEnabled } },
 @{Name = 'EnaSrdUdpEnabled'; Expression =
{ $_.Attachment.EnaSrdSpecification.EnaSrdUdpSpecification.EnaSrdUdpEnabled } }

Association :
NetworkInterfaceId : eni-0d1234e5f6a78901b
OwnerId : 111122223333
AttachTime : 6/11/2022 1:13:11 AM
AttachmentId : eni-attach-0d1234e5f6a78901b
DeleteOnTermination : True
```

```
NetworkCardIndex : 0
InstanceId : i-0d1234e5f6a78901b
InstanceOwnerId : 111122223333
Status : attached
EnaSrdEnabled : True
EnaSrdUdpEnabled : False
```

## 設定 ENA Express 設定

您可以針對支援的 EC2 執行個體類型設定 ENA Express，而無需安裝任何其他軟體。

本節介紹如何從 AWS Management Console 或設定 ENA 快速 AWS CLI。如需詳細資訊，請選擇與您將使用的方法相符的標籤。

### Console

此標籤涵蓋如何管理連接至執行個體之網路介面的 ENA Express 設定。

#### 從網路介面清單管理 ENA Express

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取要連接到執行個體的網路介面。您可以選擇 Network interface ID (網路介面 ID) 連結，以開啟詳細資訊頁面；或選取清單左側的核取方塊。
4. 從頁面右上角的 Action (動作) 功能表中選擇 Manage ENA Express (管理 ENA Express)。此操作會開啟 Manage ENA Express (管理 ENA Express) 對話方塊，並顯示選取的網路介面 ID 和目前設定。

#### Note

如果您選取的網路介面未連接至執行個體，則此動作不會顯示在功能表中。

5. 若要使用 ENA Express，請選取 Enable (啟用) 核取方塊。
6. 啟用 ENA Express 時，您可以設定 UDP 設定。若要使用 ENA Express UDP，請選取 Enable (啟用) 核取方塊。
7. 若要儲存設定，請選擇 Save (儲存)。

## 從執行個體清單管理 ENA Express

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽窗格中選擇 (執行個體)。
3. 選取您要管理的執行個體。您可以選擇 Instance ID (執行個體 ID) 連結，以開啟詳細資訊頁面；或選取清單左側的核取方塊。
4. 選取要為您的執行個體設定的 Network interface (網路介面)。
5. 從頁面右上角的 Action (動作) 功能表中選擇 Manage ENA Express (管理 ENA Express)。
6. 若要為連接至執行個體的網路介面設定 ENA Express，請從 Network interface (網路介面) 清單中進行選取。
7. 若要將 ENA Express 用於選取的網路介面附件，請選取 Enable (啟用) 核取方塊。
8. 啟用 ENA Express 時，您可以設定 UDP 設定。若要使用 ENA Express UDP，請選取 Enable (啟用) 核取方塊。
9. 若要儲存設定，請選擇 Save (儲存)。

當您將網路介面連接至 EC2 執行個體時，請設定 ENA Express

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取未連接到執行個體的網路介面 (Status (狀態) 為 Available (可用)。您可以選擇 Network interface ID (網路介面 ID) 連結，以開啟詳細資訊頁面；或選取清單左側的核取方塊。
4. 選取您要連接的 Instance (執行個體)。
5. 若要在您將網路介面連接至執行個體後使用 ENA Express，請選取 Enable (啟用) 核取方塊。
6. 啟用 ENA Express 時，您可以設定 UDP 設定。若要使用 ENA Express UDP，請選取 Enable (啟用) 核取方塊。
7. 若要將網路介面連接至執行個體並儲存 ENA Express 設定，請選擇 Attach (連接)。

## AWS CLI

此標籤涵蓋如何在 AWS CLI 中設定 ENA Express 設定。

當您連接網路介面時設定 ENA Express

若要在將網路介面連接至執行個體時設定 ENA Express，請執行中的 [attach-network-interface](#) 命令 AWS CLI，如下列範例所示：

### 範例 1：將 ENA Express 用於 TCP 流量而非 UDP 流量

在此範例中，我們會將 `EnaSrdEnabled` 設定為 `true`，並且我們允許將 `EnaSrdUdpEnabled` 預設為 `false`。

```
[ec2-user ~]$ aws ec2 attach-network-interface --network-interface-id eni-0123f4567890a1b23 --instance-id i-0f1a234b5cd67e890 --device-index 1 --ena-srd-specification 'EnaSrdEnabled=true'
{
 "AttachmentId": "eni-attach-012c3d45e678f9012"
}
```

### 範例 2：將 ENA Express 用於 TCP 流量和 UDP 流量

在此範例中，我們將 `EnaSrdEnabled` 和 `EnaSrdUdpEnabled` 都設定為 `true`。

```
[ec2-user ~]$ aws ec2 attach-network-interface --network-interface-id eni-0123f4567890a1b23 --instance-id i-0f1a234b5cd67e890 --device-index 1 --ena-srd-specification 'EnaSrdEnabled=true,EnaSrdUdpSpecification={EnaSrdUdpEnabled=true}'
{
 "AttachmentId": "eni-attach-012c3d45e678f9012"
}
```

### 更新網路介面附件的 ENA Express 設定

若要更新連接至執行個體之網路介面的 ENA Express 設定，請執行中的 [modify-network-interface-attribute](#) 命令 AWS CLI，如下列範例所示：

### 範例 1：將 ENA Express 用於 TCP 流量而非 UDP 流量

在此範例中，我們會將 `EnaSrdEnabled` 設定為 `true`，並且我們允許將 `EnaSrdUdpEnabled` 預設為 `false`，如果之前並未設定。

```
[ec2-user ~]$ aws ec2 modify-network-interface-attribute --network-interface-id eni-0123f4567890a1b23 --ena-srd-specification 'EnaSrdEnabled=true'
```

### 範例 2：將 ENA Express 用於 TCP 流量和 UDP 流量

在此範例中，我們將 `EnaSrdEnabled` 和 `EnaSrdUdpEnabled` 都設定為 `true`。

```
[ec2-user ~]$ aws ec2 modify-network-interface-attribute --
network-interface-id eni-0123f4567890a1b23 --ena-srd-specification
'EnaSrdEnabled=true,EnaSrdUdpSpecification={EnaSrdUdpEnabled=true}'
```

範例 3：停止將 ENA Express 用於 UDP 流量

在此範例中，我們會將 `EnaSrdUdpEnabled` 設定為 `false`。

```
[ec2-user ~]$ aws ec2 modify-network-interface-attribute --
network-interface-id eni-0123f4567890a1b23 --ena-srd-specification
'EnaSrdUdpSpecification={EnaSrdUdpEnabled=false}'
```

## PowerShell

此索引標籤說明如何使用來設定 ENA 快速設定 PowerShell。

當您連接網路介面時設定 ENA Express

若要設定網路介面的 ENA Express 設定，請[Add-EC2NetworkInterface Cmdlet](#)使用的工具執行，PowerShell 如下列範例所示：

範例 1：將 ENA Express 用於 TCP 流量而非 UDP 流量

在此範例中，我們會將 `EnaSrdEnabled` 設定為 `true`，並且我們允許將 `EnaSrdUdpEnabled` 預設為 `false`。

```
PS C:\> Add-EC2NetworkInterface `
-NetworkInterfaceId eni-0123f4567890a1b23 `
-InstanceId i-0f1a234b5cd67e890 `
-DeviceIndex 1 `
-EnaSrdSpecification_EnaSrdEnabled $true

eni-attach-012c3d45e678f9012
```

範例 2：將 ENA Express 用於 TCP 流量和 UDP 流量

在此範例中，我們將 `EnaSrdEnabled` 和 `EnaSrdUdpEnabled` 都設定為 `true`。

```
PS C:\> Add-EC2NetworkInterface `
-NetworkInterfaceId eni-0123f4567890a1b23 `
-InstanceId i-0f1a234b5cd67e890 `
```



```
-DeviceIndex 1 `
-EnaSrdSpecification_EnaSrdEnabled $true `
-EnaSrdSpecification_EnaSrdUdpEnabled $true

eni-attach-012c3d45e678f9012
```

## 更新網路介面附件的 ENA Express 設定

若要更新連接至執行個體之網路介面的 ENA Express 設定，請執行工具中的 [Add-EC2NetworkInterface Cmdlet](#) 命令 PowerShell，如下列範例所示：

### 範例 1：將 ENA Express 用於 TCP 流量而非 UDP 流量

在此範例中，我們會將 EnaSrdEnabled 設定為 true，並且我們允許將 EnaSrdUdpEnabled 預設為 false，如果之前並未設定。

```
PS C:\> Edit-EC2NetworkInterfaceAttribute `
-NetworkInterfaceId eni-0123f4567890a1b23 `
-EnaSrdSpecification_EnaSrdEnabled $true ;
Get-EC2NetworkInterface -NetworkInterfaceId eni-0123f4567890a1b23 | `
Select-Object `
 NetworkInterfaceId,
 @{Name = 'EnaSrdEnabled'; Expression =
 { $_.Attachment.EnaSrdSpecification.EnaSrdEnabled }},
 @{Name = 'EnaSrdUdpEnabled'; Expression =
 { $_.Attachment.EnaSrdSpecification.EnaSrdUdpSpecification.EnaSrdUdpEnabled }} | `
Format-List

NetworkInterfaceId : eni-0123f4567890a1b23
EnaSrdEnabled : True
EnaSrdUdpEnabled : False
```

### 範例 2：將 ENA Express 用於 TCP 流量和 UDP 流量

在此範例中，我們將 EnaSrdEnabled 和 EnaSrdUdpEnabled 都設定為 true。

```
PS C:\> Edit-EC2NetworkInterfaceAttribute `
-NetworkInterfaceId eni-0123f4567890a1b23 `
-EnaSrdSpecification_EnaSrdEnabled $true `
-EnaSrdSpecification_EnaSrdUdpSpecification_EnaSrdUdpEnabled $true ;
Get-EC2NetworkInterface -NetworkInterfaceId eni-0123f4567890a1b23 | `
Select-Object `
```

```

 NetworkInterfaceId,
 @{Name = 'EnaSrdEnabled'; Expression =
 { $_.Attachment.EnaSrdSpecification.EnaSrdEnabled }},
 @{Name = 'EnaSrdUdpEnabled'; Expression =
 { $_.Attachment.EnaSrdSpecification.EnaSrdUdpSpecification.EnaSrdUdpEnabled }} | `
Format-List

NetworkInterfaceId : eni-0123f4567890a1b23
EnaSrdEnabled : True
EnaSrdUdpEnabled : True

```

### 範例 3：停止將 ENA Express 用於 UDP 流量

在此範例中，我們會將 `EnaSrdUdpEnabled` 設定為 `false`。

```

PS C:\> Edit-EC2NetworkInterfaceAttribute `
-NetworkInterfaceId eni-0123f4567890a1b23 `
-EnaSrdSpecification_EnaSrdUdpSpecification_EnaSrdUdpEnabled $false ;
Get-EC2NetworkInterface -NetworkInterfaceId eni-0123f4567890a1b23 | `
Select-Object `
 NetworkInterfaceId,
 @{Name = 'EnaSrdEnabled'; Expression =
 { $_.Attachment.EnaSrdSpecification.EnaSrdEnabled }},
 @{Name = 'EnaSrdUdpEnabled'; Expression =
 { $_.Attachment.EnaSrdSpecification.EnaSrdUdpSpecification.EnaSrdUdpEnabled }} | `
Format-List

NetworkInterfaceId : eni-0123f4567890a1b23
EnaSrdEnabled : True
EnaSrdUdpEnabled : False

```

## 啟動時設定 ENA 快速

從 AWS Management Console 啟動執行個體時，您可以使用下列其中一種方法來為 AMI 設定 ENA Express。

- 使用啟動執行個體精靈啟動執行個體時，您可以為 AMI 設定 ENA Express。如需組態詳細資訊，請參閱啟動執行個體精靈的 [Network settings \(網路設定\)](#) 中的進階網路組態。
- 使用啟動範本時，您可以為 AMI 設定 ENA Express。如需啟動範本組態的相關資訊，請參閱啟動範本的 [Network settings \(網路設定\)](#) 中的進階網路組態。

## 監控 ENA Express 效能

在傳送執行個體和接收執行個體上啟用網路介面附件的 ENA Express 之後，您可以使用 ENA Express 指標來協助確保您的執行個體充分利用 SRD 技術提供的效能改進。

若要查看針對 ENA Express 篩選的指標清單，請遵循適用於您的網路介面的 `ethtool` 命令 (在這裡顯示為 `eth0`)：

```
[ec2-user ~]$ ethtool -S eth0 | grep ena_srd
NIC statistics:
 ena_srd_mode: 0
 ena_srd_tx_pkts: 0
 ena_srd_eligible_tx_pkts: 0
 ena_srd_rx_pkts: 0
 ena_srd_resource_utilization: 0
```

### 確認執行個體的 ENA Express 設定

若要確認執行個體上網路介面附件目前的 ENA Express 設定，請執行 `ethtool` 命令列出 ENA Express 指標，並記下 `ena_srd_mode` 指標的值。相關值如下：

- 0 = ENA Express 關閉，UDP 關閉
- 1 = ENA Express 開啟，UDP 關閉
- 2 = ENA Express 關閉，UDP 開啟

#### Note

僅在原先啟用 ENA Express，且將 UDP 設定為使用它時，才會發生這種情況。會保留 UDP 流量的先前值。

- 3 = ENA Express 開啟，UDP 開啟

在執行個體上為網路介面附件啟用 ENA Express 之後，傳送執行個體會啟動與接收執行個體的通訊，而 SRD 會偵測 ENA Express 是否同時在傳送執行個體和接收執行個體上運作。如果 ENA Express 正在運作，則通訊可以使用 SRD 傳輸。如果 ENA Express 尚未運作，則通訊會回復為標準 ENA 傳輸。若要確認封包傳輸是否使用 SRD，您可以比較指定期間內的合格封包數目 (`ena_srd_eligible_tx_pkts` 指標) 與傳輸的 SRD 封包數目 (`ena_srd_tx_pkts` 指標)。

您可以使用 `ena_srd_resource_utilization` 指標監控 SRD 資源使用率。如果您的執行個體即將耗盡 SRD 資源，便會知道此時需要橫向擴展執行個體了。

如需 ENA Express 指標的詳細資訊，請參閱 [ENA Express 的指標](#)。

## 調整 ENA 快速設定的效能

若要檢查 Linux 執行個體組態以獲得最佳的 ENA Express 效能，您可以執行下列 Amazon GitHub 儲存庫上提供的指令碼：

<https://github.com/amzn/amzn-ec2-ena-utilities/blob/main/ena-express/check-ena-express-settings.sh>

該腳本運行一系列測試，並建議建議和必要的配置更改。

## 使用 EC2 執行個體上的英特爾 82599 VF 介面啟用增強型聯網

Amazon EC2 可透過 Intel 82599 VF 介面提供增強型聯網功能，而該介面使用 Intel `ixgbevf` 驅動程式。

### 目錄

- [要求](#)
- [確認驅動程式已安裝](#)
- [測試是否已啟用增強型網路](#)
- [在執行個體上啟用增強型網路](#)
- [疑難排解連線問題](#)

### 要求

若要準備使用 Intel 82599 VF 介面的增強型聯網，請依下列方式設定您的執行個體：

- 選取下列支援的執行個體類型：C3、C4、D2、I2、M4 (不含 `m4.16xlarge`) 和 R3。
- 確定該執行個體具有網際網路連線能力。
- 如果您的執行個體上有想要保留的重要資料，您現在應從執行個體建立 AMI，以備份這些資料。更新核心與核心模組，以及啟用 `sriovNetSupport` 屬性，可能使執行個體變得不相容或使作業系統變得無法連線。如果您有較新的備份，發生這些狀況時資料便能獲得保留。
- Linux 執行個體 — 使用 Linux 核心版本 2.6.32 或更新版本，從 HVM AMI 啟動執行個體。最新的 Amazon Linux HVM AMI 擁有安裝增強型聯網所需的模組和必要的屬性集。因此，如果您使用目前

的 Amazon Linux HVM AMI 啟動 Amazon EBS 支援的增強型聯網所支援的執行個體，則您的執行個體已啟用增強型聯網。

### Warning

增強型聯網僅支援於 HVM 執行個體。啟用 PV 執行個體的增強型聯網可能使其變為無法連線。在沒有正確的模組或模組版本下設定此屬性，也可能使執行個體變為無法連線。

- 視窗執行個體 — 從 64 位元 HVM AMI 啟動執行個體。您無法在視窗伺服器 2008 上啟用增強型網路功能。Windows Server 2012 R2 和 Windows Server 2016 和更新版本 AMI 已啟用增強型聯網。Windows Server 2012 R2 內含 Intel 驅動程式 1.0.15.3，因此我們建議您使用 Pnputil.exe 公用程式將驅動程式升級至最新版本。
- 在您選擇的任何電腦 [AWS Tools for Windows PowerShell](#) 上使用 [AWS CloudShell](#) AWS Management Console，[AWS CLI](#) 或安裝並設定或，最好是您的本機桌上型電腦或筆記型電腦。如需詳細資訊，請參閱 [存取 Amazon EC2](#) 和 [AWS CloudShell 使用者指南](#)。增強型聯網無法從 Amazon EC2 主控台管理。

## 確認驅動程式已安裝

確認驅動程式已安裝在您的執行個體上。

### Linux 網路介面驅動程式

請用下列命令確認模組使用於特定介面，並將介面名稱換成您想要檢查的介面。如果您使用單一介面 (預設)，則這會是 eth0。如果作業系統支援 [可預測的網路名稱](#)，則名稱可能像是 ens5。

在下列範例中，不會載入 ixgbev 模組，因為列出的驅動程式為 vif。

```
[ec2-user ~]$ ethtool -i eth0
driver: vif
version:
firmware-version:
bus-info: vif-0
supports-statistics: yes
supports-test: no
supports-eprom-access: no
supports-register-dump: no
supports-priv-flags: no
```

在此範例中，將載入 ixgbev 模組。此執行個體已正確設定增強型網路。

```
[ec2-user ~]$ ethtool -i eth0
driver: ixgbevf
version: 4.0.3
firmware-version: N/A
bus-info: 0000:00:03.0
supports-statistics: yes
supports-test: yes
supports-eeprom-access: no
supports-register-dump: yes
supports-priv-flags: no
```

## 視窗網路介面卡

若要確定已安裝驅動程式，請連線到執行個體，並開啟裝置管理員。Network adapters (網路轉接器) 底下應該會列出「Intel(R) 82599 虛擬功能」。

## 測試是否已啟用增強型網路

確認sriovNetSupport屬性已設定。

### 實例屬性 (系統NetSupport)

若要查看執行個體是否設定增強型聯網的 sriovNetSupport 屬性，請使用下列其中一項命令。如果已設定屬性，則值為simple。

- [describe-instance-attribute](#) (AWS CLI) (AWS CLI/AWS CloudShell)

```
aws ec2 describe-instance-attribute --instance-id instance_id --attribute
sriovNetSupport
```

- [Get-EC2InstanceAttribute](#) (AWS Tools for Windows PowerShell)

```
Get-EC2InstanceAttribute -InstanceId instance-id -Attribute sriovNetSupport
```

### 影像屬性 (檔案NetSupport)

若要檢查 AMI 是否已設定增強型網路sriovNetSupport屬性，請使用下列其中一個指令。如果已設定屬性，則值為simple。

- [describe-images](#) (AWS CLI)

```
aws ec2 describe-images --image-id ami_id --query "Images[].SriovNetSupport"
```

- [Get-EC2Image](#) (AWS Tools for Windows PowerShell)

```
(Get-EC2Image -ImageId ami-id).SriovNetSupport
```

## 在執行個體上啟用增強型網路

您使用的程序視執行個體的作業系統而定。

### Warning

增強型聯網屬性一經啟用後，便無法停用。

## Amazon Linux

最新的 Amazon Linux HVM AMI 擁有安裝增強型網路所需的 `ixgbevf` 模組和必要的 `sriovNetSupport` 屬性集。因此，如果您使用目前的 Amazon Linux HVM AMI 啟動執行個體類型，則您的執行個體已啟用增強型聯網。如需詳細資訊，請參閱 [測試是否已啟用增強型網路](#)。

如果您使用較舊的 Amazon Linux AMI 啟動執行個體，且其尚未啟用增強型聯網，請用下列程序啟用增強型聯網。

### 啟用增強型聯網

1. 連線到您的執行個體。
2. 在執行個體上，執行下列命令，將您的執行個體更新為最新的核心與核心模組，包括 `ixgbevf`：

```
[ec2-user ~]$ sudo yum update
```

3. 從本機電腦中，使用 Amazon EC2 主控台或下列其中一個命令來重新啟動執行個體：[reboot-instances](#) (AWS CLI)、[Restart-EC2Instance](#) (AWS Tools for Windows PowerShell)。
4. 再次連接至執行個體，並使用 `ixgbevf` 中的 `modinfo ixgbevf` 命令確認 [測試是否已啟用增強型網路](#) 模組已安裝且為最低建議版本。
5. [EBS 支援的執行個體] 從本機電腦中，使用 Amazon EC2 主控台或下列其中一個命令來停止執行個體：[stop-instances](#) (AWS CLI)、[Stop-EC2Instance](#) (AWS Tools for Windows PowerShell)。如

果執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中停止執行個體，讓執行個體狀態保持同步。

[執行個體存放區後端執行個體] 您無法停止執行個體來修改屬性。請改為執行此程序：[啟用增強型聯網 \(執行個體後端執行個體\)](#)。

6. 在本機電腦上使用下列其中一個命令啟用增強型聯網屬性。

## AWS CLI

### [modify-instance-attribute](#) (AWS CLI)

```
aws ec2 modify-instance-attribute --instance-id instance_id --sriov-net-support simple
```

## PowerShell

### [Edit-EC2InstanceAttribute](#) (AWS Tools for Windows PowerShell)

```
Edit-EC2InstanceAttribute -InstanceId instance_id -SriovNetSupport "simple"
```

7. (選擇性) 從執行個體建立 AMI，如 [創建一個 Amazon EBS-backed AMI](#) 所述。AMI 將從執行個體繼承增強型聯網屬性。因此，您預設可用此 AMI 啟動其他啟用增強型聯網的執行個體。
8. 從本機電腦中，使用 Amazon EC2 主控台或下列其中一個命令來啟動執行個體：[start-instances](#) (AWS CLI)、[Start-EC2Instance](#) (AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中啟動執行個體，讓執行個體狀態保持同步。
9. 連接至執行個體，並使用 `ixgbev` 中的 `ethtool -i ethn` 命令確認 [測試是否已啟用增強型網路](#) 模組已在網路介面上安裝及載入。

## 啟用增強型聯網 (執行個體後端執行個體)

遵循先前的程序，一直進行到您停止執行個體的那個步驟。如 [建立執行個體存放區後端 Linux AMI](#) 中所述建立新的 AMI，且務必在註冊 AMI 時啟用增強型聯網屬性。

## AWS CLI

### [register-image](#) (AWS CLI/AWS CloudShell)

```
aws ec2 register-image --sriov-net-support simple ...
```



## PowerShell

### [Register-EC2Image](#) (AWS Tools for Windows PowerShell)

```
Register-EC2Image -SriovNetSupport "simple" ...
```

## Ubuntu

開始之前，[請先檢查您的執行個體是否啟用增強型聯網](#)。

Quick Start Ubuntu HVM AMI 包含增強型聯網必要的驅動程式。如果您有 2.16.4 版以前的 `ixgbevf`，則可以安裝 `linux-aws` 核心套件來取得最新的增強型聯網驅動程式。

以下程序提供在 Ubuntu 執行個體上編譯 `ixgbevf` 模組的一般步驟。

### 安裝 `linux-aws` 核心套件

1. 連線到您的執行個體。
2. 更新套件快取和套件。

```
ubuntu:~$ sudo apt-get update && sudo apt-get upgrade -y linux-aws
```

#### Important

如果更新程序期間提示您安裝 `grub`，請用 `/dev/xvda` 安裝 `grub`，然後選擇保留目前版本的 `/boot/grub/menu.lst`。

## 其他 Linux 發行版本

開始之前，[請先檢查您的執行個體是否啟用增強型聯網](#)。最新的 Quick Start HVM AMI 包含增強型聯網必要的驅動程式，因此您不需要執行其他步驟。

如果您需要在 Amazon Linux 或 Ubuntu 以外的其他 Linux 發行版本上使用 Intel 82599 VF 介面啟用增強型聯網，以下程序提供一般步驟。如需像是詳細命令語法、檔案位置或套件和工具支援等詳細資訊，請參閱 Linux 發行版本的特定文件。

### 在 Linux 上啟用增強型聯網

1. 連線到您的執行個體。

2. 在您的執行個體上從 Sourceforge (<https://sourceforge.net/projects/e1000/files/ixgbevf%20stable/>) 下載 ixgbevf 模組來源。

2.16.4 之前版本的 ixgbevf，包括 2.14.2 版，無法在某些 Linux 發行版本上正確建置，包括特定版本的 Ubuntu。

3. 在您的執行個體上編譯及安裝 ixgbevf 模組。

#### Warning

如果您為目前核心編譯 ixgbevf 模組，接著升級核心，但未針對新核心重建驅動程式，系統可能在下次重新啟動時還原為發行版本專屬的 ixgbevf 模組。如果發行版本專屬版本與增強型聯網不相容，則系統可能變得無法連線。

4. 執行 `sudo depmod` 命令更新模組相依性。
5. 在執行個體上更新 `initramfs`，確定開機時載入新模組。
6. 判斷系統是否依預設使用可預測的網路介面名稱。使用 `systemd` 或 `udev` 197 或以上版本的系統可重新命名乙太網路裝置，但不保證單一網路介面會命名為 `eth0`。該行為會造成連線至執行個體時發生問題。如需詳細資訊及查看其他的組態選項，請參閱 [freedesktop.org](http://freedesktop.org) 網站上的 [可預測的網路介面名稱](#)。
  - a. 您可用下列命令在 RPM 系統上檢查 `systemd` 或 `udev` 的版本：

```
[ec2-user ~]$ rpm -qa | grep -e '^systemd-[0-9]\+\|^udev-[0-9]\+'
systemd-208-11.el7_0.2.x86_64
```

在上述的 Red Hat Enterprise Linux 7 範例中，`systemd` 版本為 208，因此可預測的網路介面名稱已停用。

- b. 若要停用可預測網路介面名稱，請將 `net.ifnames=0` 選項新增至 `GRUB_CMDLINE_LINUX` 的 `/etc/default/grub` 行中。

```
[ec2-user ~]$ sudo sed -i '/^GRUB_CMDLINE_LINUX/s/\ "$\ net.ifnames=0\ "/' /
etc/default/grub
```

- c. 重建 `grub` 組態檔案。

```
[ec2-user ~]$ sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

7. [EBS 後端執行個體] 從本機電腦中，使用 Amazon EC2 主控台或下列其中一個命令來停止執行個體：[stop-instances](#) (AWS CLI/AWS CloudShell)、[Stop-EC2Instance](#) (AWS Tools for Windows PowerShell)。如果執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中停止執行個體，讓執行個體狀態保持同步。

[執行個體存放區後端執行個體] 您無法停止執行個體來修改屬性。請改為執行此程序：[啟用增強型聯網 \(執行個體存放區支援的執行個體\)](#)。

8. 在本機電腦上使用下列其中一個命令啟用增強型聯網屬性。

## AWS CLI

[modify-instance-attribute](#) (AWS CLI/AWS CloudShell)

```
aws ec2 modify-instance-attribute --instance-id instance_id --sriov-net-support simple
```

## PowerShell

[Edit-EC2InstanceAttribute](#) (AWS Tools for Windows PowerShell)

```
Edit-EC2InstanceAttribute -InstanceId instance_id -SriovNetSupport "simple"
```

9. (選擇性) 從執行個體建立 AMI，如 [創建一個 Amazon EBS-backed AMI](#) 所述。AMI 將從執行個體繼承增強型聯網屬性。因此，您預設可用此 AMI 啟動其他啟用增強型聯網的執行個體。

如果您執行個體的作業系統包含 `/etc/udev/rules.d/70-persistent-net.rules` 檔案，您必須先刪除檔案，然後再建立 AMI。此檔案包含原始執行個體乙太網路卡的 MAC 地址。如果其他執行個體使用此檔案開機，作業系統將無法找到設備，且 `eth0` 可能會失敗，因而造成開機問題。此檔案會在下次開機週期時重新產生，且從 AMI 啟動的任何執行個體將會建立自己的檔案版本。

10. 從本機電腦中，使用 Amazon EC2 主控台或下列其中一個命令來啟動執行個體：[start-instances](#) (AWS CLI)、[Start-EC2Instance](#) (AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中啟動執行個體，讓執行個體狀態保持同步。
11. (選擇性) 連線到執行個體，並確認模組已安裝。

啟用增強型聯網 (執行個體存放區支援的執行個體)

遵循先前的程序，一直進行到您停止執行個體的那個步驟。如 [建立執行個體存放區後端 Linux AMI](#) 中所述建立新的 AMI，且務必在註冊 AMI 時啟用增強型聯網屬性。

## AWS CLI

[register-image](#) (AWS CLI/AWS CloudShell)

```
aws ec2 register-image --sriov-net-support simple ...
```

## PowerShell

[Register-EC2Image](#) (AWS Tools for Windows PowerShell)

```
Register-EC2Image -SriovNetSupport "simple" ...
```

## Windows

如果您啟動執行個體，且其尚未啟用增強型聯網，您必須在執行個體上下載並安裝必要的網路轉接器驅動程式，然後設定 sriovNetSupport 執行個體屬性，以啟動增強型聯網。您只能在支援的執行個體類型上啟用此屬性。如需詳細資訊，請參閱 [增強的網路支援](#)。

### Important

若要檢視視窗 AMI 中的最新驅動程式更新，請參閱視窗 AMI 參考資料中的 [AWS Windows AMI 版本歷程記錄](#)。

## 啟用增強型聯網

1. 連線到您的執行個體，並以本機管理員登入。
2. [Windows 伺服器 2016 年及更新版本] 執行下列 EC2 啟動 PowerShell 指令碼，以在安裝驅動程式後設定執行個體。

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 - Schedule
```

**⚠ Important**

當您啟用初始化執行個體 EC2 Launch 指令碼時，管理員密碼將重設。您可以修改組態檔來停用管理員密碼重設，方法為在初始化任務的設定中指定它。

**3. 從執行個體中，下載適用於您的作業系統的 Intel 網路轉接器驅動程式：**

- Windows Server 2022

請造訪[下載頁面](#)並下載Wired\_driver\_*version*\_x64.zip。

- Windows Server 2019 (包括 Server 版本 1809 和更新版本\*)

請造訪[下載頁面](#)並下載Wired\_driver\_*version*\_x64.zip。

- Windows Server 2016 (包括 Server 版本 1803 和更早版本\*)

請造訪[下載頁面](#)並下載Wired\_driver\_*version*\_x64.zip。

- Windows Server 2012 R2

請造訪[下載頁面](#)並下載Wired\_driver\_*version*\_x64.zip。

- Windows Server 2012

請造訪[下載頁面](#)並下載Wired\_driver\_*version*\_x64.zip。

- Windows Server 2008 R2

請造訪[下載頁面](#)並下載PROWinx64Legacy.exe。

\* Intel 驅動程式與軟體頁面並未特別說明伺服器版本 1803 和更早版本，以及 1809 和更新版本。

**4. 安裝適用於您的作業系統的 Intel 網路轉接器驅動程式。**

- Windows Server 2008 R2

1. 在 Downloads (下載) 資料夾中，找出 PROWinx64Legacy.exe 檔案並將其重新命名為 PROWinx64Legacy.zip。
2. 解壓縮 PROWinx64Legacy.zip 檔案的內容。
3. 開啟命令列，導覽至解壓縮的資料夾，並執行下面的命令，使用 pnputil 公用程式新增及安裝驅動程式存放區中的 INF 檔案。

```
C:\> pnputil -a PROXGB\Winx64\NDIS62\vxn62x64.inf
```

- Windows Server 2022、Windows Server 2019、Windows Server 2016、Windows Server 2012 R2 和 Windows Server 2012
  1. 在 Downloads (下載) 資料夾中，解壓縮 Wired\_driver\_ *version* \_x64.zip 檔案的內容。
  2. 在解壓縮的資料夾中，找出 Wired\_driver\_ *version* \_x64.exe 檔案並將其重新命名為 Wired\_driver\_ *version* \_x64.zip。
  3. 解壓縮 Wired\_driver\_ *version* \_x64.zip 檔案的內容。
  4. 開啟命令列，導覽至解壓縮的資料夾，並執行下面其中一個命令，使用 pnputil 公用程式新增及安裝驅動程式存放區中的 INF 檔案。
    - Windows Server 2022

```
C:\> pnputil -i -a PROXGB\Winx64\WS2022\vx.s.inf
```

- Windows Server 2019

```
C:\> pnputil -i -a PROXGB\Winx64\NDIS68\vxn68x64.inf
```

- Windows Server 2016

```
C:\> pnputil -i -a PROXGB\Winx64\NDIS65\vxn65x64.inf
```

- Windows Server 2012 R2

```
C:\> pnputil -i -a PROXGB\Winx64\NDIS64\vxn64x64.inf
```

- Windows Server 2012

```
C:\> pnputil -i -a PROXGB\Winx64\NDIS63\vxn63x64.inf
```

5. 在本機電腦上使用下列其中一個命令啟用增強型聯網屬性。

## AWS CLI

### [modify-instance-attribute](#) (AWS CLI/AWS CloudShell)

```
aws ec2 modify-instance-attribute --instance-id instance_id --sriov-net-support simple
```

## PowerShell

### [Edit-EC2InstanceAttribute](#) (AWS Tools for Windows PowerShell)

```
Edit-EC2InstanceAttribute -InstanceId instance_id -SriovNetSupport "simple"
```

- (選擇性) 從執行個體建立 AMI，如 [創建一個 Amazon EBS-backed AMI](#) 所述。AMI 將從執行個體繼承增強型聯網屬性。因此，您預設可用此 AMI 啟動其他啟用增強型聯網的執行個體。
- 從本機電腦中，使用 Amazon EC2 主控台或下列其中一個命令來啟動執行個體：[start-instances](#) (AWS CLI)、[Start-EC2Instance](#) (AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中啟動執行個體，讓執行個體狀態保持同步。

## 疑難排解連線問題

如果您在啟用增強型聯網時失去連線，則 ixgbevf 模組可能與核心不相容。請嘗試安裝您執行個體適用之 Linux 發行版本隨附的 ixgbevf 模組版本。

如果您為 PV 執行個體或 AMI 啟用增強聯網，可能導致無法連線到您的執行個體。

如需詳細資訊，請參閱[如何在 EC2 執行個體上開啟和設定增強型聯網？](#)

## 監控 EC2 執行個體的網路效能

彈性網路介面 (ENA) 驅動程式會從啟用它們的執行個體發行網路效能指標。您可以使用這些指標來進行執行個體效能的故障診對、為工作負載選擇正確的執行個體大小、主動計劃擴展活動，以及基準化分析應用程式，以判斷它們是否能夠最大化執行個體上的可用效能。

Amazon EC2 定義了執行個體層級的網路最大值，以確保高品質的聯網體驗，包括跨執行個體大小的一致網路效能。AWS 為每個執行個體提供下列項目的上限：

- 頻寬能力 - 每個 EC2 執行個體都具有最大頻寬，用於彙總傳入和傳出流量 (根據執行個體類型和大小)。一些執行個體使用網路 I/O 額度機制，根據平均頻寬使用率來配置網路頻寬。Amazon EC2 還具有進入和互聯網流量 AWS Direct Connect 的最大帶寬。如需詳細資訊，請參閱 [Amazon EC2 執行個體網路頻寬](#)。
- Packet-per-second (PPS) 效能 — 根據執行個體類型和大小，每個 EC2 執行個體都有最高的 PPS 效能。
- 追蹤的連線 - 安全性群組會追蹤每個建立的連線，以確保傳回封包如預期般交付。每個執行個體可追蹤的連線數目上限。如需詳細資訊，請參閱 [安全群組連線追蹤](#)。

- 連結本機服務存取 - Amazon EC2 可為 DNS 服務、執行個體中繼資料服務和 Amazon Time Sync Service 等服務的流量提供每個網路介面的 PPS 上限。

當執行個體的網路流量超過上限時，會排入佇列，然後卸除網路封包，以 AWS 塑造超過上限的流量。您可以使用網路效能指標監控流量何時超過上限。這些指標可即時通知您網路流量的影響，以及可能的網路效能問題。

## 目錄

- [要求](#)
- [ENA 驅動程式的指標](#)
- [檢視 執行個體的網路效能指標](#)
- [ENA Express 的指標](#)
- [適用於 ENA 的網路效能度量 \(具有 DPDK 驅動程式\)](#)
- [執行 FreeBSD 執行個體的指標](#)

## 要求

### Linux 執行個體

- 安裝 ENA 驅動程式 2.2.10 或更高版本。若要確認已安裝的版本，請使用 ethtool 命令。在下列範例中，版本符合最低需求。

```
[ec2-user ~]$ ethtool -i eth0 | grep version
version: 2.2.10
```

若要升級您的 ENA 驅動程式，請參閱[增強型聯網](#)。

- 若要將這些指標匯入 Amazon CloudWatch，請安裝 CloudWatch 代理程式。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的收集網路效能指標。
- 若要支援 contrack\_allowance\_available 指標，請安裝 ENA 驅動程式 2.8.1 版。

### Windows 執行個體

- 安裝 ENA 驅動程式 2.2.2 版或更高版本。若要確認已安裝的版本，請使用如下所示的 Device Manager (裝置管理員)。
  1. 執行 devmgmt.msc 以開啟 Device Manager (裝置管理員)。



2. 展開 Network Adapters (網路介面卡)。
3. 選擇 Amazon Elastic Network Adapter (Amazon 彈性網路介面卡)、Properties (屬性)。
4. 在 Driver (驅動程式)索引標籤上，找出 Driver Version (驅動程式版本)

若要升級您的 ENA 驅動程式，請參閱[增強型聯網](#)。

- 若要將這些指標匯入 Amazon CloudWatch，請安裝 CloudWatch 代理程式。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的「[收集進階網路指標](#)」。

## ENA 驅動程式的指標

ENA 驅動程式會即時將下列指標交付至執行個體。它們提供自上次重設驅動程式後，每個網路介面上排入佇列或丟棄的累積封包數目。

指標	描述	支援於
bw_in_allowance_exceeded	因傳入的彙總頻寬超過執行個體的上限而排入佇列或丟棄的封包數目。	所有執行個體類型
bw_out_allowance_exceeded	因傳出的彙總頻寬超過執行個體的上限而排入佇列或丟棄的封包數目。	所有執行個體類型
conntack_allowance_exceeded	因為連線追蹤超過執行個體的上限且無法建立新的連線，而丟棄的封包數目。這可能會導致傳送或傳回執行個體流量的封包遺失。	所有執行個體類型
conntack_allowance_available	在達到該執行個體類型的追蹤連線限額之前，執行個體可建立的追蹤連線數目。	僅在 <a href="#">AWS Nitro 系統上建置的執行個體</a>
linklocal_allowance_exceeded	由於本機代理伺服器服務的流量 PPS 超過網路介面上限而丟棄的封包數目。這會影響 DNS 服務、執行個體中繼資料服務和	所有執行個體類型

指標	描述	支援於
	Amazon Time Sync Service 的流量。	
pps_allowance_exceeded	因雙向 PPS 超過執行個體的上限而排入佇列或丟棄的封包數目。	所有執行個體類型

## 檢視 執行個體的網路效能指標

您使用的程序視執行個體的作業系統而定。

### Linux 執行個體

您可以將指標發佈至您最愛的工具，以視覺化方式呈現指標資料。例如，您可以 CloudWatch 使用 CloudWatch 代理程式將指標發佈到 Amazon。代理程式可讓您選取個別指標並控制發佈。

您也可以使用 `ethtool` 以擷取每個網路介面的指標，例如下方所示的 `eth0`。

```
[ec2-user ~]$ ethtool -S eth0
 bw_in_allowance_exceeded: 0
 bw_out_allowance_exceeded: 0
 pps_allowance_exceeded: 0
 contrack_allowance_exceeded: 0
 linklocal_allowance_exceeded: 0
 contrack_allowance_available: 136812
```

### Windows 執行個體

您可以使用任何 Windows 效能計數器的取用者來檢視指標。數據可以根據 `EnaPerfCounters` 清單進行解析。這是定義效能計數器提供者及其計數器集的 XML 檔案。

### 若要安裝資訊清單

如果您使用包含 ENA 驅動程式 2.2.2 或更高版本的 AMI 啟動執行個體，或使用 ENA 驅動程式 2.2.2 驅動程式套件中的安裝指令碼，則資訊清單已安裝。若要手動安裝資訊清單，請使用下列步驟：

1. 使用下列命令移除現有的資訊清單：

```
unlodctr /m:EnaPerfCounters.man
```

2. 將資訊清單檔案 `EnaPerfCounters.man` 從驅動程式安裝套件複製到 `%SystemRoot%\System32\drivers`。
3. 使用下列命令安裝新的資訊清單：

```
lodctr /m:EnaPerfCounters.man
```

### 使用效能監視器檢視測量結果

1. 開啟效能監控器。
2. 按 `Ctrl+N` 以新增計數器。
3. 從清單中選擇 `ENA Packets Shaping (ENA 封包成形)`。
4. 選取要監控的執行個體，然後選取 `Add (新增)`。
5. 選擇 `OK (確定)`。

## ENA Express 的指標

ENA Express 由可 AWS 擴展的可靠數據報 (SRD) 技術提供支持。SRD 是一種高效能的網路傳輸通訊協定，其使用動態路由來增加輸送量並將網路流量尾延遲降至最低。您可以使用 ENA Express 指標來協助確保執行個體會充分利用 SRD 技術提供的效能改進，例如：

- 評估您的資源，以確保其有足夠的容量來建立更多 SRD 連線。
- 識別存在防止合格傳出封包使用 SRD 的潛在問題的位置。
- 計算對執行個體使用 SRD 的傳出流量百分比。
- 計算對執行個體使用 SRD 的傳入流量百分比。


### Note

若要產生指標，請使用 2.8 或更新版本的驅動程式。

下列 ENA Express 指標可透過 Linux 型執行個體使用 `ethtool` 命令來取得。


- `ena_srd_mode` – 描述已啟用哪些 ENA Express 功能。相關值如下：
  - `0` = ENA Express 關閉，UDP 關閉

- 1 = ENA Express 開啟，UDP 關閉
- 2 = ENA Express 關閉，UDP 開啟

 Note

僅在原先啟用 ENA Express，且將 UDP 設定為使用它時，才會發生這種情況。會保留 UDP 流量的先前值。

- 3 = ENA Express 開啟，UDP 開啟
- `ena_srd_eligible_tx_pkts` – 在指定時間段內傳送符合 SRD 資格要求的網路封包數目，如下所示：
  - 支援傳送和接收執行個體類型。如需詳細資訊，請參閱 [ENA Express 支援的執行個體類型](#) 資料表。
  - 傳送和接收執行個體都必須設定 ENA Express。
  - 傳送和接收執行個體必須在相同的可用區域中執行。
  - 執行個體之間的網路路徑不得包含中介軟體方塊。ENA Express 目前不支援中介軟體。

 Note

ENA Express 資格指標涵蓋來源和目的地需求，以及兩個端點之間的網路。在計算後，合格封包仍可能會被取消資格。例如，如果合格封包超過最大傳輸單位 (MTU) 限制，則其會回復為標準 ENA 傳輸，儘管封包在計數器中仍會反映為合格。

- `ena_srd_tx_pkts` – 在給定時間段內傳輸的 SRD 封包數目。
- `ena_srd_rx_pkts` – 在給定時間段內接收的 SRD 封包數目。
- `ena_srd_resource_utilization` – 執行個體使用的並行 SRD 連線允許的最大記憶體使用率百分比。

若要查看針對 ENA Express 篩選的指標清單，請遵循適用於您的網路介面的 `ethtool` 命令 (在這裡顯示為 `eth0`)：

```
[ec2-user ~]$ ethtool -S eth0 | grep ena_srd
NIC statistics:
 ena_srd_mode: 0
 ena_srd_tx_pkts: 0
 ena_srd_eligible_tx_pkts: 0
```

```
ena_srd_rx_pkts: 0
ena_srd_resource_utilization: 0
```

## 出口流量 (傳出封包)

若要確保您的出口流量如預期般使用 SRD，請比較 SRD 合格封包數目 (ena\_srd\_eligible\_tx\_pkts) 與指定期間內傳送的 SRD 封包數目 (ena\_srd\_tx\_pkts)。

合格封包數目與傳送的 SRD 封包數目之間的顯著差異通常是資源使用率問題所造成的。當連接到執行個體的網路卡已耗盡其最大資源，或封包超過 MTU 限制時，合格封包無法透過 SRD 傳輸，而且必須回復為標準 ENA 傳輸。在即時移轉或即時伺服器更新期間，封包也可能會落入此間隙。需要額外的疑難排解，才能判斷根本原因。

### Note

您可以忽略合格封包數目和 SRD 封包數目之間偶爾的細微差異。例如，當您的執行個體為 SRD 流量建立與另一個執行個體的連線時，就可能會發生這種情況。

若要了解指定期間內的總出口流量使用 SRD 的百分比，請比較傳出的 SRD 封包數目 (ena\_srd\_tx\_pkts) 與該執行個體在該時間內傳送的封包總數 (NetworkPacketOut)。

## 輸入流量 (傳入封包)

若要了解傳入流量使用 SRD 的百分比，請比較指定期間內接收的 SRD 封包數目 (ena\_srd\_rx\_pkts) 與該執行個體在該時間內接收的封包總數 (NetworkPacketIn)。

## 資源使用率

資源使用率是根據單一執行個體在指定時間可以保留的並行 SRD 連線數目而定。資源使用率指標 (ena\_srd\_resource\_utilization) 會追蹤您目前執行個體的使用率。當使用率接近 100% 時，您預期將會看到效能問題。ENA Express 從 SRD 回復為標準 ENA 傳輸，且捨棄封包的可能性增加。高資源使用率表示是時候擴展執行個體，以改善網路效能。

### Note

當執行個體的網路流量超過上限時，會排入佇列，然後卸除網路封包，以 AWS 塑造超過上限的流量。

## Persistence

在啟用執行個體的 ENA Express 時，會累積輸出和輸入指標。如果停用 ENA Express，則指標就會停止累積，但只要執行個體仍在執行，就會持續存在。如果執行個體重新啟動或終止，或者如果網路介面已從執行個體中斷連接，就會重設指標。

## 適用於 ENA 的網路效能度量 (具有 DPDK 驅動程式)

ENA 驅動程式 2.2.0 版及更高版本支援網路指標報告。DPDK 20.11 包含 ENA 驅動程式 2.2.0，是第一個支援此功能的 DPDK 版本。

您可以使用範例應用程式來檢視 DPDK 統計資料。若要啟動範例應用程式的互動式版本，請執行下列命令。

```
./app/dpdk-testpmd -- -i
```

在此互動式工作階段中，您可以輸入命令來擷取連接埠的延伸統計資料。下列範例命令會擷取連接埠 0 的統計資料。

```
show port xstats 0
```

以下是包含 DPDK 範例應用程式的互動式工作階段範例。

```
[root@ip-192.0.2.0 build]# ./app/dpdk-testpmd -- -i
EAL: Detected 4 lcore(s)
EAL: Detected 1 NUMA nodes
EAL: Multi-process socket /var/run/dpdk/rte/mp_socket
EAL: Selected IOVA mode 'PA'
EAL: Probing VFIO support...
EAL: Invalid NUMA socket, default to 0
EAL: Invalid NUMA socket, default to 0
EAL: Probe PCI driver: net_ena (1d0f:ec20) device: 0000:00:06.0
(socket 0)
EAL: No legacy callbacks, legacy socket not created
Interactive-mode selected

Port 0: link state change event
testpmd: create a new mbuf pool <mb_pool_0>: n=171456,
size=2176, socket=0
testpmd: preferred mempool ops selected: ring_mp_mc

Warning! port-topology=paired and odd forward ports number, the
last port will pair with itself.
```

```
Configuring Port 0 (socket 0)
Port 0: 02:C7:17:A2:60:B1
Checking link statuses...
Done
Error during enabling promiscuous mode for port 0: Operation
not supported - ignore
testpmd> show port xstats 0
NIC extended statistics for port 0
rx_good_packets: 0
tx_good_packets: 0
rx_good_bytes: 0
tx_good_bytes: 0
rx_missed_errors: 0
rx_errors: 0
tx_errors: 0
rx_mbuf_allocation_errors: 0
rx_q0_packets: 0
rx_q0_bytes: 0
rx_q0_errors: 0
tx_q0_packets: 0
tx_q0_bytes: 0
wd_expired: 0
dev_start: 1
dev_stop: 0
tx_drops: 0
bw_in_allowance_exceeded: 0
bw_out_allowance_exceeded: 0
pps_allowance_exceeded: 0
contrack_allowance_exceeded: 0
linklocal_allowance_exceeded: 0
rx_q0_cnt: 0
rx_q0_bytes: 0
rx_q0_refill_partial: 0
rx_q0_bad_csum: 0
rx_q0_mbuf_alloc_fail: 0
rx_q0_bad_desc_num: 0
rx_q0_bad_req_id: 0
tx_q0_cnt: 0
tx_q0_bytes: 0
tx_q0_prepare_ctx_err: 0
tx_q0_linearize: 0
tx_q0_linearize_failed: 0
tx_q0_tx_poll: 0
tx_q0_doorbells: 0
```

```
tx_q0_bad_req_id: 0
tx_q0_available_desc: 1023
testpmd>
```

如需範例應用程式及使用它來擷取延伸統計資料的詳細資訊，請參閱 DPDK 文件中的 [Testpmd 應用程式使用者指南](#)。

## 執行 FreeBSD 執行個體的指標

從 2.3.0 版開始，ENA FreeBSD 驅動程式支援收集執行 FreeBSD 執行個體的網路效能指標。若要啟用收集 FreeBSD 指標，請輸入下列命令，並將 *Interval* (間隔) 設定為介於 1 到 3600 之間的值。這會指定收集 FreeBSD 指標的頻率 (以秒為單位)。

```
sysctl dev.ena.network_interface.eni_metrics.sample_interval=interval
```

例如，下列命令會將驅動程式設定為每 10 秒在網路介面 1 上收集一次 FreeBSD 指標：

```
sysctl dev.ena.1.eni_metrics.sample_interval=10
```

若要關閉收集 FreeBSD 指標，您可以執行上述命令並指定 0 作為 *interval* (間隔)。

啟用收集 FreeBSD 指標之後，您可以執行下列命令來擷取最新一組收集的量度。

```
sysctl dev.ena.network_interface.eni_metrics
```

## 疑難排解 Linux 上的彈性網路介面卡

彈性網路轉接器 (ENA) 設計旨在提高作業系統運作狀態，並減少由於預料之外硬體行為及/或故障而導致長期中斷之可能性。ENA 架構盡可能使裝置或驅動程式的故障對系統保持透明。本主題提供 ENA 的故障診斷資訊。

如果您無法連線至您的執行個體，請由 [疑難排解連線問題](#) 章節開始。

如果您在遷移到第六代執行個體類型後遇到效能降低的情況，請參閱 [將 EC2 執行個體遷移到第六代執行個體之前，我需要做些什麼，以確保我獲得最大的網路效能？](#)

如果您能夠連線至您的執行個體，則可以使用本主題後面章節中說明的故障診斷和復原機制來收集診斷資訊。

### 內容

- [疑難排解連線問題](#)



- [保持有效機制](#)
- [註冊讀取逾時](#)
- [統計資料](#)
- [syslog 中的驅動程式錯誤日誌](#)
- [非最佳化組態通知](#)

## 疑難排解連線問題

如果您在啟用增強聯網時失去連線，則 ena 模組可能與您的執行個體目前執行之核心不相容。如果您為特定核心版本 (不含 dkms 或具有設定不正確的 dkms.conf 檔案) 安裝模組並更新執行個體核心，則可能會發生此情況。如果在開機階段載入的執行個體核心沒有正確安裝 ena 模組，則您的執行個體將無法識別網路轉接器，且將無法連接到您的執行個體。

如果您為 PV 執行個體或 AMI 啟用增強聯網，也可能導致無法連接到您的執行個體。

如果您的執行個體在啟用增強聯網與 ENA 後即無法連接，您可以為您的執行個體停用 enaSupport 屬性，以將其回復到原始網路轉接器。

### 停用增強聯網與 ENA (EBS 後端執行個體)

1. 在本機電腦上，使用 Amazon EC2 主控台或下列其中一個命令停止執行個體：[停止執行個體](#) (AWS CLI)、[Stop-EC2Instance](#)(AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中停止執行個體，讓執行個體狀態保持同步。

#### Important

如果您使用執行個體後端執行個體，則無法停止該執行個體。請改為繼續執行 [停用增強聯網與 ENA \(執行個體後端執行個體\)](#)。

2. 在本機電腦上使用下列命令停用增強聯網屬性。

- [modify-instance-attribute](#) (AWS CLI)

```
$ C:\> aws ec2 modify-instance-attribute --instance-id instance_id --no-ena-support
```

3. 從本機電腦使用 Amazon EC2 主控台或下列其中一個命令啟動執行個體：[啟動執行個體](#) (AWS CLI)、[Start-EC2Instance](#)(AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中啟動執行個體，讓執行個體狀態保持同步。

4. (選用) 請連線到您的執行個體，並嘗試遵循 `ena` 中的步驟使用目前核心版本來重新安裝 [使用 EC2 執行個體上的彈性網路介面卡 \(ENA\) 啟用增強型聯網](#) 模組。

停用增強聯網與 ENA (執行個體後端執行個體)

如果您的執行個體為執行個體後端執行個體，請建立新的 AMI，如 [建立執行個體存放區後端 Linux AMI](#) 中所述。請確保在註冊 AMI 時停用增強聯網 `enaSupport` 屬性。

- [register-image](#) (AWS CLI)

```
$ C:\> aws ec2 register-image --no-ena-support ...
```

- [Register-EC2Image](#) (AWS Tools for Windows PowerShell)

```
C:\> Register-EC2Image -EnaSupport $false ...
```

## 保持有效機制

ENA 裝置以固定速率張貼保持有效事件 (通常每秒一次)。ENA 驅動程式實作監視程式機制，該機制會檢查這些保持有效訊息的存在。如果有訊息存在，監視程式會重新啟動，否則驅動程式會認為裝置出現故障並執行下列作業：

- 傾印目前統計資料至 `syslog`
- 重設 ENA 裝置
- 重設 ENA 驅動程式狀態

上述重設程序可能會在短時間內導致一些流量遺失 (TCP 連線應該能夠復原)，但不應影響使用者。

ENA 裝置也可以透過不傳送保持有效通知來間接請求裝置重設程序，例如，如果 ENA 裝置在載入無法恢復的組態之後成為未知狀態。

以下是重設程序的範例：

```
[18509.800135] ena 0000:00:07.0 eth1: Keep alive watchdog timeout. // The watchdog
process initiates a reset
[18509.815244] ena 0000:00:07.0 eth1: Trigger reset is on
[18509.825589] ena 0000:00:07.0 eth1: tx_timeout: 0 // The driver logs the current
statistics
[18509.834253] ena 0000:00:07.0 eth1: io_suspend: 0
```

```
[18509.842674] ena 0000:00:07.0 eth1: io_resume: 0
[18509.850275] ena 0000:00:07.0 eth1: wd_expired: 1
[18509.857855] ena 0000:00:07.0 eth1: interface_up: 1
[18509.865415] ena 0000:00:07.0 eth1: interface_down: 0
[18509.873468] ena 0000:00:07.0 eth1: admin_q_pause: 0
[18509.881075] ena 0000:00:07.0 eth1: queue_0_tx_cnt: 0
[18509.888629] ena 0000:00:07.0 eth1: queue_0_tx_bytes: 0
[18509.895286] ena 0000:00:07.0 eth1: queue_0_tx_queue_stop: 0
.....
.....
[18511.280972] ena 0000:00:07.0 eth1: free uncompleted tx skb qid 3 idx 0x7 // At the
end of the down process, the driver discards incomplete packets.
[18511.420112] [ENA_COM: ena_com_validate_version] ena device version: 0.10 //The
driver begins its up process
[18511.420119] [ENA_COM: ena_com_validate_version] ena controller version: 0.0.1
implementation version 1
[18511.420127] [ENA_COM: ena_com_admin_init] ena_defs : Version:[b9692e8] Build date
[Wed Apr 6 09:54:21 IDT 2016]
[18512.252108] ena 0000:00:07.0: Device watchdog is Enabled
[18512.674877] ena 0000:00:07.0: irq 46 for MSI/MSI-X
[18512.674933] ena 0000:00:07.0: irq 47 for MSI/MSI-X
[18512.674990] ena 0000:00:07.0: irq 48 for MSI/MSI-X
[18512.675037] ena 0000:00:07.0: irq 49 for MSI/MSI-X
[18512.675085] ena 0000:00:07.0: irq 50 for MSI/MSI-X
[18512.675141] ena 0000:00:07.0: irq 51 for MSI/MSI-X
[18512.675188] ena 0000:00:07.0: irq 52 for MSI/MSI-X
[18512.675233] ena 0000:00:07.0: irq 53 for MSI/MSI-X
[18512.675279] ena 0000:00:07.0: irq 54 for MSI/MSI-X
[18512.772641] [ENA_COM: ena_com_set_hash_function] Feature 10 isn't supported
[18512.772647] [ENA_COM: ena_com_set_hash_ctrl] Feature 18 isn't supported
[18512.775945] ena 0000:00:07.0: Device reset completed successfully // The reset
process is complete
```

## 註冊讀取逾時

ENA 架構暗示記憶體映射之 I/O (MMIO) 讀取操作的限量使用。ENA 裝置驅動程式僅在其初始化程序中存取 MMIO 註冊。

如果驅動程式日誌 (在 `dmesg` 輸出中可用) 指出讀取操作失敗，則可能是由不相容或編譯錯誤的驅動程式、忙碌的硬體裝置或硬體故障引起的。

表示讀取操作失敗的間歇日誌項目不應視為問題；驅動程式會在此情況下將其重試。但是，包含讀取失敗的一系列日誌項目則會指出驅動程式或硬體問題。

以下是驅動程序日誌項目的範例，指出由於逾時而導致的讀取操作失敗：

```
[47.113698] [ENA_COM: ena_com_reg_bar_read32] reading reg failed for timeout.
expected: req id[1] offset[88] actual: req id[57006] offset[0]
[47.333715] [ENA_COM: ena_com_reg_bar_read32] reading reg failed for timeout.
expected: req id[2] offset[8] actual: req id[57007] offset[0]
[47.346221] [ENA_COM: ena_com_dev_reset] Reg read32 timeout occurred
```

## 統計資料

如果您遇到網路效能不足或延遲問題，則應擷取並檢查裝置統計資料。您可以使用 `ethtool` 來獲得這些統計資料，如下所示。

```
[ec2-user ~]$ ethtool -S ethN
NIC statistics:
tx_timeout: 0
suspend: 0
resume: 0
wd_expired: 0
interface_up: 1
interface_down: 0
admin_q_pause: 0
bw_in_allowance_exceeded: 0
bw_out_allowance_exceeded: 0
pps_allowance_exceeded: 0
contrack_allowance_available: 450878
contrack_allowance_exceeded: 0
linklocal_allowance_exceeded: 0
queue_0_tx_cnt: 4329
queue_0_tx_bytes: 1075749
queue_0_tx_queue_stop: 0
...
```

以下說明下列命令輸出參數：

`tx_timeout` : *N*

Netdev 監視程式啟用的次數。

`suspend` : *N*

驅動程式執行暫停操作的次數。

`resume` : *N*

驅動程式執行繼續操作的次數。

`wd_expired` : *N*

驅動程式在前 3 秒內未收到保持有效事件的次數。

`interface_up` : *N*

ENA 介面出現的次數。

`interface_down` : *N*

ENA 介面關閉的次數。

`admin_q_pause` : *N*

在執行中狀態下找不到管理員佇列的次數。

`bw_in_allowance_exceeded` : *N*

因傳入的彙總頻寬超過執行個體的上限而排入佇列或丟棄的封包數目。

`bw_out_allowance_exceeded` : *N*

因傳出的彙總頻寬超過執行個體的上限而排入佇列或丟棄的封包數目。

`pps_allowance_exceeded` : *N*

因雙向 PPS 超過執行個體的上限而排入佇列或丟棄的封包數目。

`contrack_allowance_available` : *N*

在達到該執行個體類型的追蹤連線限額之前，執行個體可建立的追蹤連線數目。僅適用於 Nitro 型執行個體。不支援 FreeBSD 執行個體或 DPDK 環境。

`contrack_allowance_exceeded` : *N*

因為連線追蹤超過執行個體的上限且無法建立新的連線，而丟棄的封包數目。這可能會導致傳送或傳回執行個體流量的封包遺失。

`linklocal_allowance_exceeded` : *N*

由於本機代理伺服器服務的流量 PPS 超過網路介面上限而丟棄的封包數目。這會影響 DNS 服務、執行個體中繼資料服務和 Amazon Time Sync Service 的流量。

`queue_N_tx_cnt` : *N*

此佇列的已傳輸封包數目。

`queue_N_tx_bytes` : *N*

此佇列的已傳輸位元組數目。

`queue_N_tx_queue_stop` : *N*

佇列 *N* 已滿並停止的次數。

`queue_N_tx_queue_wakeup` : *N*

佇列 *N* 在停止後恢復的次數。

`queue_N_tx_dma_mapping_err` : *N*

直接記憶體存取錯誤計數。如果此值為 0，則指出系統資源不足。

`queue_N_tx_linearize` : *N*

此佇列的 SKB 線性化嘗試次數。

`queue_N_tx_linearize_failed` : *N*

此佇列的 SKB 線性化失敗次數。

`queue_N_tx_napi_comp` : *N*

napi 處理常式為此佇列呼叫 `napi_complete` 的次數。

`queue_N_tx_tx_poll` : *N*

napi 處理常式為此佇列排程的次數。

`queue_N_tx_doorbells` : *N*

此佇列的傳輸門鈴數目。

`queue_N_tx_prepare_ctx_err` : *N*

此佇列的 `ena_com_prepare_tx` 失敗次數。

`queue_N_tx_bad_req_id` : *N*

此佇列的無效 `req_id`。有效的 `req_id` 為零，減去 `queue_size`，減去 1。

`queue_N_tx_llq_buffer_copy` : *N*

其標頭大小大於此佇列 llq 項目的封包數目。

`queue_N_tx_missed_tx` : *N*

此佇列未完成的封包數目。

`queue_N_tx_unmask_interrupt` : *N*

此佇列 tx 中斷取消遮罩的次數。

`queue_N_rx_cnt` : *N*

此佇列接收的封包數目。

`queue_N_rx_bytes` : *N*

此佇列接收的位元組數目。

`queue_N_rx_rx_copybreak_pkt` : *N*

此佇列的 rx 佇列接收到小於 rx\_copybreak 封包大小之封包的次數。

`queue_N_rx_csum_good` : *N*

此佇列的 rx 佇列接收到檢查總和已檢查且正確之封包的次數。

`queue_N_rx_refil_partial` : *N*

驅動程式未成功以此佇列緩衝區重新填滿 rx 佇列之空白部分的次數。如果值不為 0，則指出記憶體資源不足。

`queue_N_rx_bad_csum` : *N*

rx 佇列對此佇列有不良檢查總和的次數 (僅當支援 rx 檢查總和卸載時)。

`queue_N_rx_page_alloc_fail` : *N*

此佇列頁面配置失敗的次數。如果值不為 0，則指出記憶體資源不足。

`queue_N_rx_skb_alloc_fail` : *N*

此佇列 SKB 配置失敗的次數。如果值不為 0，則指出系統資源不足。

`queue_N_rx_dma_mapping_err` : *N*

直接記憶體存取錯誤計數。如果此值為 0，則指出系統資源不足。

`queue_N_rx_bad_desc_num` : *N*

每個封包的緩衝區過多。如果值不為 0，則表示使用非常小的緩衝區。

`queue_N_rx_bad_req_id` : *N*

此佇列的 req\_id 無效。有效的 req\_id 是從 [0, queue\_size - 1] 開始。

`queue_N_rx_empty_rx_ring` : *N*

此佇列的 rx 佇列為空白的次數。

`queue_N_rx_csum_unchecked` : *N*

此佇列的 rx 佇列接收到檢查總和尚未檢查之封包的次數。

`queue_N_rx_xdp_aborted` : *N*

XDP 封包歸類為 XDP\_ABORT 的次數。

`queue_N_rx_xdp_drop` : *N*

XDP 封包歸類為 XDP\_DROP 的次數。

`queue_N_rx_xdp_pass` : *N*

XDP 封包歸類為 XDP\_PASS 的次數。

`queue_N_rx_xdp_tx` : *N*

XDP 封包歸類為 XDP\_TX 的次數。

`queue_N_rx_xdp_invalid` : *N*

封包的 XDP 傳回碼無效的次數。

`queue_N_rx_xdp_redirect` : *N*

XDP 封包歸類為 XDP\_REDIRECT 的次數。

`queue_N_xdp_tx_cnt` : *N*

此佇列的已傳輸封包數目。

`queue_N_xdp_tx_bytes` : *N*

此佇列的已傳輸位元組數目。

`queue_N_xdp_tx_queue_stop` : *N*

此佇列已滿並停止的次數。



`queue_N_xdp_tx_queue_wakeup` : *N*

此佇列在停止後恢復的次數。

`queue_N_xdp_tx_dma_mapping_err` : *N*

直接記憶體存取錯誤計數。如果此值為 0，則指出系統資源不足。

`queue_N_xdp_tx_linearize` : *N*

此佇列的 XDP 緩衝區線性化嘗試次數。

`queue_N_xdp_tx_linearize_failed` : *N*

此佇列的 XDP 緩衝區線性化失敗次數。

`queue_N_xdp_tx_napi_comp` : *N*

napi 處理常式為此佇列呼叫 `napi_complete` 的次數。

`queue_N_xdp_tx_tx_poll` : *N*

napi 處理常式為此佇列排程的次數。

`queue_N_xdp_tx_doorbells` : *N*

此佇列的傳輸門鈴數目。

`queue_N_xdp_tx_prepare_ctx_err` : *N*

此佇列的 `ena_com_prepare_tx` 失敗次數。此值應始終為 0；如不是，請查看驅動程式日誌。

`queue_N_xdp_tx_bad_req_id` : *N*

此佇列的 `req_id` 無效。有效的 `req_id` 是從 `[0, queue_size - 1]` 開始。

`queue_N_xdp_tx_llq_buffer_copy` : *N*

此佇列使用 llq 緩衝區複本複製其標頭的封包數目。

`queue_N_xdp_tx_missed_tx` : *N*

tx 佇列項目錯過此佇列完成逾時的次數。

`queue_N_xdp_tx_unmask_interrupt` : *N*

此佇列 tx 中斷取消遮罩的次數。

`ena_admin_q_aborted_cmd` : *N*

管理命令中止次數。此情況通常發生在自動復原程序中。

ena\_admin\_q\_submitted\_cmd : *N*

管理佇列門鈴數目。

ena\_admin\_q\_completed\_cmd : *N*

管理佇列完成數目。

ena\_admin\_q\_out\_of\_space : *N*

驅動程式嘗試提交新管理命令但佇列已滿的次數。

ena\_admin\_q\_no\_completion : *N*

驅動程式未取得命令的管理完成之次數。

## syslog 中的驅動程式錯誤日誌

在系統開機期間，ENA 驅動程式會將日誌訊息寫入至 syslog。如果您遇到問題，可以檢查這些日誌以尋找錯誤。以下是系統開機時，ENA 驅動程式在 syslog 中記錄的資訊，以及選取訊息的一些註釋範例。

```
Jun 3 22:37:46 ip-172-31-2-186 kernel: [478.416939] [ENA_COM:
 ena_com_validate_version] ena device version: 0.10
Jun 3 22:37:46 ip-172-31-2-186 kernel: [478.420915] [ENA_COM:
 ena_com_validate_version] ena controller version: 0.0.1 implementation version 1
Jun 3 22:37:46 ip-172-31-2-186 kernel: [479.256831] ena 0000:00:03.0: Device
 watchdog is Enabled
Jun 3 22:37:46 ip-172-31-2-186 kernel: [479.672947] ena 0000:00:03.0: creating 8 io
 queues. queue size: 1024
Jun 3 22:37:46 ip-172-31-2-186 kernel: [479.680885] [ENA_COM:
 ena_com_init_interrupt_moderation] Feature 20 isn't supported // Interrupt moderation
 is not supported by the device
Jun 3 22:37:46 ip-172-31-2-186 kernel: [479.691609] [ENA_COM:
 ena_com_get_feature_ex] Feature 10 isn't supported // RSS HASH function configuration
 is not supported by the
 device
Jun 3 22:37:46 ip-172-31-2-186 kernel: [479.694583] [ENA_COM:
 ena_com_get_feature_ex] Feature 18 isn't supported //RSS HASH input source
 configuration is not supported by the device
Jun 3 22:37:46 ip-172-31-2-186 kernel: [479.697433] [ENA_COM:
 ena_com_set_host_attributes] Set host attribute isn't supported
Jun 3 22:37:46 ip-172-31-2-186 kernel: [479.701064] ena 0000:00:03.0 (unnamed
 net_device) (uninitialized): Cannot set host attributes
```

```
Jun 3 22:37:46 ip-172-31-2-186 kernel: [479.704917] ena 0000:00:03.0: Elastic
Network Adapter (ENA) found at mem f3000000, mac addr 02:8a:3c:1e:13:b5 Queues 8
Jun 3 22:37:46 ip-172-31-2-186 kernel: [480.805037] EXT4-fs (xvda1): re-mounted.
Opts: (null)
Jun 3 22:37:46 ip-172-31-2-186 kernel: [481.025842] NET: Registered protocol family
10
```

我可以忽略哪些錯誤？

彈性網路轉接器可以忽略下列可能出現在系統錯誤日誌中的警告：

Set host attribute isn't supported

此裝置不支援主機屬性。

failed to alloc buffer for rx queue

此為可復原的錯誤，指出在錯誤拋出時可能存在記憶體壓力問題。

Feature **X** isn't supported

參考的功能不受彈性網路轉接器支援。**X** 可能的值包括：

- **10**：此裝置不支援 RSS 雜湊功能組態。
- **12**：此裝置不支援 RSS 間接資料表組態。
- **18**：此裝置不支援 RSS 雜湊輸入組態。
- **20**：此裝置不支援中斷管制。
- **27**：彈性網路轉接器驅動程式不支援從 snmpd 中輪詢乙太網路功能。

Failed to config AENQ

彈性網路轉接器不支援 AENQ 組態。

Trying to set unsupported AENQ events

此錯誤指出嘗試設定彈性網路轉接器不支援的 AENQ 事件群組。

## 非最佳化組態通知

ENA 裝置會偵測驅動程式中您可以變更的非最佳化組態設定。裝置會通知 ENA 驅動程式，並將警告記錄到主控台。下列範例顯示警告訊息的格式。

```
Sub-optimal configuration notification code: 1. Refer to AWS ENA documentation for
additional details and mitigation options.
```

下列清單顯示通知程式碼詳細資訊，以及非最佳化組態調查結果的建議動作。

- 程式碼 1：不建議使用帶寬 LLQ 組態的 ENA Express

ENA Express ENI 設定有寬 LLQ。此組態不是最佳化的，可能會影響 ENA Express 的效能。我們建議您在使用 ENA Express ENI 時停用寬 LLQ 設定，如下所示。

```
sudo rmmod ena && sudo modprobe ena force_large_llq_header=0
```

如需 ENA Express 最佳化組態的詳細資訊，請參閱 [在 EC2 執行個體上使用 ENA 快速提升網路效能](#)。

- 代碼 2：不建議使用具有次佳 Tx 佇列深度的 ENA Express ENI

設定 ENA Express ENI 為具有次佳 Tx 佇列深度。此組態可能會影響 ENA Express 的效能。使用 ENA Express ENI 時，建議您將所有 Tx 佇列擴大至網路介面的最大值，如下所示。

您可以運行以下 ethtool 命令來調整 LLQ 大小。若要進一步了解如何控制、查詢和啟用寬 LLQ，請參閱 Amazon 驅動程式存放庫中 ENA 的 Linux 核心驅動程式的 [大型低延遲佇列 \(大型 LLQ\)](#) 主題。  
GitHub

```
ethtool -g interface
```

將您的 Tx 隊列設置為最大深度：

```
ethtool -G interface tx depth
```

如需 ENA Express 最佳化組態的詳細資訊，請參閱 [在 EC2 執行個體上使用 ENA 快速提升網路效能](#)。

- 代碼 3：具有常規 LLQ 大小的 ENA 和 Tx 數據包流量超過標頭支持的最大大小

根據預設，ENA LLQ 支援 Tx 封包標頭大小，最大可達 96 個位元組。如果封包標頭大小超過 96 個位元組，則會捨棄封包。若要緩解此問題，建議您啟用 Wide LLQ，如此可將支援的 Tx 封包標頭大小增加到最多 224 個位元組。

但是，當您啟用寬範圍時，Tx 環的最大尺寸從 1000 減少到 512 個項目。根據預設，所有 Nitro v4 及更新版本的執行個體類型都會啟用寬範圍。

- Nitro v4 實例類型的默認最大寬 LLQ Tx 環形大小為 512 個項目，無法更改。

- ~~硝基 v5 實例類型具有 512 個項目的默認寬 LLQ Tx 環形大小，最多可以增加 1000 個項目。~~

您可以運行以下 `ethtool` 命令來調整 LLQ 大小。若要進一步了解如何控制、查詢和啟用寬 LLQ，請參閱 Amazon 驅動程式存放庫中 ENA 的 Linux 核心驅動程式的 [大型低延遲佇列 \(大型 LLQ\)](#) 主題。  
GitHub

查找 Tx 隊列的最大深度：

```
ethtool -g interface
```

將您的 Tx 隊列設置為最大深度：

```
ethtool -G interface tx depth
```

## 排解彈性網路介面卡 Windows 驅動程式

彈性網路轉接器 (ENA) 設計旨在提高作業系統運作狀態，並減少可能中斷 Windows 執行個體操作的預料之外硬體行為或故障。ENA 架構盡可能使裝置或驅動程式的故障對作業系統保持透明。

### 安裝彈性網路介面卡 (ENA) 驅動程式

如果您的執行個體不是以 Amazon 提供的最新 Windows Amazon Machine Image (AMI) 為基礎，則請使用下列程序在執行個體上安裝目前的 ENA 驅動程式。您應該在方便重新啟動執行個體時，執行這項更新。如果安裝指令碼未自動重新啟動執行個體，則建議您重新啟動執行個體作為最後一步。

如果在執行個體執行時使用執行個體儲存體磁碟區來儲存資料，則當停止執行個體時會清除該資料。停止執行個體之前，請確認您已將所需的任何資料從執行個體儲存體磁碟區複製到永久性儲存，例如 Amazon EBS 或 Amazon S3。

#### 必要條件

若要安裝或升級 ENA 驅動程式，Windows 執行個體必須符合以下先決條件：

- 已安裝 3.0 或更新 PowerShell 版本

#### 步驟 1：備份資料

建議您建立備份 AMI，以防您無法透過 Device Manager 復原變更。若要使用建立備份 AMI AWS Management Console，請依照下列步驟執行：

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取需要升級驅動程式的執行個體，並從執行個體狀態選單中選擇停止執行個體。
4. 在停止執行個體後，再次選取執行個體。若要建立備份，請從動作選單中選擇映像和範本，然後選擇建立映像。
5. 若要重新啟動執行個體，請從執行個體狀態選單中選擇啟動執行個體。

## 步驟 2：安裝或升級 ENA 驅動程式

您可以使用 AWS Systems Manager 散發者或 PowerShell 指令程式來安裝或升級 ENA 驅動程式。如需進一步指示，請選取與您要使用之方法相符的索引標籤。

### Systems Manager Distributor

您可以使用 Systems Manager Distributor 功能，將套件部署至 Systems Manager 管理的節點。您可以使用 Systems Manager Distributor 一次安裝 ENA 驅動程式套件，或使用排程的更新進行安裝。如需有關如何使用 Systems Manager Distributor 安裝 ENA 驅動程式套件 (AwsEnaNetworkDriver) 的詳細資訊，請參閱《AWS Systems Manager 使用者指南》中的 [安裝或更新套件](#)。

### PowerShell

本節說明如何使用 PowerShell cmdlet 在執行個體上下載及安裝 ENA 驅動程式套件。

#### 選項 1：下載並解壓縮最新版本

1. 連線到您的執行個體，並以本機管理員登入。
2. 使用 `invoke-webrequest` cmdlet 下載最新的驅動程式套件：

```
PS C:\> invoke-webrequest https://ec2-windows-drivers-
downloads.s3.amazonaws.com/ENA/Latest/AwsEnaNetworkDriver.zip -
outfile $env:USERPROFILE\AwsEnaNetworkDriver.zip
```

#### Note

如果您在下載檔案時收到錯誤訊息，而且您使用的是 Windows 伺服器 2016 或更早版本，則可能需要為您的 PowerShell 終端機啟用 TLS 1.2。您可以使用下列命令為目前 PowerShell 工作階段啟用 TLS 1.2，然後再試一次：

```
[Net.ServicePointManager]::SecurityProtocol =
[Net.SecurityProtocolType]::Tls12
```

或者，可以從執行個體的瀏覽器視窗下載最新的驅動程式套件。

3. 使用 `expand-archive cmdlet` 來解壓縮下載至執行個體的 zip 封存檔：

```
PS C:\> expand-archive $env:userprofile\AwsEnaNetworkDriver.zip -
DestinationPath $env:userprofile\AwsEnaNetworkDriver
```

### 選項 2：下載並解壓縮特定版本

1. 連線到您的執行個體，並以本機管理員登入。
2. 從 [視窗 ENA 驅動](#) 表格中的版本連結下載您想要的特定版本的 ENA 驅動程式套件。
3. 將 zip 封存檔解壓縮至執行個體。

### 使用安裝 ENA 驅動程式 PowerShell

無論您是下載最新的驅動程式還是特定版本，安裝步驟都相同。若要安裝 ENA 驅動程式，請遵循下列步驟。

1. 若要安裝驅動程式，請從 `install.ps1` PowerShell 執行個體上的 `AwsEnaNetworkDriver` 目錄執行指令碼。如果出現錯誤，請確定您使用的是 PowerShell 3.0 或更新版本。
2. 如果安裝程式未自動重新啟動執行個體，請執行 `Restart-Computer` PowerShell 指令碼。


```
PS C:\> Restart-Computer
```

### 步驟 3 (選用)：安裝後驗證 ENA 驅動程式版本

若要確保 ENA 驅動程式套件已成功安裝在執行個體上，可以依照下列步驟驗證新版本：

1. 連線到您的執行個體，並以本機管理員登入。
2. 若要開啟 Windows Device Manager (Windows 裝置管理員)，請在 Run (執行) 對話方塊中輸入 `devmgmt.msc`。

3. 選擇確定。這將開啟 Device Manager (裝置管理員) 視窗。
4. 選取 Network adapters (網路轉接器) 左側的箭頭以展開清單。
5. 選取名稱，或開啟 Amazon Elastic Network Adapter (Amazon 彈性網路轉接器) 的內容功能表，然後選擇 Properties (屬性)。這會開啟 Amazon 彈性網路介面卡屬性對話方塊。

 Note


ENA 介面卡都使用相同的驅動程式。如果您有多個 ENA 介面卡，則可以選取其中一個來更新所有 ENA 介面卡的驅動程式。

6. 若要驗證目前安裝的版本，請開啟驅動程式索引標籤並檢查驅動程式版本。如果目前版本與目標版本不相符，則請參閱 [排解彈性網路介面卡 Windows 驅動程式](#)。

### 復原 ENA 驅動程式安裝

如果安裝出現任何問題，則您可能需要復原驅動程序。請依照下列步驟復原至執行個體上安裝的 ENA 驅動程式的先前版本。

1. 連線到您的執行個體，並以本機管理員登入。
2. 若要開啟 Windows Device Manager (Windows 裝置管理員)，請在 Run (執行) 對話方塊中輸入 `devmgmt.msc`。
3. 選擇確定。這將開啟 Device Manager (裝置管理員) 視窗。
4. 選取 Network adapters (網路轉接器) 左側的箭頭以展開清單。
5. 選取名稱，或開啟 Amazon Elastic Network Adapter (Amazon 彈性網路轉接器) 的內容功能表，然後選擇 Properties (屬性)。這會開啟 Amazon 彈性網路介面卡屬性對話方塊。

 Note

ENA 介面卡都使用相同的驅動程式。如果您有多個 ENA 介面卡，則可以選取其中一個來更新所有 ENA 介面卡的驅動程式。

6. 若要復原驅動程式，請開啟驅動程式索引標籤並選擇復原驅動程式。這會開啟驅動程式套件復原視窗。



**Note**

如果驅動程式索引標籤未顯示復原驅動程式動作，或者動作無法使用，則表示執行個體上的 [驅動程式存放區](#) 不包含先前安裝的驅動程式套件。若要疑難排解此問題，請參閱 [故障診斷方案](#) 並展開非預期的 ENA 驅動程式版本已安裝部分。如需有關裝置驅動程式套件選取程序的詳細資訊，請參閱 Microsoft 文件網站上的 [Windows 如何為裝置選取驅動程式套件](#)。

## 收集有關執行個體的診斷資訊

開啟 Windows 作業系統 (OS) 工具的步驟會有所不同，具體取決於您執行個體上安裝的作業系統版本。在以下各部分，我們使用 Run (執行) 對話方塊開啟工具，該工具在所有作業系統版本的操作方式均相同。但是，您可以使用偏好的任何方法來存取這些工具。

### 存取 Run (執行) 對話方塊

- 使用 Windows 標誌鍵組合：Windows + R
- 使用搜尋列：
  - 在搜尋列中，輸入 run。
  - 從搜尋結果中選擇 Run (執行) 應用程式。

某些步驟需要內容功能表才能存取屬性或內容相關動作。根據您的作業系統版本和硬體，可透過多種方式執行此作業。

### 存取內容功能表

- 使用滑鼠：在項目上按一下滑鼠右鍵以顯示其內容功能表。
- 使用鍵盤：
  - 根據您的作業系統版本，使用 Shift + F10 或 Ctrl + Shift + F10。
  - 如果鍵盤上有內容鍵 (方塊中帶有三條水平線)，請選擇所需項目，然後按內容鍵。

如果可以連接到自己的執行個體，請使用以下技術收集診斷資訊以進行故障診斷。

## 檢查 ENA 裝置狀態

若要使用 Windows Device Manager 檢查 ENA Windows 驅動程式的狀態，請按照下列步驟操作：

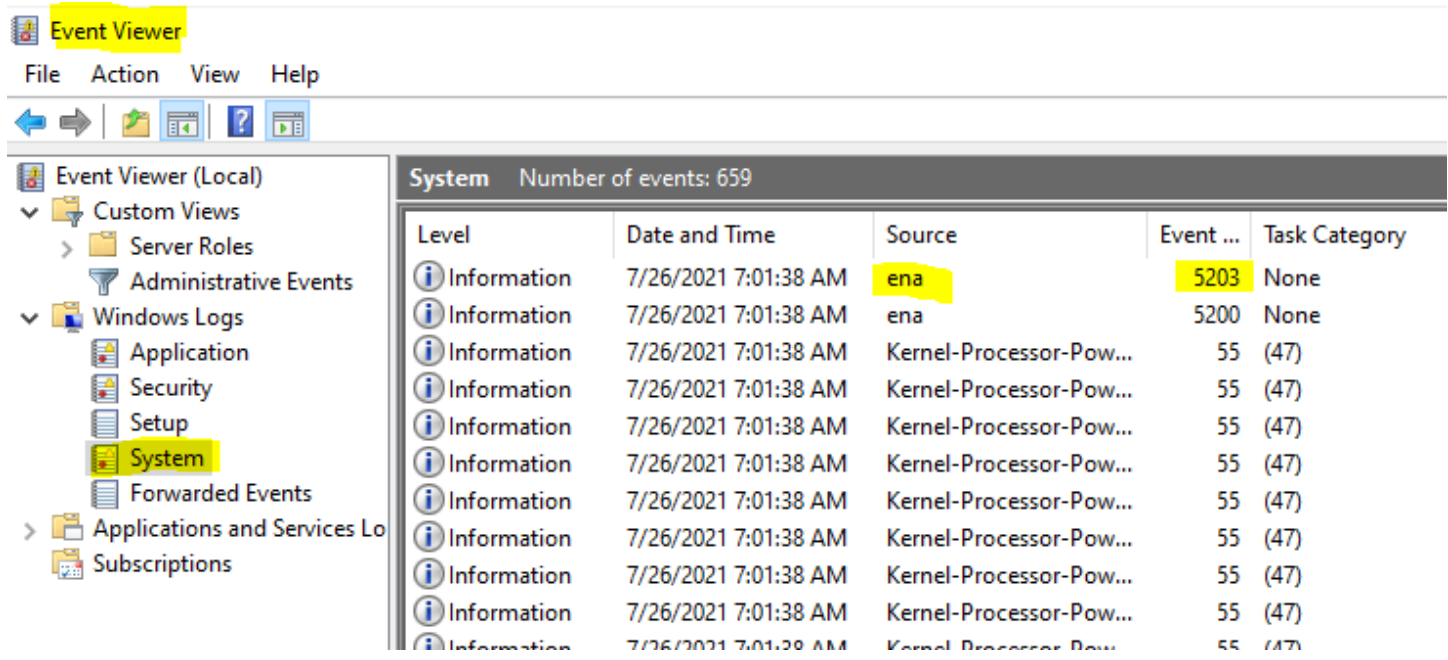
1. 使用上一部分中描述的其中一種方法開啟 Run (執行) 對話方塊。
2. 若要開啟 Windows Device Manager (Windows 裝置管理員)，請在 Run (執行) 對話方塊中輸入 `devmgmt.msc`。
3. 選擇確定。這將開啟 Device Manager (裝置管理員) 視窗。
4. 選取 Network adapters (網路轉接器) 左側的箭頭以展開清單。
5. 選取名稱，或開啟 Amazon Elastic Network Adapter (Amazon 彈性網路轉接器) 的內容功能表，然後選擇 Properties (屬性)。這會開啟 Amazon 彈性網路介面卡屬性對話方塊。
6. 確認一般索引標籤中的訊息顯示「此裝置運作正常」。

### 調查驅動程式事件訊息

若要使用 Windows 事件檢視器檢視 ENA Windows 驅動程式事件日誌，請按照下列步驟操作：

1. 使用上一部分中描述的其中一種方法開啟 Run (執行) 對話方塊。
2. 若要開啟 Windows 事件檢視器，請在 Run (執行) 對話方塊中輸入 `eventvwr.msc`。
3. 選擇確定。Event Viewer (事件檢視器) 視窗隨即開啟。
4. 展開 Windows Logs (Windows 日誌) 功能表，然後選擇 System (系統)。
5. 在右上方面板中的 Action (動作) 下，選擇 Filter Current Log (篩選當前日誌)。此時將顯示篩選對話方塊。
6. 在 Event sources (事件來源) 方塊中，輸入 `ena`。這會將結果限制為由 ENA Windows 驅動程式產生的事件。
7. 選擇確定。此時將在視窗的詳細資訊部分顯示篩選的事件日誌結果。
8. 要深入查看詳細資訊，請從清單中選擇一條事件訊息。

下面的範例顯示了 Windows 事件檢視器系統事件清單中的 ENA 驅動程式事件：



## 事件訊息摘要

如下資料表顯示了 ENA Windows 驅動程式產生的事件訊息。

## 輸入

事件 ID	ENA 驅動程式事件描述	Type
5001	硬體資源不足	錯誤
5002	轉接器偵測到硬體錯誤	錯誤
5005	對於未及時完成的 NDIS 操作，轉接器存在逾時	錯誤
5032	轉接器未能重設裝置	錯誤
5200	已初始化轉接器	資訊
5201	已停止轉接器	資訊
5202	已暫停轉接器	資訊
5203	已重新啟動轉接器	資訊
5204	已關閉轉接器	資訊

事件 ID	ENA 驅動程式事件描述	Type
5205	已重設轉接器	錯誤
5206	已意外刪除轉接器	錯誤
5208	轉接器初始化常式失敗	錯誤
5210	轉接器遇到內部問題並已成功恢復	錯誤

## 檢閱效能指標

ENA Windows 驅動程式會從啟用其的執行個體發佈網路效能指標。您可以使用本機效能監控應用程式檢視和啟用執行個體上的指標。如需 ENA Windows 驅動程式所產生指標的詳細資訊，請參閱 [監控 EC2 執行個體的網路效能](#)。

在啟用 ENA 指標並安裝 Amazon CloudWatch 代理程式的執行個體上，CloudWatch 會收集與 Windows 效能監視器中計數器相關聯的指標，以及 ENA 的一些進階指標。除了在 EC2 執行個體上預設啟用的指標以外，還會收集以下這些指標。如需有關指標的詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的 [CloudWatch 代理程式收集的指標](#)。

### Note

效能指標適用於 ENA 驅動程式 2.4.0 版及更新版本 (也適用於 2.2.3 版)。因第六代 EC2 執行個體的潛在效能下降，ENA 驅動程式 2.2.4 版已回復。建議您升級至驅動程式的目前版本，確保您擁有最新的更新項目。

使用效能指標的一些方式包括：

- 對執行個體效能問題執行故障診斷。
- 為工作負載選擇合適的執行個體大小。
- 主動規劃擴展活動。
- 基準化應用程式，以確定它們是否最大限度地提高了執行個體上可用的效能。

## 重新整理速率

根據預設，驅動程式使用 1 秒的間隔重新整理指標。但是，擷取指標的應用程式可能會使用不同的間隔進行輪詢。您可以使用驅動程式的進階屬性，在 Device Manager (裝置管理員) 中變更重新整理間隔。

若要變更 ENA Windows 驅動程式的指標重新整理間隔，請按照下列步驟操作：

1. 使用上一部分中描述的其中一種方法開啟 Run (執行) 對話方塊。
2. 若要開啟 Windows Device Manager (Windows 裝置管理員)，請在 Run (執行) 對話方塊中輸入 `devmgmt.msc`。
3. 選擇確定。這將開啟 Device Manager (裝置管理員) 視窗。
4. 選取 Network adapters (網路轉接器) 左側的箭頭以展開清單。
5. 選取名稱，或開啟 Amazon Elastic Network Adapter (Amazon 彈性網路轉接器) 的內容功能表，然後選擇 Properties (屬性)。這會開啟 Amazon 彈性網路介面卡屬性對話方塊。
6. 在彈出式視窗中開啟 Advanced (進階) 索引標籤。
7. 從 Property (屬性) 清單中，選擇 Metrics Refresh Interval (指標重新整理間隔) 以變更值。
8. 完成後，選擇 OK (確定)。

## ENA 轉接器重設

當 ENA Windows 驅動程式偵測到轉接器上的錯誤並將轉接器標記為運作狀態不良時，重設程序就會啟動。驅動程式無法自行重設，因此它依賴於作業系統來檢查轉接器的運作狀態，並叫用 ENA Windows 驅動程式的重設處理常式。重設程序可能會導致短時間內出現流量遺失。但是，TCP 連接應該能夠復原。

ENA 轉接器也可能透過傳送保持有效通知失敗而間接請求裝置重設程序。例如，如果 ENA 轉接器在載入不可恢復的組態之後達到未知狀態，它可能會停止傳送保持有效通知。

### ENA 轉接器重設的常見原因

- 缺少保持有效訊息

ENA 轉接器以固定速率張貼保持有效事件 (通常每秒一次)。ENA Windows 驅動程式實作監視程式機制，該機制會檢查這些保持有效訊息的存在。如果自上次檢查以來偵測到一條或多條新訊息，它就會記錄成功結果。否則，驅動程式得出裝置出現故障的結論，並啟動重設序列。

- 封包卡在傳輸佇列中

ENA 轉接器驗證封包是否按預期方式通過傳輸佇列。ENA Windows 驅動程式偵測封包是否卡住，如果已卡住，則啟動重設序列。

- 記憶體映射 I/O (MMIO) 暫存器的讀取逾時

若要限制記憶體映射 I/O (MMIO) 讀取操作，ENA Windows 驅動程式僅在初始化和重設過程中存取 MMIO 暫存器。如果驅動程式偵測到逾時，它將執行以下動作之一，具體取決於正在執行的程序：

- 如果在初始化過程中偵測到逾時，它會讓流程失敗，從而導致驅動程式在 Windows Device Manager (裝置管理員) 中的 ENA 轉接器上顯示黃色的驚歎號。
- 如果在重設過程中偵測到逾時，則會讓流程失敗。然後，作業系統會啟動 ENA 轉接器的意外刪除，同時透過停止並啟動已刪除的轉接器來進行復原。有關意外刪除網路介面卡 (NIC) 的詳細資訊，請參閱 Microsoft Windows 硬體開發人員文件中的 [處 NIC 的意外刪除](#)。

## 故障診斷方案

下列方案可以幫助您對 ENA Windows 驅動程式可能發生的問題進行故障診斷：如果您沒有最新版本，我們建議您從升級 ENA 驅動程式開始操作。若要尋找 Windows 作業系統版本的最新驅動程式，請參閱 [視窗 ENA 驅動](#)。

### 非預期的 ENA 驅動程式版本已安裝

#### 描述

在完成安裝特定版本 ENA 驅動程式的步驟之後，Windows Device Manager 會顯示 Windows 安裝了不同版本的 ENA 驅動程式。

#### 原因

當您執行驅動程式套件的安裝時，Windows 會在開始安裝之前，對本機 [驅動程式存放區](#) 中指定裝置的全部有效驅動程式套件進行排名。然後其會選取最低等級值的套件作為最佳相符項目。這可能與您要安裝的套件不同。如需有關裝置驅動程式套件選取程序的詳細資訊，請參閱 Microsoft 文件網站上的 [Windows 如何為裝置選取驅動程式套件](#)。

#### 解決方案

若要確保 Windows 會安裝您選擇的驅動程式套件版本，可以使用 [PnPUtil](#) 命令列工具從驅動程式存放區移除排名較低的驅動程式套件。

請遵循下列步驟更新 ENA 驅動程式：

1. 連線到您的執行個體，並以本機管理員登入。
2. 開啟 Device Manager (裝置管理員) 屬性視窗，如 [檢查 ENA 裝置狀態](#) 部分中所述。這會開啟 Amazon 彈性網路介面卡屬性視窗的一般索引標籤。

3. 開啟 Driver (驅動程式) 索引標籤。
4. 選擇 Update Driver (更新驅動程式)。這會開啟更新驅動程式軟體 – Amazon 彈性網路介面卡對話方塊。
  - a. 在您要如何搜尋驅動程式軟體？區段中，選擇瀏覽電腦上的驅動程式軟體。
  - b. 在瀏覽電腦上的驅動程式軟體頁面中，選擇位於搜尋欄下方的讓我從電腦上的裝置驅動程式清單挑選。
  - c. 在選取您要為此硬體安裝的裝置驅動程式頁面中，選擇擁有磁碟...
  - d. 在從磁碟安裝視窗中，選擇下拉式清單中檔案位置旁邊的瀏覽...
  - e. 導覽至下載目標 ENA 驅動程式套件的位置。選取名為 `ena.inf` 的檔案，然後選擇開啟。
  - f. 若要開始安裝，請選擇確定，然後選擇下一步。
5. 如果安裝程式未自動重新啟動執行個體，請執行指 `Restart-Computer PowerShell` 令程式。

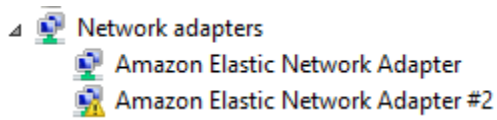
```
PS C:\> Restart-Computer
```

## ENA 驅動程式的裝置警告

### 描述

Device Manager (裝置管理員) 的 Network adapters (網路轉接器) 部分中的 ENA 轉接器圖示顯示一個警告符號 (內部帶有驚歎號的黃色三角形)。

下面的範例顯示了在 Windows Device Manager (裝置管理員) 中帶有警告圖示的 ENA 轉接器：



### 原因

此裝置警告通常由環境問題引起，此問題可能需要進行更多的研究，並且通常需要執行消除程序來確定根本原因。如需裝置錯誤的完整清單，請參閱 Microsoft Windows 硬體開發人員文件中的 [Device Manager \(裝置管理員\) 錯誤訊息](#)。

### 解決方案

此裝置警告的解決方案取決於根本原因。此處介紹的消除程序包括幾個基本步驟，用於幫助確定和解決可能具有簡單解決方案的最常見問題。如果這些步驟無法解決問題，則需要進行其他根本原因分析。

請按照以下步驟幫助確定和解決常見問題：

### 1. 停止並啟動裝置。

開啟 Device Manager (裝置管理員) 屬性視窗，如 [檢查 ENA 裝置狀態](#) 部分中所述。這將開啟 Amazon Elastic Network Adapter Properties (Amazon 彈性網路轉接器屬性) 視窗中的 General (常規) 索引標籤，其中 Device status (裝置狀態) 顯示錯誤代碼和一條簡短訊息。

- a. 開啟 Driver (驅動程式) 索引標籤。
- b. 選擇 Disable Device (停用裝置)，在顯示的警告訊息中回應 Yes (是)。
- c. 選擇 Enable Device (啟用裝置)。

### 2. 停止並啟動 EC2 執行個體

如果轉接器仍在 Device Manager (裝置管理員) 中顯示警告圖示，則下一步是停止並啟動 EC2 執行個體。在大多數情況下，這會在不同硬體上重新啟動執行個體。

### 3. 調查可能的執行個體資源問題

如果您已停止並啟動 EC2 執行個體，但問題仍然存在，這可能表明您的執行個體存在資源問題，例如記憶體不足。

## 轉接器重設時連接逾時 (錯誤代碼 5007、5205)

### 描述

Windows 事件檢視器顯示 ENA 轉接器的適配器逾時以及結合其發生的事件。消息類似以下範例：

- 事件 ID 5007：Amazon 彈性網路轉接器：操作過程中逾時。
- 事件 ID 5205：Amazon 彈性網路轉接器：轉接器重設已啟動。

轉接器重設會導致最小規模的流量中斷。即使執行多個重設，它們也很少會導致任何嚴重的網路中斷。

### 原因

此事件序列表示 ENA Windows 驅動程式為無回應的 ENA 轉接器啟動了重設。但是，裝置驅動程式用於偵測此問題的機制受到 CPU 0 不足導致的誤報約束。

### 解決方案

如果這種錯誤組合頻繁發生，請檢查您的資源分配，以了解調整可能發揮作用的位置。



1. 使用上一部分中描述的其中一種方法開啟 Run (執行) 對話方塊。
2. 若要開啟 Windows Resource Monitor (資源監視器)，請在 Run (執行) 對話方塊中輸入 resmon。
3. 選擇確定。這將開啟 Resource Monitor (資源監視器) 視窗。
4. 開啟 CPU 索引標籤。每個 CPU 的用量圖表顯示在 Resource Monitor (資源監視器) 視窗的右側。
5. 檢查 CPU 0 的用量級別，了解它們是否太高。

我們建議您設定 RSS 以排除較大執行個體類型 (超過 16 個 vCPU) 上適用於 ENA 轉接器的 CPU 0。對於較小的執行個體類型，設定 RSS 可能會改善體驗，但由於可用核心數量較少，因此必須進行測試，以確保限制 CPU 核心不會對效能產生負面影響。

使用 Set-NetAdapterRss 命令為 ENA 轉接器設定 RSS，如下列範例所示。

```
Set-NetAdapterRss -name (Get-NetAdapter | Where-Object {$_.InterfaceDescription -like
"*Elastic*"}).Name -Baseprocessorgroup 0 -BaseProcessorNumber 1
```

## 遷移到第六代執行個體基礎設施會影響效能或配件

### 描述

如果遷移到第六代 EC2 執行個體，並且您尚未更新 ENA Windows 驅動程式版本，則可能會遇到效能下降或 ENA 配件故障。

### 原因

依據執行個體作業系統 (OS)，第六代 EC2 執行個體類型需要以下 ENA Windows 驅動程式的最低版本。

### 最低版本

Windows Server 版本	ENA 驅動程式版本
Windows Server 2008 R2	2.2.3 或 2.4.0
Windows Server 2012 及更新版本	2.2.3 和更新版本
Windows 工作站	2.2.3 和更新版本

## 解決方案

在升級至第六代 EC2 執行個體之前，請確定您用來啟動的 AMI 具有相容驅動程式 (依據執行個體作業系統)，如上表所示。如需詳細資訊，請參閱 AWS re:Post 知識中心文章 [在將 EC2 執行個體遷移至第六代執行個體之前，我需要做些什麼來確保獲得最高網路效能？](#)。

### 彈性網路介面效能不佳

#### 描述

ENA 介面未按預期發揮效能。

#### 原因

效能問題的根本原因分析是一種消除程序。其中涉及太多的變數，無法指定常見原因。

#### 解決方案

根本原因分析的第一步是檢閱未按預期發揮效能的執行個體的診斷資訊，以確定是否存在可能導致問題的錯誤。如需詳細資訊，請參閱 [收集有關執行個體的診斷資訊](#) 一節。

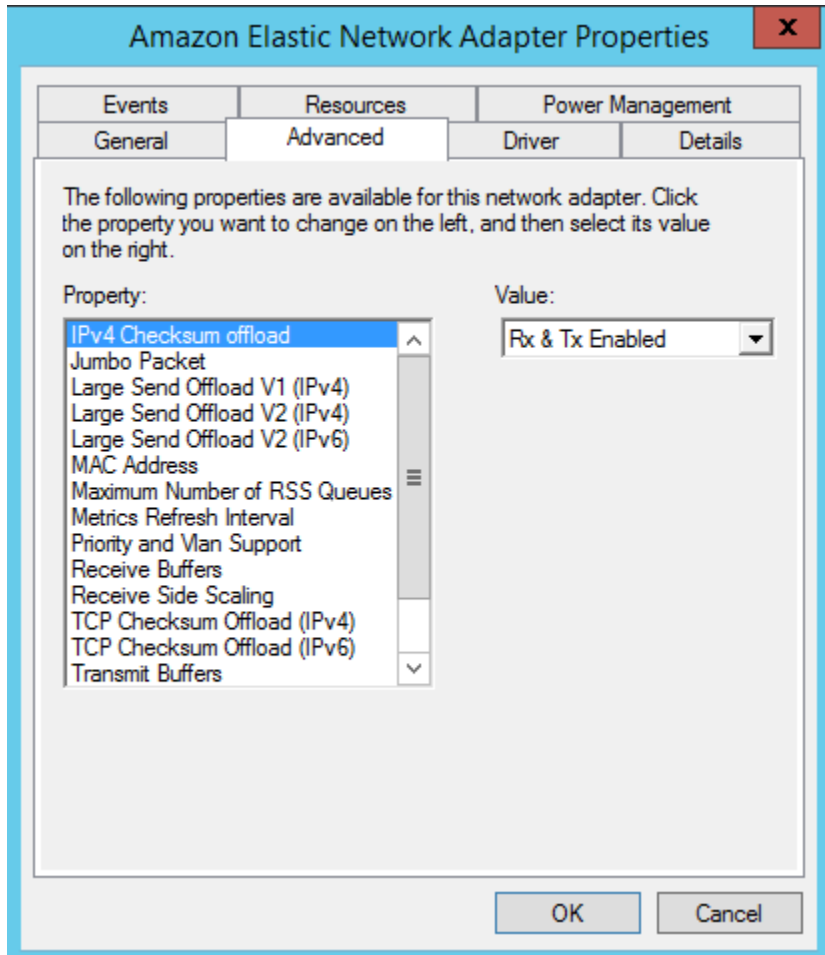
為了在執行個體上達到使用增強型聯網的最大網路效能，您可能需要修改預設的作業系統組態。根據預設，某些最佳化 (例如開啟檢查總和卸載和啟用 RSS) 已經在官方 Windows AMI 上設定。有關可套用至 ENA 轉接器的其他最佳化，請參閱 [ENA 轉接器效能調整](#) 中所示的效能調整。

我們建議您謹慎進行，並將裝置內容調整限制為本節所列的內容，或是 AWS 支援團隊建議的特定變更。

若要變更 ENA 轉接器屬性，請按照下列步驟操作：

1. 使用上一部分中描述的其中一種方法開啟 Run (執行) 對話方塊。
2. 若要開啟 Windows Device Manager (Windows 裝置管理員)，請在 Run (執行) 對話方塊中輸入 `devmgmt.msc`。
3. 選擇確定。這將開啟 Device Manager (裝置管理員) 視窗。
4. 選取 Network adapters (網路轉接器) 左側的箭頭以展開清單。
5. 選取名稱，或開啟 Amazon Elastic Network Adapter (Amazon 彈性網路轉接器) 的內容功能表，然後選擇 Properties (屬性)。這會開啟 Amazon 彈性網路介面卡屬性對話方塊。
6. 若要進行變更，請開啟進階索引標籤。
7. 完成後，請選擇確定以儲存變更。

下面的範例顯示了 Windows Device Manager (裝置管理員) 中的 ENA 轉接器屬性：



## ENA 轉接器效能調整

下表包含可調整以提高 ENA 介面效能的屬性。

### 輸入

屬性	描述	預設值	調整
接收緩衝區	控制軟體接收佇列中的項目數目。	1024	可提高至上限 8192。
接收端調整 (RSS)	支援在多處理器系統中跨多個 CPU 高效分配網路接收處理。	已啟用	您可以將負載分散到多個處理器中。如需進一步了解，請參閱

屬性	描述	預設值	調整
			<a href="#">最佳化 Windows 執行個體的網路效能。</a>
最大 RSS 佇列數	設定在 RSS 已啟用時允許的 RSS 佇列數上限。	32	<p>RSS 佇列的數量是在驅動程式初始化過程中確定的，其包括以下限制 (除其他限制外)：</p> <ul style="list-style-type: none"> <li>• 由此屬性設定的 RSS 佇列限制</li> <li>• 執行個體限制 (vCPU 計數)</li> <li>• 硬體世代限制 (ENAv1 中最多 8 個 RSS 佇列，ENAv2 中最多 32 個 RSS 佇列)</li> </ul> <p>根據執行個體和硬體世代限制，您可以設定 1-32 的值。如需進一步了解，請參閱<a href="#">最佳化 Windows 執行個體的網路效能。</a></p>

屬性	描述	預設值	調整
巨型封包	允許使用巨型乙太網路訊框 (超過 1500 個位元組的承載)。	已停用 (這將承載限制為 1500 個位元組或更少)	值可以設定為最高 9015，即轉換為 9001 個位元組的承載。這是巨型乙太網路訊框的最大承載。請參閱 <a href="#">使用巨型乙太網路訊框的考量事項</a> 。

### 使用巨型乙太網路訊框的考量事項

巨型訊框可增加單一封包的承載大小，這會增加不屬於封包成本的封包比例，藉此允許超過 1500 位元組的資料。傳送相同數量的可用資料，所需的封包數即會減少。但是，在下列情況下，流量限制為 MTU 最大為 1500：

- EC2 傳統版特定 AWS 區域之外的流量。
- 單一 VPC 外的流量。
- 跨區域 VPC 對等互連的流量。
- VPN 連線的流量。
- 網際網路閘道的流量。

#### Note

超過 1500 個位元組的封包會分段。如果 IP 標頭設有 Don't Fragment 旗標，這些封包將被捨棄。

巨型訊框應謹慎用於進出網路的流量，或任何離開 VPC 的流量。封包由中介系統切割，因此拖累流量速度。要在 VPC 內使用巨型訊框而不影響離開 VPC 的傳出流量，請嘗試以下選項之一：

- 按路由設定 MTU 大小。
- 使用具有不同 MTU 大小和不同路由的多個網路介面。

### 巨型訊框的建議使用案例

巨型訊框對於 VPC 內部和之間的流量非常有用。我們建議在以下使用案例中使用巨型訊框：

- 以配置於叢集置放群組內的執行個體而言，巨型訊框有助實現最大網路傳輸量。如需詳細資訊，請參閱 [置放群組](#)。
- 您可以針對透過 AWS Direct Connect 的 VPC 和現場部署網路之間的流量，使用 Jumbo Frame。如需有關使用 AWS Direct Connect 和驗證 Jumbo 框架功能的詳細資訊，請參閱《使用指南》中的 [< 設定私有虛擬介面的網路 MTU 或傳輸虛擬介面 >](#)。AWS Direct Connect
- 如需傳輸閘道的支援 MTU 大小的詳細資訊，請參閱 Amazon VPC 傳輸閘道中的 [傳輸閘道的配額](#)。

## 改善基於 Linux 的 Amazon EC2 執行個體的網路延遲

網路延遲是指資料封包從其來源傳送到目的地所需的時間。透過網路傳送資料的應用程式仰賴及時回應為使用者提供正面體驗。高網路延遲可能會導致各種問題，例如：

- 網頁載入時間緩慢
- 影片串流延遲
- 難以存取線上資源

本節針對在 Linux 上執行的 Amazon EC2 執行個體，列出可改善網路延遲的步驟。若要達到最佳延遲，請依照下列步驟設定執行個體、核心和 ENA 驅動程式設定。如需其他組態指引，請參閱上的 [《ENA Linux 驅動程式最佳作法和效能最佳化指南》](#) GitHub。

### Note

視網路硬體、啟動執行個體的 AMI 以及應用程式使用案例而定，步驟和設定可能會略有不同。變更之前，請徹底測試並監控網路效能，確保可獲得理想結果。

## 減少網路躍點

資料封包在從路由器之間移動時所經過的每個躍點都會增加網路延遲。通常，流量必須經過多個躍點才能到達目的地。有兩種方法可以減少 Amazon EC2 執行個體經過的網路躍點：

- 集群置放群組 – When 若指定 [集群置放群組](#)，Amazon EC2 會啟動彼此非常靠近的執行個體，其實際位置就在同一個可用區域 (AZ) 內，封裝更緊密。群組中執行個體的實體接近性可讓其充分利用高速連線，進而實現低延遲和高單流輸送量。

- 專用執行個體—[專用執行個體](#)是專供您使用的實體伺服器。使用專用執行個體，可以啟動執行個體在相同實體伺服器上執行。在相同專用執行個體上執行的執行個體之間的通訊可以不需要額外的躍點。

## Linux 核心設定

Linux 核心組態可以增加或減少網路延遲。若要實現延遲最佳化目標，請務必根據工作負載的特定需求對 Linux 核心組態進行微調。

Linux 核心的許多組態選項可能有助於降低網路延遲。最具影響力的選項如下所示。

- 啟用忙碌輪詢模式 — 忙碌輪詢模式可減少網路接收路徑的延遲。若啟用忙碌輪詢模式，通訊端層代碼可以直接輪詢網路裝置的接收佇列。忙碌輪詢的缺點是主機中的 CPU 使用率較高，因為要在緊密迴圈中輪詢新資料。有兩個全域設定可控制等待所有介面封包的微秒數。

### busy\_read

通訊端讀取的低延遲忙碌輪詢逾時。這可控制等待通訊端層讀取裝置佇列上封包的微秒數。若要使用 `sysctl` 命令全域啟用此功能，Linux 核心組織建議使用 50 微秒的值。如需詳細資訊，請參閱 Linux 核心使用者和管理員指南中的 [busy\\_read](#)。

```
$ C:\> sudo sysctl -w net.core.busy_read=50
```

### busy\_poll

輪詢和選取的低延遲忙碌輪詢逾時。這可控制等待事件的微秒數。建議值介於 50-100 微秒之間，具體取決於要輪詢的通訊端數目。新增的通訊端越多，數值越高。

```
$ C:\> sudo sysctl -w net.core.busy_poll=50
```

- 設定 CPU 電源狀態 (C-state) – C-state 可控制核心未使用時要進入的休眠層級。建議您控制 C-state 來微調系統的延遲與效能。在更深層的 C-state 中，CPU 基本上處於「休眠」狀態，在喚醒並轉換回作用中狀態之前無法回應請求。核心進入休眠狀態需要時間，而且雖然休眠核心讓其他核心更有餘裕加速至更高頻率，但是欲喚醒該休眠核心並開始執行工作也需要時間。

例如，若負責處理網路封包中斷的核心處於休眠狀態，則處理此類中斷的服務就可能受到延遲。您可以設定系統，使其不會使用更深層的 C-state。不過，雖然此組態可能會減少處理器反應延遲，但同時也減少了 Turbo Boost 的其他核心可用預留空間。

若要減少處理器反應延遲，您可以限制更深層的 C-state。如需詳細資訊，請參閱 Amazon Linux 2 使用者指南中的[限制更深 C 狀態的高效能和低延遲](#)。

## ENA 驅動程式設定

ENA 網路驅動程式可啟用執行個體與網路之間的通訊。驅動程式會處理網路封包，並將它們傳遞至網路堆疊或 Nitro 卡。網路封包傳入時，Nitro 卡會產生中斷，讓 CPU 通知軟體有事件發生。

### 插斷

中斷是裝置或應用程式傳送給處理器的訊號。中斷會通知處理器已發生事件或已滿足需要立即注意的條件。中斷可以處理時間敏感任務，例如從網路介面接收資料、處理硬體事件或服務來自其他裝置的請求。

### 插斷仲裁

中斷調節是一種技術，可以透過彙總或延遲來減少裝置產生的中斷次數。中斷調節的目的是透過減少處理大量中斷所產生的額外負荷來改善系統效能。中斷次數過多會增加 CPU 使用率，對輸送量造成不利影響，而中斷次數太少會增加延遲。

### 動態中斷調節

動態中斷調節是一種增強型的中斷調節形式，可根據目前的系統負載和流量模式動態調整中斷速率。其目標是在減少中斷負荷和每秒封包數或頻寬之間取得平衡。

#### Note

在某些 AMI 中預設啟用動態中斷調節 (但在所有 AMI 中都可啟用或停用)。

為了將網路延遲降到最低，可能需要停用中斷調節。但是這也可能會增加插斷處理的開銷。在減少延遲和盡可能減少開銷之間找到適當的平衡點非常重要。ethtool 命令可協助您設定插斷仲裁。依預設，rx-usecs 設定為 20，tx-usecs 設定為 64。

若要取得目前的插斷仲裁設定，請使用下列命令。

```
$ C:\> ethtool -c interface | egrep "rx-usecs:|tx-usecs:|Adaptive RX"
Adaptive RX: on TX: off
rx-usecs: 20
tx-usecs: 64
```



若要停用中斷修改和動態中斷調節，請使用下列命令。

```
$ C:\> sudo ethtool -C interface adaptive-rx off rx-usecs 0 tx-usecs 0
```

## 對於性能調整的硝基系統考量

Nitro 系統結合一系列由 AWS 建立的硬體和軟體元件，能保障高效能、高可用性和高安全性。Nitro System 提供類似裸機的功能，可消除虛擬化額外負荷，並支援需要完全存取主機硬體的工作負載。有關更多詳細信息，請參閱[AWS 硝基系統](#)。

所有目前一代的 EC2 執行個體類型都會在 EC2 Nitro 卡上執行網路封包處理。本主題涵蓋 Nitro 卡上的高階封包處理、影響封包處理效能的網路架構和組態的常見方面，以及您可以採取哪些動作來達到 Nitro 型執行個體的峰值效能。

Nitro 卡可處理所有輸入和輸出 (I/O) 介面，例如虛擬私有雲 (VPC) 所需的介面。對於透過網路傳送或接收資訊的所有元件，Nitro 卡會充當 I/O 流量的獨立運算裝置，實際上與執行客戶工作負載的系統主機板分開。

### 硝基卡上的網路數據包流

建立在 Nitro 系統上的 EC2 執行個體具有硬體加速功能，可實現更快的封包處理速度，這些功能是以每秒封包 (PPS) 輸送量率來衡量的。當 Nitro 卡對新流程執行初始評估時，它會儲存流程中所有封包的相同資訊，例如安全群組、存取控制清單和路由表項目。當它處理相同流程的其他封包時，可以使用儲存的資訊來減少這些封包的額外負荷。

您的連線速率是以每秒連線數 (CPS) 量度來測量。每個新連線都需要額外的處理額外負荷，這些負荷必須納入工作負載能力的估計。設計工作負載時，請務必同時考慮 CPS 和 PPS 指標。

### 如何建立連線

在 Nitro 型執行個體與另一個端點之間建立連線時，Nitro 卡會評估兩個端點之間傳送或接收的第一個封包的完整流程。對於相同流程的後續封包，通常不需要完全重新評估。但是，也有例外。如需例外狀況的詳細資訊，請參閱[不使用硬體加速的封包](#)。

下列內容定義兩個端點及它們之間的封包流量。這五個屬性一起被稱為 5 元組流程。

- 來源 IP
- 來源連接埠
- 目標 IP

- 目標連接埠
- 通訊協定

封包流程的方向稱為入口 (入埠) 和出口 (出埠)。下列高階說明摘要說明端對端網路封包流量。

- Ingress — 當 Nitro 卡處理輸入網路封包時，會根據狀態防火牆規則和存取控制清單來評估封包。它會追蹤連線、測量連線，並視情況執行其他動作。然後它將數據包轉發到主機 CPU 上的目的地。
- 輸出 — 當 Nitro 卡處理輸出網路封包時，它會查詢遠端介面目的地、評估各種 VPC 功能、套用速率限制，以及執行其他適用的動作。然後它會將封包轉送到網路上的下一個躍點目的地。

## 最佳效能的設計

要利用 Nitro 系統的性能功能，您必須了解網路處理需求以及這些需求如何影響 Nitro 資源的工作負載。然後，您可以針對網路環境設計最佳效能。您的基礎結構設定以及應用程式工作負載的設計和組態可能會影響封包處理和連線速率。例如，如果您的應用程式具有較高的連線建立速率 (例如 DNS 服務、防火牆或虛擬路由器)，則只有在建立連線之後才會發生的硬體加速，就會有較少的機會。

您可以設定應用程式和基礎架構設定，以簡化工作負載並改善網路效能。不過，並非所有封包都符合加速的資格。Nitro 系統將完整的網路流程用於新連線，以及不符合加速資格的封包。

本節的其餘部分將著重於應用程式和基礎結構設計考量，以協助確保封包盡可能在加速路徑內流通。

### 考量事項

當您為執行個體設定網路流量時，需要考量許多方面，這些方面可能會影響 PPS 效能。建立流程之後，大多數定期進出的封包都有資格加速。但是，存在例外情況，以確保基礎架構設計和封包流程持續符合通訊協定標準。

為了從 Nitro 卡獲得最佳性能，您應該仔細考慮基礎架構和應用程序的以下配置詳細信息的優缺點。

### 基礎架構考

您的基礎架構組態可能會影響封包流量和處理效率。下列清單包含一些重要的考量。

#### 具有不對稱性的網路接口配置

安全性群組會使用連線追蹤來追蹤進出執行個體的流量相關資訊。非對稱路由是流量透過一個網路介面進入執行個體，並透過不同的網路介面離開，可降低執行個體在追蹤流程時所能達到的尖峰效能。如需有關安全性群組連線追蹤、未追蹤連線以及自動追蹤連線的詳細資訊，請參閱[安全群組連線追蹤](#)。

## 網絡驅動

網絡驅動程序會定期更新和發布。如果您的驅動程序過時，可能會嚴重損害性能。讓驅動程式保持在最新狀態，以確保您擁有最新的修補程式，並可利用效能改進的優勢，例如僅適用於最新一代驅動程式的加速路徑功能。較早的驅動程序不支持加速路徑功能。

若要利用加速路徑功能，建議您在執行個體上安裝最新的 ENA 驅動程式。

Linux 執行個體 — ENA Linux 驅動程式 2.2.9 或更新版本。若要從 Amazon 驅動程式 GitHub 儲存庫安裝或更新 ENA Linux 驅動程式，請參閱我檔案的[驅動程式編譯](#)一節。

視窗執行個體 — 視窗驅動程式 2.0.0 或更新版本。若要安裝或更新 ENA 視窗驅動程式，請參閱[安裝彈性網路介面卡 \(ENA\) 驅動程式](#)。

## 端點之間的距離

由於在應用程式層進行 TCP 視窗，因此相同可用區域中兩個執行個體之間的連線每秒處理封包數量，因為應用程式層上的 TCP 視窗會決定在任何指定時間可以進行中的資料量。執行個體之間的長距離會增加延遲，並減少端點可處理的封包數目。

## 應用程式的設計

應用程式設計和組態的某些方面可能會影響您的處理效率。下列清單包含一些重要的考量。

### 封包大小

較大的封包大小可增加執行個體在網路上傳送和接收的資料輸送量。較小的封包大小可以增加封包處理速率，但是當封包數量超過 PPS 容量時，這可以減少達到的最大頻寬。

如果封包的大小超過網路躍點的最大傳輸單位 (MTU)，路徑上的路由器可能會將其分段。產生的封包片段會被視為例外狀況，並以標準速率處理 (非加速)。這可能會導致您的表現變化。Amazon EC2 支援 9001 位元組的巨型框架，但並非所有服務都支援它。我們建議您在設定 MTU 時評估拓撲。

### 協議權衡

像 TCP 這樣的可靠協議比 UDP 等不可靠的協議具有更多的開銷。UDP 傳輸通訊協定的較低額外負荷和簡化的網路處理可能會導致更高的 PPS 速率，但會犧牲可靠的封包傳遞。如果可靠的數據包傳遞對您的應用程序來說並不重要，UDP 可能是一個不錯的選擇。

### 微爆破

當流量在短時間內超過限額而非均勻分佈時，就會發生微爆炸。這通常發生在微秒的範圍內。

例如，假設您有一個最多可傳送 10 Gbps 的執行個體，而您的應用程式會在半秒內傳送完整的 10 Gb。這種微爆在上半年第二次超過了津貼，並且在第二個的剩餘部分沒有留下任何東西。即使您在 1 秒的時間範圍內傳送 10Gb，前半秒的限額也可能導致封包排入佇列或捨棄。

您可以使用網路排程器 (例如 Linux 流量控制) 來協助調整輸送量，並避免因為微量爆發造成佇列或丟棄封包。

## 流量數

單一流程的限制為 5 Gbps，除非它位於支援高達 10 Gbps 的叢集置放群組內，或使用 ENA Express (最高支援 25 Gbps)。

同樣地，與使用單一流程相比，Nitro 卡可以跨多個流程處理更多封包。為了達到每個執行個體的尖峰封包處理速率，我們建議在 100 Gbps 或更高彙總頻寬的執行個體上至少 100 個流量。隨著彙總頻寬功能的增加，達到尖峰處理速率所需的流量數也會增加。基準測試可協助您判斷網路上達到最高速率所需的組態。

## 彈性網路介面卡 (ENA) 佇列數目

根據預設，ENA 佇列數目上限會根據您的執行個體大小和類型分配給網路介面。減少佇列數量可以降低可達到的最高繳費靈費率。我們建議使用預設佇列配置以獲得最佳效能。

對於 Linux，網路接口默認配置為最大值。對於以「資料平面開發套件」(DPDK) 為基礎的應用程式，建議您設定可用的佇列數目上限。

## 功能過程開銷

流量鏡像和 ENA Express 等功能可以增加更多的處理額外負荷，從而降低絕對封包處理效能。您可以限制功能使用或停用功能，以提高封包處理速率。

## 連線追蹤以維持狀態

您的安全性群組會使用連線追蹤來儲存進出執行個體的流量相關資訊。連線追蹤會針對每個個別網路流量套用規則，以判斷流量是否允許或拒絕。Nitro 卡使用流量追蹤來維持流量的狀態。隨著套用的安全性群組規則越多，評估流程需要更多工作。

### Note

並非所有網路流量都會追蹤。如果使用設定安全性群組規則 [未追蹤的連線](#)，則除了自動追蹤的連線以確保在有多個有效回覆路徑時對稱路由，則不需要其他工作。

## 不使用硬體加速的封包

並非所有封包都能利用硬體加速。處理這些例外狀況涉及一些處理額外負荷，這是確保網路流程健全狀況所必需的。網路流程必須可靠地符合通訊協定標準、符合 VPC 設計中的變更，並且僅將封包路由到允許的目的地。但是，開銷會降低您的性能。

### 封包片段

如應用程式考量所述，超過網路 MTU 的封包所產生的封包片段會作為例外狀況處理，而且無法利用硬體加速的優勢。

### 閒置連線

當連線有一段時間沒有活動時，即使連線尚未達到其逾時限制，系統也可以取消其優先順序。然後，如果在取消連接優先級之後傳入數據，則系統需要將其作為異常處理才能重新連接。

若要管理連線，您可以使用連線追蹤逾時來關閉閒置連線。您也可以使用 TCP 保持連線來保持閒置連線的開啟狀態。如需詳細資訊，請參閱 [閒置連線追蹤逾時](#)。

### VPC 突變

安全性群組、路由表和存取控制清單的更新都需要在處理路徑中重新評估，以確保路由項目和安全性群組規則仍如預期般套用。

### ICMP 流程

網際網路控制訊息通訊協定 (ICMP) 是網路裝置用來診斷網路通訊問題的網路層通訊協定。這些封包一律會使用完整流程。

## 最大限度地提高 Nitro 系統的網路性能

在您對執行個體做出任何設計決策或調整任何網路設定之前，我們建議您採取下列步驟，以協助確保您獲得最佳結果：

1. 透過檢閱，瞭解您可以採取的行動的優缺點 [考量事項](#)。

如需執行個體設定的更多考量和最佳做法，請參閱：

Linux 執行個體 — GitHub 網站上的 [ENA Linux 驅動程式最佳實務和效能最佳化指南](#)。

視窗執行個體 — [設定網路介面的最佳實務](#)。

2. 使用尖峰作用中流量計數對工作負載進行基準測試，以決定應用程式效能的基準。使用效能基準，您可以測試設定或應用程式設計中的變化，以瞭解哪些考量會產生最大的影響，尤其是在您打算擴充或向外擴充時。

下列清單包含可根據系統需求調整 PPS 效能的動作。

- 減少兩個例證之間的物理距離。當傳送和接收執行個體位於相同的可用區域或使用叢集置放群組時，您可以減少封包從一個端點移動到另一個端點所需採取的躍點數目。
- 請使用 [未追蹤的連線](#)。
- 針對網路流量使用 UDP 通訊協定。
- 對於彙總頻寬為 100 Gbps 或以上的 EC2 執行個體，請將工作負載分配超過 100 個或更多個別流程，以將工作平均分散到 Nitro 卡上。

### 監控 Linux 執行個體的效能

您可以在 Linux 執行個體上使用 Ethtool 指標來監控執行個體網路效能指標，例如頻寬、封包速率和連線追蹤。如需詳細資訊，請參閱 [監控 EC2 執行個體的網路效能](#)。

## 最佳化 Windows 執行個體的網路效能

若要在具有增強型聯網功能的 Windows 執行個體上達到最大的網路效能，您可能需要修改預設作業系統組態。建議您針對需要高網路效能的應用程式進行下列組態變更。其他最佳化 (例如，開啟總和檢查碼卸載和啟用 RSS) 已在官方 Windows AMI 上設定。

#### Note

大多數使用案例都應該停用 TCP 煙囪卸載；此功能在 Windows Server 2016 中已被移除。

除了這些作業系統最佳化以外，您也應該考慮網路流量的最大傳輸單位 (MTU)，並根據您的工作負載和網路架構來調整。如需詳細資訊，請參閱 [EC2 執行個體的網路最大傳輸單位 \(MTU\)](#)。

AWS 定期測量在 50us 叢集置放群組中啟動的執行個體之間的平均往返延遲，以及 99.9 百分位數為 200us 的尾部延遲。如果您的應用程式需要始終很低的延遲，我們建議在 Nitro 系統中建構的固定效能執行個體上使用最新版本的 ENA 驅動程式。

## 設定 RSS CPU 親和性

接收端調整 (RSS) 可用來將網路流量 CPU 負載分配至多個處理器。根據預設，Amazon 官方 Windows AMI 會設為啟用 RSS。ENA ENI 最多提供 8 個 RSS 佇列。藉由為 RSS 佇列以及其他系統處理程序定義 CPU 親和性，您可以將 CPU 負載分配至多核心系統，以便處理更多網路流量。對於擁有超過 16 個 vCPUs 的執行個體類型，建議您使用 `Set-NetAdapterRSS` PowerShell 指令程式，該指令程式會從所有 ENI 的 RSS 組態中手動排除開機處理器 (啟用超執行緒時，邏輯處理器 0 和 1)，以避免與各種系統元件發生爭用。

Windows 會察覺超執行緒，並確保單一 NIC 的 RSS 佇列總是置於不同的實體核心。因此，除非停用超執行緒，否則為了完全避免與 NIC 競爭，會將各 NIC 的 RSS 組態分配在 16 個邏輯處理器之間。指 `Set-NetAdapterRss` 令程式可讓您定義 `BaseProcessorGroup`、`BaseProcessorNumber`、`MaxProcessorNumber` 和 `NumaNode` (選用) 的值，以定義每個 NIC 範圍的有效邏輯處理器。如果無足夠的實體核心可以完全消除 NIC 間的競爭，視 ENI 預期的工作負載而定，請盡量減少重疊範圍，或減少 ENI 範圍中的邏輯處理器數量 (換言之，少量管理網路 ENI 可能不需要許多已指派的 RSS 佇列)。此外，如先前所述，各種元件必須在 CPU 0 上執行，因此當可使用足夠的 vCPU 時，我們建議從所有 RSS 組態中加以排除。

例如，當有 2 個 NUMA 節點並啟用超執行緒的 72 vCPU 執行個體出現三個 ENI 時，下列命令會將網路負載分配在兩顆 CPU 之間，而不會重疊並避免完全使用核心 0。

```
Set-NetAdapterRss -Name NIC1 -BaseProcessorGroup 0 -BaseProcessorNumber 2 -
MaxProcessorNumber 16
Set-NetAdapterRss -Name NIC2 -BaseProcessorGroup 1 -BaseProcessorNumber 0 -
MaxProcessorNumber 14
Set-NetAdapterRss -Name NIC3 -BaseProcessorGroup 1 -BaseProcessorNumber 16 -
MaxProcessorNumber 30
```

注意，這些設定對於每個網路轉接器都是持續有效。如果有個執行個體調整大小成不同 vCPU 數的執行個體，您應針對每個已啟用 ENI 重新評估 RSS 組態。您可於以下網站找到 `Set-NetAdapterRss` cmdlet 完整的 Microsoft 文件：<https://docs.microsoft.com/en-us/powershell/module/netadapter/set-netadapterrss>。

SQL 工作負載的特別注意事項：我們還建議您檢查輸入/輸出執行緒同質性設定和 ENI RSS 組態，以盡量減少相同 CPU 的輸入/輸出和網路爭用。請參閱 [親和性遮罩伺服器組態選項](#)。

# Elastic Fabric Adapter

Elastic Fabric Adapter (EFA) 是可連接到 Amazon EC2 執行個體的網路裝置，用以加速高效能運算 (HPC) 和機器學習應用程式。EFA 可讓您透過雲端提供的延展性、彈性和彈性，達到內部部署 HPC 叢集的應用程式效能。AWS

與雲端式 HPC 系統中傳統上使用的 TCP 傳輸相比，EFA 可提供更低和更一致的延遲及更高的輸送量。它可增強執行個體間通訊的效能，在擴展 HPC 和機器學習應用程式時不可或缺。它經過優化，可以在現有的 AWS 網路基礎架構上工作，並且可以根據應用程序需求進行擴展。

EFA 整合 Libfabric 1.7.0 和更新版本，並支援適用於 HPC 應用程式的 Open MPI 5 和更新版本及 Intel MPI 2019 Update 5 和更新版本，以及適用於機器學習應用程式的 Nvidia Collective Communications Library (NCCL)。

## Note

Windows 執行個體上不支援 EFAs 的 OS-bypass 功能。如果您將 EFA 連接到 Windows 執行個體，該執行個體會充當「彈性網路轉接器」，但沒有額外的 EFA 功能。

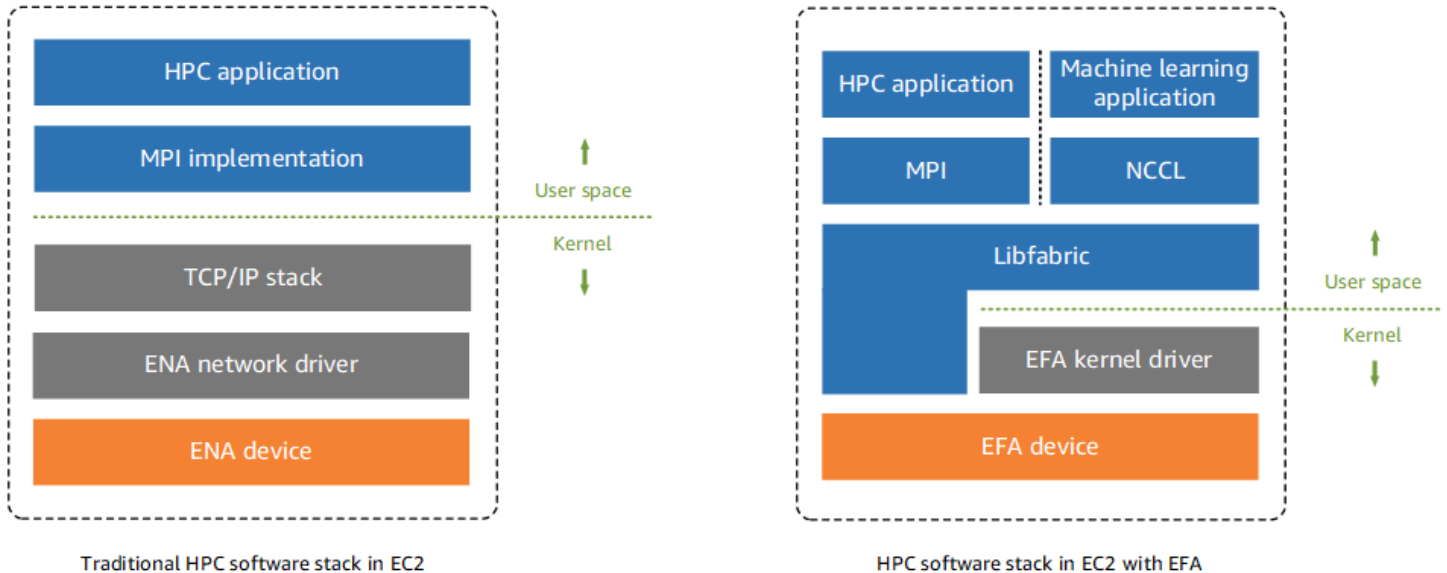
## 內容

- [EFA 基本概念](#)
- [支援的介面和程式庫](#)
- [支援的執行個體類型](#)
- [支援的作業系統](#)
- [EFA 限制](#)
- [EFA 定價](#)
- [開始使用 P5 執行個體和 EFA](#)
- [開始使用 EFA 和 MPI](#)
- [開始使用 EFA 和 NCCL](#)
- [使用 EFA](#)
- [監控 EFA](#)
- [使用檢查總和驗證 EFA 安裝程式](#)



## EFA 基本概念

EFA 是有額外功能的彈性網路轉接器 (ENA)。它提供 ENA 的所有功能，還有額外的 OS-bypass 功能。OS-bypass 是一種存取模式，可讓 HPC 和機器學習應用程式直接與網路介面硬體通訊，以提供低延遲、可靠的傳輸功能。



傳統上，HPC 應用程式使用 Message Passing Interface (MPI) 來與系統的網路傳輸連接。在 AWS 雲端中，這表示應用程式與 MPI 進行介面，接著會使用作業系統的 TCP/IP 堆疊和 ENA 裝置驅動程式來啟用執行個體之間的網路通訊。

透過 EFA，HPC 應用程式使用 MPI 或 NCCL 來與 Libfabric API 互動。Libfabric API 會繞過作業系統的核心，直接與 EFA 裝置通訊來將封包放到網路上。這樣可降低額外負荷，讓 HPC 應用程式執行更有效率。

### Note

Libfabric 是 OpenFabrics 界面 ( OFI ) 框架的核心組件，它定義和導出 OFI 的用戶空間 API。如需詳細資訊，請參閱 [LibFabric OpenFabrics](#) 網站。

## EFA 與 ENA 之間的差異

Elastic Network Adapters (ENA) 提供傳統 IP 聯網功能，是支援 VPC 聯網不可或缺的功能。EFA 提供與 ENA 相同的所有傳統 IP 聯網功能，它們還支援 OS-bypass 功能。OS-bypass 可讓 HPC 和機器學習應用程式繞過作業系統核心，直接與 EFA 裝置通訊。

## 支援的介面和程式庫

EFA 支援下列介面和程式庫：

- 開啟 Open MPI 5 和更新版本
- Graviton 首選 Open MPI 4.0 或更高版本
- Intel MPI 2019 Update 5 及更新版本
- NVIDIA Collective Communications Library (NCCL) 2.4.2 和更新版本

## 支援的執行個體類型

下列執行個體類型支援 EFAs：

- 一般用途：m5dn.24xlarge m5dn.metal m5n.24xlarge m5n.metal | m5zn.12xlarge | m5zn.metal | m6a.48xlarge | m6a.metal | m6i.32xlarge | m6i.metal | m6id.32xlarge | m6id.metal | m6idn.32xlarge | m6idn.metal | m6in.32xlarge | m6in.metal | m7a.48xlarge | m7a.metal-48x1 | m7g.16xlarge | m7g.metal | m7gd.16xlarge | m7gd.metal | m7i.48xlarge | m7i.metal-48x1
- 運算最佳化：c5n.9xlarge c5n.18xlarge c5n.metal c6a.48xlarge c6a.metal c6gn.16xlarge c6i.32xlarge | c6i.metal | c6id.32xlarge | c6id.metal | c6in.32xlarge | c6in.metal | c7a.48xlarge | c7a.metal-48x1 | c7g.16xlarge | c7g.metal | c7gd.16xlarge | c7gd.metal | c7gn.16xlarge | c7gn.metal | c7i.48xlarge | c7i.metal-48x1
- 記憶體最佳化：r5dn.24xlarge r5dn.metal r5n.24xlarge r5n.metal r6a.48xlarge r6a.metal r6i.32xlarge | r6idn.32xlarge | r6idn.metal | r6in.32xlarge | r6in.metal | r6id.32xlarge | r6id.metal | r7a.48xlarge | r7a.metal-48x1 | r7g.16xlarge | r7g.metal | r7gd.16xlarge | r7gd.metal | r7i.48xlarge | r7i.metal-48x1 | r7iz.32xlarge | r7iz.metal-32x1 | u7i-12tb.224xlarge | u7in-16tb.224xlarge | u7in-24tb.224xlarge | u7in-32tb.224xlarge | x2idn.32xlarge | x2idn.metal | x2iedn.32xlarge | x2iedn.metal | x2iezn.12xlarge | x2iezn.metal
- 儲存最佳化：i3en.12xlarge | i3en.24xlarge | i3en.metal | i4g.16xlarge | i4i.32xlarge | i4i.metal | im4gn.16xlarge
- 加速運算：d11.24xlarge d12q.24xlarge g4dn.8xlarge g4dn.12xlarge g4dn.16xlarge g4dn.metal g5trn1.32xlarge | trn1n.32xlarge | vt1.24xlarge

- 高效能運算 : hpc6a.48xlarge|hpc6id.32xlarge|hpc7a.12xlarge|hpc7a.24xlarge | hpc7a.48xlarge | hpc7a.96xlarge | hpc7g.4xlarge | hpc7g.8xlarge | hpc7g.16xlarge

查看特定區域中支援 EFA 的可用執行個體類型

可用的執行個體類型因區域而異。若要查看區域中支援 EFA 的可用執行個體類型，請使用 [describe-instance-types](#) 命令，搭配 `--region` 參數。包含 `--filters` 參數以將結果範圍限定為支援 EFA 的執行個體類型，以及包含 `--query` 參數以將輸出範圍限定為 InstanceType 的值。

```
aws ec2 describe-instance-types --region us-east-1 --filters Name=network-info.efa-supported,Values=true --query "InstanceTypes[*].[InstanceType]" --output text | sort
```

## 支援的作業系統

以下作業系統支援 Intel/AMD x86 執行個體類型的 EFA：

- Amazon Linux 2023
- Amazon Linux 2
- CentOS 7
- RHEL 7、8 和 9
- Debian 10 和 11
- Rocky Linux 8 和 9
- Ubuntu 20.04 和 22.04
- SUSE Linux Enterprise 15 SP2 及更新版本
- OpenSUSE Leap 15.4 及更新版本

### Note

Ubuntu 20.04 與 d11.24xlarge 執行個體搭配使用時支援對等直接支援。

以下作業系統支援 Arm (Graviton) 執行個體類型的 EFA：

- Amazon Linux 2023
- Amazon Linux 2

- RHEL 8/9 和 Rocky Linux 8/9
- Debian 10 和 11
- Ubuntu 20.04 和 22.04
- SUSE Linux Enterprise 15 SP2 及更新版本

## EFA 限制

EFA 具有下列限制：

- 所有 P4d 和 P5 執行個體類型都支援 NVIDIA GPUDirect Remote Direct Memory Access (RDMA)。
- 目前不支援 P4d/P4de/DL1 執行個體與其他執行個體類型之間的 EFA 流量。
- [支援多個網路卡的執行個體類型](#)可以設定為每個網路卡一個 EFA。所有其他支援的執行個體類型每個執行個體只支援一個 EFA。
- 對於 c7g.16xlarge，在連接 EFA 時，不支援 m7g.16xlarge 和 r7g.16xlarge 專用執行個體和專用主機。
- EFA OS-bypass 流量限於單一子網。換言之，無法在兩個子網之間傳送 EFA 流量。兩個子網之間可以傳送來自 EFA 的標準 IP 流量。
- EFA OS-bypass 流量不可路由傳送。來自 EFA 的標準 IP 流量仍可路由傳送。
- EFA 必須是安全群組的成員，而該安全群組允許往返於其本身的所有傳入和傳出流量。
- 視窗執行個體不支援 EFA。
- AWS [Out posts](#) 不支持 EFA。

## EFA 定價

EFA 是選用的 Amazon EC2 聯網功能，您可以在任何支援的執行個體上啟用該功能，而無需額外付費。

## 開始使用 P5 執行個體和 EFA

P5 執行個體使用多個 EFA 介面來提供 3200 Gbps 的聯網頻寬。P5 執行個體支援 32 個網路卡。如需 P5 執行個體的入門詳細資訊，請參閱 [開始使用適用於 Linux 的 P5 執行個體](#)。

我們建議您為每張網路卡定義單一 EFA 網路介面。若要在啟動時設定這些介面，我們建議您進行下列設定：

- 對於網路介面 0，指定裝置索引 0

- 對於網路介面 1 至 31，指定裝置索引 1

如果您使用的是 Amazon EC2 主控台，請在啟動執行個體精靈的網路設定區段中選擇編輯。展開進階網路組態並選擇新增網路介面以新增所需數量的網路介面。對於每個網路介面，針對 EFA 選取啟用。對於所有網路介面 (主要網路介面除外)，針對裝置索引指定 1。視需要進行其餘的設定。

如果您正在使用 AWS CLI，請使用 [run-instance](#) 指令 `--network-interfaces`，指定所需的網路介面數。對於每個網路介面，針對 `InterfaceType` 指定 `efa`。對於主要網路介面，針對 `NetworkCardIndex` 和 `DeviceIndex` 指定 0。對於剩餘的網路介面，針對 `NetworkCardIndex` 指定從 1 至 31 的唯一值，以及針對 `DeviceIndex` 指定 1。

下列範例命令程式碼片段顯示具有 32 個 EFA 網路介面的請求。

```
$ aws --region $REGION ec2 run-instances \
--instance-type p5.48xlarge \
--count 1 \
--key-name key_pair_name \
--image-id ami_id \
--network-interfaces
"NetworkCardIndex=0,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa" \
\br/>"NetworkCardIndex=1,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa" \
\br/>"NetworkCardIndex=2,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa" \
\br/>"NetworkCardIndex=3,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa" \
\br/>"NetworkCardIndex=4,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa" \
\br/>"NetworkCardIndex=5,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa" \
\br/>"NetworkCardIndex=6,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa" \
\br/>"NetworkCardIndex=7,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa" \
\
```

```
"NetworkCardIndex=8,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
\
"NetworkCardIndex=9,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
\
"NetworkCardIndex=10,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\
"NetworkCardIndex=11,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\
"NetworkCardIndex=12,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\
"NetworkCardIndex=13,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\
"NetworkCardIndex=14,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\
"NetworkCardIndex=15,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\
"NetworkCardIndex=16,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\
"NetworkCardIndex=17,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\
"NetworkCardIndex=18,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\
"NetworkCardIndex=19,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\
"NetworkCardIndex=20,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\
"NetworkCardIndex=21,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\

```

```
"NetworkCardIndex=22,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
\

"NetworkCardIndex=23,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
\

"NetworkCardIndex=24,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
\

"NetworkCardIndex=25,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
\

"NetworkCardIndex=26,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
\

"NetworkCardIndex=27,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
\

"NetworkCardIndex=28,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
\

"NetworkCardIndex=29,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
\

"NetworkCardIndex=30,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
\

"NetworkCardIndex=31,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
...

```

如果您使用的是啟動範本，請在啟動範本中指定所需的網路介面數量。對於每個網路介面，針對 `InterfaceType` 指定 `efa`。對於主要網路介面，針對 `NetworkCardIndex` 和 `DeviceIndex` 指定 `0`。對於剩餘的網路介面，針對 `NetworkCardIndex` 指定從 1 至 31 的唯一值，以及針對 `DeviceIndex` 指定 1。下面的程式碼片段顯示了一個範例，其中包含可能的 32 個網路介面中的 3 個網路介面。

```
"NetworkInterfaces": [
{
 "NetworkCardIndex": 0,
 "DeviceIndex": 0,
 "InterfaceType": "efa",
 "AssociatePublicIpAddress": false,

```

```
"Groups":[
 "security_group_id"
],
"DeleteOnTermination":true
},
{
 "NetworkCardIndex": 1,
 "DeviceIndex": 1,
 "InterfaceType": "efa",
 "AssociatePublicIpAddress":false,
 "Groups":[
 "security_group_id"
],
 "DeleteOnTermination":true
},
{
 "NetworkCardIndex": 2,
 "DeviceIndex": 1,
 "InterfaceType": "efa",
 "AssociatePublicIpAddress":false,
 "Groups":[
 "security_group_id"
],
 "DeleteOnTermination":true
}
...

```

啟動具有多個網路介面的 P5 執行個體時，您無法自動指派公有 IP 地址。不過，您可以在啟動網際網路連線後將彈性 IP 位址附加至主要網路介面 (NetworkCardIndex DeviceIndex =0, =0)。Ubuntu 20.04 及更新版本以及 Amazon Linux 2 及更新版本均設定為在執行個體啟動時使用主要網路介面處理網際網路流量，如上所述。

## 開始使用 EFA 和 MPI

本教學課程協助您啟動 EFA 和 MPI 啟用的執行個體叢集用於 HPC 工作負載。在本教學課程中，您將執行下列步驟：

### 內容

- [步驟 1：準備具備 EFA 功能的安全群組](#)
- [步驟 2：啟動暫時執行個體](#)
- [步驟 3：安裝 EFA 軟體](#)



- [步驟 4：\(選用\) 啟用 Open MPI 5](#)
- [步驟 5：\(選用\) 安裝 Intel MPI](#)
- [步驟 6：停用 ptrace 保護](#)
- [步驟 7. 確認安裝](#)
- [步驟 8：安裝 HPC 應用程式](#)
- [步驟 9：建立已啟用 EFA 的 AMI](#)
- [步驟 10：在集群放置群組中啟動已啟用 EFA 的執行個體](#)
- [步驟 11：終止暫時執行個體](#)
- [步驟 12：啟用無密碼 SSH](#)

## 步驟 1：準備具備 EFA 功能的安全群組

EFA 需要有安全群組來允許往返安全群組自身的所有傳入和傳出流量。下列程序會建立一個安全群組，該安全群組允許所有傳入和傳出流量，並允許來自任何 IPv4 地址的傳入 SSH 流量以進行 SSH 連線。

### Important

此安全群組僅供測試之用。對於生產環境，我們建議您建立傳入 SSH 規則，該規則僅允許來自連線 IP 地址的流量，例如電腦的 IP 地址或區域網路中的一系列 IP 地址。

如需其他案例，請參閱[不同使用案例的安全群組規則](#)。

### 建立具備 EFA 功能的安全群組

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Security Groups (安全群組)，然後選擇 Create Security Group (建立安全群組)。
3. 在 Create Security Group (建立安全群組) 視窗中，執行下列動作：
  - a. 對於 Security group name (安全群組名稱)，輸入安全群組的描述性名稱，例如 EFA-enabled security group。
  - b. (選用) 對於 Description (描述)，輸入安全群組的簡短描述。
  - c. 對於 VPC，選取您打算讓具備 EFA 功能的執行個體在其中啟動的 VPC。

- d. 選擇 Create Security Group (建立安全群組)。
4. 選取您建立的安全群組，在 Details (詳細資料) 索引標籤上，複製 Security group ID (安全群組 ID)。
5. 在仍然選取安全群組的情況下，依序選取 Actions (動作) 和 Edit inbound rules (編輯傳入規則)，然後執行下列動作。
  - a. 選擇新增規則。
  - b. 針對類型，選擇所有流量。
  - c. 若為 Source type (來源類型)，選擇 Custom (自訂)，然後將您複製的安全群組 ID 貼到欄位中。
  - d. 選擇新增規則。
  - e. 針對 Type (類型)，選擇 SSH。
  - f. 針對 Source type (來源類別)，選擇 Anywhere-IPv4(隨處 - IPv4)。
  - g. 選擇儲存規則。
6. 在仍然選取安全群組的情況下，依序選取 Actions (動作) 和 Edit outbound rules (編輯傳出規則)，然後執行下列動作。
  - a. 選擇新增規則。
  - b. 針對類型，選擇所有流量。
  - c. 若為 Destination type (目的地類型)，選擇 Custom (自訂)，然後將您複製的安全群組 ID 貼到欄位中。
  - d. 選擇儲存規則。

## 步驟 2：啟動暫時執行個體

啟動暫時執行個體，以用來安裝和設定 EFA 軟體元件。您可以使用此執行個體來建立具備 EFA 功能的 AMI，再從中啟動具備 EFA 功能的執行個體。

### 啟動暫時執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)，接著選擇 Launch Instances (啟動執行個體) 來開啟新的啟動執行個體精靈。
3. (選用) 在 Name and tags (名稱和標籤) 區段中，為執行個體命名，例如，EFA-instance。該名稱將指派作為執行個體的資源標籤 (Name=*EFA-instance*)。

4. 在 Application and OS Images (應用程式和作業系統映像) 區段中，為其中一個[支援的作業系統](#) 選取 AMI。
5. 在 Instance type (執行個體類型) 區段中，選取[支援的執行個體類型](#)。
6. 在 Key pair (金鑰對) 區段中，選取要用於執行個體的金鑰對。
7. 在 Network settings (網路設定) 區段，選擇 Edit (編輯)，接著執行下列動作：
  - a. 對於 Subnet (子網)，請選擇要在其中啟動執行個體的字網。若您未選取子網，便無法啟用 EFA 的執行個體。
  - b. 針對 Firewall (security groups) (防火牆 (安全群組))，選取 Select existing security group (選取現有的安全群組)，接著選取您在前一個步驟中建立的安全群組。
  - c. 展開 Advanced network configuration (進階網路組態) 區段，並針對 Elastic Fabric Adapter，選取 Enable (啟用)。
8. 在 儲存 區段中，根據需求設定磁碟區。
9. 在右邊的 Summary (摘要) 面板中，選擇 Launch instance (啟動執行個體)。

### 步驟 3：安裝 EFA 軟體

在暫存執行個體上安裝支援 EFA 所需的具備 EFA 功能的核心、EFA 驅動程式、Libfabric 和 Open MPI 堆疊。

該步驟會根據您是否打算將 EFA 與 Open MPI、Intel MPI 或 Open MPI 與 Intel MPI 搭配使用而有所不同。

#### 安裝 EFA 軟體

1. 連接至您啟動的執行個體。如需詳細資訊，請參閱 [連接至您的 Linux 執行個體](#)。
2. 為確保所有軟體套件皆為最新版本，請對您的執行個體執行快速軟體更新。此程序可能需要幾分鐘的時間。

- Amazon Linux 2023, Amazon Linux 2, RHEL 7/8/9, CentOS 7, 洛基 Linux 8/9

```
$ sudo yum update -y
```

- Ubuntu 20.04/22.04 和 Debian 10/11

```
$ sudo apt-get update && sudo apt-get upgrade -y
```

- SUSE Linux Enterprise

```
$ sudo zypper update -y
```

3. 重新啟動執行個體並重新連線至其中。
4. 下載 EFA 軟體安裝檔案。軟體安裝檔案已封裝成壓縮 tarball (.tar.gz) 檔案。若要下載最新穩定版本，請使用下列命令：

```
$ C:\> curl -O https://efa-installer.amazonaws.com/aws-efa-installer-1.32.0.tar.gz
```

您也可以在之前的命令中以 latest 取代版本號碼，以取得最新版本。

5. (選用) 驗證 EFA tarball (.tar.gz) 檔案的真確性及完整性。

我們建議您執行這項操作來確認軟體發行者的身分，並檢查檔案自發行以來並未遭到變更或損毀。如果您不想驗證 tarball 檔案，請略過此步驟。

#### Note

或者，如果您偏好使用 MD5 或 SHA256 檢查總和來驗證 tarball 檔案，請參閱[使用檢查總和驗證 EFA 安裝程式](#)。

- a. 下載公有 GPG 金鑰並匯入至您的 keyring。

```
$ wget https://efa-installer.amazonaws.com/aws-efa-installer.key && gpg --import aws-efa-installer.key
```

命令應傳回金鑰值。請記下金鑰的值，因為下一個步驟將需要它。

- b. 驗證 GPG 金鑰的指紋。執行以下命令，並指定上一步驟中的金鑰值。

```
$ gpg --fingerprint key_value
```

命令應傳回與 4E90 91BC BB97 A96B 26B1 5E59 A054 80B1 DD2D 3CCC 相同的指紋。如果指紋不相符，請勿執行 EFA 安裝指令碼，並聯絡 AWS Support。

- c. 下載簽章檔案並驗證 EFA tarball 檔案的簽章。

```
$ wget https://efa-installer.amazonaws.com/aws-efa-installer-1.32.0.tar.gz.sig && gpg --verify ./aws-efa-installer-1.32.0.tar.gz.sig
```

下面顯示了範例輸出。

```
gpg: Signature made Wed 29 Jul 2020 12:50:13 AM UTC using RSA key ID DD2D3CCC
gpg: Good signature from "Amazon EC2 EFA <ec2-efa-maintainers@amazon.com>"
gpg: WARNING: This key is not certified with a trusted signature!
gpg: There is no indication that the signature belongs to the owner.
Primary key fingerprint: 4E90 91BC BB97 A96B 26B1 5E59 A054 80B1 DD2D 3CCC
```

如果結果包含 Good signature，而指紋與上一個步驟中傳回的指紋相符，請繼續下一個步驟。如果指紋不相符，請勿執行 EFA 安裝指令碼，並聯絡 AWS Support。

6. 從壓縮的 .tar.gz 檔案中解壓縮檔案，然後導覽至解壓縮的目錄。

```
$ C:\> tar -xf aws-efa-installer-1.32.0.tar.gz && cd aws-efa-installer
```

7. 安裝 EFA 軟體。根據使用案例而定，執行以下其中一個命令。

#### Note

EFA 不支援使用 SUSE Linux 的 NVIDIA GPUDirect。如果您使用的是 SUSE Linux，您必須另外指定 `--skip-kmod` 選項來防止 kmod 安裝。根據預設，SUSE Linux 不允許 out-of-tree 核心模組。

## Open MPI and Intel MPI

如果您打算將 EFA 與 Open MPI 和 Intel MPI 搭配使用，您必須使用 Libfabric 和 Open MPI 安裝 EFA 軟體，且必須完成「步驟 5：安裝 Intel MPI」。

若要使用 Libfabric 和 Open MPI 安裝 EFA 軟體，請執行下列命令。

#### Note

從 EFA 1.30.0 開始，根據預設會安裝 Open MPI 4 和 Open MPI 5。您可以選擇是否指定您要安裝的 Open MPI 版本。若僅要安裝 Open MPI 4，請包括 `--mpi=openmpi4`。若要安裝 Open MPI 5，請納入 `--mpi=openmpi5`。若要同時安裝這兩種版本，請忽略 `--mpi` 選項。

```
$ C:\> sudo ./efa_installer.sh -y
```

Libfabric 會安裝到 `/opt/amazon/efa`。Open MPI 4 會安裝到 `/opt/amazon/openmpi`。Open MPI 5 會安裝到 `/opt/amazon/openmpi5`。

### Open MPI only

如果您打算將 EFA 與 Open MPI 搭配使用，您必須使用 Libfabric 和 Open MPI 安裝 EFA 軟體，且必須跳過「步驟 5：安裝 Intel MPI」。若要使用 Libfabric 和 Open MPI 安裝 EFA 軟體，請執行下列命令。

#### Note

從 EFA 1.30.0 開始，根據預設會安裝 Open MPI 4 和 Open MPI 5。您可以選擇是否指定您要安裝的 Open MPI 版本。若僅要安裝 Open MPI 4，請包括 `--mpi=openmpi4`。若要安裝 Open MPI 5，請納入 `--mpi=openmpi5`。若要同時安裝這兩種版本，請忽略 `--mpi` 選項。

```
$ C:\> sudo ./efa_installer.sh -y
```

Libfabric 會安裝到 `/opt/amazon/efa`。Open MPI 4 會安裝到 `/opt/amazon/openmpi`。Open MPI 5 會安裝到 `/opt/amazon/openmpi5`。

### Intel MPI only

如果您打算只將 EFA 與 Intel MPI 搭配使用，您無需使用 Libfabric 和 Open MPI 即可安裝 EFA 軟體。在這種情況下，Intel MPI 會使用其內嵌 Libfabric。如果您選擇這麼做，您必須完成步驟 5：安裝 Intel MPI。

若要在不使用 Libfabric 和 Open MPI 的情況下安裝 EFA 軟體，請執行下列命令。

```
$ C:\> sudo ./efa_installer.sh -y --minimal
```

8. 如果 EFA 安裝程式提示您重新啟動執行個體，請執行這項操作，然後重新連線至執行個體。否則，請登出執行個體，然後重新登入以完成安裝。

## 步驟 4 : (選用) 啟用 Open MPI 5

### Note

只有在您打算使用 Open MPI 5 時，才執行此步驟。

從 EFA 1.30.0 開始，根據預設會安裝 Open MPI 4 和 Open MPI 5。或者，您也可以選擇僅安裝 Open MPI 4 或 Open MPI 5。

如果您在步驟 3：安裝 EFA 軟體中選擇安裝且有意使用 Open MPI 5，則必須執行下列步驟以啟用 Open MPI 5。

### 啟用 Open MPI 5

1. 將 Open MPI 5 新增至 PATH 環境變數。

```
$ module load openmpi5
```

2. 確認是否已啟用 Open MPI 5 以供使用。

```
$ which mpicc
```

此命令應會傳回 Open MPI 5 安裝目錄 - /opt/amazon/openmpi5。

3. (選用) 若要確保在每次執行個體啟動時，Open MPI 5 都會新增至 PATH 環境變數，請執行下列動作：

bash shell

將 `module load openmpi5` 新增至 `/home/username/.bashrc` 和 `/home/username/.bash_profile`。

csh and tcsh shells

將 `module load openmpi5` 新增至 `/home/username/.cshrc`。

如果您需要從 PATH 環境變數移除 Open MPI 5，請執行下列命令，並從 shell 啟動指令碼中移除該命令。

```
$ module unload openmpi5
```

## 步驟 5 : (選用) 安裝 Intel MPI

### Important

只有在您打算使用 Intel MPI 時，才執行此步驟。如果您打算使用 Open MPI，請略過此步驟。

Intel MPI 需要額外的安裝和環境變數組態。

### 先決條件

確保執行下列步驟的使用者具有 sudo 許可。

### 安裝 Intel MPI

1. 如要下載 Intel MPI 安裝指令碼，請執行下列步驟
  - a. 請造訪 [Intel 網站](#)。
  - b. 在該網頁的 Intel MPI Library 區段中，選擇 Intel MPI Library for Linux Offline 安裝程式。
2. 執行您在上一個步驟中下載的安裝指令碼。

```
$ C:\> sudo bash installation_script_name.sh
```

3. 在安裝程式中，選擇 Accept & install (接受並安裝)。
4. 閱讀 Intel 改進計畫，選擇適當的選項，然後選擇 Begin Installation (開始安裝)。
5. 完成安裝時，請選擇 Close (關閉)。
6. 根據預設，Intel MPI 會使用其內嵌 (內部) Libfabric。您可以將 Intel MPI 設定為使用 EFA 安裝程式隨附的 LibFabric。通常，EFA 安裝程式隨附的 Libfabric 版本比 Intel MPI 版本更高。在某些情況下，EFA 安裝程式隨附的 Libfabric 比 Intel MPI 效能更佳。若要將 Intel MPI 設定為使用 EFA 安裝程式隨附的 Libfabric，請根據您的 shell 執行下列其中一項操作。

### bash shells

將以下陳述式新增至 `/home/username/.bashrc` 和 `/home/username/.bash_profile`。

```
export I_MPI_OFI_LIBRARY_INTERNAL=0
```



## csch and tcsh shells

將以下陳述式新增至 `/home/username/.cshrc`。

```
setenv I_MPI_OFI_LIBRARY_INTERNAL 0
```

7. 新增下列 `source` 命令到您的 shell 指令碼，從安裝目錄中獲取 `vars.sh` 的指令碼，從而在每次啟動執行個體時設定編譯器環境。根據 shell 而定，執行以下其中一個命令。

## bash shells

將以下陳述式新增至 `/home/username/.bashrc` 和 `/home/username/.bash_profile`。

```
source /opt/intel/oneapi/mpi/latest/env/vars.sh
```

## csch and tcsh shells

將以下陳述式新增至 `/home/username/.cshrc`。

```
source /opt/intel/oneapi/mpi/latest/env/vars.csh
```

8. 根據預設，如果因為設定錯誤而無法使用 EFA，Intel MPI 會預設使用 TCP/IP 網路堆疊，這可能會導致應用程式效能降低。透過將 `I_MPI_OFI_PROVIDER` 設定為 `efa`，可防止此情況發生。如果無法使用 EFA，這會導致 Intel MPI 失敗，並出現下列錯誤：

```
Abort (XXXXXX) on node 0 (rank 0 in comm 0): Fatal error in PMPI_Init: OtherMPI
error,
MPIR_Init_thread (XXX).....:
MPID_Init (XXXX).....:
MPIDI_OFI_mpi_init_hook (XXXX):
open_fabric (XXXX).....:
find_provider (XXXX).....:
OFI fi_getinfo() failed (ofi_init.c:2684:find_provider:
```

根據 shell 而定，執行以下其中一個命令。

## bash shells

將以下陳述式新增至 `/home/username/.bashrc` 和 `/home/username/.bash_profile`。

```
export I_MPI_OFI_PROVIDER=efa
```

## csh and tcsh shells

將以下陳述式新增至 `/home/username/.cshrc`。

```
setenv I_MPI_OFI_PROVIDER efa
```

9. 根據預設，Intel MPI 不會列印偵錯資訊。您可以指定不同的詳細資訊等級來控制偵錯資訊。可能的值 (依提供的詳細資訊量排序) 為：0 (預設)、1、2、3、4、5。等級 1 和更高等級會列印 `libfabric version` 和 `libfabric provider`。使用 `libfabric version` 檢查 Intel MPI 使用的是內部 LibFabric 還是 EFA 安裝程式隨附的 LibFabric。如果它使用的是內部 Libfabric，則版本字尾為 `impi`。使用 `libfabric provider` 檢查 Intel MPI 是否的是 EFA 還是或 TCP/IP 網絡。如果使用的是 EFA，則值為 `efa`。如果使用的是 TCP/IP，則值為 `tcp;ofi_rxm`。

若要啟用偵錯資訊，請根據您的 shell 執行以下其中一個命令。

## bash shells

將以下陳述式新增至 `/home/username/.bashrc` 和 `/home/username/.bash_profile`。

```
export I_MPI_DEBUG=value
```

## csh and tcsh shells

將以下陳述式新增至 `/home/username/.cshrc`。

```
setenv I_MPI_DEBUG value
```

10. 根據預設，Intel MPI 會使用作業系統的共用記憶體 (shm) 進行節點內通訊，並且僅將 Libfabric (ofi) 用於節點間通訊。通常，此組態可提供最佳效能。不過，在某些情況下，Intel MPI shm 結構可能會導致某些應用程式無限期中止。

若要解決此問題，您可以強制 Intel MPI 使用 Libfabric 進行節點內和節點間通訊。為此，請根據您的 shell，執行以下其中一個命令。

### bash shells

將以下陳述式新增至 `/home/username/.bashrc` 和 `/home/username/.bash_profile`。

```
export I_MPI_FABRICS=ofi
```

### csh and tcsh shells

將以下陳述式新增至 `/home/username/.cshrc`。

```
setenv I_MPI_FABRICS ofi
```

#### Note

EFA Libfabric 提供者會使用作業系統的共用記憶體進行節點內通訊。這表示將 `I_MPI_FABRICS` 設定為 `ofi` 可產生與預設 `shm:ofi` 組態類似的效能。

11. 登出執行個體，然後重新登入。

如果您不要再使用 Intel MPI，請從 shell 啟動指令碼中移除環境變數。

## 步驟 6：停用 ptrace 保護

為了改善 HPC 應用程式的效能，在相同執行個體上執行處理序時，Libfabric 會使用執行個體的本機記憶體進行處理序間通訊。

共用記憶體功能使用 Cross Memory Attach (CMA)，其未受 ptrace 保護支援。如果您使用的 Linux 發行套件依預設已啟用 ptrace 防護，例如 Ubuntu，您必須停用該保護。如果您的 Linux 發行套件依預設未啟用 ptrace 保護，請略過此步驟。

### 停用 ptrace 保護

請執行下列其中一項：

- 如果出於測試用途要暫時停用 ptrace 保護，請執行下列命令。

```
$ sudo sysctl -w kernel.yama.ptrace_scope=0
```

- 若要永久停用 ptrace 保護，請將 `kernel.yama.ptrace_scope = 0` 新增至 `/etc/sysctl.d/10-ptrace.conf`，並重新啟動執行個體。

## 步驟 7. 確認安裝

### 確認已成功安裝

1. 若要確認已成功安裝 MPI，請執行下列命令：

```
$ which mpicc
```

- 若為 Open MPI，傳回的路徑應包括 `/opt/amazon/`
  - 若為 Intel MPI，傳回的路徑應包括 `/opt/intel/`。如果沒有得到預期輸出，請確保已獲取 Intel MPI `vars.sh` 指令碼。
2. 若要確認是否已成功安裝 EFA 軟體元件和 Libfabric，請執行下列命令。

```
$ C:\> fi_info -p efa -t FI_EP_RDM
```

此命令應該會傳回 Libfabric EFA 介面的相關資訊。以下範例顯示命令輸出。

```
provider: efa
 fabric: EFA-fe80::94:3dff:fe89:1b70
 domain: efa_0-rdm
 version: 2.0
 type: FI_EP_RDM
 protocol: FI_PROTO_EFA
```

## 步驟 8：安裝 HPC 應用程式

在暫時執行個體上安裝 HPC 應用程式。安裝程序依特定的 HPC 應用程式而異。如需詳細資訊，請參閱 Amazon Linux 2 使用者指南中的 [管理 AL2 執行個體上的軟體](#)。

**Note**

參考 HPC 應用程式的文件，以取得安裝指示。

## 步驟 9：建立已啟用 EFA 的 AMI

安裝所需的軟體元件之後，您需要建立 AMI，以重複用來啟動具備 EFA 功能的執行個體。

### 從暫時執行個體建立 AMI

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取您建立的暫時執行個體，然後選取 Actions (動作)、Image (映像)、Create image (建立映像)。
4. 對於 Create image (建立映像)，執行下列動作：
  - a. 對於 Image name (映像名稱)，輸入 AMI 的描述性名稱。
  - b. (選用) 對於 Image description (映像描述)，輸入 AMI 的簡短描述。
  - c. 選擇 Create image。
5. 在導覽窗格中，選擇 AMIs (AMI)。
6. 在清單中找出您建立的 AMI。等待狀態從 pending 變為 available，再繼續進行下一個步驟。

## 步驟 10：在集群放置群組中啟動已啟用 EFA 的執行個體

使用具備 EFA 功能的 AMI (在步驟 7 中建立) 和具備 EFA 功能的安全群組 (在步驟 1 中建立)，在集群放置群組中啟動具備 EFA 功能的執行個體。

**Note**

- 在集群放置群組中啟動具備 EFA 的執行個體並非絕對必要。不過，建議在集群放置群組中執行具備 EFA 功能的執行個體，因為這樣會在單一可用區域的低延遲群組中啟動執行個體。
- 要確保在擴展叢集執行個體時容量可用，您可以為集群放置群組建立容量保留。如需詳細資訊，請參閱 [集群放置群組中的容量保留](#)。

## 啟動暫時執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)，接著選擇 Launch Instances (啟動執行個體) 來開啟新的啟動執行個體精靈。
3. (選用) 在 Name and tags (名稱和標籤) 區段中，為執行個體命名，例如，EFA-instance。該名稱將指派作為執行個體的資源標籤 (Name=*EFA-instance*)。
4. 在 Application and OS Images (應用程式和作業系統映像) 區段中，選取 My AMIs (我的 AMI)，接著選取您在前一個步驟中建立的 AMI。
5. 在 Instance type (執行個體類型) 區段中，選取 [支援的執行個體類型](#)。
6. 在 Key pair (金鑰對) 區段中，選取要用於執行個體的金鑰對。
7. 在 Network settings (網路設定) 區段，選擇 Edit (編輯)，接著執行下列動作：
  - a. 對於 Subnet (子網)，請選擇要在其中啟動執行個體的字網。若您未選取子網，便無法啟用 EFA 的執行個體。
  - b. 針對 Firewall (security groups) (防火牆 (安全群組))，選取 Select existing security group (選取現有的安全群組)，接著選取您在前一個步驟中建立的安全群組。
  - c. 展開 Advanced network configuration (進階網路組態) 區段，並針對 Elastic Fabric Adapter，選取 Enable (啟用)。
8. (選用) 在 儲存 區段中，根據需求設定磁碟區。
9. 在 進階詳細資訊 區段中，針對 配置群組名稱，選取要在其中啟動執行個體的集群放置群組。如果您需要建立新的集群放置群組，請選取 建立新的配置群組。
10. 在右邊的 摘要 面板，針對 執行個體的數目，輸入要啟動的具備 EFA 功能的執行個體數量，接著選擇 啟動執行個體。

## 步驟 11：終止暫時執行個體

此時，您不再需要您啟動的暫時執行個體。您可以終止執行個體，以停止產生費用。

### 終止暫時執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取您建立的暫時執行個體，然後選取動作、執行個體狀態、終止執行個體。
4. 出現確認提示時，請選擇終止。

## 步驟 12：啟用無密碼 SSH

若要讓您的應用程式能夠跨越叢集中的所有執行個體執行，您必須啟用從領導節點到成員節點的無密碼 SSH 存取。領導節點是您在其中執行應用程式的執行個體。叢集中的其餘執行個體為成員節點。

在叢集的執行個體之間啟用無密碼 SSH

1. 選取叢集中的一個執行個體作為領導節點，並與它連線。
2. 在領導節點上停用 `strictHostKeyChecking` 並啟用 `ForwardAgent`。使用您偏好的文字編輯器開啟 `~/.ssh/config`，然後新增下列命令。

```
Host *
 ForwardAgent yes
Host *
 StrictHostKeyChecking no
```

3. 產生 RSA 金鑰對。

```
$ ssh-keygen -t rsa -N "" -f ~/.ssh/id_rsa
```

在 `$HOME/.ssh/` 目錄中建立金鑰對。

4. 變更領導節點上私有金鑰的許可。

```
$ chmod 600 ~/.ssh/id_rsa
chmod 600 ~/.ssh/config
```

5. 使用您偏好的文字編輯器開啟 `~/.ssh/id_rsa.pub` 並複製該金鑰。
6. 對於叢集中的每個成員節點，執行以下操作：
  - a. 連線到執行個體。
  - b. 使用您偏好的文字編輯器開啟 `~/.ssh/authorized_keys`，然後新增您先前複製的公有金鑰。
7. 若要測試無密碼的 SSH 是否如預期運作，請連線至領導節點並執行下列命令。

```
$ ssh member_node_private_ip
```

您應該連線至成員節點，而不會提示您輸入金鑰或密碼。

## 開始使用 EFA 和 NCCL

NVIDIA Collective Communications Library (NCCL) 是一種標準集體溝通常式程式庫，用於多個 GPU 橫跨單一節點或多個節點。NCCL 可與 EFA、Libfabric 和 MPI 一同使用來支援不同的機器學習工作負載。如需詳細資訊，請參閱 [NCCL](#) 網站。

### Note

- 使用 EFA 的 NCCL 僅支援 p3dn.24xlarge、p4d.24xlarge 和 p5.48xlarge。
- EFA 僅支援 NCCL 2.4.2 和更新版本。

下列教學課程協助您啟動 EFA 和 NCCL 啟用的執行個體叢集用於機器學習工作負載。

- [使用基礎 AMI](#)
- [使用 AWS 深度學習 AMI](#)

### 使用基礎 AMI

下列步驟可協助您透過其中一個[支援的基本作業系統](#)，開始搭配使用 Elastic Fabric Adapter 和 AMI。

### Note

- 只支援 p3dn.24xlarge、p4d.24xlarge 和 p5.48xlarge 執行個體類型。
- 只支援 Amazon Linux 2、RHEL 7/8/9、CentOS 7、岩石 Linux 8/9 和 Ubuntu 20.04/22.04 基本 AMI。

### 目錄

- [步驟 1：準備具備 EFA 功能的安全群組](#)
- [步驟 2：啟動暫時執行個體](#)
- [步驟 3：安裝 Nvidia GPU 驅動程式、Nvidia CUDA 工具組和 cuDNN](#)
- [步驟 4：安裝 GDRY](#)
- [步驟 5：安裝 EFA 軟體](#)



- [步驟 6：安裝 NCCL](#)
- [步驟 7：安裝 aws-ofi-nccl 插件](#)
- [步驟 8：安裝 NCCL Test](#)
- [步驟 9：測試您的 EFA 和 NCCL 組態](#)
- [步驟 10：安裝您的機器學習應用程式](#)
- [步驟 11：建立啟用 EFA 和 NCCL 的 AMI](#)
- [步驟 12：終止暫時執行個體](#)
- [步驟 13：在集群放置群組中啟動已啟用 EFA 和 NCCL 的執行個體](#)
- [步驟 14：啟用無密碼 SSH](#)

### 步驟 1：準備具備 EFA 功能的安全群組

EFA 需要有安全群組來允許往返安全群組自身的所有傳入和傳出流量。下列程序會建立一個安全群組，該安全群組允許所有傳入和傳出流量，並允許來自任何 IPv4 地址的傳入 SSH 流量以進行 SSH 連線。

#### Important

此安全群組僅供測試之用。對於生產環境，我們建議您建立傳入 SSH 規則，該規則僅允許來自連線 IP 地址的流量，例如電腦的 IP 地址或區域網路中的一系列 IP 地址。

如需其他案例，請參閱[不同使用案例的安全群組規則](#)。

### 建立具備 EFA 功能的安全群組

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Security Groups (安全群組)，然後選擇 Create Security Group (建立安全群組)。
3. 在 Create Security Group (建立安全群組) 視窗中，執行下列動作：
  - a. 對於 Security group name (安全群組名稱)，輸入安全群組的描述性名稱，例如 EFA-enabled security group。
  - b. (選用) 對於 Description (描述)，輸入安全群組的簡短描述。
  - c. 對於 VPC，選取您打算讓具備 EFA 功能的執行個體在其中啟動的 VPC。

- d. 選擇 Create Security Group (建立安全群組)。
4. 選取您建立的安全群組，在 Details (詳細資料) 索引標籤上，複製 Security group ID (安全群組 ID)。
5. 在仍然選取安全群組的情況下，依序選取 Actions (動作) 和 Edit inbound rules (編輯傳入規則)，然後執行下列動作。
  - a. 選擇新增規則。
  - b. 針對類型，選擇所有流量。
  - c. 若為 Source type (來源類型)，選擇 Custom (自訂)，然後將您複製的安全群組 ID 貼到欄位中。
  - d. 選擇新增規則。
  - e. 針對 Type (類型)，選擇 SSH。
  - f. 針對 Source type (來源類別)，選擇 Anywhere-IPv4(隨處 - IPv4)。
  - g. 選擇儲存規則。
6. 在仍然選取安全群組的情況下，依序選取 Actions (動作) 和 Edit outbound rules (編輯傳出規則)，然後執行下列動作。
  - a. 選擇新增規則。
  - b. 針對類型，選擇所有流量。
  - c. 若為 Destination type (目的地類型)，選擇 Custom (自訂)，然後將您複製的安全群組 ID 貼到欄位中。
  - d. 選擇儲存規則。

## 步驟 2：啟動暫時執行個體

啟動暫時執行個體，以用來安裝和設定 EFA 軟體元件。您可以使用此執行個體來建立具備 EFA 功能的 AMI，再從中啟動具備 EFA 功能的執行個體。

### 啟動暫時執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)，接著選擇 Launch Instances (啟動執行個體) 來開啟新的啟動執行個體精靈。
3. (選用) 在 Name and tags (名稱和標籤) 區段中，為執行個體命名，例如，EFA-instance。該名稱將指派作為執行個體的資源標籤 (Name=*EFA-instance*)。

- 在 Application and OS Images (應用程式和作業系統映像) 區段中，為其中一個[支援的作業系統](#) 選取 AMI。
- 在執行個體類型區段中，選取 p3dn.24xlarge、p4d.24xlarge 或 p5.48xlarge。
- 在 Key pair (金鑰對) 區段中，選取要用於執行個體的金鑰對。
- 在 Network settings (網路設定) 區段，選擇 Edit (編輯)，接著執行下列動作：
  - 對於 Subnet (子網)，請選擇要在其中啟動執行個體子網。若您未選取子網，便無法啟用 EFA 的執行個體。
  - 針對 Firewall (security groups) (防火牆 (安全群組))，選取 Select existing security group (選取現有的安全群組)，接著選取您在前一個步驟中建立的安全群組。
  - 展開 Advanced network configuration (進階網路組態) 區段，並針對 Elastic Fabric Adapter，選取 Enable (啟用)。
- 在 儲存 區段中，根據需求設定磁碟區。

#### Note

您必須為 Nvidia CUDA 工具組佈建額外的 10 到 20 GiB 儲存空間。如果您沒有佈建足夠的儲存空間，您會在嘗試安裝 Nvidia 驅動程式和 CUDA 工具組時，收到 `insufficient disk space` 錯誤。

- 在右邊的 Summary (摘要) 面板中，選擇 Launch instance (啟動執行個體)。

### 步驟 3：安裝 Nvidia GPU 驅動程式、Nvidia CUDA 工具組和 cuDNN

#### Amazon Linux 2

##### 安裝 Nvidia GPU 驅動程式、Nvidia CUDA 工具組和 cuDNN

- 為確保所有軟體套件皆為最新版本，請對您的執行個體執行快速軟體更新。

```
$ sudo yum upgrade -y && sudo reboot
```

重新啟動執行個體後，請重新與它連線。

- 請安裝 Nvidia GPU 驅動程式和 Nvidia CUDA 工具組需要的公用程式。

```
$ sudo yum groupinstall 'Development Tools' -y
```

### 3. 停用 nouveau 開放原始碼驅動程式。

- a. 針對您目前執行的核心版本，安裝需要的公用程式和核心標頭套件。

```
$ sudo yum install -y wget kernel-devel-$(uname -r) kernel-headers-$(uname -r)
```

- b. 新增 nouveau 至 `/etc/modprobe.d/blacklist.conf` 拒絕清單檔案。

```
$ cat << EOF | sudo tee --append /etc/modprobe.d/blacklist.conf
blacklist vga16fb
blacklist nouveau
blacklist rivafb
blacklist nvidiafb
blacklist rivatv
EOF
```

- c. 將 `GRUB_CMDLINE_LINUX="rdblacklist=nouveau"` 附加至 `grub` 檔案並重建 `Grub` 組態。

```
$ echo 'GRUB_CMDLINE_LINUX="rdblacklist=nouveau"' | sudo tee -a /etc/default/grub \
&& sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

### 4. 重新啟動執行個體並重新連線至其中。

### 5. 準備所需的儲存庫

- a. 安裝 DKMS 的 EPEL 儲存庫，並為您的 Linux 發行版本啟用任何可選儲存庫。

```
$ sudo yum install -y https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
```

- b. 安裝 CUDA 儲存庫公有 GPG 金鑰。

```
$ distribution='rhel7'
```

- c. 設定 CUDA 網路儲存庫並更新儲存庫快取。

```
$ ARCH=$(/bin/arch) \
&& sudo yum-config-manager --add-repo http://developer.download.nvidia.com/compute/cuda/repos/${distribution}/${ARCH}/cuda-${distribution}.repo \
```

```
&& sudo yum clean expire-cache
```

- d. (僅限核心版本 5.10) 唯有當您使用 Amazon Linux 2 搭配核心版本 5.10 時，才執行下列步驟。如果您是使用 Amazon Linux 2 搭配核心版本 4.12，請跳過下列步驟。若要檢查核心版本，請執行 `uname -r`。
  - i. 建立名為 `/etc/dkms/nvidia.conf` 的 Nvidia 驅動程式組態檔案。

```
$ sudo mkdir -p /etc/dkms \
&& echo "MAKE[0]=\''make' -j2 module SYSSRC=\${kernel_source_dir}
IGNORE_XEN_PRESENCE=1 IGNORE_PREEMPT_RT_PRESENCE=1 IGNORE_CC_MISMATCH=1
CC=/usr/bin/gcc10-gcc\" | sudo tee /etc/dkms/nvidia.conf
```

- ii. (僅限 p4d.24xlarge 和 p5.48xlarge) 複製 Nvidia 驅動程序組態檔案。

```
$ sudo cp /etc/dkms/nvidia.conf /etc/dkms/nvidia-open.conf
```

## 6. 安裝 Nvidia GPU 驅動程式、NVIDIA CUDA 工具組和 cuDNN。

- p3dn.24xlarge

```
$ sudo yum clean all \
&& sudo yum -y install kmod-nvidia-latest-dkms nvidia-driver-latest-dkms \
&& sudo yum -y install cuda-drivers-fabricmanager cuda lib cudnn8-devel
```

- p4d.24xlarge 和 p5.48xlarge

```
$ sudo yum clean all \
&& sudo yum -y install kmod-nvidia-open-dkms nvidia-driver-latest-dkms \
&& sudo yum -y install cuda-drivers-fabricmanager cuda lib cudnn8-devel
```

7. 重新啟動執行個體並重新連線至其中。
8. (僅限 p4d.24xlarge 和 p5.48xlarge) 啟動 Nvidia Fabric Manager 服務，並確保它會在執行個體啟動時自動啟動。NV Switch Management 需要使用 Nvidia Fabric Manager。

```
$ sudo systemctl enable nvidia-fabricmanager && sudo systemctl start nvidia-fabricmanager
```

## 9. 確認每次執行個體啟動時都已設定 CUDA 路徑。

- 針對 bash shell，新增以下陳述式到 `/home/username/.bashrc` 和 `/home/username/.bash_profile`。

```
export PATH=/usr/local/cuda/bin:$PATH
export LD_LIBRARY_PATH=/usr/local/cuda/lib64:/usr/local/cuda/extras/CUPTI/
lib64:$LD_LIBRARY_PATH
```

- 針對 tcsh shell，新增以下陳述式到 `/home/username/.cshrc`。

```
setenv PATH=/usr/local/cuda/bin:$PATH
setenv LD_LIBRARY_PATH=/usr/local/cuda/lib64:/usr/local/cuda/extras/CUPTI/
lib64:$LD_LIBRARY_PATH
```

10. 若要確認 Nvidia GPU 驅動程式正確運作，請執行下列命令。

```
$ nvidia-smi -q | head
```

命令應該回傳 Nvidia GPU、Nvidia GPU 驅動程式和 Nvidia CUDA 工具組的資訊。

## CentOS 7

### 安裝 Nvidia GPU 驅動程式、Nvidia CUDA 工具組和 cuDNN

1. 為確保所有軟體套件皆為最新版本，請對您的執行個體執行快速軟體更新。

```
$ sudo yum upgrade -y && sudo reboot
```

重新啟動執行個體後，請重新與它連線。

2. 請安裝 Nvidia GPU 驅動程式和 Nvidia CUDA 工具組需要的公用程式。

```
$ sudo yum groupinstall 'Development Tools' -y \
&& sudo yum install -y tar bzip2 make automake pciutils elfutils-libelf-devel
libglvnd-devel iptables firewalld vim bind-utils
```

3. 若要使用 Nvidia GPU 驅動程式，您必須先停用 nouveau 開源驅動程式。

- a. 針對您目前執行的核心版本，安裝需要的公用程式和核心標頭套件。

```
$ sudo yum install -y wget kernel-devel-$(uname -r) kernel-headers-$(uname -
r)
```

- b. 新增 nouveau 至 `/etc/modprobe.d/blacklist.conf` 拒絕清單檔案。

```
$ cat << EOF | sudo tee --append /etc/modprobe.d/blacklist.conf
blacklist vga16fb
blacklist nouveau
blacklist rivafb
blacklist nvidiafb
blacklist rivatv
EOF
```

- c. 使用您偏好的文字編輯器開啟 `/etc/default/grub`，然後新增下列命令。

```
GRUB_CMDLINE_LINUX="rdblacklist=nouveau"
```

- d. 重建 Grub 組態。

```
$ sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

4. 重新啟動執行個體並重新連線至其中。
5. 安裝 Nvidia GPU 驅動程式、NVIDIA CUDA 工具組和 cuDNN。

- a. 安裝 DKMS 的 EPEL 儲存庫，並為您的 Linux 發行版本啟用任何可選儲存庫。

```
$ sudo yum install -y https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
```

- b. 安裝 CUDA 儲存庫公有 GPG 金鑰。

```
$ distribution='rhel7'
```

- c. 設定 CUDA 網路儲存庫並更新儲存庫快取。

```
$ ARCH=$(/bin/arch) \
&& sudo yum-config-manager --add-repo http://developer.download.nvidia.com/
compute/cuda/repos/$distribution/${ARCH}/cuda-$distribution.repo \
&& sudo yum clean expire-cache
```

- d. 安裝 NVIDIA、CUDA 驅動程式和 cuDNN。

```
$ sudo yum clean all \
&& sudo yum -y install cuda-drivers-fabricmanager cuda lib cudnn8-devel
```

6. 重新啟動執行個體並重新連線至其中。

7. (僅限 p4d.24xlarge 和 p5.48xlarge) 啟動 Nvidia Fabric Manager 服務，並確保它會在執行個體啟動時自動啟動。NV Switch Management 需要使用 Nvidia Fabric Manager。

```
$ sudo systemctl start nvidia-fabricmanager \
&& sudo systemctl enable nvidia-fabricmanager
```

8. 確認每次執行個體啟動時都已設定 CUDA 路徑。

- 針對 bash shell，新增以下陳述式到 `/home/username/.bashrc` 和 `/home/username/.bash_profile`。

```
export PATH=/usr/local/cuda/bin:$PATH
export LD_LIBRARY_PATH=/usr/local/cuda/lib64:/usr/local/cuda/extras/CUPTI/
lib64:$LD_LIBRARY_PATH
```

- 針對 tcsh shell，新增以下陳述式到 `/home/username/.cshrc`。

```
setenv PATH=/usr/local/cuda/bin:$PATH
setenv LD_LIBRARY_PATH=/usr/local/cuda/lib64:/usr/local/cuda/extras/CUPTI/
lib64:$LD_LIBRARY_PATH
```

9. 若要確認 Nvidia GPU 驅動程式正確運作，請執行下列命令。

```
$ nvidia-smi -q | head
```

命令應該回傳 Nvidia GPU、Nvidia GPU 驅動程式和 Nvidia CUDA 工具組的資訊。

## RHEL 7/8/9 and Rocky Linux 8/9

### 安裝 Nvidia GPU 驅動程式、Nvidia CUDA 工具組和 cuDNN

1. 為確保所有軟體套件皆為最新版本，請對您的執行個體執行快速軟體更新。

```
$ sudo yum upgrade -y && sudo reboot
```

重新啟動執行個體後，請重新與它連線。

2. 請安裝 Nvidia GPU 驅動程式和 Nvidia CUDA 工具組需要的公用程式。

```
$ sudo yum groupinstall 'Development Tools' -y
```



3. 若要使用 Nvidia GPU 驅動程式，您必須先停用 nouveau 開源驅動程式。

- a. 針對您目前執行的核心版本，安裝需要的公用程式和核心標頭套件。

```
$ sudo yum install -y wget kernel-devel-$(uname -r) kernel-headers-$(uname -r)
```

- b. 新增 nouveau 至 `/etc/modprobe.d/blacklist.conf` 拒絕清單檔案。

```
$ cat << EOF | sudo tee --append /etc/modprobe.d/blacklist.conf
blacklist vga16fb
blacklist nouveau
blacklist rivafb
blacklist nvidiafb
blacklist rivatv
EOF
```

- c. 使用您偏好的文字編輯器開啟 `/etc/default/grub`，然後新增下列命令。

```
GRUB_CMDLINE_LINUX="rdblacklist=nouveau"
```

- d. 重建 Grub 組態。

```
$ sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

4. 重新啟動執行個體並重新連線至其中。

5. 安裝 Nvidia GPU 驅動程式、NVIDIA CUDA 工具組和 cuDNN。

- a. 安裝 DKMS 的 EPEL 儲存庫，並為您的 Linux 發行版本啟用任何可選儲存庫。

- RHEL 7

```
$ sudo yum install -y https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
```

- RHEL 8 和 Rocky Linux 8/9

```
$ sudo yum install -y https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
```

- RHEL 9

```
$ sudo yum install -y https://dl.fedoraproject.org/pub/epel/epel-release-latest-9.noarch.rpm
```

- b. 安裝 CUDA 儲存庫公有 GPG 金鑰。

```
$ distribution=$(. /etc/os-release;echo $ID`rpm -E "%{?rhel}%{?fedora}"`)
```

- c. 設定 CUDA 網路儲存庫並更新儲存庫快取。

```
$ ARCH=$(/bin/arch) \
&& sudo yum-config-manager --add-repo http://developer.download.nvidia.com/compute/cuda/repos/$distribution/${ARCH}/cuda-$distribution.repo \
&& sudo yum clean expire-cache
```

- d. 安裝 NVIDIA、CUDA 驅動程式和 cuDNN。

```
$ sudo yum clean all \
&& sudo yum -y install cuda-drivers-fabricmanager cuda libcuda8-devel
```

6. 重新啟動執行個體並重新連線至其中。

7. (僅限 p4d.24xlarge 和 p5.48xlarge) 啟動 Nvidia Fabric Manager 服務，並確保它會在執行個體啟動時自動啟動。NV Switch Management 需要使用 Nvidia Fabric Manager。

```
$ sudo systemctl start nvidia-fabricmanager \
&& sudo systemctl enable nvidia-fabricmanager
```

8. 確認每次執行個體啟動時都已設定 CUDA 路徑。

- 針對 bash shell，新增以下陳述式到 `/home/username/.bashrc` 和 `/home/username/.bash_profile`。

```
export PATH=/usr/local/cuda/bin:$PATH
export LD_LIBRARY_PATH=/usr/local/cuda/lib64:/usr/local/cuda/extras/CUPTI/
lib64:$LD_LIBRARY_PATH
```

- 針對 tcsh shell，新增以下陳述式到 `/home/username/.cshrc`。

```
setenv PATH=/usr/local/cuda/bin:$PATH
setenv LD_LIBRARY_PATH=/usr/local/cuda/lib64:/usr/local/cuda/extras/CUPTI/
lib64:$LD_LIBRARY_PATH
```

- 若要確認 Nvidia GPU 驅動程式正確運作，請執行下列命令。

```
$ nvidia-smi -q | head
```

命令應該回傳 Nvidia GPU、Nvidia GPU 驅動程式和 Nvidia CUDA 工具組的資訊。

## Ubuntu 20.04/22.04

### 安裝 Nvidia GPU 驅動程式、Nvidia CUDA 工具組和 cuDNN

- 為確保所有軟體套件皆為最新版本，請對您的執行個體執行快速軟體更新。

```
$ sudo apt-get update && sudo apt-get upgrade -y
```

- 請安裝 Nvidia GPU 驅動程式和 Nvidia CUDA 工具組需要的公用程式。

```
$ sudo apt-get update && sudo apt-get install build-essential -y
```

- 若要使用 Nvidia GPU 驅動程式，您必須先停用 nouveau 開源驅動程式。

- 針對您目前執行的核心版本，安裝需要的公用程式和核心標頭套件。

```
$ sudo apt-get install -y gcc make linux-headers-$(uname -r)
```

- 新增 nouveau 至 `/etc/modprobe.d/blacklist.conf` 拒絕清單檔案。

```
$ cat << EOF | sudo tee --append /etc/modprobe.d/blacklist.conf
blacklist vga16fb
blacklist nouveau
blacklist rivafb
blacklist nvidiafb
blacklist rivatv
EOF
```

- 使用您偏好的文字編輯器開啟 `/etc/default/grub`，然後新增下列命令。

```
GRUB_CMDLINE_LINUX="rdblacklist=nouveau"
```

- 重建 Grub 組態。

```
$ sudo update-grub
```

4. 重新啟動執行個體並重新連線至其中。
5. 新增 CUDA 儲存庫並安裝 Nvidia GPU 驅動程式、NVIDIA CUDA 工具組和 cuDNN。

- p3dn.24xlarge

```
$ sudo apt-key adv --fetch-keys http://developer.download.nvidia.com/compute/machine-learning/repos/ubuntu2004/x86_64/7fa2af80.pub \
&& wget -O /tmp/deeplearning.deb http://developer.download.nvidia.com/compute/machine-learning/repos/ubuntu2004/x86_64/nvidia-machine-learning-repo-ubuntu2004_1.0.0-1_amd64.deb \
&& sudo dpkg -i /tmp/deeplearning.deb \
&& wget -O /tmp/cuda.pin https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2004/x86_64/cuda-ubuntu2004.pin \
&& sudo mv /tmp/cuda.pin /etc/apt/preferences.d/cuda-repository-pin-600 \
&& sudo apt-key adv --fetch-keys https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2004/x86_64/3bf863cc.pub \
&& sudo add-apt-repository 'deb http://developer.download.nvidia.com/compute/cuda/repos/ubuntu2004/x86_64/ /' \
&& sudo apt update \
&& sudo apt install nvidia-dkms-535 \
&& sudo apt install -o Dpkg::Options::='--force-overwrite' cuda-drivers-535 cuda-toolkit-12-3 libcudnn8 libcudnn8-dev -y
```

- p4d.24xlarge 和 p5.48xlarge

```
$ sudo apt-key adv --fetch-keys http://developer.download.nvidia.com/compute/machine-learning/repos/ubuntu2004/x86_64/7fa2af80.pub \
&& wget -O /tmp/deeplearning.deb http://developer.download.nvidia.com/compute/machine-learning/repos/ubuntu2004/x86_64/nvidia-machine-learning-repo-ubuntu2004_1.0.0-1_amd64.deb \
&& sudo dpkg -i /tmp/deeplearning.deb \
&& wget -O /tmp/cuda.pin https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2004/x86_64/cuda-ubuntu2004.pin \
&& sudo mv /tmp/cuda.pin /etc/apt/preferences.d/cuda-repository-pin-600 \
&& sudo apt-key adv --fetch-keys https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2004/x86_64/3bf863cc.pub \
&& sudo add-apt-repository 'deb http://developer.download.nvidia.com/compute/cuda/repos/ubuntu2004/x86_64/ /' \
&& sudo apt update \
&& sudo apt install nvidia-kernel-open-535 \

```

```
&& sudo apt install -o Dpkg::Options::='--force-overwrite' cuda-drivers-535
cuda-toolkit-12-3 libcudnn8 libcudnn8-dev -y
```

6. 重新啟動執行個體並重新連線至其中。
7. (僅限 p4d.24xlarge 和 p5.48xlarge) 安裝 Nvidia Fabric Manager。
  - a. 您安裝的 Nvidia Fabric Manager 版本必須與您在上一個步驟中安裝的 Nvidia 核心模組版本一致。

執行以下命令來判斷 Nvidia 核心模組的版本。

```
$ cat /proc/driver/nvidia/version | grep "Kernel Module"
```

下列為範例輸出。

```
NVRM version: NVIDIA UNIX x86_64 Kernel Module 450.42.01 Tue Jun 15
21:26:37 UTC 2021
```

在上述範例中，已安裝核心模組主要版本 450。這表示您需要安裝 Nvidia Fabric Manager 版本 450。

- b. 安裝 Nvidia Fabric Manager。執行以下命令，並指定在上一步驟中使用的主要版本。

```
$ sudo apt install -o Dpkg::Options::='--force-overwrite' nvidia-
fabricmanager-major_version_number
```

例如，如果已安裝核心模組的主要版本 450，請使用下列命令來安裝同樣版本的 Nvidia Fabric Manager。

```
$ sudo apt install -o Dpkg::Options::='--force-overwrite' nvidia-
fabricmanager-450
```

- c. 啟動服務，並確保服務會在執行個體啟動時自動啟動。NV Switch Management 需要使用 Nvidia Fabric Manager。

```
$ sudo systemctl start nvidia-fabricmanager && sudo systemctl enable nvidia-
fabricmanager
```

8. 確認每次執行個體啟動時都已設定 CUDA 路徑。

- 針對 bash shell，新增以下陳述式到 `/home/username/.bashrc` 和 `/home/username/.bash_profile`。

```
export PATH=/usr/local/cuda/bin:$PATH
export LD_LIBRARY_PATH=/usr/local/cuda/lib64:/usr/local/cuda/extras/CUPTI/
lib64:$LD_LIBRARY_PATH
```

- 針對 tcsh shell，新增以下陳述式到 `/home/username/.cshrc`。

```
setenv PATH=/usr/local/cuda/bin:$PATH
setenv LD_LIBRARY_PATH=/usr/local/cuda/lib64:/usr/local/cuda/extras/CUPTI/
lib64:$LD_LIBRARY_PATH
```

9. 若要確認 Nvidia GPU 驅動程式正確運作，請執行下列命令。

```
$ nvidia-smi -q | head
```

命令應該回傳 Nvidia GPU、Nvidia GPU 驅動程式和 Nvidia CUDA 工具組的資訊。

#### 步驟 4：安裝 GDRY

安裝 GDRCopy 以提高 Libfabric 的效能。如需更多關於 GDRCopy 的詳細資訊，請參閱 [GDRCopy 儲存庫](#)。

Amazon Linux 2, CentOS 7, RHEL 7/8/9, and Rocky Linux 8/9

若要安裝 GDRCopy

1. 安裝所需的依存項目。

```
$ sudo yum -y install dkms rpm-build make check check-devel subunit subunit-
devel
```

2. 下載並解壓縮 GDRCopy 套件。

```
$ wget https://github.com/NVIDIA/gdrcopy/archive/refs/tags/v2.4.tar.gz \
&& tar xf v2.4.tar.gz ; cd gdrcopy-2.4/packages
```

3. 建立 GDRCopy RPM 套件。

```
$ CUDA=/usr/local/cuda ./build-rpm-packages.sh
```

#### 4. 安裝 GDRCopy RPM 套件。

```
$ sudo rpm -Uvh gdrdrv-kmod-2.4-1dkms.noarch*.rpm \
&& sudo rpm -Uvh gdrdrv-2.4-1.x86_64*.rpm \
&& sudo rpm -Uvh gdrdrv-devel-2.4-1.noarch*.rpm
```

## Ubuntu 20.04/22.04

### 若要安裝 GDRCopy

#### 1. 安裝所需的依存項目。

```
$ sudo apt -y install build-essential devscripts debhelper check libsubunit-dev \
fakeroot pkg-config dkms
```

#### 2. 下載並解壓縮 GDRCopy 套件。

```
$ wget https://github.com/NVIDIA/gdrdrv/archive/refs/tags/v2.4.tar.gz \
&& tar xf v2.4.tar.gz \
&& cd gdrdrv-2.4/packages
```

#### 3. 建立 GDRCopy RPM 套件。

```
$ CUDA=/usr/local/cuda ./build-deb-packages.sh
```

#### 4. 安裝 GDRCopy RPM 套件。

```
$ sudo dpkg -i gdrdrv-dkms_2.4-1_amd64.*.deb \
&& sudo dpkg -i libgdrapi_2.4-1_amd64.*.deb \
&& sudo dpkg -i gdrdrv-tests_2.4-1_amd64.*.deb \
&& sudo dpkg -i gdrdrv_2.4-1_amd64.*.deb
```

## 步驟 5：安裝 EFA 軟體

在暫存執行個體上安裝支援 EFA 所需的具備 EFA 功能的核心、EFA 驅動程式、Libfabric 和 Open MPI 堆疊。

## 安裝 EFA 軟體

1. 連接至您啟動的執行個體。如需詳細資訊，請參閱 [連接至您的 Linux 執行個體](#)。
2. 下載 EFA 軟體安裝檔案。軟體安裝檔案已封裝成壓縮 tarball (.tar.gz) 檔案。若要下載最新穩定版本，請使用下列命令：

```
$ C:\> curl -O https://efa-installer.amazonaws.com/aws-efa-installer-1.32.0.tar.gz
```

您也可以在此前的命令中以 latest 取代版本號碼，以取得最新版本。

3. (選用) 驗證 EFA tarball (.tar.gz) 檔案的真確性及完整性。

我們建議您執行這項操作來確認軟體發行者的身分，並檢查檔案自發行以來並未遭到變更或損毀。如果您不想驗證 tarball 檔案，請略過此步驟。

### Note

或者，如果您偏好使用 MD5 或 SHA256 檢查總和來驗證 tarball 檔案，請參閱 [使用檢查總和驗證 EFA 安裝程式](#)。

- a. 下載公有 GPG 金鑰並匯入至您的 keyring。

```
$ wget https://efa-installer.amazonaws.com/aws-efa-installer.key && gpg --import aws-efa-installer.key
```

命令應傳回金鑰值。請記下金鑰的值，因為下一個步驟將需要它。

- b. 驗證 GPG 金鑰的指紋。執行以下命令，並指定上一步驟中的金鑰值。

```
$ gpg --fingerprint key_value
```

命令應傳回與 4E90 91BC BB97 A96B 26B1 5E59 A054 80B1 DD2D 3CCC 相同的指紋。如果指紋不相符，請勿執行 EFA 安裝指令碼，並聯絡 AWS Support。

- c. 下載簽章檔案並驗證 EFA tarball 檔案的簽章。

```
$ wget https://efa-installer.amazonaws.com/aws-efa-installer-1.32.0.tar.gz.sig && gpg --verify ./aws-efa-installer-1.32.0.tar.gz.sig
```



下面顯示了範例輸出。

```
gpg: Signature made Wed 29 Jul 2020 12:50:13 AM UTC using RSA key ID DD2D3CCC
gpg: Good signature from "Amazon EC2 EFA <ec2-efa-maintainers@amazon.com>"
gpg: WARNING: This key is not certified with a trusted signature!
gpg: There is no indication that the signature belongs to the owner.
Primary key fingerprint: 4E90 91BC BB97 A96B 26B1 5E59 A054 80B1 DD2D 3CCC
```

如果結果包含 Good signature，而指紋與上一個步驟中傳回的指紋相符，請繼續下一個步驟。如果指紋不相符，請勿執行 EFA 安裝指令碼，並聯絡 AWS Support。

4. 從壓縮的 .tar.gz 檔案中解壓縮檔案，然後導覽至解壓縮的目錄。

```
$ C:\> tar -xf aws-efa-installer-1.32.0.tar.gz && cd aws-efa-installer
```

5. 執行 EFA 軟體安裝指令碼。

#### Note

從 EFA 1.30.0 開始，根據預設會安裝 Open MPI 4 和 Open MPI 5。除非您需要 Open MPI 5，否則我們會建議您僅安裝 Open MPI 4。下列命令僅會安裝 Open MPI 4。如果您想安裝 Open MPI 4 和 Open MPI 5，請移除 --mpi=openmpi4。

```
$ C:\> sudo ./efa_installer.sh -y --mpi=openmpi4
```

Libfabric 安裝在 /opt/amazon/efa 目錄中，而 Open MPI 則是安裝在 /opt/amazon/openmpi 目錄中。

6. 如果 EFA 安裝程式提示您重新啟動執行個體，請執行這項操作，然後重新連線至執行個體。否則，請登出執行個體，然後重新登入以完成安裝。
7. 確認已成功安裝 EFA 軟體元件。

```
$ C:\> fi_info -p efa -t FI_EP_RDM
```

此命令應該會傳回 Libfabric EFA 介面的相關資訊。以下範例顯示命令輸出。

- p3dn.24xlarge 具有單一網路介面

```
provider: efa
fabric: EFA-fe80::94:3dff:fe89:1b70
domain: efa_0-rdm
version: 2.0
type: FI_EP_RDM
protocol: FI_PROTO_EFA
```

- 具有多個網路介面的 p4d.24xlarge 和 p5.48xlarge

```
provider: efa
fabric: EFA-fe80::c6e:8fff:fef6:e7ff
domain: efa_0-rdm
version: 111.0
type: FI_EP_RDM
protocol: FI_PROTO_EFA
provider: efa
fabric: EFA-fe80::c34:3eff:feb2:3c35
domain: efa_1-rdm
version: 111.0
type: FI_EP_RDM
protocol: FI_PROTO_EFA
provider: efa
fabric: EFA-fe80::c0f:7bff:fe68:a775
domain: efa_2-rdm
version: 111.0
type: FI_EP_RDM
protocol: FI_PROTO_EFA
provider: efa
fabric: EFA-fe80::ca7:b0ff:fea6:5e99
domain: efa_3-rdm
version: 111.0
type: FI_EP_RDM
protocol: FI_PROTO_EFA
```

## 步驟 6：安裝 NCCL

安裝 NCCL。如需更多關於 NCCL 詳細資訊，請參閱 [NCCL 存放庫](#)。

若要安裝 NCCL

1. 請前往 /opt 目錄。

```
$ cd /opt
```

- 複製官方的 NCCL 存放庫至執行個體，並導覽至本地複製的存放庫。

```
$ sudo git clone https://github.com/NVIDIA/nccl.git && cd nccl
```

- 建立並安裝 NCCL 並指定 CUDA 安裝目錄。

```
$ sudo make -j src.build CUDA_HOME=/usr/local/cuda
```

### 步驟 7：安裝 aws-ofi-nccl 插件

此 aws-ofi-nccl 外掛程式會將 NCCL 的連線導向傳輸 API 對應至 Libfabric 的無連線可靠介面。這可讓您在執行以 NCCL 為基礎的應用程式時以網路提供者的方式使用 Libfabric。如需有關 aws-ofi-nccl 外掛程式的詳細資訊，請參閱[aws-ofi-nccl 儲存庫](#)。

若要安裝 aws-ofi-nccl 外掛程式

- 導覽至您的主目錄。

```
$ cd $HOME
```

- ( 僅限 Amazon Linux 2 和 Ubuntu ) 安裝所需的實用程序。

- Amazon Linux 2

```
$ sudo yum install hwloc-devel
```

- Ubuntu 20.04

```
$ sudo apt-get install libhwloc-dev
```

- 下載 aws-ofi-nccl 插件文件。檔案已封裝成壓縮 tarball (.tar.gz)。

```
$ wget https://github.com/aws/aws-ofi-nccl/releases/download/v1.9.1-aws/aws-ofi-nccl-1.9.1-aws.tar.gz
```

- 從壓縮的 .tar.gz 檔案中解壓縮檔案，然後導覽至解壓縮的目錄。

```
$ tar -xf aws-ofi-nccl-1.9.1-aws.tar.gz && cd aws-ofi-nccl-1.9.1-aws
```

- 若要產生製作檔案，請執行 `configure` 指令碼和指定 MPI、Libfabric、NCCL 和 CUDA 安裝目錄。

```
$./configure --prefix=/opt/aws-ofi-nccl --with-mpi=/opt/amazon/openmpi \
--with-libfabric=/opt/amazon/efa \
--with-cuda=/usr/local/cuda \
--enable-platform-aws
```

- 將 Open MPI 目錄新增至 PATH 變數。

```
$ export PATH=/opt/amazon/openmpi/bin/:$PATH
```

- 安裝 `aws-ofi-nccl` 插件。

```
$ make && sudo make install
```

## 步驟 8：安裝 NCCL Test

安裝 NCCL test。NCCL test 可讓您確認 NCCL 已正確安裝，並如預期中執行。如需更多關於 NCCL test 的詳細資訊，請參閱 [nccl-test 存放庫](#)。

若要安裝 NCCL test

- 導覽至您的主目錄。

```
$ cd $HOME
```

- 複製官方的 `nccl-tests` 存放庫至執行個體，並導覽至本地複製的存放庫。

```
$ git clone https://github.com/NVIDIA/nccl-tests.git && cd nccl-tests
```

- 新增 Libfabric 目錄至 `LD_LIBRARY_PATH` 變數。

- Amazon Linux、Amazon Linux 2、RHEL、Rocky Linux 8/9 和 CentOS

```
$ export LD_LIBRARY_PATH=/opt/amazon/efa/lib64:$LD_LIBRARY_PATH
```

- Ubuntu

```
$ export LD_LIBRARY_PATH=/opt/amazon/efa/lib:$LD_LIBRARY_PATH
```

#### 4. 安裝 NCCL test 並指定 MPI、NCCL 和 CUDA 安裝目錄。

```
$ make MPI=1 MPI_HOME=/opt/amazon/openmpi NCCL_HOME=/opt/nccl/build CUDA_HOME=/usr/local/cuda
```

### 步驟 9：測試您的 EFA 和 NCCL 組態

執行測試以確保已對 EFA 和 NCCL 正確設定暫時執行個體。

若要測試您的 EFA 和 NCCL 組態

1. 建立主機檔案指定要執行測試的主機。下列命令另建立名為 my-hosts 的主機檔案，包含執行個體本身的參考資料。

#### IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://169.254.169.254/latest/meta-data/local-ipv4 >> my-hosts
```

#### IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/local-ipv4 >> my-hosts
```

2. 執行測試並指定主機檔案 (--hostfile) 以及要使用的 GPU 數量 (-n)。下列命令在執行個體本身為 8 GPU，和下列指定的環境變數上執行 all\_reduce\_perf 測試。
  - FI\_EFA\_USE\_DEVICE\_RDMA=1—(僅限 p4d.24xlarge) 使用裝置的 RDMA 功能進行單面和雙面傳輸。
  - NCCL\_DEBUG=INFO - 啟用詳細的除錯輸出。您也可以在測試開始時指定 VERSION 來列印只有 NCCL 版本，或 WARN 只接收錯誤訊息。

如需更多關於 NCCL 測試引數的詳細資訊，請參閱官方 nccl-test 存放庫中的 [NCCL Test README](#)。

- p3dn.24xlarge

```
$ /opt/amazon/openmpi/bin/mpirun \
-x LD_LIBRARY_PATH=/opt/nccl/build/lib:/usr/local/cuda/lib64:/opt/amazon/efa/
lib:/opt/amazon/openmpi/lib:/opt/aws-ofi-nccl/lib:$LD_LIBRARY_PATH \
-x NCCL_DEBUG=INFO \
--hostfile my-hosts -n 8 -N 8 \
--mca pml ^cm --mca btl tcp,self --mca btl_tcp_if_exclude lo,docker0 --bind-
to none \
$HOME/nccl-tests/build/all_reduce_perf -b 8 -e 1G -f 2 -g 1 -c 1 -n 100
```

- p4d.24xlarge 和 p5.48xlarge

```
$ /opt/amazon/openmpi/bin/mpirun \
-x FI_EFA_USE_DEVICE_RDMA=1 \
-x LD_LIBRARY_PATH=/opt/nccl/build/lib:/usr/local/cuda/lib64:/opt/amazon/efa/
lib:/opt/amazon/openmpi/lib:/opt/aws-ofi-nccl/lib:$LD_LIBRARY_PATH \
-x NCCL_DEBUG=INFO \
--hostfile my-hosts -n 8 -N 8 \
--mca pml ^cm --mca btl tcp,self --mca btl_tcp_if_exclude lo,docker0 --bind-
to none \
$HOME/nccl-tests/build/all_reduce_perf -b 8 -e 1G -f 2 -g 1 -c 1 -n 100
```

### 3. 列印 NCCL\_DEBUG 記錄時，您可以確認 EFA 是積極作為 NCCL 的基礎提供者。

```
ip-192-168-2-54:14:14 [0] NCCL INFO NET/OFI Selected Provider is efa*
```

使用 p4d.24xlarge 執行個體時會顯示下列其他資訊。

```
ip-192-168-2-54:14:14 [0] NCCL INFO NET/OFI Running on P4d platform, Setting
NCCL_TOPO_FILE environment variable to /home/ec2-user/install/plugin/share/aws-
ofi-nccl/xml/p4d-24x1-topo.xml
```

## 步驟 10：安裝您的機器學習應用程式

在暫時執行個體上安裝機器學習應用程式。安裝程序依特定的機器學習應用程式而異。如需在 Linux 執行個體上安裝軟體的詳細資訊，請參閱[管理 Amazon Linux 2 執行個體上的軟體](#)。

**Note**

參考機器學習應用程式的文件，以取得安裝指示。

**步驟 11：建立啟用 EFA 和 NCCL 的 AMI**

安裝所需的軟體元件之後，您需要建立 AMI，以重複用來啟動具備 EFA 功能的執行個體。

**從暫時執行個體建立 AMI**

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取您建立的暫時執行個體，然後選取 Actions (動作)、Image (映像)、Create image (建立映像)。
4. 對於 Create image (建立映像)，執行下列動作：
  - a. 對於 Image name (映像名稱)，輸入 AMI 的描述性名稱。
  - b. (選用) 對於 Image description (映像描述)，輸入 AMI 的簡短描述。
  - c. 選擇 Create image。
5. 在導覽窗格中，選擇 AMIs (AMI)。
6. 在清單中找出您建立的 AMI。等待狀態從 pending 變為 available，再繼續進行下一個步驟。

**步驟 12：終止暫時執行個體**

此時，您不再需要您啟動的暫時執行個體。您可以終止執行個體，以停止產生費用。

**終止暫時執行個體**

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取您建立的暫時執行個體，然後選取動作、執行個體狀態、終止執行個體。
4. 出現確認提示時，請選擇終止。

## 步驟 13：在集群放置群組中啟動已啟用 EFA 和 NCCL 的執行個體

使用具備 EFA 功能的 AMI 和具備 EFA 功能的安全群組，在集群放置群組中啟動具備 EFA 和 NCCL 功能的執行個體。

### Note

- 在集群放置群組中啟動具備 EFA 的執行個體並非絕對必要。不過，建議在集群放置群組中執行具備 EFA 功能的執行個體，因為這樣會在單一可用區域的低延遲群組中啟動執行個體。
- 要確保在擴展叢集執行個體時容量可用，您可以為集群放置群組建立容量保留。如需詳細資訊，請參閱 [集群放置群組中的容量保留](#)。

## New console

### 啟動暫時執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)，接著選擇 Launch Instances (啟動執行個體) 來開啟新的啟動執行個體精靈。
3. (選用) 在 Name and tags (名稱和標籤) 區段中，為執行個體命名，例如，EFA-instance。該名稱將指派作為執行個體的資源標籤 (Name=*EFA-instance*)。
4. 在 Application and OS Images (應用程式和作業系統映像) 區段中，選取 My AMIs (我的 AMI)，接著選取您在前一個步驟中建立的 AMI。
5. 在執行個體類型 區段中，選取 p3dn.24xlarge 或者 p4d.24xlarge。
6. 在 Key pair (金鑰對) 區段中，選取要用於執行個體的金鑰對。
7. 在 Network settings (網路設定) 區段，選擇 Edit (編輯)，接著執行下列動作：
  - a. 對於 Subnet (子網)，請選擇要在其中啟動執行個體子網。若您未選取子網，便無法啟用 EFA 的執行個體。
  - b. 針對 Firewall (security groups) (防火牆 (安全群組))，選取 Select existing security group (選取現有的安全群組)，接著選取您在前一個步驟中建立的安全群組。
  - c. 展開 Advanced network configuration (進階網路組態) 區段，並針對 Elastic Fabric Adapter，選取 Enable (啟用)。
8. (選用) 在 儲存 區段中，根據需求設定磁碟區。



9. 在進階詳細資訊區段中，針對置放群組名稱，選取要在其中啟動執行個體的集群放置群組。如果您需要建立新的集群放置群組，請選取建立新的配置群組。
10. 在右邊的摘要面板，針對執行個體的數目，輸入要啟動的具備 EFA 功能的執行個體數量，接著選擇啟動執行個體。

## Old console

在集群放置群組中啟動具備 EFA 和 NCCL 功能的執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 選擇 Launch Instance (啟動執行個體)。
3. 在 Choose an AMI (選取 AMI 頁面) 上，選取 My AMIs (我的 AMI)，找出您之前建立的 AMI，然後選取 Select (選取)。
4. 在 Choose an Instance Type (選取執行個體類型) 頁面上，選取 p3dn.24xlarge，然後選取 Next: Configure Instance Details (下一步：設定執行個體詳細資訊)。
5. 在 Configure Instance Details (設定執行個體詳細資訊) 頁面上，執行下列操作：
  - a. 對於 Number of instances (執行個體的數目)，輸入要啟動的具備 EFA 和 NCCL 功能的執行個體數。
  - b. 對於 Network (網路) 和 Subnet (子網)，選取要在其中啟動執行個體的 VPC 和子網。
  - c. 對於 Placement group (置放群組)，選取 Add instance to placement group (將執行個體新增至置放群組)。
  - d. 對於 Placement group name (置放群組名稱)，選取 Add to a new placement group (新增至新的置放群組)，然後輸入置放群組的描述性名稱。然後為 Placement group strategy (置放群組策略) 選取 cluster (叢集)。
  - e. 對於 EFA，選擇 Enable (啟用)。
  - f. 在 Network Interfaces (網路介面) 區段中，針對裝置 eth0，選擇 New network interface (新網路介面)。您可以選擇性指定一個主要 IPv4 地址，以及一或多個次要 IPv4 地址。如果您在有相關聯 IPv6 CIDR 區塊的子網中啟動執行個體，您可以選擇性指定一個主要 IPv6 地址，以及一或多個次要 IPv6 地址。
  - g. 選擇 Next: Add Storage (下一步：新增儲存體)。
6. 在 Add Storage (新增儲存體) 頁面上，除了 AMI 指定的磁碟區 (例如根設備磁碟區)，指定要連接到執行個體的磁碟區。然後選擇 Next: Add Tags (下一步：新增標籤)。
7. 在 Add Tags (新增標籤) 頁面上，為執行個體指定標籤 (例如使用者易記的名稱)，然後選擇 Next: Configure Security Group (下一步：設定安全群組)。

- 在 Configure Security Group (設定安全群組) 頁面上，針對 Assign a security group (指派安全群組)，選取 Select an existing security group (選取現有安全群組)，然後選取您之前建立的安全群組。
- 選擇 Review and Launch (檢閱和啟動)。
- 在 Review Instance Launch (檢閱執行個體啟動) 頁面上，檢閱設定，然後選擇 Launch (啟動)，以選擇金鑰對並啟動執行個體。

## 步驟 14：啟用無密碼 SSH

若要讓您的應用程式能夠跨越叢集中的所有執行個體執行，您必須啟用從領導節點到成員節點的無密碼 SSH 存取。領導節點是您在其中執行應用程式的執行個體。叢集中的其餘執行個體為成員節點。

### 在叢集的執行個體之間啟用無密碼 SSH

- 選取叢集中的一個執行個體作為領導節點，並與它連線。
- 在領導節點上停用 `strictHostKeyChecking` 並啟用 `ForwardAgent`。使用您偏好的文字編輯器開啟 `~/.ssh/config`，然後新增下列命令。

```
Host *
 ForwardAgent yes
Host *
 StrictHostKeyChecking no
```

- 產生 RSA 金鑰對。

```
$ ssh-keygen -t rsa -N "" -f ~/.ssh/id_rsa
```

在 `$HOME/.ssh/` 目錄中建立金鑰對。

- 變更領導節點上私有金鑰的許可。

```
$ chmod 600 ~/.ssh/id_rsa
chmod 600 ~/.ssh/config
```

- 使用您偏好的文字編輯器開啟 `~/.ssh/id_rsa.pub` 並複製該金鑰。
- 對於叢集中的每個成員節點，執行以下操作：
  - 連線到執行個體。

- b. 使用您偏好的文字編輯器開啟 `~/.ssh/authorized_keys`，然後新增您先前複製的公有金鑰。
7. 若要測試無密碼的 SSH 是否如預期運作，請連線至領導節點並執行下列命令。

```
$ ssh member_node_private_ip
```

您應該連線至成員節點，而不會提示您輸入金鑰或密碼。

## 使用 AWS 深度學習 AMI

下列步驟可協助您開始使用下列其中一種 AWS 深度學習 AMI：

- 深度學習 AMI (Amazon Linux 2)
- 深度學習 AMI (Ubuntu 20.04)

如需詳細資訊，請參閱 [《AWS Deep Learning AMI 使用者指南》](#)。

### Note

只支援 p3dn.24xlarge 和 p4d.24xlarge 執行個體類型。

## 內容

- [步驟 1：準備具備 EFA 功能的安全群組](#)
- [步驟 2：啟動暫時執行個體](#)
- [步驟 3：測試您的 EFA 和 NCCL 組態](#)
- [步驟 4：安裝您的機器學習應用程式](#)
- [步驟 5：建立 EFA 和 NCCL 啟用的 AMI](#)
- [步驟 6：終止暫時執行個體](#)
- [步驟 7：在集群放置群組中啟動 EFA 和具備 NCCL 功能的執行個體](#)
- [步驟 8：啟用無密碼 SSH](#)

## 步驟 1：準備具備 EFA 功能的安全群組

EFA 需要有安全群組來允許往返安全群組自身的所有傳入和傳出流量。下列程序會建立一個安全群組，該安全群組允許所有傳入和傳出流量，並允許來自任何 IPv4 地址的傳入 SSH 流量以進行 SSH 連線。

### Important

此安全群組僅供測試之用。對於生產環境，我們建議您建立傳入 SSH 規則，該規則僅允許來自連線 IP 地址的流量，例如電腦的 IP 地址或區域網路中的一系列 IP 地址。

如需其他案例，請參閱[不同使用案例的安全群組規則](#)。

### 建立具備 EFA 功能的安全群組

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Security Groups (安全群組)，然後選擇 Create Security Group (建立安全群組)。
3. 在 Create Security Group (建立安全群組) 視窗中，執行下列動作：
  - a. 對於 Security group name (安全群組名稱)，輸入安全群組的描述性名稱，例如 EFA-enabled security group。
  - b. (選用) 對於 Description (描述)，輸入安全群組的簡短描述。
  - c. 對於 VPC，選取您打算讓具備 EFA 功能的執行個體在其中啟動的 VPC。
  - d. 選擇 Create Security Group (建立安全群組)。
4. 選取您建立的安全群組，在 Details (詳細資料) 索引標籤上，複製 Security group ID (安全群組 ID)。
5. 在仍然選取安全群組的情況下，依序選取 Actions (動作) 和 Edit inbound rules (編輯傳入規則)，然後執行下列動作。
  - a. 選擇新增規則。
  - b. 針對類型，選擇所有流量。
  - c. 若為 Source type (來源類型)，選擇 Custom (自訂)，然後將您複製的安全群組 ID 貼到欄位中。
  - d. 選擇新增規則。
  - e. 針對 Type (類型)，選擇 SSH。

- f. 針對 Source type (來源類別)，選擇 Anywhere-IPv4(隨處 - IPv4)。
  - g. 選擇儲存規則。
6. 在仍然選取安全群組的情況下，依序選取 Actions (動作) 和 Edit outbound rules (編輯傳出規則)，然後執行下列動作。
- a. 選擇新增規則。
  - b. 針對類型，選擇所有流量。
  - c. 若為 Destination type (目的地類型)，選擇 Custom (自訂)，然後將您複製的安全群組 ID 貼到欄位中。
  - d. 選擇儲存規則。

## 步驟 2：啟動暫時執行個體

啟動暫時執行個體，以用來安裝和設定 EFA 軟體元件。您可以使用此執行個體來建立具備 EFA 功能的 AMI，再從中啟動具備 EFA 功能的執行個體。

### 啟動暫時執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)，接著選擇 Launch Instances (啟動執行個體) 來開啟新的啟動執行個體精靈。
3. (選用) 在 Name and tags (名稱和標籤) 區段中，為執行個體命名，例如，EFA-instance。該名稱將指派作為執行個體的資源標籤 (Name=*EFA-instance*)。
4. 在 Application and OS Images (應用程式和作業系統映像) 區段中，選取支援的 AWS Deep Learning AMI Version 25.0 or later (深度學習 AMI 版本 25.0 或更新版本)。
5. 在執行個體類型 區段中，選取 p3dn.24xlarge 或者 p4d.24xlarge。
6. 在 Key pair (金鑰對) 區段中，選取要用於執行個體的金鑰對。
7. 在 Network settings (網路設定) 區段，選擇 Edit (編輯)，接著執行下列動作：
  - a. 對於 Subnet (子網)，請選擇要在其中啟動執行個體子網。若您未選取子網，便無法啟用 EFA 的執行個體。
  - b. 針對 Firewall (security groups) (防火牆 (安全群組))，選取 Select existing security group (選取現有的安全群組)，接著選取您在前一個步驟中建立的安全群組。
  - c. 展開 Advanced network configuration (進階網路組態) 區段，並針對 Elastic Fabric Adapter，選取 Enable (啟用)。

- 在儲存區段中，根據需求設定磁碟區。

**Note**

您必須為 Nvidia CUDA 工具組佈建額外的 10 到 20 GiB 儲存空間。如果您沒有佈建足夠的儲存空間，您會在嘗試安裝 Nvidia 驅動程式和 CUDA 工具組時，收到 `insufficient disk space` 錯誤。

- 在右邊的 Summary (摘要) 面板中，選擇 Launch instance (啟動執行個體)。

### 步驟 3：測試您的 EFA 和 NCCL 組態

執行測試以確保已對 EFA 和 NCCL 正確設定暫時執行個體。

若要測試您的 EFA 和 NCCL 組態

- 建立主機檔案指定要執行測試的主機。下列命令另建立名為 `my-hosts` 的主機檔案，包含執行個體本身的參考資料。

#### IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://169.254.169.254/latest/meta-data/local-ipv4 >> my-hosts
```

#### IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/local-ipv4 >> my-hosts
```

- 執行測試並指定主機檔案 (`--hostfile`) 以及要使用的 GPU 數量 (`-n`)。下列命令在執行個體本身為 8 GPU，和下列指定的環境變數上執行 `all_reduce_perf` 測試。
  - `FI_EFA_USE_DEVICE_RDMA=1`—(僅限 `p4d.24xlarge`) 使用裝置的 RDMA 功能進行單面和雙面傳輸。
  - `NCCL_DEBUG=INFO` - 啟用詳細的除錯輸出。您也可以在測試開始時指定 `VERSION` 來列印只有 NCCL 版本，或 `WARN` 只接收錯誤訊息。

如需更多關於 NCCL 測試引數的詳細資訊，請參閱官方 `nccl-test` 存放庫中的 [NCCL Test README](#)。

- `p3dn.24xlarge`

```
$ /opt/amazon/openmpi/bin/mpirun \
-x LD_LIBRARY_PATH=/opt/nccl/build/lib:/usr/local/cuda/lib64:/opt/amazon/efa/
lib:/opt/amazon/openmpi/lib:/opt/aws-ofi-nccl/lib:$LD_LIBRARY_PATH \
-x NCCL_DEBUG=INFO \
--hostfile my-hosts -n 8 -N 8 \
--mca pml ^cm --mca btl tcp,self --mca btl_tcp_if_exclude lo,docker0 --bind-
to none \
$HOME/nccl-tests/build/all_reduce_perf -b 8 -e 1G -f 2 -g 1 -c 1 -n 100
```

- `p4d.24xlarge`

```
$ /opt/amazon/openmpi/bin/mpirun \
-x FI_EFA_USE_DEVICE_RDMA=1 \
-x LD_LIBRARY_PATH=/opt/nccl/build/lib:/usr/local/cuda/lib64:/opt/amazon/efa/
lib:/opt/amazon/openmpi/lib:/opt/aws-ofi-nccl/lib:$LD_LIBRARY_PATH \
-x NCCL_DEBUG=INFO \
--hostfile my-hosts -n 8 -N 8 \
--mca pml ^cm --mca btl tcp,self --mca btl_tcp_if_exclude lo,docker0 --bind-
to none \
$HOME/nccl-tests/build/all_reduce_perf -b 8 -e 1G -f 2 -g 1 -c 1 -n 100
```

3. 列印 `NCCL_DEBUG` 記錄時，您可以確認 EFA 是積極作為 NCCL 的基礎提供者。

```
ip-192-168-2-54:14:14 [0] NCCL INFO NET/OFI Selected Provider is efa*
```

使用 `p4d.24xlarge` 執行個體時會顯示下列其他資訊。

```
ip-192-168-2-54:14:14 [0] NCCL INFO NET/OFI Running on P4d platform, Setting
NCCL_TOPO_FILE environment variable to /home/ec2-user/install/plugin/share/aws-
ofi-nccl/xml/p4d-24x1-topo.xml
```

## 步驟 4：安裝您的機器學習應用程式

在暫時執行個體上安裝機器學習應用程式。安裝程序依特定的機器學習應用程式而異。如需在 Linux 執行個體上安裝軟體的詳細資訊，請參閱[管理 Amazon Linux 2 執行個體上的軟體](#)。

### Note

參考機器學習應用程式的文件，以取得安裝指示。

## 步驟 5：建立 EFA 和 NCCL 啟用的 AMI

安裝所需的軟體元件之後，您需要建立 AMI，以重複用來啟動具備 EFA 功能的執行個體。

### 從暫時執行個體建立 AMI

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取您建立的暫時執行個體，然後選取 Actions (動作)、Image (映像)、Create image (建立映像)。
4. 對於 Create image (建立映像)，執行下列動作：
  - a. 對於 Image name (映像名稱)，輸入 AMI 的描述性名稱。
  - b. (選用) 對於 Image description (映像描述)，輸入 AMI 的簡短描述。
  - c. 選擇 Create image。
5. 在導覽窗格中，選擇 AMIs (AMI)。
6. 在清單中找出您建立的 AMI。等待狀態從 pending 變為 available，再繼續進行下一個步驟。

## 步驟 6：終止暫時執行個體

此時，您不再需要您啟動的暫時執行個體。您可以終止執行個體，以停止產生費用。

### 終止暫時執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取您建立的暫時執行個體，然後選取動作、執行個體狀態、終止執行個體。
4. 出現確認提示時，請選擇終止。



## 步驟 7：在集群放置群組中啟動 EFA 和具備 NCCL 功能的執行個體

使用具備 EFA 功能的 AMI 和具備 EFA 功能的安全群組，在集群放置群組中啟動具備 EFA 和 NCCL 功能的執行個體。

### Note

- 在集群放置群組中啟動具備 EFA 的執行個體並非絕對必要。不過，建議在集群放置群組中執行具備 EFA 功能的執行個體，因為這樣會在單一可用區域的低延遲群組中啟動執行個體。
- 要確保在擴展叢集執行個體時容量可用，您可以為集群放置群組建立容量保留。如需詳細資訊，請參閱 [集群放置群組中的容量保留](#)。

## New console

### 啟動暫時執行個體

- 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
- 在導覽窗格中，選擇 Instances (執行個體)，接著選擇 Launch Instances (啟動執行個體) 來開啟新的啟動執行個體精靈。
- (選用) 在 Name and tags (名稱和標籤) 區段中，為執行個體命名，例如，EFA-instance。該名稱將指派作為執行個體的資源標籤 (Name=*EFA-instance*)。
- 在 Application and OS Images (應用程式和作業系統映像) 區段中，選取 My AMIs (我的 AMI)，接著選取您在前一個步驟中建立的 AMI。
- 在執行個體類型 區段中，選取 p3dn.24xlarge 或者 p4d.24xlarge。
- 在 Key pair (金鑰對) 區段中，選取要用於執行個體的金鑰對。
- 在 Network settings (網路設定) 區段，選擇 Edit (編輯)，接著執行下列動作：
  - 對於 Subnet (子網)，請選擇要在其中啟動執行個體子網。若您未選取子網，便無法啟用 EFA 的執行個體。
  - 針對 Firewall (security groups) (防火牆 (安全群組))，選取 Select existing security group (選取現有的安全群組)，接著選取您在前一個步驟中建立的安全群組。
  - 展開 Advanced network configuration (進階網路組態) 區段，並針對 Elastic Fabric Adapter，選取 Enable (啟用)。
- (選用) 在 儲存 區段中，根據需求設定磁碟區。

9. 在進階詳細資訊 區段中，針對 置放群組名稱，選取要在其中啟動執行個體的集群放置群組。如果您需要建立新的集群放置群組，請選取 建立新的配置群組。
10. 在右邊的 摘要 面板，針對 執行個體的數目，輸入要啟動的具備 EFA 功能的執行個體數量，接著選擇 啟動執行個體。

## Old console

在集群放置群組中啟動具備 EFA 和 NCCL 功能的執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 選擇 Launch Instance (啟動執行個體)。
3. 在 Choose an AMI (選取 AMI 頁面) 上，選取 My AMIs (我的 AMI)，找出您之前建立的 AMI，然後選取 Select (選取)。
4. 在 Choose an Instance Type (選取執行個體類型) 頁面上，選取 p3dn.24xlarge，然後選取 Next: Configure Instance Details (下一步：設定執行個體詳細資訊)。
5. 在 Configure Instance Details (設定執行個體詳細資訊) 頁面上，執行下列操作：
  - a. 對於 Number of instances (執行個體的數目)，輸入要啟動的具備 EFA 和 NCCL 功能的執行個體數。
  - b. 對於 Network (網路) 和 Subnet (子網)，選取要在其中啟動執行個體的 VPC 和子網。
  - c. 對於 Placement group (置放群組)，選取 Add instance to placement group (將執行個體新增至置放群組)。
  - d. 對於 Placement group name (置放群組名稱)，選取 Add to a new placement group (新增至新的置放群組)，然後輸入置放群組的描述性名稱。然後為 Placement group strategy (置放群組策略) 選取 cluster (叢集)。
  - e. 對於 EFA，選擇 Enable (啟用)。
  - f. 在 Network Interfaces (網路介面) 區段中，針對裝置 eth0，選擇 New network interface (新網路介面)。您可以選擇性指定一個主要 IPv4 地址，以及一或多個次要 IPv4 地址。如果您在有相關聯 IPv6 CIDR 區塊的子網中啟動執行個體，您可以選擇性指定一個主要 IPv6 地址，以及一或多個次要 IPv6 地址。
  - g. 選擇 Next: Add Storage (下一步：新增儲存體)。
6. 在 Add Storage (新增儲存體) 頁面上，除了 AMI 指定的磁碟區 (例如根設備磁碟區)，指定要連接到執行個體的磁碟區。然後選擇 Next: Add Tags (下一步：新增標籤)。
7. 在 Add Tags (新增標籤) 頁面上，為執行個體指定標籤 (例如使用者易記的名稱)，然後選擇 Next: Configure Security Group (下一步：設定安全群組)。

8. 在 Configure Security Group (設定安全群組) 頁面上，針對 Assign a security group (指派安全群組)，選取 Select an existing security group (選取現有安全群組)，然後選取您之前建立的安全群組。
9. 選擇 Review and Launch (檢閱和啟動)。
10. 在 Review Instance Launch (檢閱執行個體啟動) 頁面上，檢閱設定，然後選擇 Launch (啟動)，以選擇金鑰對並啟動執行個體。

## 步驟 8：啟用無密碼 SSH

若要讓您的應用程式能夠跨越叢集中的所有執行個體執行，您必須啟用從領導節點到成員節點的無密碼 SSH 存取。領導節點是您在其中執行應用程式的執行個體。叢集中的其餘執行個體為成員節點。

### 在叢集的執行個體之間啟用無密碼 SSH

1. 選取叢集中的一個執行個體作為領導節點，並與它連線。
2. 在領導節點上停用 `strictHostKeyChecking` 並啟用 `ForwardAgent`。使用您偏好的文字編輯器開啟 `~/.ssh/config`，然後新增下列命令。

```
Host *
 ForwardAgent yes
Host *
 StrictHostKeyChecking no
```

3. 產生 RSA 金鑰對。

```
$ ssh-keygen -t rsa -N "" -f ~/.ssh/id_rsa
```

在 `$HOME/.ssh/` 目錄中建立金鑰對。

4. 變更領導節點上私有金鑰的許可。

```
$ chmod 600 ~/.ssh/id_rsa
chmod 600 ~/.ssh/config
```

5. 使用您偏好的文字編輯器開啟 `~/.ssh/id_rsa.pub` 並複製該金鑰。
6. 對於叢集中的每個成員節點，執行以下操作：
  - a. 連線到執行個體。

- b. 使用您偏好的文字編輯器開啟 `~/.ssh/authorized_keys`，然後新增您先前複製的公有金鑰。
7. 若要測試無密碼的 SSH 是否如預期運作，請連線至領導節點並執行下列命令。

```
$ ssh member_node_private_ip
```

您應該連線至成員節點，而不會提示您輸入金鑰或密碼。

## 使用 EFA

您可以以非常類似 Amazon EC2 中的其他任何彈性網絡介面的方式建立、使用和管理 EFA。不過，與彈性網絡介面不同，EFAs 不能與處於執行中狀態的執行個體連接或分離。

### EFA 要求

若要使用 EFA，您必須執行下列動作：

- 選擇其中一個[支援的執行個體類型](#)。
- 對其中一個[支援的作業系統](#)使用 AMI。
- 安裝 EFA 軟體元件。如需詳細資訊，請參閱 [步驟 3：安裝 EFA 軟體](#) 和 [步驟 5：\(選用\) 安裝 Intel MPI](#)。
- 使用安全群組，而該安全群組允許往返於其本身的所有傳入和傳出流量。如需詳細資訊，請參閱 [步驟 1：準備具備 EFA 功能的安全群組](#)。

### 內容

- [建立 EFA](#)
- [將 EFA 連接至已停止的執行個體](#)
- [啟動執行個體時連接 EFA](#)
- [將 EFA 新增至啟動範本](#)
- [管理 EFA 的 IP 地址](#)
- [變更 EFA 的安全群組](#)
- [分離 EFA](#)
- [檢視 EFAs](#)

- [刪除 EFA](#)

## 建立 EFA

您可以在 VPC 的子網中建立 EFA。EFA 建立後即無法移至另一個子網，而且只能連接到同一可用區域中已停止的執行個體。

使用主控台建立新的 EFA 資料表

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選擇 Create Network Interface (建立網路介面)。
4. 針對 Description (描述)，輸入 EFA 的描述性名稱。
5. 針對 Subnet (子網)，選取要在其中建立 EFA 的子網。
6. 針對 Private IP (私有 IP)，輸入主要私有 IPv4 地址。如果您不指定 IPv4 地址，我們會從選取子網中選取可用的私有 IPv4 地址。
7. (限 IPv6) 如果已選取有關聯 IPv6 CIDR 區塊的子網，您可在 IPv6 IP 欄位中選取性指定 IPv6 地址。
8. 針對 Security groups (安全群組)，選取一個或多個安全群組。
9. 對於 EFA，選擇 Enable (啟用)。
10. 選擇 Yes, Create (是，建立)。

若要使用建立新的 EFA AWS CLI

使用 [create-network-interface](#) 命令，並對於 interface-type，指定 efa，如下列範例所示。

```
aws ec2 create-network-interface --subnet-id subnet-01234567890 --
description example_efa --interface-type efa
```

## 將 EFA 連接至已停止的執行個體

您可以將 EFA 連接至任何處於 stopped 狀態的受支援執行個體。您無法將 EFA 連接至處於 running 狀態的執行個體。如需支援的執行個體類型的詳細資訊，請參閱[支援的執行個體類型](#)。

就像將網路介面連接至執行個體，您以同樣的方式將 EFA 連接至執行個體。如需詳細資訊，請參閱[將網路介面連接到執行個體](#)。

## 啟動執行個體時連接 EFA

### 啟動執行個體時連接現有的 EFA (AWS CLI)

使用 [執行例項](#) 指令，對於 NetworkInterfaceId，指定 EFA 的識別碼，如下列範例所示。

```
aws ec2 run-instances --image-id ami_id --count 1 --instance-type c5n.18xlarge --key-name my_key_pair --network-interfaces DeviceIndex=0,NetworkInterfaceId=efa_id,Groups=sg_id,SubnetId=subnet_id
```

### 啟動執行個體時連接新的 EFA (AWS CLI)

使用 [run-instance](#) 指令，並指定 InterfaceType，如下列範例所示。

```
aws ec2 run-instances --image-id ami_id --count 1 --instance-type c5n.18xlarge --key-name my_key_pair --network-interfaces DeviceIndex=0,InterfaceType=efa,Groups=sg_id,SubnetId=subnet_id
```

## 將 EFA 新增至啟動範本

您可以建立啟動範本，其中包含啟動具備 EFA 功能的執行個體所需的組態資訊。若要建立具備 EFA 功能的啟動範本，請建立新的啟動範本，並指定支援的執行個體類型、具備 EFA 功能的 AMI，以及具備 EFA 功能的安全群組。如需詳細資訊，請參閱 [開始使用 EFA 和 MPI](#)。

您可以利用啟動範本，啟動具備 EFA 功能的執行個體及其他 AWS 服務，例如 [AWS Batch](#) 或 [AWS ParallelCluster](#)。

如需建立啟動範本的詳細資訊，請參閱 [建立啟動範本](#)。

## 管理 EFA 的 IP 地址

您可以變更與 EFA 相關聯的 IP 地址。如果您有彈性 IP 地址，您可以將它與 EFA 建立關聯。如果您的 EFA 佈建在有相關聯 IPv6 CIDR 區塊的子網中，您可以將一或多個 IPv6 地址指派給 EFA。

將彈性 IP (IPv4) 和 IPv6 地址指派給 EFA 的方式，就像將 IP 地址指派給彈性網絡介面一樣。如需詳細資訊，請參閱 [管理 IP 地址](#)。

## 變更 EFA 的安全群組

您可以變更與 EFA 相關聯的安全群組。若要啟用 OS-bypass 功能，EFA 必須是安全群組的成員，而該安全群組允許往返於其本身的所有傳入和傳出流量。

就像變更與彈性網絡介面相關聯的安全群組，您以同樣的方式變更與 EFA 相關聯的安全群組。如需詳細資訊，請參閱[變更安全群組](#)。

## 分離 EFA

若要從執行個體分離 EFA，您必須先停止執行個體。您無法從處於執行中狀態的執行個體分離 EFA。

就像從執行個體分離彈性網絡介面，您以同樣的方式從執行個體分離 EFA。如需詳細資訊，請參閱[從執行個體分離網路介面](#)。

## 檢視 EFAs

您可以檢視您帳戶中的所有 EFAs。

就像檢視彈性網絡介面一樣，您以同樣的方式檢視 EFAs。如需詳細資訊，請參閱[檢視網路介面的詳細資訊](#)。

## 刪除 EFA

若要刪除 EFA，您必須先從執行個體分離它。您無法刪除仍連接至執行個體的 EFA。

就像刪除彈性網絡介面一樣，您以同樣的方式刪除 EFAs。如需詳細資訊，請參閱[刪除網路介面](#)。

## 監控 EFA

您可以使用下列功能來監控 Elastic Fabric Adapter 的效能。

### Amazon VPC 流程日誌

您可以建立 Amazon VPC 流程日誌，以擷取進出 EFA 的流量的相關資訊。流程日誌資料可以發佈到 Amazon CloudWatch 日誌和 Amazon S3。建立流量日誌之後，您可以在選擇的目的地中擷取及檢視其資料。如需詳細資訊，請參閱《Amazon VPC 使用者指南》<https://docs.aws.amazon.com/vpc/latest/userguide/flow-logs.html>中的 VPC 流程日誌。

就像為彈性網絡介面建立流程日誌，您以同樣的方式為 EFA 建立流程日誌。如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的[建立流量日誌](#)。

在流程日誌項目中，EFA 流量是以 `srcAddress` 和 `destAddress` 來識別，兩者的格式都是 MAC 地址，如下列範例所示。

version	accountId	eniId	srcAddress	destAddress	sourcePort	destPort
protocol	packets	bytes	start	end	action	log-status
2	3794735123	eni-10000001	01:23:45:67:89:ab	05:23:45:67:89:ab	-	-
-	9	5689	1521232534	1524512343	ACCEPT	OK

## Amazon CloudWatch

Amazon CloudWatch 提供的指標可讓您即時監控您的 EFA。您可以收集和追蹤指標、建立自訂儀板表，以及設定警示，在特定指標達到您指定的閾值時通知您或採取動作。如需詳細資訊，請參閱 [使用監控執行個體 CloudWatch](#)。

## 使用檢查總和驗證 EFA 安裝程式

您可以選擇使用 MD5 或 SHA256 檢查總和來驗證 EFA tarball (.tar.gz 檔案)。我們建議您執行這項操作來確認軟體發行者的身分識別，並檢查應用程式自發行以來並未遭到變更或損毀。

### 驗證 tarball 的步驟

使用 md5sum 公用程式做為 MD5 檢查總和，或使用 Sha256sum 公用程式做為 SHA256 檢查總和，並指定 tarball 檔案名稱。您必須從儲存 tarball 檔案的目錄執行命令。

- MD5

```
$ md5sum tarball_filename.tar.gz
```

- SHA256

```
$ sha256sum tarball_filename.tar.gz
```

這些命令應該會傳回下列格式的檢查總和值。

```
checksum_value tarball_filename.tar.gz
```

將命令傳回的檢查總和值與下表所提供的檢查總和值進行比較。如果檢查總和相符，則可以安全地執行安裝指令碼。如果檢查總和不相符，請勿執行安裝指令碼，然後聯絡 AWS Support。

例如，下列指令會使用 SHA256 檢查總和來驗證 EFA 1.9.4 tarball。

```
$ sha256sum aws-efa-installer-1.9.4.tar.gz
```



```
1009b5182693490d908ef0ed2c1dd4f813cc310a5d2062ce9619c4c12b5a7f14 aws-efa-
installer-1.9.4.tar.gz
```

下表列出 EFA 最新版本的檢查總和。

版本	下載 URL	檢查總和
EFA	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.32.0.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.32.0.tar.gz</a>	MD5: db8d65cc028d8d08b5 a9f2d88881c1b1  SHA256 : 5f7233760be57f6fee 6de8c09acbfbf59238 de848e06048dc54d15 6ef578fc66
美国国家法	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.31.0.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.31.0.tar.gz</a>	MD5: 856352f12bef2ccbad cd75e35aa52aaf  SHA256 : 943325bd37902a4300 ac9e5715163537d56e cb4e7b87b37827c3e5 47aa1897bf
EFA 1.30.0	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.30.0.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.30.0.tar.gz</a>	MD5: 31f48e1a47fe93ede8 ebd273fb747358  SHA256 : 876ab9403e07a0c3c9 1a1a34685a52eced89 0ae052df94857f6081 c5f6c78a0a
EFA 1.29.1	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.29.1.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.29.1.tar.gz</a>	MD5: e1872ca815d752c1d7 c2b5c175e52a16  SHA256 : 178b263b8c25845b63 dc93b25bcdff5870df 5204ec509af26f43e8 d283488744

版本	下載 URL	檢查總和
EFA 1.29.0	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.29.0.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.29.0.tar.gz</a>	MD5: 39d06a002154d94cd9 82ed348133f385  SHA256 : 836655f87015547e73 3e7d9f7c760e4e2469 7f8bbc261bb5f3560a bd4206bc36
EFA 1.28.0	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.28.0.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.28.0.tar.gz</a>	MD5: 9dc13b744666582260 5e66febe074035  SHA256 : 2e625d2d6d3e073b51 78e8e861891273d896 b66d03cb1a32244fd5 6789f1c435
EFA 1.27.0	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.27.0.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.27.0.tar.gz</a>	MD5: 98bfb515ea3e8d93f5 54020f3837fa15  SHA256 : 1d49a97b0bf8d964d9 1652a79ac851f2550e 33a5bf9d0cf86ec935 7ff6579aa3
EFA 1.26.1	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.26.1.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.26.1.tar.gz</a>	MD5: 884e74671fdef47255 01f7cd2d451d0c  SHA256 : c616994c924f54ebfa bfab32b7fe8ac56947 fae00a0ff453d975e2 98d174fc96

版本	下載 URL	檢查總和
EFA 1.26.0	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.26.0.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.26.0.tar.gz</a>	MD5: f8839f12ff2e3b9ba0 9ae8a82b30e663  SHA256 : bc1abc1f76e97d204d 3755d2a9ca307fc423 e51c63141f798c2f15 be3715aa11
EFA 1.25.1	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.25.1.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.25.1.tar.gz</a>	MD5: 6d876b894547847a45 bb8854d4431f18  SHA256 : d2abc553d22b89a4ce 92882052c1fa6de450 d3a801fe005da718b7 d4b9602b06
EFA 1.25.0	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.25.0.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.25.0.tar.gz</a>	MD5: 1993836ca749596051 da04694ea0d00c  SHA256 : 98b7b26ce031a2d6a9 3de2297cc71b03af64 7194866369ca53b60d 82d45ad342
EFA 1.24.1	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.24.1.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.24.1.tar.gz</a>	MD5: 211b249f39d53086f3 cb0c07665f4e6f  SHA256 : 120cfeec233af09556 23ac7133b674143329 f9561a9a8193e47306 0f596aec62

版本	下載 URL	檢查總和
EFA 1.24.0	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.24.0.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.24.0.tar.gz</a>	MD5: 7afe0187951e2dd2c9 cc4b572e62f924  SHA256 : 878623f819a0d9099d 76ecd41cf4f569d4c3 aac0c9bb7ba9536347 c50b6bf88e
EFA 1.23.1	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.23.1.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.23.1.tar.gz</a>	MD5: 22491e114b6ee7160a 8290145dca0c28  SHA256 : 5ca848d8e0ff4d1571 cd443c36f8d27c8cdf 2a0c97e9068ebf000c 303fc40797
EFA 1.23.0	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.23.0.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.23.0.tar.gz</a>	MD5: 38a6d7c1861f5038db a4e441ca7683ca  SHA256 : 555d497a60f22e3857 fdeb3dfc53aa86d059 26023c68c916d15d2d c3df6525bd
EFA 1.22.1	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.22.1.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.22.1.tar.gz</a>	MD5: 600c0ad7cdbc06e8e8 46cb763f92901b  SHA256 : f90f3d5f59c031b9a9 64466b5401e86fd042 9272408f6c207c3f90 48254e9665

版本	下載 URL	檢查總和
EFA 1.22.0	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.22.0.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.22.0.tar.gz</a>	MD5: 8f100c93dc8ab519c2 aeb5dab89e98f8  SHA256 : f329e7d54a86a03ea5 1da6ea9a5b68fb354f bae4a57a02f9592e21 fce431dc3a
EFA 1.21.0	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.21.0.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.21.0.tar.gz</a>	MD5: 959ccc3a4347461909 ec02ed3ba7c372  SHA256 : c64e6ca34ccfc3ebe8 e82d08899ae8442b3e f552541cf5429c43d1 1a04333050
EFA 1.20.0	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.20.0.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.20.0.tar.gz</a>	MD5: 7ebfbb8e85f1b94709 df4ab3db47913b  SHA256 : aeefd2681ffd5c4c63 1d1502867db5b83162 1d6eb85b61fe3ec80d f983d1dcf0
EFA 1.19.0	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.19.0.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.19.0.tar.gz</a>	MD5: 2fd45324953347ec55 18da7e3fefa0ec  SHA256 : 99b77821b9e72c8dea 015cc92c96193e8db3 07deee05b91a58094c c331f16709

版本	下載 URL	檢查總和
EFA 1.18.0	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.18.0.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.18.0.tar.gz</a>	MD5: fc2571a72f5d3c7b7b 576ce2de38d91e  SHA256 : acb18a0808aedb9a5e 485f1469225b9ac97f 21db9af78e4cd69397 00debe1cb6
EFA 1.17.3	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.17.3.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.17.3.tar.gz</a>	MD5: 0517df4a190356ab55 9235147174cafd  SHA256 : 5130998b0d2883bbae 189b21ab215ecbc1b0 1ae0231659a9b4a17b 0a33ebc6ca
EFA 1.17.2	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.17.2.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.17.2.tar.gz</a>	MD5: a329dedab53c4832df 218a24449f4c9a  SHA256 : bca1fdde8b32b00346 e175e597ffab32a09a 08ee9ab136875fb382 83cc4cd099
EFA 1.17.1	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.17.1.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.17.1.tar.gz</a>	MD5: 733ae2cfc9d14b5201 7eaf0a2ab6b0ff  SHA256 : f29322640a88ae9279 805993cb836276ea24 0623820848463ca686 c8ce02136f

版本	下載 URL	檢查總和
EFA 1.17.0	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.17.0.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.17.0.tar.gz</a>	MD5: d430fc841563c11c38 05c5f82a4746b1  SHA256 : 75ab0cee4fb6bd3888 9dce313183f5d3a83b d233e0a6ef6205d835 2821ea901d
EFA 1.16.0	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.16.0.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.16.0.tar.gz</a>	MD5: 399548d3b0d2e812d7 4dd67937b696b4  SHA256 : cecec36495a1bc6fdc 82f97761a541e4fb6c 9a3cbf3cfc145acf2 5ea5dbd45b
EFA 1.15.2	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.15.2.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.15.2.tar.gz</a>	MD5: 955fea580d5170b058 23d51acde7ca21  SHA256 : 84df4fbc1b3741b6c0 73176287789a601a58 9313accc8e6653434e 8d4c20bd49
EFA 1.15.1	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.15.1.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.15.1.tar.gz</a>	MD5: c4610267039f72bbe4 e35d7bf53519bc  SHA256 : be871781a1b9a15fca 342a9d169219260069 942a8bda7a8ad06d4b aeb5e2efd7

版本	下載 URL	檢查總和
EFA 1.15.0	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.15.0.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.15.0.tar.gz</a>	MD5: 9861694e1cc00d884f adac07d22898be  SHA256 : b329862dd5729d2d09 8d0507fb486bf859d7 c70ce18b61c3029822 34a3a5c88f
EFA 1.14.1	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.14.1.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.14.1.tar.gz</a>	MD5: 50ba56397d359e5787 2fde1f74d4168a  SHA256 : c7b1b48e86fe4b3eaa 4299d3600930919c4f e6d88cc6e2c7e4a408 a3f16452c7
EFA 1.14.0	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.14.0.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.14.0.tar.gz</a>	MD5: 40805e7fd842c36ece cb9fd7f921b1ae  SHA256 : 662d62c12de85116df 33780d40e0533ef7da d92709f4f613907475 a7a1b60a97
EFA 1.13.0	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.13.0.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.13.0.tar.gz</a>	MD5: c91d16556f4fd53bec adbb345828221e  SHA256 : ad6705eb23a3fce44a f3afc0f76430915956 53a723ad0374084f4f 2b715192e1



版本	下載 URL	檢查總和
EFA 1.12.3	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.12.3.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.12.3.tar.gz</a>	MD5: 818aee81f097918cfa ebd724eddea678  SHA256 : 2c225321824788b8ca 3fbc118207b944cdb0 96b847e1e0d1d853ef 2f0d727172
EFA 1.12.2	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.12.2.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.12.2.tar.gz</a>	MD5: 956bb1fc5ae0d6f0f8 7d2e481d49fccf  SHA256: 083a868a2c212a5a4f cf3e4d732b685ce39c ceb3ca7e5d50d0b74e 7788d06259
EFA 1.12.1	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.12.1.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.12.1.tar.gz</a>	MD5: f5bfe52779df435188 b0a2874d0633ea  SHA256: 5665795c2b4f09d5f3 f767506d4d4c429695 b36d4a17e5758b27f0 33aee58900
EFA 1.12.0	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.12.0.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.12.0.tar.gz</a>	MD5: d6c6b49fafb39b7702 97e1cc44fe68a6  SHA256: 28256c57e9ecc0b077 8b41c1f777a9982b4e 8eae782343dfe12460 79933dca59

版本	下載 URL	檢查總和
EFA 1.11.2	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.11.2.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.11.2.tar.gz</a>	MD5 : 2376cf18d1353a4551 e35c33d269c404  SHA256: a25786f98a3628f7f5 4f7f74ee2b39bc6734 ea9374720507d37d3e 8bf8ee1371
EFA 1.11.1	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.11.1.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.11.1.tar.gz</a>	MD5: 026b0d9a0a48780cc7 406bd51997b1c0  SHA256: 6cb04baf5ffc58ddf3 19e956b5461289199c 8dd805fe216f8f9ab8 d102f6d02a
EFA 1.11.0	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.11.0.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.11.0.tar.gz</a>	MD5: 7d9058e010ad65bf2e 14259214a36949  SHA256: 7891f6d45ae33e8221 89511c4ea1d14c9d54 d000f6696f97be54e9 15ce2c9dfa
EFA 1.10.1	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.10.1.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.10.1.tar.gz</a>	MD5: 78521d3d668be22976 f46c6fecc7b730  SHA256: 61564582de7320b21d e319f532c3a677d26c c46785378eb3b95c63 6506b9bcb4

版本	下載 URL	檢查總和
EFA 1.10.0	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.10.0.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.10.0.tar.gz</a>	MD5 : 46f73f5a7afe41b4bb 918c81888fef9a9  SHA256: 136612f96f2a085a7d 98296da0afb6fa807b 38142e2fc0c548fa98 6c41186282
EFA 1.9.5	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.9.5.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.9.5.tar.gz</a>	MD5: 95edb8a209c18ba8d2 50409846eb6ef4  SHA256: a4343308d7ea4dc943 ccc21bcebed913e886 8e59bfb2ac93599c61 a7c87d7d25
EFA 1.9.4	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.9.4.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.9.4.tar.gz</a>	MD5: f26dd5c350422c1a98 5e35947fa5aa28  SHA256: 1009b5182693490d90 8ef0ed2c1dd4f813cc 310a5d2062ce9619c4 c12b5a7f14
EFA 1.9.3	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.9.3.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.9.3.tar.gz</a>	MD5: 95755765a097802d3e 6d5018d1a5d3d6  SHA256: 46ce732d6f3fcc9edf 6a6e9f9df0ad136054 328e24675567f7029e dab90c68f1

版本	下載 URL	檢查總和
EFA 1.8.4	<a href="https://efa-installer.amazonaws.com/aws-efa-installer-1.8.4.tar.gz">https://efa-installer.amazonaws.com/ aws-efa-installer-1.8.4.tar.gz</a>	MD5: 85d594c41e831afc6c 9305263140457e  SHA256 : 0d974655a09b213d78 59e658965e56dc4f23 a0eee2dc44bb41b6d0 39cc5bab45

## Amazon EC2 執行個體拓撲

說明您的執行個體拓撲可提供執行個體之間相對鄰近度的階層式檢視。您可以使用這些資訊來管理大規模的高效能運算 (HPC) 和機器學習 (ML) 運算基礎架構，同時最佳化工作安排。HPC 和 ML 任務對延遲和輸送量很敏感。您可以使用執行個體拓撲來偵測執行個體的位置，然後使用這項資訊，在實際上彼此較近的執行個體上執行 HPC 和 ML 工作，以最佳化 HPC 和 ML 工作。

您可以使用執行個體拓撲來偵測現有執行個體的位置，但無法使用它來選擇實際啟動靠近現有執行個體的新執行個體。若要影響例證放置，您可以使用[集群放置群組中的容量保留](#)。

### 定價

說明您的執行個體拓撲不需額外費用。

### 目錄

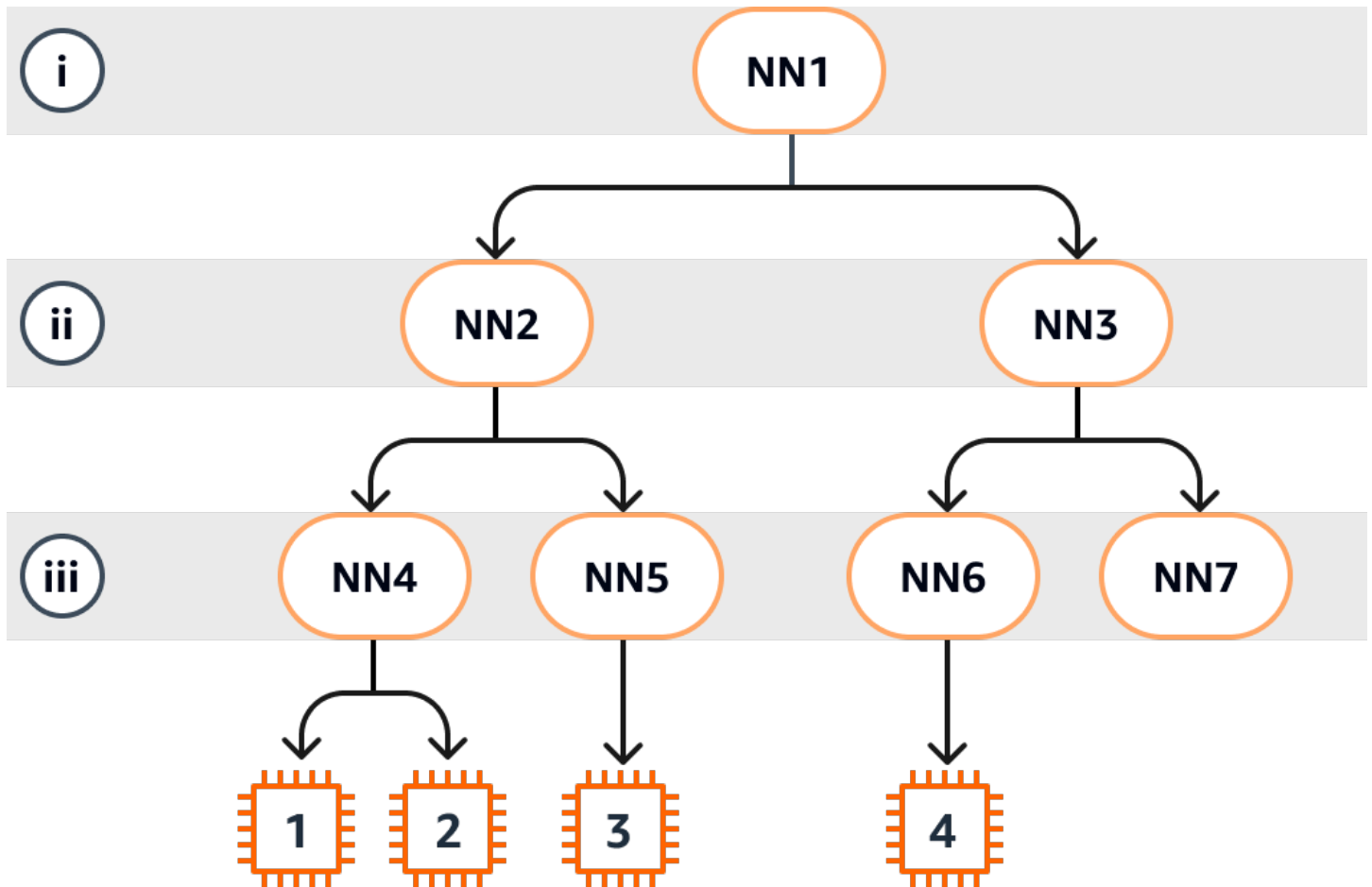
- [例項拓撲的運作方式](#)
- [執行個體拓撲的必要](#)
- [Amazon EC2 實例拓撲示例](#)

## 例項拓撲的運作方式

每個 EC2 執行個體都連接到一個節點集。節點集由三個網路節點組成，每個節點代表 AWS 網路中的不同圖層。網路層是以 3 層或更多層的階層進行排列。節點集合提供此階層的由上而下檢視，其底層與執行個體最接近。

有關節點集的資訊稱為執行個體拓撲。

下圖提供可用來瞭解執行個體拓撲的視覺化表示法。網路節點會被識別為 NN1 — NN7。數字 i、ii 和 iii 可識別網路層。數字 1、2、3 和 4 可識別 EC2 執行個體。例證連接到底層中的節點，由 iii 標識。可有多個執行個體連接到同一個節點。



在此範例中：

- 執行個體 1 會連線至第三層中的網路節點 4 (NN4)。NN4 連接到第 ii 層中的網路節點 2 (NN2)，NN2 連接到第 i 層中的網路節點 1 (NN1)，也就是此範例中的網路階層頂端。網路節點集由 NN1、NN2 和 NN4 組成，以上層到底層的階層表示。
- 執行個體 2 也連接至網路節點 4 (NN4)。執行個體 1 和執行個體 2 有共同的網路節點集：NN1、NN2 和 NN4。
- 執行個體 3 連接至網路節點 5 (NN5)。NN5 連接到 NN2，而 NN2 連接到 NN1。執行個體 3 的網路節點集為 NN1、NN2 和 NN5。
- 執行個體 4 連接至網路節點 6 (NN6)。其網路節點集為 NN1、NN3 和 NN6。

考量執行個體 1、2 和 3 的鄰近性時，執行個體 1 和 2 彼此更接近，因為兩者連接到相同的網路節點 (NN4)，而執行個體 3 連接至不同的網路節點 (NN5) 因此較遠。

考量此圖中所有執行個體的鄰近性時，比起與執行個體 4 之間的距離，執行個體 1、2 和 3 彼此的距離更接近，因為這幾個執行個體的網路節點集中都有 NN2。

一般而言，如果連接到任何兩個執行個體的網路節點相同，這兩個執行個體實際位置彼此相近，就像執行個體 1 和 2 的情況一樣。此外，網路節點之間的躍點數越少，執行個體彼此之間的距離就越接近。例如，執行個體 1 和 3 到共同網路節點 (NN2) 的躍點，少於與執行個體 4 的共同網路節點 (NN1) 之間的躍點，因此它們之間的距離比執行個體 4 更近。

在此範例中，沒有在網路節點 7 (NN7) 下執行的執行個體，因此 API 輸出不會包含 NN7。

## 如何解釋輸出

您可以使用拓撲 API 取得執行個體 [DescribeInstance 拓撲](#) 資訊。輸出會提供執行個體之基礎網路拓撲的階層式檢視。

下列輸出範例對應上圖中四個執行個體的網路拓撲資訊。為了說明本範例，輸出範例中會包含註解。

輸出中的以下資訊非常重要：

- NetworkNodes 說明執行個體的網路節點集。
- 在每個網路節點集中，網路節點會以由上至下的階層順序列出。
- 連接至執行個體的網路節點是清單中的最後一個網路節點 (底層)。
- 若要找出哪些執行個體彼此靠近，請先在底層中尋找一般網路節點。如果底層沒有共同網路節點，請往上層尋找共同網路節點。

在下列範例中，輸出 `i-1111111111example` 和 `i-2222222222example` 位於與此範例中的其他執行個體相較之間最接近的位置，因為它們在底層 `nn-4444444444example` 中具有共同的網路節點。

```
{
 "Instances": [
 {
 "InstanceId": "i-1111111111example", //Corresponds to instance 1
 "InstanceType": "p4d.24xlarge",
 "GroupName": "ML-group",
 "NetworkNodes": [
 "nn-1111111111example", //Corresponds to NN1 in layer i
 "nn-2222222222example", //Corresponds to NN2 in layer ii
]
 }
]
}
```

```

 "nn-4444444444example" //Corresponds to NN4 in layer iii -
bottom layer, connected to the instance
],
 "ZoneId": "usw2-az2",
 "AvailabilityZone": "us-west-2a"
},
{
 "InstanceId": "i-2222222222example", //Corresponds to instance 2
 "InstanceType": "p4d.24xlarge",
 "NetworkNodes": [
 "nn-1111111111example", //Corresponds to NN1 - layer i
 "nn-2222222222example", //Corresponds to NN2 - layer ii
 "nn-4444444444example" //Corresponds to NN4 - layer iii -
connected to instance
],
 "ZoneId": "usw2-az2",
 "AvailabilityZone": "us-west-2a"
},
{
 "InstanceId": "i-3333333333example", //Corresponds to instance 3
 "InstanceType": "trn1.32xlarge",
 "NetworkNodes": [
 "nn-1111111111example", //Corresponds to NN1 - layer i
 "nn-2222222222example", //Corresponds to NN2 - layer ii
 "nn-5555555555example" //Corresponds to NN5 - layer iii -
connected to instance
],
 "ZoneId": "usw2-az2",
 "AvailabilityZone": "us-west-2a"
},
{
 "InstanceId": "i-4444444444example", //Corresponds to instance 4
 "InstanceType": "trn1.2xlarge",
 "NetworkNodes": [
 "nn-1111111111example", //Corresponds to NN1 - layer i
 "nn-3333333333example", //Corresponds to NN3 - layer ii
 "nn-6666666666example" //Corresponds to NN6 - layer iii -
connected to instance
],
 "ZoneId": "usw2-az2",
 "AvailabilityZone": "us-west-2a"
}
],
"NextToken": "SomeEncryptedToken"

```

```
}
```

## 限制

有下列限制：

- 執行個體必須處於running狀態。
- 每個執行個體拓撲檢視對每個帳戶都是唯一的。
- AWS Management Console 不支援檢視例證拓撲。

## 執行個體拓撲的必要

在說明執行個體的執行個體拓撲之前，請確定您的執行個體符合下列需求。

說明執行個體拓撲的需求

- [AWS 區域](#)
- [執行個體類型](#)
- [執行個體狀態](#)
- [IAM 許可](#)

## AWS 區域

支援 AWS 區域：

- 美國東部 (維吉尼亞北部)、美國東部 (俄亥俄)、美國西部 (加利佛尼亞北部)、美國西部 (奧勒岡)
- 亞太區域 (首爾)、亞太區域 (東京)
- 加拿大 (中部)
- 歐洲 (法蘭克福)、歐洲 (愛爾蘭)、歐洲 (斯德哥爾摩)

## 執行個體類型

支援的執行個體類型：

- hpc6a.48xlarge | hpc6id.32xlarge | hpc7a.12xlarge | hpc7a.24xlarge  
| hpc7a.48xlarge | hpc7a.96xlarge | hpc7g.4xlarge | hpc7g.8xlarge |  
hpc7g.16xlarge



- p3dn.24xlarge | p4d.24xlarge | p4de.24xlarge | p5.48xlarge
- trn1.2xlarge | trn1.32xlarge | trn1n.32xlarge

### 查看特定區域的可用執行個體類型

可用的執行個體類型因區域而異。若要查看「區域」中是否有執行個體類型可用，請將指[describe-instance-types-offerings](#) 令與 `--region` 參數搭配使用。包含 `--filters` 參數可將結果範圍限定為您想使用的執行個體系列或執行個體類型，包含 `--query` 參數則可將輸出範圍限定為 `InstanceType` 的值。

```
aws ec2 describe-instance-type-offerings \
 --region us-east-2 \
 --filters 'Name=instance-type, Values=trn1*' \
 --query 'InstanceTypeOfferings[].InstanceType'
```

### 預期的輸出結果

```
[
 "trn1.2xlarge",
 "trn1.32xlarge",
 "trn1n.32xlarge"
]
```

### 執行個體狀態

執行個體必須處於 `running` 狀態。若執行個體處於其他狀態，您無法取得其執行個體拓撲資訊。

### IAM 許可

您的 IAM 身分 (使用者、使用者群組或角色) 需要下列 IAM 權限：

- `ec2:DescribeInstanceTopology`

## Amazon EC2 實例拓撲示例

您可以使用 [describe-instance-topology](#) CLI 命令描述 EC2 執行個體的執行個體拓撲。

若不搭配參數或篩選條件使用 `describe-instance-topology` 命令，回應會包含指定區域中符合此命令所支援執行個體類型的所有執行個體。您可以藉由包含 `--region` 參數或設定預設區域來指定區域。如需設定預設區域的詳細資訊，請參閱：[指定資源的區域](#)。

您可以加入參數以傳回符合指定執行個體 ID 或放置群組名稱的執行個體。您也可以加入篩選條件，以傳回符合指定執行個體類型或執行個體系列的執行個體，或是指定可用區域或本地區域中的執行個體。您可以包含單一參數或篩選條件，或是參數和篩選條件的組合。

系統會將輸出會分頁，預設每頁最多 20 個執行個體。您可以使用 `--max-results` 參數指定每頁最多 100 個執行個體。

如需詳細資訊，請參閱 AWS CLI 命令參考中的 [describe-instance-topology](#)。

所需的許可

描述執行個體拓撲需要下列權限：

- `ec2:DescribeInstanceTopology`

範例

- [範例 1：無參數或篩選條件](#)
- [範例 2：instance-type 篩選條件](#)
  - [範例 2a：指定執行個體類型的完全相符篩選條件](#)
  - [範例 2b：執行個體系列的萬用字元篩選條件](#)
  - [範例 2c：結合執行個體系列和完全相符篩選條件](#)
- [範例 3：zone-id 篩選條件](#)
  - [範例 3a：可用區域篩選條件](#)
  - [範例 3b：本地區域篩選條件](#)
  - [範例 3c：結合可用區域與本地區域篩選條件](#)
- [範例 4：結合 instance-type 和 zone-id 篩選條件](#)
- [範例 5：放置群組名稱參數](#)
- [範例 6：執行個體 ID](#)

範例 1：無參數或篩選條件

說明所有執行個體的執行個體拓撲

使用 [describe-instance-topology](#) CLI 命令而不指定任何參數或篩選條件。

```
aws ec2 describe-instance-topology --region us-west-2
```

回應只會傳回符合此 API 所支援執行個體類型的執行個體。執行個體可以位於不同可用區域、本地區域 (ZoneId) 和放置群組 (GroupName)。如果執行個體不在放置群組中，GroupName 欄位就不會出現在輸出中。在下列輸出範例中，只有一個執行個體位於放置群組中。

### 範例輸出

```
{
 "Instances": [
 {
 "InstanceId": "i-1111111111example",
 "InstanceType": "p4d.24xlarge",
 "GroupName": "my-m1-cpg",
 "NetworkNodes": [
 "nn-1111111111example",
 "nn-2222222222example",
 "nn-3333333333example"
],
 "ZoneId": "usw2-az2",
 "AvailabilityZone": "us-west-2a"
 },
 {
 "InstanceId": "i-2222222222example",
 "InstanceType": "p4d.24xlarge",
 "NetworkNodes": [
 "nn-1111111111example",
 "nn-2222222222example",
 "nn-3333333333example"
],
 "ZoneId": "usw2-az2",
 "AvailabilityZone": "us-west-2a"
 },
 {
 "InstanceId": "i-3333333333example",
 "InstanceType": "trn1.32xlarge",
 "NetworkNodes": [
 "nn-1212121212example",
 "nn-1211122211example",
 "nn-1311133311example"
],
 "ZoneId": "usw2-az4",
 "AvailabilityZone": "us-west-2d"
 },
 {
```

```

 "InstanceId": "i-44444444example",
 "InstanceType": "trn1.2xlarge",
 "NetworkNodes": [
 "nn-11111111example",
 "nn-5434334334example",
 "nn-1235301234example"
],
 "ZoneId": "usw2-az2",
 "AvailabilityZone": "us-west-2a"
 }
],
"NextToken": "SomeEncryptedToken"
}

```

## 範例 2 : instance-type 篩選條件

您可以依指定的執行個體類型進行篩選 (完全相符)，或依執行個體系列進行篩選 (使用萬用字元)。您也可以結合指定的執行個體類型篩選器和執行個體族群篩選。

### 範例 2a : 指定執行個體類型的完全相符篩選條件

說明符合指定執行個體類型的所有執行個體之執行個體拓撲

使用 [describe-instance-topology](#) CLI 命令搭配 instance-type 篩選條件。在此範例中，系統會針對 trn1n.32xlarge 執行個體篩選輸出。回應只會傳回符合指定執行個體類型的執行個體。

```

aws ec2 describe-instance-topology \
 --region us-west-2 \
 --filters Name=instance-type,Values=trn1n.32xlarge

```

### 範例輸出

```

{
 "Instances": [
 {
 "InstanceId": "i-22222222example",
 "InstanceType": "trn1n.32xlarge",
 "NetworkNodes": [
 "nn-11111111example",
 "nn-22222222example",
 "nn-33333333example"
]
 }
]
}

```

```

],
 "ZoneId": "usw2-az2",
 "AvailabilityZone": "us-west-2a"
 }
],
"NextToken": "SomeEncryptedToken"
}

```

## 範例 2b：執行個體系列的萬用字元篩選條件

說明符合特定執行個體系列的所有執行個體之執行個體拓撲

使用 [describe-instance-topology](#) CLI 命令搭配 `instance-type` 篩選條件。在此範例中，系統會針對 `trn1*` 執行個體篩選輸出。回應只會傳回符合指定執行個體系列的執行個體。

```

aws ec2 describe-instance-topology \
 --region us-west-2 \
 --filters Name=instance-type,Values=trn1*

```

## 範例輸出

```

{
 "Instances": [
 {
 "InstanceId": "i-222222222example",
 "InstanceType": "trn1n.32xlarge",
 "NetworkNodes": [
 "nn-111111111example",
 "nn-222222222example",
 "nn-333333333example"
],
 "ZoneId": "usw2-az2",
 "AvailabilityZone": "us-west-2a"
 },
 {
 "InstanceId": "i-333333333example",
 "InstanceType": "trn1.32xlarge",
 "NetworkNodes": [
 "nn-121212121example",
 "nn-1211122211example",
 "nn-1311133311example"
],
 },
],
}

```

```

 "ZoneId": "usw2-az4",
 "AvailabilityZone": "us-west-2d"
 },
 {
 "InstanceId": "i-444444444example",
 "InstanceType": "trn1.2xlarge",
 "NetworkNodes": [
 "nn-111111111example",
 "nn-5434334334example",
 "nn-1235301234example"
],
 "ZoneId": "usw2-az2",
 "AvailabilityZone": "us-west-2a"
 }
],
"NextToken": "SomeEncryptedToken"
}

```

### 範例 2c：結合執行個體系列和完全相符篩選條件

說明符合特定執行個體系列或指定執行個體類型的所有執行個體之執行個體拓撲

使用 [describe-instance-topology](#) CLI 命令搭配 `instance-type` 篩選條件。在此範例中，系統會針對 `pd4d*` 或 `trn1n.32xlarge` 執行個體篩選輸出。回應只會傳回符合任一個篩選條件的執行個體。

```

aws ec2 describe-instance-topology \
 --region us-west-2 \
 --filters "Name=instance-type,Values=p4d*,trn1n.32xlarge"

```

### 範例輸出

```

{
 "Instances": [
 {
 "InstanceId": "i-111111111example",
 "InstanceType": "p4d.24xlarge",
 "GroupName": "ML-group",
 "NetworkNodes": [
 "nn-111111111example",
 "nn-222222222example",
 "nn-333333333example"
],
 },
],
}

```

```

 "ZoneId": "usw2-az2",
 "AvailabilityZone": "us-west-2a"
 },
 {
 "InstanceId": "i-2222222222example",
 "InstanceType": "trn1n.32xlarge",
 "NetworkNodes": [
 "nn-1111111111example",
 "nn-2222222222example",
 "nn-4343434343example"
],
 "ZoneId": "usw2-az2",
 "AvailabilityZone": "us-west-2a"
 }
],
"NextToken": "SomeEncryptedToken"
}

```

### 範例 3 : zone-id 篩選條件

您可以使用 `zone-id` 篩選條件來依可用區域或本地區域進行篩選。您也可以結合可用區域篩選條件和本地區域篩選條件。

#### 範例 3a : 可用區域篩選條件

說明符合指定可用區域的所有執行個體之執行個體拓撲

使用 [describe-instance-topology](#) CLI 命令搭配 `zone-id` 篩選條件。在此範例中，輸出會使用可用區域 ID 篩選 `use1-az1`。回應只會傳回符合指定可用區域的執行個體。

```

aws ec2 describe-instance-topology \
 --region us-east-1 \
 --filters Name=zone-id,Values=use1-az1

```

#### 範例輸出

```

{
 "Instances": [
 {
 "InstanceId": "i-2222222222example",
 "InstanceType": "trn1n.32xlarge",

```

```

 "NetworkNodes": [
 "nn-1111111111example",
 "nn-2222222222example",
 "nn-3214313214example"
],
 "ZoneId": "use1-az1",
 "AvailabilityZone": "us-east-1a"
 }
],
"NextToken": "SomeEncryptedToken"
}

```

### 範例 3b：本地區域篩選條件

說明符合指定本地區域的所有執行個體之執行個體拓撲

使用 [describe-instance-topology](#) CLI 命令搭配 `zone-id` 篩選條件。在此範例中，會使用本機區域 ID 篩選輸出 `use1-atl2-az1`。回應只會傳回符合指定本地區域的執行個體。

```

aws ec2 describe-instance-topology \
 --region us-east-1 \
 --filters Name=zone-id,Values=use1-atl2-az1

```

### 範例輸出

```

{
 "Instances": [
 {
 "InstanceId": "i-1111111111example",
 "InstanceType": "p4d.24xlarge",
 "GroupName": "ML-group",
 "NetworkNodes": [
 "nn-1111111111example",
 "nn-2222222222example",
 "nn-3333333333example"
],
 "ZoneId": "use1-atl2-az1",
 "AvailabilityZone": "us-east-1-atl-2a"
 }
],
 "NextToken": "SomeEncryptedToken"
}

```



### 範例 3c：結合可用區域與本地區域篩選條件

說明符合指定可用區域或本地區域的所有執行個體之執行個體拓撲

使用 [describe-instance-topology](#) CLI 命令搭配 `zone-id` 篩選條件。在此範例中，輸出會使用可用區域 ID `use1-az1` 和本機區域 ID 進行篩選 `use1-atl2-az1`。回應只會傳回符合任一個篩選條件的執行個體。

```
aws ec2 describe-instance-topology \
 --region us-east-1 \
 --filters Name=zone-id,Values=use1-az1,use1-atl2-az1
```

### 範例輸出

```
{
 "Instances": [
 {
 "InstanceId": "i-1111111111example",
 "InstanceType": "p4d.24xlarge",
 "GroupName": "ML-group",
 "NetworkNodes": [
 "nn-1111111111example",
 "nn-2222222222example",
 "nn-3333333333example"
],
 "ZoneId": "use1-atl2-az1",
 "AvailabilityZone": "us-east-1-atl-2a"
 },
 {
 "InstanceId": "i-2222222222example",
 "InstanceType": "trn1n.32xlarge",
 "NetworkNodes": [
 "nn-1111111111example",
 "nn-2222222222example",
 "nn-3214313214example"
],
 "ZoneId": "use1-az1",
 "AvailabilityZone": "us-east-1a"
 }
],
 "NextToken": "SomeEncryptedToken"
}
```

## 範例 4：結合 instance-type 和 zone-id 篩選條件

您可將所有篩選條件結合為單一命令。

說明符合指定執行個體類型、執行個體系列、可用區域或本地區域的所有執行個體之執行個體拓撲

使用 [describe-instance-topology](#) CLI 命令搭配 instance-type 和 zone-id 篩選條件。在此範例中，系統會針對執行個體 p4d\* 系列、trn1n.32xlarge 執行個體類型、use1-az1 可用區域 ID 和 use1-atl2-az1 本地區域 ID 篩選輸出。回應會傳回符合 us-east-1a 或 us-east-1-atl-2a 區域中 p4d\* 或 trn1n.32xlarge 執行個體的執行個體。

```
aws ec2 describe-instance-topology \
 --region us-east-1 \
 --filters "Name=instance-type,Values=p4d*,trn1n.32xlarge" "Name=zone-
id,Values=use1-az1,use1-atl2-az1"
```

### 範例輸出

```
{
 "Instances": [
 {
 "InstanceId": "i-1111111111example",
 "InstanceType": "p4d.24xlarge",
 "GroupName": "ML-group",
 "NetworkNodes": [
 "nn-1111111111example",
 "nn-2222222222example",
 "nn-3333333333example"
],
 "ZoneId": "use1-atl2-az1",
 "AvailabilityZone": "us-east-1-atl-2a"
 },
 {
 "InstanceId": "i-2222222222example",
 "InstanceType": "trn1n.32xlarge",
 "NetworkNodes": [
 "nn-1111111111example",
 "nn-2222222222example",
 "nn-3214313214example"
],
 "ZoneId": "use1-az1",
 "AvailabilityZone": "us-east-1a"
 }
]
}
```

```
],
 "NextToken": "SomeEncryptedToken"
}
```

## 範例 5：放置群組名稱參數

描述指定放置群組中所有執行個體的執行個體拓撲

使用 [describe-instance-topology](#) CLI 命令搭配 `group-names` 參數。在下列範例中，執行個體可以位於 `ML-group` 或 `HPC-group` 置群組中。回應會傳回位於其中一個放置群組中的執行個體。

```
aws ec2 describe-instance-topology \
 --region us-west-2 \
 --group-names ML-group HPC-group
```

## 範例輸出

```
{
 "Instances": [
 {
 "InstanceId": "i-1111111111example",
 "InstanceType": "p4d.24xlarge",
 "GroupName": "ML-group",
 "NetworkNodes": [
 "nn-1111111111example",
 "nn-2222222222example",
 "nn-3333333333example"
],
 "ZoneId": "usw2-az2",
 "AvailabilityZone": "us-west-2a"
 },
 {
 "InstanceId": "i-2222222222example",
 "InstanceType": "trn1n.32xlarge",
 "GroupName": "HPC-group",
 "NetworkNodes": [
 "nn-1111111111example",
 "nn-2222222222example",
 "nn-3214313214example"
],
 "ZoneId": "usw2-az2",
 "AvailabilityZone": "us-west-2a"
 }
]
}
```

```
],
 "NextToken": "SomeEncryptedToken"
}
```

## 範例 6：執行個體 ID

說明指定執行個體的執行個體拓撲

使用 [describe-instance-topology](#) CLI 命令搭配 `--instance-ids` 參數。回應會傳回符合指定執行個體 ID 的執行個體。

```
aws ec2 describe-instance-topology \
 --region us-west-2 \
 --instance-ids i-1111111111example i-2222222222example
```

## 範例輸出

```
{
 "Instances": [
 {
 "InstanceId": "i-1111111111example",
 "InstanceType": "p4d.24xlarge",
 "GroupName": "ML-group",
 "NetworkNodes": [
 "nn-1111111111example",
 "nn-2222222222example",
 "nn-3333333333example"
],
 "ZoneId": "usw2-az2",
 "AvailabilityZone": "us-west-2a"
 },
 {
 "InstanceId": "i-2222222222example",
 "InstanceType": "trn1n.32xlarge",
 "GroupName": "HPC-group",
 "NetworkNodes": [
 "nn-1111111111example",
 "nn-2222222222example",
 "nn-3214313214example"
],
 "ZoneId": "usw2-az2",
 "AvailabilityZone": "us-west-2a"
 }
]
}
```

```
],
 "NextToken": "SomeEncryptedToken"
}
```

## 置放群組

為了符合您的工作負載需求，您可以將一組相互依存的 EC2 執行個體啟動到置放群組中以影響其置放。

視工作負載類型而定，您可以採取下列其中一個置放策略來建立置放群組：

- **叢集**：會將執行個體緊密地包裝在一個可用區域內。此策略可讓工作負載達到高效能運算 (HPC) 應用程式典型之緊密結合 node-to-node 通訊所需的低延遲網路效能。
- **分區**：會將執行個體分散到邏輯分區，使一個分區中的執行個體群組不會與不同分區中的執行個體群組共用底層硬體。大量分散和複寫的工作負載 (例如 Hadoop、Cassandra 和 Kafka) 通常採取此策略。
- **分散**：嚴格地將一小組執行個體分散到不同的底層硬體，以減少相互關聯的故障。

置放群組是選用的程序。如果您未將執行個體啟動至置放群組中，EC2 會嘗試以將所有執行個體分散於底層硬體上的方式置放執行個體，使相互關聯的故障減到最少。

建立置放群組無需收費。

## 置放策略

您可以使用下列其中一種置放策略來建立置放群組。

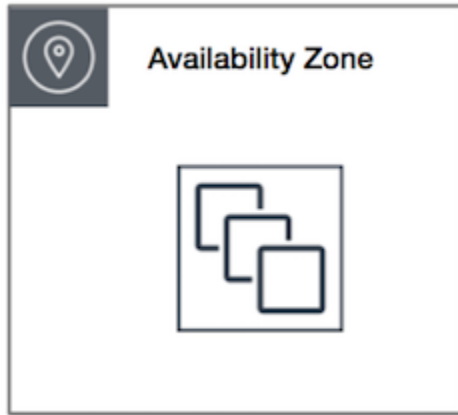
置放策略：

- [集群放置群組](#)
- [分區置放群組](#)
- [分散放置群組](#)

## 集群放置群組

集群放置群組是單一可用區域中的執行個體邏輯分組。集群放置群組可以跨越相同區域中互連的虛擬私有網路 (VPC)。相同集群放置群組中的執行個體可享受 TCP/IP 流量更高的每個流程傳輸量限制，並且放置在相同的網路高等分頻寬區段中。

下列映像會顯示放入集群放置群組中的執行個體。



能夠受惠於低網路延遲、高網路傳輸量或這兩者的應用，建議採用集群放置群組。當大部分的網路流量都在群組中的執行個體之間時，也建議使用這些群組。若要為置放群組提供最低延遲和最高的 packet-per-second 網路效能，請選擇支援增強型聯網的執行個體類型。如需詳細資訊，請參閱[增強型聯網](#)。

我們建議您以下列方式啟動執行個體：

- 使用單一啟動要求，在置放群組中啟動您所需的執行個體數目。
- 對置放群組中的所有執行個體使用相同的執行個體類型。

如果之後您試著在置放群組中加入更多執行個體，或是在置放群組中啟動超過一種執行個體類型，將會提高出現容量不足錯誤的機率。

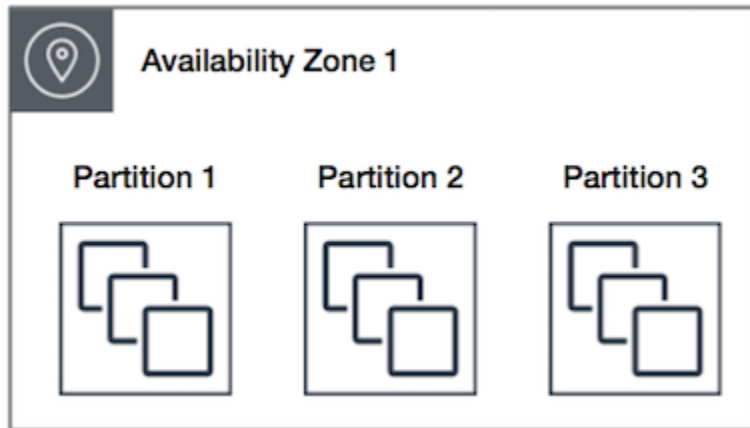
如果先停止置放群組中的執行個體，然後再啟動，此執行個體仍會在置放群組中執行。不過，如果執行個體可用的容量不足，起始作業將會失敗。

若置放群組中已經擁有正在運作的執行個體，當您在此置放群組中啟動執行個體時，如果收到容量錯誤，請停止和啟動該置放群組中的所有執行個體，然後試著再次啟動。啟動執行個體的動作，可能會讓這些執行個體轉移到容量足夠的硬體，能夠提供容量給所有請求的執行個體使用。

## 分區置放群組

分區置放群組有助於降低應用程式發生相關硬體故障的可能性。使用分區置放群組時，Amazon EC2 會將每個群組分割成稱為分區的邏輯區段。Amazon EC2 會確保置放群組內的每個分區有其自己的一組機架。每個機架有其自己的網路和電源。一個置放群組內不會有兩個分割區共用相同的機架，這樣可讓您隔離硬體故障在應用程式內造成的影響。

下圖顯示在單一可用區域中分區置放群組的簡單視覺化呈現。其中顯示放入分區置放群組中的執行個體，而該群組具有三個分區 - 分區 1、分區 2 和分區 3。每個分區包含多個執行個體。一個分割區中的執行個體不會與其他分割區中的執行個體共用機架，可讓您將單一硬體故障的影響僅限縮在相關聯的分割區中。



分區置放群組可用於跨不同機架來部署大型分散式和複寫式工作負載，例如 HDFS、HBase 和 Cassandra。將執行個體啟動到分區置放群組時，Amazon EC2 會嘗試在指定的分區數中均勻分佈執行個體。您也可以將執行個體啟動到特定的分區，讓您能夠更有效控制放置執行個體的位置。

分區置放群組的分割區可以在同一區域中的多個可用區域。分區置放群組在每個可用區域最多可有 7 個分區。可啟動到分區置放群組中的執行個體數，僅會受到帳戶限制的制。

此外，分區置放群組可提供對分區的可見性 - 您可以看到執行個體位於哪個分區中。您可以與拓撲感知應用程式 (例如 HDFS、HBase 和 Cassandra) 共用此資訊。這些應用程式會使用此資訊來進行智能資料複寫決策，以提高資料可用性與持久性。

如果開始或啟動分區置放群組中的執行個體，但是沒有足夠的唯一硬體來滿足請求，則請求會失敗。Amazon EC2 會隨著時間釋出更多可用的不同硬體，因此您可以之後再嘗試進行請求。

## 分散置放群組

分散置放群組是一組放在不同硬體上的執行個體。

若應用程式具有應該分開保存的少量重要執行個體，建議使用分散置放群組。透過分散層級置放群組來啟動執行個體，可降低同時發生故障的風險，這種情況可能會在執行個體共用相同設備時發生。分散層級置放群組提供對不同硬體的存取，因此適合混合的執行個體類型或不同時間啟動執行個體。

如果開始或啟動分散置放群組中的執行個體，但是沒有足夠的唯一硬體來滿足請求，則請求會失敗。Amazon EC2 會隨著時間釋出更多可用的不同硬體，因此您可以之後再嘗試進行請求。置放

群組可以跨機架或主機分散放置執行個體。機架層級分攤放置群組可用於 AWS 區域和開啟 AWS Outposts。主機層級分攤放置群組 AWS Outposts 只能搭配使用。

### 機架層級分攤放置群組

下列映像會顯示在單一可用區域中放入分散置放群組中的 7 個執行個體。七個執行個體放在七個不同機架上，而各機架有其自己的網路和電源。



機架層級分攤放置群組可跨越相同區域中的多個可用區域。在一個區域中，每個群組的每個可用區域，機架層級分攤置放群組最多可以有七個執行中執行個體。使用 Outposts，機架級分攤放置群組可以容納與 Outpost 部署中機架相同數量的執行個體。

### 主機層級分散置放群組

主機層級分攤放置群組僅適用於 AWS Outposts。主機分攤層級置放群組可以容納與 Outpost 部署中主機相同數量的執行個體。如需詳細資訊，請參閱 [the section called “放置群組 AWS Outposts”](#)。

## 放置群組的規則與限制

### 主題

- [一般規則與限制](#)
- [集群放置群組的規則與限制](#)
- [分區放置群組的規則與限制](#)
- [分散放置群組的規則與限制](#)

### 一般規則與限制

在使用置放群組之前，請記住下列規則：



- 您可以在每個區域的每個帳戶建立最多 500 個置放群組。
- 為置放群組所指定的名稱，在您該區域的 AWS 帳戶中必須是唯一的。
- 置放群組無法合併。
- 執行個體一次可以在一個置放群組中啟動；執行個體不能涵跨多個置放群組。
- [隨需容量保留](#)和[區域預留執行個體](#)可讓您為可用區域中的 EC2 執行個體預留容量。啟動執行個體時，如果執行個體屬性與隨需容量保留或區域預留執行個體指定的屬性相符，則該執行個體會自動使用預留容量。如果您將例證啟動到置放群組中，也是如此。

如果您打算在叢集置放群組中啟動執行個體，建議您在叢集置放群組中明確保留容量。您可以透過在[指定的叢集置放群組中建立隨需容量保留](#)來執行此操作。請注意，雖然您可以使用隨需容量保留以這種方式保留容量，但區域預留執行個體無法這樣做，因為它們無法在置放群組中明確保留容量。

- 您無法在置放群組中啟動專用執行個體。
- 您無法啟動在置放群組中設定為中斷時停止或休眠的 Spot 執行個體。

## 集群置放群組的規則與限制

集群置放群組適用的規則如下：

- 支援下列執行個體類型：
  - 目前一代執行個體，高載[效能執行個體](#) (例如 T2)、[Mac1 執行個體](#)和[M7I-Flex 執行個體](#)除外。
  - 下列上一代執行個體：A1、C3、C4、I2、M4、R3 和 R4。
- 集群置放群組不能涵跨多個可用區域。
- 在集群置放群組中，兩個執行個體之間傳送資料的最高網路傳輸速度，會受限於這兩個執行個體之中較慢者的速度。針對需要高傳輸量的應用程式，請選擇符合您需求之網路連線能力的執行個體類型。
- 下列規則適用於為增強聯網所啟用的執行個體：
  - 集群置放群組內的執行個體，單一傳輸流的速度最多可達 10 Gbps。不在集群置放群組內的執行個體，單一傳輸流的速度最高可達 5 Gbps。
  - 在同一個區域中，Amazon S3 儲存貯體之間透過公有 IP 地址空間或 VPC 端點的往來流量，可以使用所有可用的執行個體總頻寬。
- 您可以在集群置放群組中啟動多種執行個體類型。不過，這會降低啟動成功所需容量可用的機率。我們建議在集群置放群組中，針對所有執行個體使用相同的執行個體類型。
- 叢集置放群組的網際網路和透過內部部署資源 AWS Direct Connect 連線的網路流量限制為 5 Gbps。

## 分區放置群組的規則與限制

分區置放群組適用的規則如下：

- 分區置放群組在每個可用區域最多支援 7 個分區。您可以在分區置放群組中啟動的執行個體數，僅會受到帳戶限制的制限制。
- 將執行個體啟動到分區置放群組時，Amazon EC2 會嘗試在所有分區中均勻分配執行個體。Amazon EC2 不保證會在所有分區中均勻分配執行個體。
- 具有專用執行個體的分區置放群組最多可有 2 個分區。
- 容量保留不能在分區置放群組中預留容量。

## 分散放置群組的規則與限制

分散置放群組適用的規則如下：

- 機架分散置放群組支援每個可用區域最多可有七個執行中的執行個體。例如，在具有三個可用區域的區域中，您在該群組共可執行 21 個執行個體 (每個可用區域七個)。如果您嘗試在相同可用區域和在相同的分散置放群組中啟動第 8 個執行個體，則該執行個體不會啟動。如果您在一個可用區域中需要七個以上的執行個體，建議使用多重分散置放群組。使用多重分散置放群組並不能保證群組之間的執行個體分散，但可有助於確保每個群組的分散，從而限制某些類型失敗的影響。
- 專用執行個體不支援分散置放群組。
- 主機層級分攤置放群組僅支援在上的置放群組 AWS Outposts。主機層級分攤置放群組可以容納與 Outpost 部署中主機相同數量的執行個體。
- 在一個區域中，每個群組的每個可用區域，機架層級分攤置放群組最多可以有七個執行中執行個體。使用 AWS Outposts 機架層級分攤置放群組可容納與 Outpost 部署中機架相同數量的執行個體。
- 容量保留不能在分散置放群組中預留容量。

## 使用置放群組

目錄

- [建立置放群組](#)
- [檢視置放群組資訊](#)
- [標記置放群組](#)
- [在置放群組中啟動執行個體](#)

- [描述置放群組中的執行個體](#)
- [變更執行個體的置放群組](#)
- [從置放群組中移除執行個體](#)
- [刪除置放群組](#)

## 建立置放群組

您可以使用下列其中一種方法來建立置放群組。

### Console

使用主控台來建立置放群組

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Placement Groups (置放群組)。
3. 選擇建立置放群組。
4. 指定群組的名稱。
5. 選擇群組的置放策略。
  - 如果選擇 Spread (分散)，請選擇分散層級。
    - 機架：無限制
    - 主機：僅適用於 Outposts
  - 如果您選擇 Partition (分區)，請選擇群組內的分區數目。
6. 若要標記置放群組，請選擇 Add tag (新增標籤)，然後輸入金鑰和值。選擇 Add tag (新增標籤) 來逐個新增標籤。
7. 選擇 Create group (建立群組)。

### AWS CLI

若要使用建立置放群組 AWS CLI

使用 [create-placement-group](#) 命令。下列範例會建立一個名為 my-cluster 的置放群組，它使用 cluster 置放策略，並且套用一個標籤以及 purpose 的索引鍵和 production 的數值。

```
aws ec2 create-placement-group \
```

```
--group-name my-cluster \
--strategy cluster \
--tag-specifications 'ResourceType=placement-
group,Tags={Key=purpose,Value=production}'
```

## 使用建立分割區放置群組 AWS CLI

使用 [create-placement-group](#) 命令。以 `--strategy` 值指定 `partition` 參數，並以所需的分區數目指定 `--partition-count` 參數。在此範例中，分區置放群組名為 `HDFS-Group-A`，且使用五個分區建立。

```
aws ec2 create-placement-group \
 --group-name HDFS-Group-A \
 --strategy partition \
 --partition-count 5
```

## PowerShell

若要使用建立放置群組 AWS Tools for Windows PowerShell

使用 [New-EC2PlacementGroup](#) 命令。

## 檢視放置群組資訊

您可以使用下列其中一種方法來檢視所有放置群組及其相關資訊。

### Console

若要檢視一或多個置放群組的相關資訊

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在功能窗格的 [網路與安全性] 底下，選擇 [置放群組]。
3. 在「放置群組」表格中，您可以針對每個放置群組檢視下列資訊：
  - 群組名稱 — 您指定給置放群組的名稱。
  - 群組 ID — 置入群組的 ID。
  - 策略-放置群組的放置策略。
  - 狀態-放置群組的狀態。
  - 分割區 — 分割區的數目。僅當策略為分割時才有效。

- 群組 ARN — 放置群組的 Amazon 資源名稱 (ARN)。

## AWS CLI

說明您所有的置放群組

使用[描述放置](#)- AWS CLI 群組指令。

```
aws ec2 describe-placement-groups
```

回應範例

```
{
 "PlacementGroups": [
 {
 "GroupName": "my-cluster-pg",
 "State": "available",
 "Strategy": "cluster",
 "GroupId": "pg-0123456789example",
 "GroupArn": "arn:aws:ec2:eu-west-1:111111111111:placement-group/my-
cluster-pg"
 },
 ...
]
}
```

描述指定的放置群組

使用[描述放置](#)- AWS CLI 群組指令。您可以指定 `--group-id` 或 `--group-name` 參數。

指定放置群組 ID：

```
aws ec2 describe-placement-groups --group-id pg-0123456789example
```

指定放置群組名稱：

```
aws ec2 describe-placement-groups --group-name my-cluster-pg
```

回應範例

```
{
```

```
"PlacementGroups": [
 {
 "GroupName": "my-cluster-pg",
 "State": "available",
 "Strategy": "cluster",
 "GroupId": "pg-0123456789example",
 "GroupArn": "arn:aws:ec2:eu-west-1:111111111111:placement-group/my-
cluster-pg"
 }
]
```

## 標記放置群組

若要協助分類和管理現有的置放群組，您可以使用自訂中繼資料標記它們。如需有關標籤運作方式的詳細資訊，請參閱[標記您的 Amazon EC2 資源](#)。

當您為置放群組加上標籤時，啟動到置放群組的執行個體不會自動加上標籤。您需要明確標記啟動到置放群組中的執行個體。如需詳細資訊，請參閱[在您啟動執行個體時新增標籤](#)。

您可以使用下列其中一種方法來檢視、新增及刪除標籤。

### Console

#### 檢視、新增或刪除現有置放群組標籤的步驟

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Placement Groups (置放群組)。
3. 選取置放群組，然後選取 Actions (動作)、Manage tags (管理標籤)。
4. 管理標籤畫面會顯示指派給置放群組的所有標籤。
  - 若要加入標籤，請選擇 Add tag (新增標籤)，然後輸入標籤索引鍵和值。每個置放群組最多可以新增 50 個標籤。如需詳細資訊，請參閱 [標籤限制](#)。
  - 若要移除標籤，請在您要移除的標籤旁邊選擇 Remove (移除)。
5. 選擇儲存。

### AWS CLI

#### 檢視置放群組的標籤

使用 [describe-tags](#) 命令檢視指定資源的標籤。在下列範例中，您將描述所有置放群組的標籤。

```
aws ec2 describe-tags \
 --filters Name=resource-type,Values=placement-group
```

```
{
 "Tags": [
 {
 "Key": "Environment",
 "ResourceId": "pg-0123456789EXAMPLE",
 "ResourceType": "placement-group",
 "Value": "Production"
 },
 {
 "Key": "Environment",
 "ResourceId": "pg-9876543210EXAMPLE",
 "ResourceType": "placement-group",
 "Value": "Production"
 }
]
}
```

您也可以使用 [describe-tags](#) 命令，透過指定置放群組的 ID 來檢視置放群組的標籤。在下列範例中將描述 pg-0123456789EXAMPLE 的標籤。

```
aws ec2 describe-tags \
 --filters Name=resource-id,Values=pg-0123456789EXAMPLE
```

```
{
 "Tags": [
 {
 "Key": "Environment",
 "ResourceId": "pg-0123456789EXAMPLE",
 "ResourceType": "placement-group",
 "Value": "Production"
 }
]
}
```

您也可以描述置放群組來檢視置放群組的標籤。

使用 [describe-placement-group](#) 命令來檢視指定置放群組的組態，其中包括為置放群組指定的任何標籤。

```
aws ec2 describe-placement-groups \
 --group-name my-cluster
```

```
{
 "PlacementGroups": [
 {
 "GroupName": "my-cluster",
 "State": "available",
 "Strategy": "cluster",
 "GroupId": "pg-0123456789EXAMPLE",
 "Tags": [
 {
 "Key": "Environment",
 "Value": "Production"
 }
]
 }
]
}
```

使用為既有置放群組貼標籤 AWS CLI

使用 [create-tags](#) 命令來標記現有資源。在下列範例中，會使用 Key=Cost-Center 和 Value=CC-123 來標記現有置放群組。

```
aws ec2 create-tags \
 --resources pg-0123456789EXAMPLE \
 --tags Key=Cost-Center,Value=CC-123
```

若要使用刪除置放群組中的標籤 AWS CLI

您可以使用 [delete-tags](#) 命令從現有資源中刪除標籤。如需範例，請參閱 AWS CLI 命令參考中的 [範例](#)。

PowerShell

檢視置放群組的標籤

使用 [Get-EC2Tag](#) 命令。



描述特定置放群組的標籤

使用 [Get-EC2PlacementGroup](#) 命令。

標記現有的置放群組

使用 [New-EC2Tag](#) 命令。

從置放群組刪除標籤

使用 [Remove-EC2Tag](#) 命令。

## 在放置群組中啟動執行個體

如果符合[置放群組規則和限制](#)，您可以使用下列其中一種方法，將置放群組中啟動執行個體。

### Console

在置放群組中啟動執行個體

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在 EC2 主控台儀表板的啟動執行個體方塊中，選擇啟動執行個體。依照指示完成表單，請注意做到下列幾點：
  - 在 Instance Type (執行個體類型) 欄位中，選取可以在置放群組中啟動的執行個體類型。
  - 在 Number of instances (執行個體的數量) 下的 Summary (摘要) 方塊中，輸入您在這個置放群組中所需的執行個體總數，您之後可能會無法在此置放群組中加入執行個體。
  - 在 Advanced details (進階詳細資訊) 下的 Placement group name (置放群組名稱) 欄位中，可以選擇將執行個體新增至新的或現有的置放群組。如果選擇具有分區策略的置放群組，請在 Target partition (目標分割區) 欄位中選擇要在其中啟動執行個體的分區。

### AWS CLI

在置放群組中啟動執行個體

使用 [run-instances](#) 命令，並使用 `--placement "GroupName = my-cluster"` 參數指定置放群組名稱。在此範例中，置放群組會命名為 `my-cluster`。

```
aws ec2 run-instances --placement "GroupName = my-cluster"
```

使用將執行個體啟動到分割區放置群組的特定分割區 AWS CLI

使用 [run-instances](#) 命令，並使用 `--placement "GroupName = HDFS-Group-A, PartitionNumber = 3"` 參數指定置放群組名稱和分區。在此範例中，置放群組名為 HDFS-Group-A，且分區數為 3。

```
aws ec2 run-instances --placement "GroupName = HDFS-Group-A, PartitionNumber = 3"
```

## PowerShell

使用 AWS Tools for Windows PowerShell 在置放群組中啟動執行個體

使用指 [New-EC2Instance](#) 令並使用 `-Placement_GroupName` 參數指定置放群組名稱。

## 描述置放群組中的執行個體

您可以使用下列其中一種方法來檢視執行個體的置放資訊。您也可以使用 AWS CLI，依照分區號碼篩選分區置放群組。

## Console

檢視執行個體的置放群組和分區號碼

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體。
4. 在 Details (詳細資訊) 標籤的 Host and placement group (主機與置放群組) 下，尋找 Placement group (置放群組)。如果執行個體不在置放群組中，則欄位為空白。否則，它會包含置放群組的名稱。如果置放群組是分割區置放群組，則 Partition number (分割區編號) 包含執行個體的分區編號。

## AWS CLI

檢視分區置放群組中執行個體的分區號碼

使用 [describe-instances](#) 命令，並指定 `--instance-id` 參數。

```
aws ec2 describe-instances --instance-id i-0123a456700123456
```

回應包含置放資訊，其中包括置放群組名稱和執行個體分區數。

```
"Placement": {
 "AvailabilityZone": "us-east-1c",
 "GroupName": "HDFS-Group-A",
 "PartitionNumber": 3,
 "Tenancy": "default"
}
```

篩選特定分區放置群組和分區號碼的執行個體

使用 [describe-instances](#) 命令，並使用 `--filters` 和 `placement-group-name` 篩選條件來指定 `placement-partition-number` 參數。在此範例中，置放群組名為 `HDFS-Group-A`，且分區數為 7。

```
aws ec2 describe-instances --filters "Name = placement-group-name, Values = HDFS-Group-A" "Name = placement-partition-number, Values = 7"
```

回應會列出在指定分區的指定置放群組內的所有執行個體。以下範例輸出僅顯示執行個體 ID、執行個體類型和傳回的執行個體置放資訊。

```
"Instances": [
 {
 "InstanceId": "i-0a1bc23d4567e8f90",
 "InstanceType": "r4.large",
 },
 "Placement": {
 "AvailabilityZone": "us-east-1c",
 "GroupName": "HDFS-Group-A",
 "PartitionNumber": 7,
 "Tenancy": "default"
 }
 {
 "InstanceId": "i-0a9b876cd5d4ef321",
 "InstanceType": "r4.large",
 },
 "Placement": {
 "AvailabilityZone": "us-east-1c",
```

```
 "GroupName": "HDFS-Group-A",
 "PartitionNumber": 7,
 "Tenancy": "default"
 }
],
```

## 變更執行個體的置放群組

您可以使用下列任一方式變更執行個體的置放群組：

- 將現有執行個體移至置放群組
- 在置放群組之間移動執行個體

執行個體必須處於 `stopped` 狀態，您才能移動該執行個體。

### Console

#### 將執行個體移到置放群組

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體，然後選擇執行個體狀態、停止執行個體。
4. 在已選取執行個體的情況下，選擇動作、執行個體設定、修改執行個體的配置。
5. 對於置放群組，選擇要做為移動執行個體目的地的置放群組。
6. 選擇儲存。

### AWS CLI

#### 將執行個體移到置放群組

1. 使用 [stop-instances](#) 命令停止執行個體。
2. 使用 [modify-instance-placement](#) 命令，並針對要做為移動執行個體目的地的置放群組，指定群組的名稱。

```
aws ec2 modify-instance-placement \
 --instance-id i-0123a456700123456 \
 --group-name MySpreadGroup
```

3. 使用 [start-instances](#) 命令啟動執行個體。

## PowerShell

使用 AWS Tools for Windows PowerShell將執行個體移到置放群組

1. 使用 [Stop-EC2Instance](#) 指令停止執行個體。
2. 使用指 [Edit-EC2InstancePlacement](#) 令並指定要將例證移至的放置群組名稱。
3. 使用 [Start-EC2Instance](#) 指令啟動執行個體。

## 從置放群組中移除執行個體

您可以使用下列其中一種方法，從置放群組移除執行個體。

執行個體必須處於 stopped 狀態，您才能從置放群組中移除該執行個體。

## Console

從置放群組中移除執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體，然後選擇執行個體狀態、停止執行個體。
4. 在已選取執行個體的情況下，選擇動作、執行個體設定、修改執行個體的配置。
5. 對於置放群組，請選擇無。
6. 選擇儲存。

## AWS CLI

從置放群組中移除執行個體

1. 使用 [stop-instances](#) 命令停止執行個體。
2. 使用 [modify-instance-placement](#) 命令，並針對置放群組名稱指定空的字串。

```
aws ec2 modify-instance-placement \
 --instance-id i-0123a456700123456 \
 --group-name ""
```

3. 使用 [start-instances](#) 命令啟動執行個體。

## PowerShell

使用 AWS Tools for Windows PowerShell 移除置放群組中的執行個體

1. 使用 [Stop-EC2Instance](#) 指令停止執行個體。
2. 使用指 [Edit-EC2InstancePlacement](#) 令並指定置放群組名稱的空字串。
3. 使用 [Start-EC2Instance](#) 指令啟動執行個體。

## 刪除置放群組

如果需要更換或不再需要某個置放群組，可以刪除該置放群組。您可以使用下列其中一種方法來刪除置放群組。

### 先決條件

置放群組不得包含任何執行個體，您才可予以刪除。您可以 [終止](#) 在該置放群組中啟動的所有執行個體，將執行個體 [移動](#) 到另一個置放群組，或將執行個體從置放群組中 [移除](#)。

## Console

### 刪除置放群組

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Placement Groups (置放群組)。
3. 選取置放群組並依序選取 Actions (動作)、Delete (刪除)。
4. 出現確認提示時，請輸入 **Delete**，然後選擇 Delete (刪除)。

## AWS CLI

### 刪除置放群組

使用 [delete-placement-group](#) 命令並指定置放群組名稱以刪除置放群組。在此範例中，置放群組名稱為 `my-cluster`。

```
aws ec2 delete-placement-group --group-name my-cluster
```

## PowerShell

若要使用刪除放置群組 AWS Tools for Windows PowerShell

使用 [Remove-EC2PlacementGroup](#) 指令刪除放置群組。

## 共用置放群組

放置群組共用可讓您影響不同 AWS 帳戶所擁有的相互依存例證的位置。您可以跨多個 AWS 帳戶或組織內共用放置群組。您可以在共用置放群組中啟動執行個體。

置放群組擁者可與下列項目共用置放群組：

- 組織內部或外部的特定 AWS 帳戶
- 組織內的組織單位
- 整個 組織

### Note

您要共用放置群組的 AWS 帳戶必須在 IAM 政策中具有下列許可。

- `ec2:PutResourcePolicy`
- `ec2>DeleteResourcePolicy`

## 主題

- [規則與限制](#)
- [跨可用區域共用](#)
- [共用置放群組](#)
- [識別共用置放群組](#)
- [在共用置放群組中啟動執行個體](#)
- [取消共用已共用置放群組](#)

## 規則與限制

當您共用置放群組或與您共用置放群組時，下列規則和限制適用。

- 若要共用刊登位置群組，您必須在 AWS 帳戶中擁有該群組。您無法將已共用給您的置放群組再共用出去。
- 共用分割區或分散置放群組時，置放群組限制不會變更。共用分割區置放群組在每個可用區域最多可有七個分割區，而共用分散置放群組在每個可用區域最多可支援七個執行中的執行個體。
- 若要與您的組織或組織中的組織單位共用放置群組，您必須啟用與共用功能 AWS Organizations。如需詳細資訊，請參閱[共用您的 AWS 資源](#)。
- 您必須負責管理共用置放群組中您所擁有的執行個體。
- 您無法檢視或修改與共用置放群組相關聯但非您擁有的執行個體和容量保留。

## 跨可用區域共用

為確保資源分配至區域中的所有可用區域，可用區域會獨立對應至各個帳戶的名稱。這可能導致帳戶之間的可用區域命名出現差異。例如，您 AWS 帳戶的可 us-east-1a 用區域可能與其他 AWS 帳戶 us-east-1a 的位置不同。

若要基於您的帳戶來識別專用執行個體的相對位置，您必須使用可用區域 ID (AZ ID)。可用區域 ID 是可用區域在所有 AWS 帳戶之間唯一且一致的識別符。例如，use1-az1 是 us-east-1 區域的可用區域 ID，它在每一個 AWS 帳戶的位置都相同。

檢視您帳戶中可用區域的可用區域 ID

1. [請在以下位置開啟 AWS RAM 主控台](https://console.aws.amazon.com/ram)。 <https://console.aws.amazon.com/ram>
2. 右側窗格中 Your AZ ID (您的 AZ ID) 下會顯示目前區域的可用區域 ID。

## 共用置放群組

若要共用置放群組，您必須將它新增至資源共用。資源共用是一 AWS RAM 種可讓您跨 AWS 帳號共用資源的資源。資源共享指定要共用的資源，以及共用它們的消費者。

如果您屬於組織內 AWS Organizations 共用的組織已啟用，您組織中的用戶就會獲得共用置放群組的存取權。

如果放置群組與組織外部的 AWS 帳戶共用，AWS 帳戶擁有者將會收到加入資源共用的邀請。他們可以在接受邀請後存取共用置放群組。

您可以使用 <https://console.aws.amazon.com/ram> 或跨 AWS 帳戶共用放置群組 AWS CLI。



## AWS RAM console

若要使用 <https://console.aws.amazon.com/ram> 共用您擁有的置放群組，請參閱[建立資源共用](#)。

## AWS CLI

若要共用您擁有的置放群組，請使用「[建立資源共用](#)」指令。

## 識別共用置放群組

放置群組的 Amazon 資源名稱 (ARN) 包含擁有放置群組之帳戶的 12 位數帳戶 ID。您可以使用帳戶 ID 來識別與您共用的置放群組的擁有者。

您可以使用下列其中一種方法來尋找放置群組 ARN。如需詳細資訊，請參閱 [檢視放置群組資訊](#)。

### Amazon EC2 console

若要識別共用放置群組

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在功能窗格的 [網路與安全性] 底下，選擇 [置放群組]。
3. 「放置群組」(Placement group) 表格會列出您所擁有並與您共用的所有放置群組。「群組 ARN」欄會顯示放置群組 ARN。

如果看不到「群組 ARN」欄，請選擇右上角的設定



開啟「群組 ARN」，然後選擇「確認」。

### AWS CLI

若要識別共用放置群組

使用 [describe-placement-groups](#) 指令列出您所擁有並與您共用的所有放置群組。在回應中，GroupId 參數會顯示放置群組的 ARN。

## 在共用置放群組中啟動執行個體

### Important

使用啟動共 AWS CLI 用放置群組中的例證時，您必須使用 `GroupId` 參數指定放置群組 ID。

只有當您是要共用之置放群組的擁有者時，才能使用放置群組名稱。我們建議您使用放置群組 ID，以避免 AWS 帳戶之間潛在的放置群組名稱衝突。

您可以在 Amazon EC2 主控台的「放置群組」畫面或使用 [describe-placement-groups](#) AWS CLI 指令，找到放置群組的 ID。如需詳細資訊，請參閱 [檢視放置群組資訊](#)。

### Console

將例證啟動到共用放置群組

1. 請依照程序 [啟動執行個體](#)，但在完成下列步驟指定置放群組的設定之前，請勿啟動執行個體。
2. 在 Instance type (執行個體類型) 中，選取支援的執行個體類型。如需詳細資訊，請參閱 [放置群組的規則與限制](#)。
3. 展開 [進階詳細資料]，然後依下列方式配置置群組設定：
  - a. 在「放置」群組中，選取與您共用的放置群組。

### Note

如果有相同名稱的置放群組，請檢查放置群組 ID，確定您選取了正確的放置群組。

- b. 如果您選擇具有分割區策略的置放群組，請針對「目標分割區」選擇要在其中啟動執行個體的分割區。
4. 在「摘要」面板中，執行下列動作：
    - a. 對於 Number of instances (執行個體的數目)，輸入您在這個置放群組中所需的執行個體總數，因為您之後可能會無法在此置放群組中加入執行個體。
    - b. 檢閱您的執行個體組態，然後選擇 Launch 執行個體。

如需詳細資訊，請參閱 [使用新的啟動執行個體精靈啟動執行個體](#)。

## AWS CLI

若要在共用置放群組中啟動執行個體

使用指[run-instances](#) 令並指定共用放置群組的放置群組 ID。

```
aws ec2 run-instances --placement "GroupId = pg-0123456789example"
```

若要將執行個體啟動到共用分割區置放群組的特定分割區

使用指[run-instances](#) 令並指定共用置放群組的放置群組 ID 和分割區編號。

```
aws ec2 run-instances --placement "GroupId = pg-0123456789example, PartitionNumber
= 3"
```

### Tip

使用 VPC 對等連接不同 AWS 帳戶所擁有的執行個體，並取得共用叢集置放群組所提供的完整延遲優勢。如需詳細資訊，請參閱[什麼是 VPC 互連？](#)

## 取消共用已共用置放群組

置放群組擁有者可以隨時取消共用已共用置放群組。

當您取消共用已共用置放群組時，下列變更將會生效。

- 共用置放群組的 AWS 帳戶將無法再啟動執行個體或保留容量。
- 如果您的執行個體是在共用置放群組中執行，則它們會與置放群組中斷關聯，但仍會繼續在您的 AWS 帳戶中正常執行。
- 如果您在共用置放群組中有容量保留，這些保留項目將會與置放群組中斷關聯，但您仍可在 AWS 帳戶中繼續存取這些資料。

您可以使用下列其中一種方法來取消共用已共用置放群組。

### AWS RAM console

若要使用 <https://console.aws.amazon.com/ram> 取消共用已共用置放群組，請參考[刪除資料共用](#)。

## AWS CLI

若要使用取消共用放置群組 AWS Command Line Interface，請使用取消關聯-資源共用指令。

## 放置群組 AWS Outposts

AWS Outposts 是一項全受管服務，可將 AWS 基礎架構、服務、API 和工具延伸至客戶場所。透過提供 AWS 受管理基礎架構的本機存取權，AWS Outposts 讓客戶能夠使用與 AWS 區域相同的程式設計介面在內部部署建置和執行應用程式，同時使用本機運算和儲存資源來降低延遲和本機資料處理需求。

Outpost 是部署在客戶站點的 AWS 計算和儲存容量集區。AWS 作為 AWS 區域的一部分來操作、監控和管理此容量。

您可以在帳戶中建立的 Outposts 中建立置放群組。這樣，您就可以在站點上的 Outpost 中跨基礎硬體分散執行個體。在 Outposts 中建立和使用置放群組的方式，與在一般可用區域中建立和使用置放群組的方式相同。在 Outpost 中建立具有分散策略的置放群組時，可以選擇讓置放群組在主機層級或是機架層級分散執行個體。在主機層級分散執行個體可讓您透過單一機架 Outpost 使用分散策略。

### 考量

- 機架層級分攤放置群組可容納與 Outpost 部署中機架相同數量的執行個體。
- 主機層級分攤置放群組可以容納與 Outpost 部署中主機相同數量的執行個體。

### 先決條件

您的站點必須安裝 Outpost。如需詳細資訊，請參閱 AWS Outposts 使用者指南中的[建立 Outpost 並訂購 Outpost 容量](#)。

### 在 Outpost 中使用置放群組

1. 在 Outpost 上建立子網。如需詳細資訊，請參閱 AWS Outposts 使用者指南中的[建立子網](#)。
2. 在 Outpost 的關聯區域中建立置放群組。如果您使用分攤策略建立置放群組，您可以選擇主機或機架層級分攤，以決定群組如何將執行個體分散到 Outpost 上的基礎硬體。如需詳細資訊，請參閱 [the section called “建立置放群組”](#)。
3. 在置放群組中啟動執行個體。在 Subnet (子網路) 欄位中，選取您在步驟 1 建立的子網路，在 Placement group name (置放群組名稱) 欄位中，選取您在步驟 2 建立的置放群組。如需詳細資訊，請參閱 AWS Outposts 使用者指南中的[在 Outpost 上啟動執行個體](#)。

## EC2 執行個體的網路最大傳輸單位 (MTU)

網路連線的最大傳輸單位 (MTU) 係允許通過該連線的最大封包大小 (以位元組為單位)。連線的 MTU 越大，單一封包能傳遞的資料也越多。乙太網幀內含封包 (或您實際傳送的資料) 和環繞的網路額外負荷資訊。

乙太網路訊框具有不同格式，最常見的格式是標準乙太網路 v2 訊框格式。其可支援 1500 MTU，此為多數網路均支援的最大乙太網路封包大小。執行個體支援的最大 MTU 因執行個體類型而異。

下列規則適用於 Wavelength 區域中的執行個體：

- 在相同 Wavelength 區域的 VPC 內從一個執行個體傳送至另一個執行個體的流量的 MTU 為 1300。
- 從一個執行個體傳送至另一個使用 Wavelength 區域內電信業者 IP 的執行個體的流量的 MTU 為 1500。
- 在 Wavelength 區域與使用公有 IP 地址的區域之間從一個執行個體傳送至另一個執行個體的流量的流量 MTU 為 1500。
- 在 Wavelength 區域與使用私有 IP 地址的區域之間從一個執行個體傳送至另一個執行個體的流量的流量 MTU 為 1300。

下列規則適用於 Outposts 中的執行個體：

- 從 Outposts 中的執行個體流向區域中執行個體的流量的 MTU 為 1300。

### 目錄

- [巨型訊框 \(9001 MTU\)](#)
- [路徑 MTU 探索](#)
- [檢查兩個主機間的路徑 MTU](#)
- [檢查執行個體的 MTU](#)
- [為您的執行個體設定 MTU](#)
- [疑難排解](#)

## 巨型訊框 (9001 MTU)

巨型訊框可增加單一封包的承載大小，進而增加不屬於封包成本的封包比例，藉此允許超過 1500 位元組的資料。傳送相同數量的可用資料，所需的封包數即會減少。但是，在下列情況下，流量限制為 MTU 最大為 1500：

- 網際網路閘道的流量
- 跨區域 VPC 對等互連的流量
- VPN 連線的流量
- 指定 AWS 區域以外的流量

封包若超過 1500 位元組將被切割為片段，若其 IP 標頭設有 Don't Fragment 旗標，則會將其捨棄。

巨型訊框應謹慎用於進出網路的流量，或任何離開 VPC 的流量。封包由中介系統切割，因此拖累流量速度。欲在 VPC 中使用巨型訊框但不減緩流出 VPC 的流量，您可依據路由設定 MTU 大小，或使用多個具備不同 MTU 大小與路由的彈性網路介面。

以配置於集群放置群組內的執行個體而言，巨型訊框有助實現最大網路傳輸量，此情況建議使用巨型訊框。如需詳細資訊，請參閱 [置放群組](#)。

您可以針對透過 AWS Direct Connect 的 VPC 和現場部署網路之間的流量，使用 Jumbo Frame。如需詳細資訊，以及如何驗證 Jumbo Frame 功能，請參閱 AWS Direct Connect 使用者指南中的 [設定網路 MTU](#)。

所有 Amazon EC2 執行個體類型都支援 1500 MTU，目前所有的執行個體類型都支援巨型框架。下列前一代執行個體類型支援巨型框架：A1、C3、I2、M3 和 R3。

如需支援的 MTU 大小詳細資訊：

- 如需 NAT 閘道相關資訊，請參閱《Amazon VPC 使用者指南》中的 [NAT 閘道基本概念](#)。
- 如需傳輸閘道相關資訊，請參閱《Amazon VPC 傳輸閘道使用者指南》中的 [MTU](#)。
- 如需 Local Zones 相關資訊，請參閱《AWS Local Zones 使用者指南》中的 [考量](#)。

## 路徑 MTU 探索

路徑 MTU 探索 (PMTUD) 可用於確認兩個裝置間的路徑 MTU。路徑 MTU 是原始主機和接收主機之間的路徑上支援的最大封包尺寸。在兩個主機之間的網路中，當 MTU 大小出現差異時，PMTUD 會允許

接收主機使用 ICMP 訊息回應原始主機。此 ICMP 訊息指示原始主機沿著網路路徑，使用最低的 MTU 大小來重新傳送請求。如果不這樣協議，可能會因為請求太大，使得接收端主機無法接受，而發生封包捨棄。

針對 IPv4，若主機傳送的封包大小大於接收主機的 MTU，或是大於路徑上裝置的 MTU，則接收主機或裝置便會丟棄封包，然後傳回下列 ICMP 訊息：Destination Unreachable: Fragmentation Needed and Don't Fragment was Set (類型 3，代碼 4)。這會指示傳輸主機將承載分割成多個較小的封包，然後重新傳輸它們。

IPv6 通訊協定不支援網路中的分段。若主機傳送的封包大小大於接收主機的 MTU，或是大於路徑上裝置的 MTU，則接收主機或裝置便會丟棄封包，然後傳回下列 ICMP 訊息：ICMPv6 Packet Too Big (PTB) (類型 2)。這會指示傳輸主機將承載分割成多個較小的封包，然後重新傳輸它們。

透過某些元件 (例如 NAT 閘道和負載平衡器) 建立的連線會[自動追蹤](#)。這意味著系統會自動為您的傳出連線嘗試啟用[安全群組追蹤](#)。如果自動追蹤連線，或者安全群組規則允許傳入 ICMP 流量，就可以接收 PMTUD 回應。

請注意，即使在安全群組層級允許流量，也可以封鎖 ICMP 流量，例如，如果您有拒絕進入子網路的 ICMP 流量的網路存取控制清單項目。

#### Important

路徑 MTU 探索並不保證某些路由器不會捨棄巨型訊框。您 VPC 內的網際網路閘道僅能轉送至多 1500 位元組的封包。網際網路流量建議使用 1500 MTU 封包。

## 檢查兩個主機間的路徑 MTU

您可以檢查 EC2 執行個體和其他主機之間的路徑 MTU。您可以指定 DNS 名稱或 IP 位址做為目的地。如果目的地是另一個 EC2 執行個體，請確認其安全群組允許輸入 UDP 流量。

您使用的程序視執行個體的作業系統而定。

### Linux 執行個體

在執行個體上執行 `tracert` 命令，以檢查 EC2 執行個體和指定目的地之間的路徑 MTU。這個命令是軟件 `iputils` 包的一部分，默認情況下在許多 Linux 發行版中都可用。

此範例會檢查 EC2 執行個體和 `amazon.com` 之間的路徑 MTU。

```
[ec2-user ~]$ traceroute amazon.com
```

在此範例輸出中，路徑 MTU 為 1500。

```
1?: [LOCALHOST] pmtu 9001
1: ip-172-31-16-1.us-west-1.compute.internal (172.31.16.1) 0.187ms pmtu 1500
1: no reply
2: no reply
3: no reply
4: 100.64.16.241 (100.64.16.241) 0.574ms
5: 72.21.222.221 (72.21.222.221) 84.447ms asymm 21
6: 205.251.229.97 (205.251.229.97) 79.970ms asymm 19
7: 72.21.222.194 (72.21.222.194) 96.546ms asymm 16
8: 72.21.222.239 (72.21.222.239) 79.244ms asymm 15
9: 205.251.225.73 (205.251.225.73) 91.867ms asymm 16
...
31: no reply
 Too many hops: pmtu 1500
 Resume: pmtu 1500
```

## Windows 執行個體

### 使用曲線檢查路徑 MTU

1. 請從 `mturoute.exe` 以下位置下載至您的 EC2 執行個體。 <http://www.elifulkerson.com/projects/mturoute.php>
2. 開啟命令提示字元視窗，並變更至您下載 `mturoute.exe` 所在的目錄。
3. 使用以下命令檢查 EC2 實例和指定目的地之間的路徑 MTU。此範例會檢查 EC2 執行個體和 `www.elifulkerson.com` 之間的路徑 MTU。

```
.\mturoute.exe www.elifulkerson.com
```

在此範例輸出中，路徑 MTU 為 1500。

```
* ICMP Fragmentation is not permitted. *
* Speed optimization is enabled. *
* Maximum payload is 10000 bytes. *
+ ICMP payload of 1472 bytes succeeded.
- ICMP payload of 1473 bytes is too big.
Path MTU: 1500 bytes.
```



## 檢查執行個體的 MTU

您可以檢查執行個體的 MTU 值。部分執行個體設定為使用巨型訊框，其他則使用標準訊框大小。

您使用的程序視執行個體的作業系統而定。

### Linux 執行個體

#### 檢查 Linux 執行個體上的 MTU 設定

在 EC2 執行個體上執行下列ip命令。如果主要網路介面不是eth0，請eth0以您的網路介面取代。

```
[ec2-user ~]$ ip link show eth0
```

在此範例輸出中，*mtu 9001* 表示執行個體使用超大型框架。

```
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc pfifo_fast state UP mode
 DEFAULT group default qlen 1000
 link/ether 02:90:c0:b7:9e:d1 brd ff:ff:ff:ff:ff:ff
```

### Windows 執行個體

您使用的程序取決於執行個體上的驅動程式。

#### ENA driver

##### 版本 2.1.0 及更高版本

若要取得 MTU 值，請在 EC2 執行個體上使用下列Get-NetAdapterAdvancedProperty命令。使用萬用字元 (星號) 取得所有乙太網路名稱。檢查介面名稱的輸出\*JumboPacket。9015 值表示已啟用巨型訊框。巨型訊框預設為停用。

```
Get-NetAdapterAdvancedProperty -Name "Ethernet*"
```

##### 版本 1.5 及更早版本

若要取得 MTU 值，請在 EC2 執行個體上使用下列Get-NetAdapterAdvancedProperty命令。檢查介面名稱的輸出MTU。9001 值表示已啟用巨型訊框。巨型訊框預設為停用。

```
Get-NetAdapterAdvancedProperty -Name "Ethernet"
```

## Intel SRIOV 82599 driver

若要取得 MTU 值，請在 EC2 執行個體上使用下列 `Get-NetAdapterAdvancedProperty` 命令。檢查介面名稱 `*JumboPacket` 的項目。9014 值表示已啟用巨型訊框。(請注意，MTU 大小包含標頭和承載。) 巨型訊框預設為停用。

```
Get-NetAdapterAdvancedProperty -Name "Ethernet"
```

## AWS PV driver

若要取得 MTU 值，請在 EC2 執行個體上使用下列命令。介面名稱可能有所不同。請在輸出中尋找具備名稱 "Ethernet"、"Ethernet 2" 或 "Local Area Connection" 的項目。您將需要介面名稱來啟用或停用巨型訊框。9001 值表示已啟用巨型訊框。

```
netsh interface ipv4 show subinterface
```

## 為您的執行個體設定 MTU

您可能想要針對 VPC 內的網路流量使用巨型框架，並針對網際網路流量使用標準框架。無論您的使用案例為何，我們都建議您驗證執行個體是否如預期般運作。

您使用的程序視執行個體的作業系統而定。

### Linux 執行個體

#### 設定 Linux 執行個體上的 MTU 值

1. 在執行個體上執行下列 `ip` 命令。它會將所需的 MTU 值設定為 1500，但您可以改用 9001。

```
[ec2-user ~]$ sudo ip link set dev eth0 mtu 1500
```

2. (選用) 欲在重新啟動後保留您的網路 MTU 設定，請依據您的作業系統類型修改下列組態檔案。

- 若為 Amazon Linux 2，請將下列一行新增至 `/etc/sysconfig/network-scripts/ifcfg-eth0` 檔案：

```
MTU=1500
```

將下行新增至 `/etc/dhcp/dhclient.conf` 檔案：

```
request subnet-mask, broadcast-address, time-offset, routers, domain-name,
domain-search, domain-name-servers, host-name, nis-domain, nis-servers, ntp-
servers;
```

- 對於 Amazon Linux AMI，請將以下幾行添加到您的/etc/dhcp/dhclient-eth0.conf文件中。

```
interface "eth0" {
supersede interface-mtu 1500;
}
```

- 若是其他 Linux 發行版本，請參閱其特定文件。

### 3. (可選) 重新啟動執行個體，並確認 MTU 設定正確。

## Windows 執行個體

您使用的程序取決於執行個體上的驅動程式。

### ENA driver

您可以使用裝置管理員或執行個體上的Set-NetAdapterAdvancedProperty指令來變更 MTU。

#### 版本 2.1.0 及更高版本

使用下列命令來啟用巨型框架。

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "*JumboPacket" -
RegistryValue 9015
```

使用下列命令來停用巨型框架。

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "*JumboPacket" -
RegistryValue 1514
```

#### 版本 1.5 及更早版本

使用下列命令來啟用巨型框架。

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "MTU" -
RegistryValue 9001
```

使用下列命令來停用巨型框架。

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "MTU" -
RegistryValue 1500
```

### Intel SRIOV 82599 driver

您可以使用裝置管理員或執行個體上的Set-NetAdapterAdvancedProperty指令來變更 MTU。

使用下列命令來啟用巨型框架。

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "*JumboPacket" -
RegistryValue 9014
```

使用下列命令來停用巨型框架。

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "*JumboPacket" -
RegistryValue 1514
```

### AWS PV driver

您可以使用執行個體上的指netsh令來變更 MTU。您無法使用「裝置管理員」變更 MTU。

使用下列命令來啟用巨型框架。

```
netsh interface ipv4 set subinterface "Ethernet" mtu=9001
```

使用下列命令來停用巨型框架。

```
netsh interface ipv4 set subinterface "Ethernet" mtu=1500
```

## 疑難排解

如果您在使用巨型框架時，EC2 執行個體和 Amazon Redshift 叢集之間遇到[連線問題](#)，請參閱[Amazon Redshift 管理指南中的查詢似乎掛起](#)。

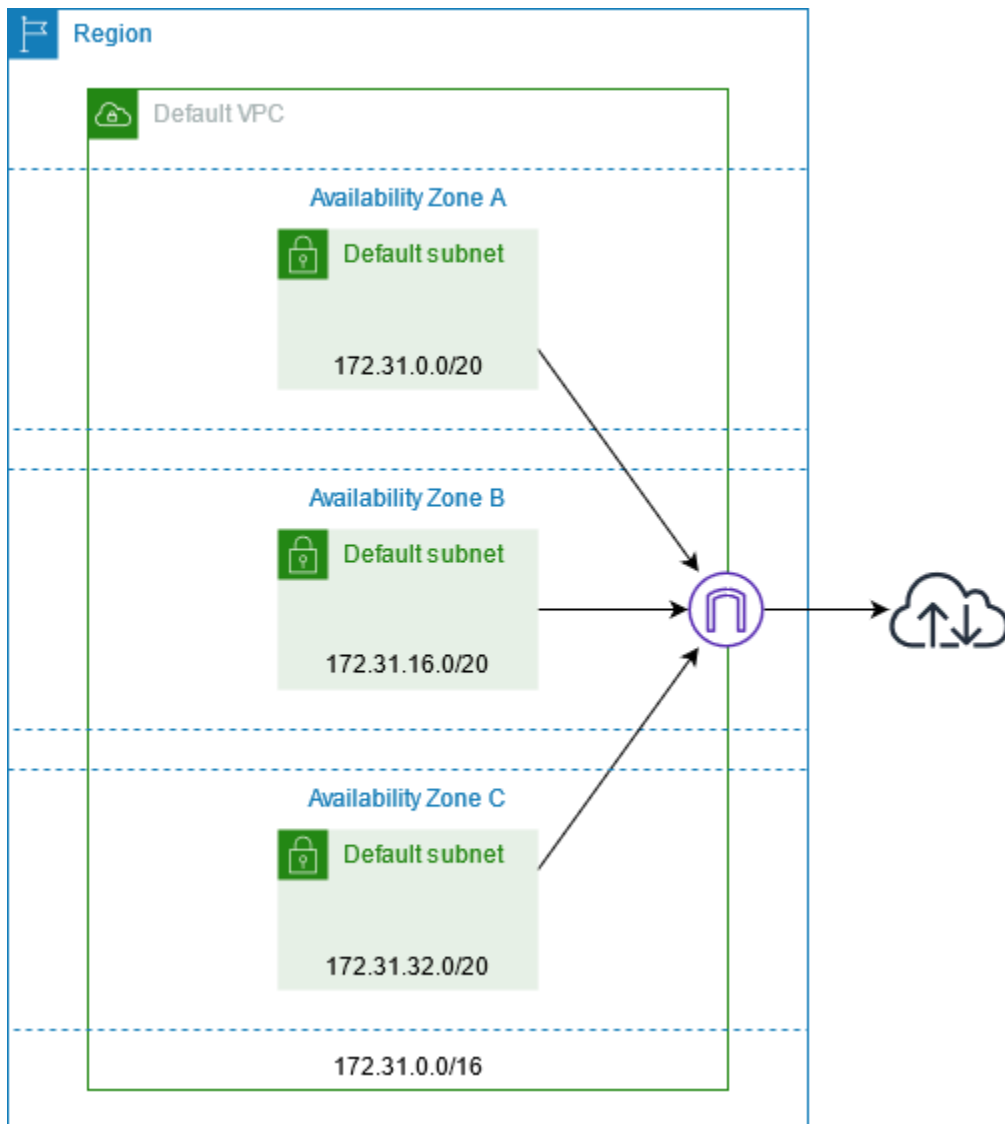
## 適用於 EC2 執行個體的虛擬私有雲

Amazon Virtual Private Cloud (Amazon VPC) 可讓您在雲端內的邏輯隔離區域中定義虛擬網路，稱為虛擬私有雲或 VPC。AWS 您可以在 VPC 的子網路中建立 AWS 資源 (例如 Amazon EC2 執行個

體)。VPC 近似於您在自有資料中心內運作的傳統網路，卻能提供 AWS 可擴展性基礎設施的效益。您可以設定您的 VPC；您可以選取其 IP 地址範圍、建立子網，以及設定路由表、網路閘道與安全設定。您可以將 VPC 中的執行個體連線至網際網路或連線至自己的資料中心。

## 您的預設 VPC

當您建立 AWS 帳戶時，我們會在每個區域建立預設 VPC。預設 VPC 是已設定好可供您使用的 VPC。例如，每個預設 VPC 的每個可用區域都有預設子網、連接到 VPC 的網際網路閘道，且在主路由表中有路由，將所有流量 (0.0.0.0/0) 傳送至網際網路閘道。或者，您可以建立自己的 VPC，並進行設定以滿足您的需要。



## 建立其他 VPC

使用下列程序建立具有您所需子網路、閘道和路由組態的 VPC。

## 建立 VPC

1. 前往 <https://console.aws.amazon.com/vpc/> 開啟 Amazon VPC 主控台。
2. 選擇建立 VPC。
3. 在要建立的資源之下，選擇 VPC 等。
4. 對於 Name tag auto-generation (自動產生名稱標籤)，請輸入 VPC 的名稱。
5. 對於 IPv4 CIDR block (IPv4 CIDR 區塊)，請保留預設建議，或輸入應用程式或網路所需的 CIDR 區塊。
6. 對於 Number of Availability Zones (可用區域數量)，請選擇 2，以便在多個可用區域中啟動執行個體，以確保高可用性。
7. 如果必須可從網際網路存取執行個體，請執行以下其中一項操作：
  - 如果您的執行個體可位於公有子網路中，請為 Number of public subnets (公有子網路數量) 選取非零值。保持選取 DNS options (DNS 選項) 下的兩個選項。您可以選擇現在或稍後新增私有子網路。
  - 如果您的執行個體必須位於私有子網路中，請為 Number of public subnets (公有子網路數量) 選取 0。對於 Number of private subnets (私有子網路數量)，請根據您的需要選取一個數字 (可能的值對應於每個可用區域 1 或 2 個私有子網路)。對於 NAT gateways (NAT 閘道)，如果兩個可用區域中的執行個體跨可用區域傳送或接收大量流量，則選取 1 per AZ (每個可用區域 1 個)。否則，請選取 In 1 AZ (在 1 個可用區域中)，然後啟動在與 NAT 閘道相同的可用區域中傳送或接收跨區域流量的執行個體。
8. 展開 Customize subnet CIDR blocks (自訂子網路 CIDR 區塊)。保留預設建議，或為每個子網路輸入 CIDR 區塊。如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的 [子網路 CIDR 區塊](#)。
9. 檢閱 Preview (預覽) 窗格，此窗格會顯示將根據您的選取項目所建立的 VPC 資源。
10. 選擇建立 VPC。

## 從您的執行個體中存取網際網路

在預設 VPC 中啟動至預設子網路的執行個體可存取網際網路，因為預設 VPC 設定為指派公用 IP 位址和 DNS 主機名稱，而主路由表則設定為連接至 VPC 之網際網路閘道的路由。

對於在非預設子網路和 VPC 中啟動的執行個體，您可以使用下列其中一個選項來確保您在這些子網路中啟動的執行個體可存取網際網路：

- 設定網際網路閘道。如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的 [使用網際網路閘道連接至網際網路](#)。

- 設定公有 NAT 閘道。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的[從私有子網存取網際網路](#)。

## 共用子網路

將 EC2 執行個體啟動到共用 VPC 子網路時，請注意以下事項：

- 參與者可以指定共用子網路的 ID，在共用子網路中執行執行個體。參與者必須擁有他們指定的任何安全群組或網路介面。
- 參與者可以啟動、停止、終止和描述他們在共用子網路中建立的執行個體。參與者無法啟動、停止、終止或描述 VPC 擁有者在共用子網路中建立的執行個體。
- VPC 擁有者無法啟動、停止、終止或描述共用子網路中參與者建立的執行個體。
- 參與者可以使用 EC2 執行個體 Connect 端點連線到共用子網路中的執行個體。參與者必須在共享子網中創建 EC2 實例 Connect 端點。參與者無法使用 VPC 擁有者在共用子網路中建立的 EC2 執行個體 Connect 端點。

如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的[與其他帳戶共享 VPC](#)。

## 僅限 IPv6 的子網路

在僅限 IPv6 之子網路中啟動的 EC2 執行個體會收到 IPv6 地址，而非 IPv4 地址。您啟動到僅限 IPv6 子網路的任何執行個體都必須是 Nitro 系統[上建置的執行個體 AWS](#)。

# Amazon EC2 中的安全

雲安全 AWS 是最高的優先級。身為 AWS 客戶，您可以從資料中心和網路架構中獲益，這些架構是專為滿足對安全性最敏感的組織的需求而打造的。

安全是 AWS 與您之間共同的責任。[共同責任模型](#)將其描述為雲端的安全性和雲端中的安全性：

- 雲端的安全性 — AWS 負責保護在 AWS 雲端中執行 AWS 服務的基礎架構。AWS 還為您提供可以安全使用的服務。若要了解適用於 Amazon EC2 的合規計劃，請參閱AWS 合規計劃的[合規計劃 AWS 服務範](#)的服務。
- 雲端內部的安全 – 您的責任包含下列領域：
  - 控制對執行個體的網路存取，例如：透過設定 VPC 和安全群組。如需詳細資訊，請參閱 [控制網路流量](#)。
  - 管理用於連線到執行個體的登入資料。
  - 管理部署至訪客作業系統的訪客作業系統和軟體，包括更新和安全性修補程式。如需詳細資訊，請參閱 [Amazon EC2 視窗執行個體的更新管理](#)。
  - 設定連接至執行個體的 IAM 角色以及與這些角色相關聯的許可。如需詳細資訊，請參閱 [Amazon EC2 的 IAM 的角色](#)。

本文件有助於您了解如何在使用 Amazon EC2 時套用共同責任模型。它會示範如何設定 Amazon EC2 以符合您的安全性和合規目標。您也會學到如何使用其他可 AWS 協助您監控和保護 Amazon EC2 資源的服務。

## 目錄

- [Amazon EC2 中的資料保護](#)
- [Amazon EC2 中的基礎設施安全](#)
- [Amazon EC2 的復原功能](#)
- [Amazon EC2 的合規驗證](#)
- [Amazon EC2 的 Identity and Access Management](#)
- [使用介面 VPC 端點存取 Amazon EC2](#)
- [Amazon EC2 視窗執行個體的更新管理](#)
- [Windows 執行個體的安全性最佳做法](#)
- [Amazon EC2 金鑰配對和 Amazon EC2 執行個體](#)



- [適用於 Amazon EC2 執行個體的亞馬遜 EC2 安全群組](#)
- [NitroTPM](#)
- [視窗執行個體的認證保護](#)

## Amazon EC2 中的資料保護

AWS [共同責任模型](#)適用於 Amazon 彈性運算雲端中的資料保護。如此模型中所述，AWS 負責保護執行所有 AWS 雲端。您負責維護在此基礎設施上託管內容的控制權。您也同時負責所使用 AWS 服務的安全組態和管理任務。如需資料隱私權的詳細資訊，請參閱[資料隱私權常見問答集](#)。如需有關歐洲資料保護的相關資訊，請參閱 AWS 安全性部落格上的 [AWS 共同的責任模型和 GDPR](#) 部落格文章。

基於資料保護目的，我們建議您使用 AWS IAM Identity Center 或 AWS Identity and Access Management (IAM) 保護 AWS 帳戶登入資料並設定個別使用者。如此一來，每個使用者都只會獲得授與完成其任務所必須的許可。我們也建議您採用下列方式保護資料：

- 每個帳戶均要使用多重要素驗證 (MFA)。
- 使用 SSL/TLS 與 AWS 資源進行通訊。我們需要 TLS 1.2 並建議使用 TLS 1.3。
- 使用設定 API 和使用者活動記錄 AWS CloudTrail。
- 使用 AWS 加密解決方案以及其中的所有默認安全控制 AWS 服務。
- 使用進階的受管安全服務 (例如 Amazon Macie)，協助探索和保護儲存在 Amazon S3 的敏感資料。
- 如果您在透過命令列介面或 API 存取時需要經 AWS 過 FIPS 140-2 驗證的加密模組，請使用 FIPS 端點。如需有關 FIPS 和 FIPS 端點的更多相關資訊，請參閱[聯邦資訊處理標準 \(FIPS\) 140-2 概觀](#)。

我們強烈建議您絕對不要將客戶的電子郵件地址等機密或敏感資訊，放在標籤或自由格式的文字欄位中，例如名稱欄位。這包括當您使用主控台、API 或 AWS 開發套件 AWS 服務使用 Amazon EC2 或其他工作時。AWS CLI您在標籤或自由格式文字欄位中輸入的任何資料都可能用於計費或診斷日誌。如果您提供外部伺服器的 URL，我們強烈建議請勿在驗證您對該伺服器請求的 URL 中包含憑證資訊。

### 目錄

- [Amazon EBS 資料安全](#)
- [靜態加密](#)
- [傳輸中加密](#)

## Amazon EBS 資料安全

Amazon EBS 磁碟區是以原始、未格式化的區塊型儲存設備型式提供給您。這些裝置是在 EBS 基礎設施上建立的邏輯裝置，Amazon EBS 服務可確保裝置在客戶進行任何使用或重複使用之前在邏輯上是空的 (也就是說，原始區塊為零或其包含加密虛擬隨機資料)。

若您的程序要求在使用後或使用前 (或兩者)，使用特定方法清除所有資料，例如 DoD 5220.22-M (國家工業安全計畫操作手冊) 或 NIST 800-88 (媒體清理準則)，您可在 Amazon EBS 上執行此作業。該區塊層級的活動將反映至 Amazon EBS 服務中的基礎儲存媒體。

### 靜態加密

#### EBS 磁碟區

Amazon EBS 加密是適用於 EBS 磁碟區和快照的加密解決方案。它使用 AWS KMS keys。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的 Amazon EBS 加密](#)。

[視窗執行個體] 您也可以使用 Microsoft EFS 和 NTFS 權限進行資料夾層級和檔案層級加密。

#### 執行個體儲存體磁碟區

NVMe 執行個體存放區磁碟區上的資料會以執行個體上的硬體模組中實作的 XTS-AES-256 區塊編碼器來加密。用於加密寫入本機連接 NVMe 儲存裝置之資料的金鑰按照每個客戶和每個磁碟區提供。這些金鑰由硬體模組產生且僅駐留在其中，AWS 人員無法存取此模組。這些加密金鑰會在執行個體停止或終止時銷毀，且無法復原。您無法停用此加密，也無法提供您自己的加密金鑰。

H1、D3 和 D3en 執行個體上的 HDD 執行個體存放區磁碟區中的資料會使用 XTS-AES-256 和一次性金鑰來加密。

當您讓執行個體停止、休眠或終止時，執行個體存放區磁碟區中的所有儲存區塊都會重設。因此，資料無法透過另一個執行個體的執行個體存放區存取。

### 記憶體

下列執行個體會啟用記憶體加密：

- 具有 AWS 重力子處理器的執行個體。AWS 引力 2、重力 on AWS 3 和 AWS 重力 on3E 支援永遠在線的記憶體加密。加密金鑰會在主機系統內安全地產生，不能離開主機系統，並在主機重新啟動或關閉電源時銷毀。如需詳細資訊，請參閱 [AWS Graviton 處理器](#)。
- 搭載第三代 Intel Xeon 可擴充處理器 (Ice Lake) 的執行個體 (例如 M6i 執行個體) 以及搭載第四代 Intel Xeon 可擴充處理器 (Sapphire Rapids) 的執行特體 (例如 M7i 執行個體)。這些處理器支援使用 Intel 總記憶體加密 (TME) 的永遠開啟記憶體加密。

- 搭載第三代 AMD EPYC 處理器 (Milan) 的執行個體 (例如 M6a 執行個體) 以及搭載第四代 AMD EPYC 處理器 (Genoa) 的執行個體 (例如 M7a 執行個體)。這些處理器使用 AMD 安全記憶體加密 (SME)，可支援永遠開啟的記憶體加密。搭載第三代 AMD EPYC 處理器 (Milan) 的執行個體也支援 AMD Secure Encrypted Virtualization-Secure Nested Paging (SEV-SNP)。

## 傳輸中加密

### 在實體層加密

在離開 AWS 安全設施之前，透過 AWS 全球網路跨 AWS 區域流動的所有資料都會在實體層自動加密。AZ 之間的所有流量都會加密。額外的加密層 (包括本節所列的加密層) 可能會提供額外的保護。

Amazon VPC 對等互連和 Transit Gateway 跨區域對等互連提供的加密

所有使用 Amazon VPC 對等互連和 Transit Gateway 對等的跨區域流量會在退出區域時自動進行大量加密。如本節先前所述，在所有流量離開 AWS 安全設施之前，會在實體層自動為所有流量提供額外的加密層。

### 執行個體間的加密

AWS 在所有類型的 EC2 執行個體之間提供安全和私有的連線。此外，某些執行個體類型使用基礎 Nitro System 硬體的卸載功能，以自動加密執行個體之間的傳輸中流量。此加密機制使用帶有關聯資料的認證加密 (AEAD) 演算法 (採用 256 位元加密)。這對網路效能沒有影響。若要支援執行個體之間額外的傳輸中流量加密，必須符合下列要求：

- 執行個體使用下列執行個體類型：
  - 一般用途：M5 DN，M5 N，兆兆安，M6i，M6i，M6 ID，M6 英寸，M7 安培，M7 克，M7gD，M7 億，M7i，M7-柔性
  - 計算優化：C5 安培，C5 公升，C5n，C6 安，C6 安，C6 標識，C6 英寸，C7 克，C7 克，C7 吉 D，氯化鎂，C7i，C7-柔性
  - 記憶體最佳化：R5 DN、R5 N、六安、R6 I、R6 英寸、R6 英寸、R7 A、R7 克、R7GD、R7i、R7IZ、U-三 TB1、U-6TB1、U -9TB1、U-十八 TB1、U U7 英寸-24 TB, 乌 7 英寸 32 TB, X2 IDN, X2 伊丹, X2 天
  - 儲存最佳化：D3、D3en、I3en、I4G、I4i、I4GN、i4 世代
  - 加速運算：DL1、DL2q、G4ad、G4 DN、G5、G6、GR6、資源 1、資料二次、P4dN、P4d、P4de、P5、中一、中一、VT1
  - 高效能運算：Hpc6a、Hpc6id、Hpc7a、Hpc7g
- 這些執行個體位於相同的區域中。

- 這些執行個體位於相同的 VPC 或對等 VPC 中，且流量不會經過虛擬網路裝置或服務，例如負載平衡器或傳輸閘道。

如本節先前所述，在所有流量離開 AWS 安全設施之前，會在實體層自動為所有流量提供額外的加密層。

使用 AWS CLI 檢視加密執行個體之間傳輸中流量的執行個體類型

使用下列 [describe-instance-types](#) 命令。

```
aws ec2 describe-instance-types \
 --filters Name=network-info.encryption-in-transit-supported,Values=true \
 --query "InstanceTypes[*].[InstanceType]" \
 --output text | sort
```

來回加密 AWS Outposts

Outpost 會建立特殊的網路連線，稱為連至其本 AWS 位目錄區域的服務連結，並選擇性地建立您指定之 VPC 子網路的私人連線。透過這些連線的所有流量都會完整加密。如需詳細資訊，請參閱 AWS Outposts 使用者指南中的 [透過服務連結連線](#) 和 [傳輸中加密](#)。

遠端存取加密

SSH 和 RDP 通訊協定為您的執行個體提供安全的通訊通道，無論是直接還是透過 EC2 執行個體 Connect 進行遠端存取。使用 AWS Systems Manager 工作階段管理員或執行命令對執行個體的遠端存取會使用 TLS 1.2 進行加密，建立連線的請求會使用 [Sigv4](#) 簽署，並經過 [AWS Identity and Access Management](#) 驗證和授權。

使用諸如 Transport Layer Security (TLS) 等加密通訊協定是您的責任，以便加密在用戶端和您的 Amazon EC2 執行個體之間傳輸的敏感資料。

(Windows 執行個體) 請務必只允許 EC2 執行個體與 AWS API 端點或其他敏感遠端網路服務之間的加密連線。您可以透過傳出安全群組或 [Windows 防火牆](#) 規則來強制執行此操作。

## Amazon EC2 中的基礎設施安全

作為受管服務，Amazon 彈性運算雲端受到 AWS 全球網路安全的保護。有關 AWS 安全服務以及如何 AWS 保護基礎架構的詳細資訊，請參閱 [AWS 雲端安全](#)。若要使用基礎架構安全性的最佳做法來設計您的 AWS 環境，請參閱 [安全性支柱](#) 架構良 AWS 好的架構中的基礎結構保護。

您可以使用 AWS 已發佈的 API 呼叫透過網路存取 Amazon EC2。使用者端必須支援下列專案：

- Transport Layer Security (TLS)。我們需要 TLS 1.2 並建議使用 TLS 1.3。
- 具備完美轉送私密(PFS)的密碼套件，例如 DHE (Ephemeral Diffie-Hellman)或 ECDHE (Elliptic Curve Ephemeral Diffie-Hellman)。現代系統(如 Java 7 和更新版本)大多會支援這些模式。

此外，請求必須使用存取金鑰 ID 和與 IAM 主體相關聯的私密存取金鑰來簽署。或者，您可以透過 [AWS Security Token Service \(AWS STS\)](#) 來產生暫時安全憑證來簽署請求。

如需詳細資訊，請參閱安全性支柱 — AWS Well-Architected 的架構中的基礎結構[保護](#)。

## 網路隔離

虛擬私有雲 (VPC) 是您自己在 AWS 雲中邏輯隔離區域中的虛擬網絡。使用不同的 VPC，依工作負載或組織實體來隔離基礎設施。

子網是您的 VPC 中的 IP 地址範圍。啟動執行個體時，您會在 VPC 的子網中啟動它。使用子網來隔離單一 VPC 內的應用程式層 (例如，Web、應用程式及資料庫)。如果不應該從網際網路直接存取，則針對您的執行個體使用私有子網。

若要使用私有 IP 地址從 VPC 呼叫 Amazon EC2 API，請使用 AWS PrivateLink。如需詳細資訊，請參閱 [使用介面 VPC 端點存取 Amazon EC2](#)。

## 實體主機上的隔離

已經將相同實體主機上的不同 EC2 執行個體彼此隔離，如同他們位於不同實體主機一樣。Hypervisor 會隔離 CPU 和記憶體，而且執行個體式提供的虛擬化磁碟，而非存取原始磁碟裝置。

當您停止或終止執行個體時，Hypervisor 會先清除配置到該執行個體的記憶體 (設定為零)，然後再將之配置到新執行個體，而且會重設儲存體的每個區塊。這可確保不會蓄意地向其他執行個體公開您的資料。

網路 MAC 位址是由網 AWS 路基礎結構動態指派給執行個體。IP 地址會由 AWS 網路基礎設施動態地指派給執行個體，或是由 EC2 管理員透過驗證的 API 請求進行指派。AWS 網路允許執行個體僅從指派給執行個體的 MAC 和 IP 位址傳送流量。否則，流量會遭到捨棄。

依預設，執行個體無法接收不是特別定址給它的流量。如果您需要在執行個體上執行網路位置轉譯 (NAT)、路油或防火牆服務，您可以針對網路介面停用來源/目的地檢查。

## 控制網路流量

請考慮下列選項來控制 EC2 執行個體的網路流量：

- 使用[安全性群組](#)限制您執行個體的存取權限。設定允許最低所需網路流量的規則。例如，您可以僅允許來自公司網路位址範圍的流量，或僅允許特定通訊協定 (例如 HTTPS) 的流量。對於 Windows 執行個體，請允許 Windows 管理流量和最小輸出連線。
- 運用安全群組作為控制 Amazon EC2 執行個體網路存取的主要機制。必要時，請謹慎使用網路 ACL 來提供無狀態、粗糙的網路控制。安全群組比網路 ACL 更多用途，因為它們能夠執行有狀態封包篩選，並建立參考其他安全群組的規則。不過，網路 ACL 可以有效地作為次要控制項，以拒絕特定流量子集或提供高階子網防護護欄。此外，由於網路 ACL 適用於整個子網路，因此可以像 defense-in-depth 在沒有正確安全性群組的情況下無意中啟動執行個體一樣使用。
- [Windows 執行個體] 使用群組原則物件 (GPO) 集中管理 Windows 防火牆設定，進一步強化網路控制。客戶通常會使用 Windows 防火牆進一步了解網路流量，以及補充安全群組篩選條件，以建立進階規則以封鎖特定應用程式存取網路，或篩選來自子集 IP 地址的流量。例如，Windows 防火牆可以限制只有特定使用者或應用程式才能存取 EC2 中繼資料服務 IP 地址。或者，面向公眾的服務可能會使用安全群組來限制特定連接埠的流量，並使用 Windows 防火牆來維護明確封鎖之 IP 地址的清單。
- 如果不應該從網際網路直接存取，則針對您的執行個體使用私有子網。使用堡壘主機或 NAT 閘道，以取得來自私人子網中執行個體的網際網路存取權限。
- [視窗執行個體] 使用安全的管理通訊協定，例如透過 SSL/TLS 進行 RDP 封裝。遠端桌面閘道 Quick Start 提供部署遠端桌面閘道的最佳實務，包括將 RDP 設定為使用 SSL/TLS。
- [Windows 執行個體] 使用「作用中目錄」，或 AWS Directory Service 嚴格集中控制和監控互動式使用者和群組對 Windows 執行個體的存取，並避免本機使用者權限。亦請避免使用網域管理員，而是建立更精細、以應用程式特定角色為基礎的帳戶。Just Enough Administration (JEA) 允許管理 Windows 執行個體的變更，無需互動或管理員存取。此外，JEA 可讓組織鎖定執行個體管理所需之 Windows PowerShell 命令子集的管理存取權。如需詳細資訊，請參閱 [AWS 安全最佳實務](#) 白皮書中的「管理 Amazon EC2 的作業系統存取」。
- [Windows 執行個體] 系統管理員應使用存取權限有限的 Windows 帳戶來執行日常活動，並且僅在必要時提升存取權限以執行特定組態變更。此外，只有在絕對必要時才能直接存取 Windows 執行個體。而是利用 EC2 執行命令、系統中心組態管理員 (SCCM)、Windows PowerShell DSC 或 Amazon EC2 系統管理員 (SSM) 等中央組態管理系統，將變更推送到 Windows 伺服器。
- 設定必要網路路由最少的 Amazon VPC 子網路路由表。例如，只將重新報告直接網際網路存取的 Amazon EC2 執行個體放入具有連線至網際網路閘道的子網路中，並且只將需要直接存取內部網路的 Amazon EC2 執行個體放入子網路中，並使用路由到虛擬私有閘道。
- 請考慮使用其他安全群組或網路介面，來控制和稽核 Amazon EC2 執行個體管理流量，並與一般應用程式流量分開。這種方法可讓客戶實作變更控制的特殊 IAM 政策，讓稽核安全群組規則或自動規則驗證指令碼的變更變得更輕鬆。使用多個網路介面還提供了控制網路流量的其他選項，包括建立以主機為基礎的路由原則的能力，或根據網路介面的指派子網路運用不同的 VPC 子網路路由規則。

- 使用 AWS Virtual Private Network 或 AWS Direct Connect 建立從遠端網路到 VPC 的私人連線。如需詳細資訊，請參閱 [Network-to-Amazon VPC 連線選項](#)。
- 使用 [VPC 流程日誌](#) 監控到達您執行個體的流量。
- 使用 [GuardDuty 惡意程式碼防護](#) 來識別執行個體上指示惡意軟體的可疑行為，這些行為可能會損害您的工作負載、重新利用資源以供惡意使用，以及取得未經授權的資料存取權。
- 使用 [GuardDuty 行階段監控](#) 來識別並回應執行個體的潛在威脅。如需詳細資訊，請參閱 [執行階段監控如何搭配 Amazon EC2 執行個體運作](#)。
- 使用 [AWS Security Hub](#)、可 [Reachability Analyzer](#) 或 [網路存取分析器](#) 來檢查執行個體是否有意外的網路可存取性。
- 使用 [EC2 Instance Connect](#) 連線至您的執行個體，其會使用 Secure Shell (SSH)，而不需要分享和管理 SSH 金鑰。
- 使用 [AWS Systems Manager 工作階段管理員](#) 從遠端存取執行個體，而不是開啟輸入 SSH 或 RDP 連接埠，以及管理金鑰配對。
- 使用 [AWS Systems Manager 執行命令](#) 自動執行一般管理工作，而不是連線至執行個體。
- [Windows 執行個體] 許多 Windows 作業系統角色和 Microsoft 商務應用程式也提供增強的功能，例如 IIS 內的 IP 位址範圍限制、Microsoft SQL Server 中的 TCP/IP 篩選原則，以及 Microsoft Exchange 中的連線篩選原則。應用程式層內的網路限制功能可為重要的商務應用程式伺服器提供額外的防禦層。

Amazon VPC 支援其他網路安全控制，例如閘道、代理伺服器和網路監控選項。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的 [控制網路流量](#)。

## Amazon EC2 的復原功能

AWS 全球基礎架構是圍繞區 AWS 域和可用區域建立的。區域提供多個分開且隔離的實際可用區域，並以低延遲、高輸送量和高度備援網路連線相互連結。透過可用區域，您可以設計與操作的應用程式和資料庫，在可用區域之間自動容錯移轉而不會發生中斷。可用區域的可用性、容錯能力和擴充能力，均較單一或多個資料中心的傳統基礎設施還高。

如果您需要在更大的地理距離內複製資料或應用程式，請使用 AWS Local Zones。「AWS 本地區域」是您使用者所在地理位置附近的「AWS 區域」的延伸。Local Zones 有自己的網際網路連線，並支援 AWS Direct Connect。與所有區 AWS 域一樣，AWS Local Zones 與其他 AWS 區域完全隔離。

如果您需要在 AWS 本機區域中複寫資料或應用程式，AWS 建議您使用下列其中一個區域做為容錯移轉區域：

- 另一個 Local Zone
- 區域中不是父區域的可用區域。您可以使用 [describe-availability-zones](#) 命令來檢視父區域。

如需區域和可用區域的相關 AWS 資訊，請參閱[AWS 全域基礎結構](#)。

除了 AWS 全球基礎設施之外，Amazon EC2 還提供下列功能來支援您的資料彈性：

- 跨區域複製 AMI
- 跨區域複製 EBS 快照
- 使用 Amazon Data Lifecycle Manager 自動化 EBS 後端 AMI
- 使用 Amazon Data Lifecycle Manager 自動化 EBS
- 使用 Amazon EC2 Auto Scaling 維護機群的運作狀態及可用性
- 使用 Elastic Load Balancing 將輸入流量分配到單一可用區域或多個可用區域中的多個執行個體。

## Amazon EC2 的合規驗證

若要瞭解 AWS 服務 是否屬於特定規範遵循方案的範圍內，請參閱[AWS 服務 遵循規範計劃](#)方案中的，並選擇您感興趣的合規方案。如需一般資訊，請參閱[AWS 規範計劃](#)。

您可以使用下載第三方稽核報告 AWS Artifact。如需詳細資訊，請參閱[下載中的報告中的 AWS Artifact](#)。

您在使用時的合規責任取決 AWS 服務 於資料的敏感性、公司的合規目標以及適用的法律和法規。AWS 提供下列資源以協助遵循法規：

- [安全性與合規性快速入門指南](#) — 這些部署指南討論架構考量，並提供部署以安全性和合規性 AWS 為重點的基準環境的步驟。
- [在 Amazon Web Services 上架構 HIPAA 安全性與合規性](#) — 本白皮書說明公司如何使用建立符合 HIPAA 資格的應 AWS 應用程式。

### Note

並非所有人 AWS 服務 都符合 HIPAA 資格。如需詳細資訊，請參閱 [HIPAA 資格服務參照](#)。

- [AWS 合規資源](#) — 此工作簿和指南集合可能適用於您的產業和所在地。



- [AWS 客戶合規指南](#) — 透過合規的角度瞭解共同的責任模式。這份指南總結了在多個架構 (包括美國國家標準技術研究所 (NIST)、支付卡產業安全標準委員會 (PCI) 和國際標準化組織 (ISO)) 中，保 AWS 服務 護指引並對應至安全控制的最佳做法。
- [使用 AWS Config 開發人員指南中的規則評估資源](#) — 此 AWS Config 服務會評估您的資源組態符合內部實務、產業準則和法規的程度。
- [AWS Security Hub](#) — 這 AWS 服務 提供了內部安全狀態的全面視圖 AWS。Security Hub 使用安全控制，可評估您的 AWS 資源並檢查您的法規遵循是否符合安全業界標準和最佳實務。如需支援的服務和控制清單，請參閱 [Security Hub controls reference](#)。
- [Amazon GuardDuty](#) — 透過監控環境中的 AWS 帳戶可疑和惡意活動，藉此 AWS 服務 偵測您的工作負載、容器和資料的潛在威脅。GuardDuty 可協助您因應各種合規性需求，例如 PCI DSS，滿足特定合規性架構所規定的入侵偵測需求。
- [AWS Audit Manager](#) — 這 AWS 服務 有助於您持續稽核您的 AWS 使用情況，以簡化您管理風險的方式，以及遵守法規和業界標準的方式。

## Amazon EC2 的 Identity and Access Management

您的安全登入資料可識別您的服務，AWS 並授予您無限制使用 AWS 資源 (例如 Amazon EC2 資源)。您可使用 Amazon EC2 和 AWS Identity and Access Management (IAM) 的功能來允許其他使用者、服務和應用程式使用您的 Amazon EC2 資源，但不共用您的安全憑證。您可以使用 IAM 控制其他使用者如何使用您 AWS 帳戶中的資源，也可以使用安全群組控制對 Amazon EC2 執行個體的存取。您可選擇授予 Amazon EC2 資源的完全使用或有限制使用的權限。

如需使用 IAM 保護資 AWS 源安全的最佳實務，請參閱 [IAM 中的安全性最佳實務](#)。

### 目錄

- [網路存取您的執行個體](#)
- [Amazon EC2 許可屬性](#)
- [IAM 和 Amazon EC2](#)
- [適用於 Amazon EC2 的 IAM 政策](#)
- [AWS Amazon EC2 的受管政策](#)
- [Amazon EC2 的 IAM 的角色](#)

## 網路存取您的執行個體

安全群組就像是防火牆，控制允許傳到一個以上執行個體的流量。啟動執行個體時，您會為其指派一個或多個安全群組。您須於每個安全群組新增規則，藉此控制執行個體的流量。您可以隨時修改安全群組的規則。新規則會自動套用至指派安全群組的所有執行個體。

如需詳細資訊，請參閱 [安全群組規則](#)。

## Amazon EC2 許可屬性

您的組織可能有多個 AWS 帳戶。Amazon EC2 可讓您指定可以使用 Amazon 機器映像 (AMI) 和 Amazon EBS 快照的其他 AWS 帳戶。這些權限僅適用於 AWS 帳戶層級；您無法限制指定 AWS 帳戶內特定使用者的權限。您在 AWS 帳戶內指定的所有使用者，都能夠使用 AMI 或快照。

每個 AMI 都具有 LaunchPermission 屬性，可用此控制能夠存取該 AMI 的 AWS 帳戶。如需詳細資訊，請參閱 [使 AMI 公有化](#)。

每個 Amazon EBS 快照都有一個 createVolumePermission 屬性，可控制哪些 AWS 帳戶可以使用快照。如需詳細資訊，請參閱 [Amazon EBS 使用者指南中的共用 Amazon EBS 快照](#)。

## IAM 和 Amazon EC2

IAM 可讓您執行以下項目：

- 在您的下方建立使用者和群組 AWS 帳戶
- 為您下的每個用戶分配唯一的安全憑據 AWS 帳戶
- 控制每個使用者使用 AWS 資源執行工作的權限
- 允許其他 AWS 帳戶 用戶共享您的 AWS 資源
- 為您的角色建立角色，AWS 帳戶 並定義可以擔任他們的使用者或服務
- 使用企業的現有身分識別授與使用 AWS 資源執行工作的權限

搭配 Amazon EC2 使用 IAM，您可以控制組織中的使用者是否可以使用特定的 Amazon EC2 API 動作來執行任務，以及是否可以使用特定的 AWS 資源。

本主題能夠回答下列問題：

- 如何在 IAM 內建立群組與使用者？
- 如何建立政策？

- 在 Amazon EC2 內執行任務需要什麼 IAM 政策？
- 如何授予在 Amazon EC2 內執行動作的許可？
- 如何授予在 Amazon EC2 內特定資源上執行動作的許可？

## 建立使用者、群組和角色

您可以為您的建立使用者 AWS 帳戶 和群組，然後為他們指派所需的權限。作為最佳實務，使用者應透過擔任 IAM 角色來取得許可。

[IAM 角色](#)是您可以在帳戶中建立的另一種 IAM 身分，具有特定的許可。IAM 角色與 IAM 使用者類似，因為它是具有許可政策的 AWS 身分識別，可決定身分可以執行和不能在其中執行的操作 AWS。但是，角色的目的是讓需要它的任何人可代入，而不是單獨地與某個人員關聯。此外，角色沒有與之關聯的標準長期憑證，例如密碼或存取金鑰。反之，當您擔任角色時，其會為您的角色工作階段提供臨時安全性憑證。如需如何建立 IAM 角色並向其授與許可的詳細資訊，請參閱[the section called “IAM 角色”](#)。

## 相關主題

如需 IAM 的詳細資訊，請參閱下列各項：

- [適用於 Amazon EC2 的 IAM 政策](#)
- [Amazon EC2 的 IAM 的角色](#)
- [AWS Identity and Access Management \(IAM\)](#)
- [IAM 使用者指南](#)

## 適用於 Amazon EC2 的 IAM 政策

依預設，使用者無權建立或修改 Amazon EC2 資源，或使用 Amazon EC2 API、Amazon EC2 主控台或 CLI 執行任務。若要允許使用者建立或修改資源並執行任務，您必須建立授予使用者許可的 IAM 政策，以使用他們需要的特定資源和 API 動作，然後將這些政策連接到需要這些許可的使用者、群組或 IAM 角色。

將政策連接到使用者、使用者群組或角色時，政策會允許或拒絕使用者在特定資源上執行特定任務的許可。如需 IAM 政策的一般詳細資訊，請參閱《IAM 使用者指南》中 [IAM 中的政策和許可](#)。如需管理和建立自訂 IAM 政策的詳細資訊，請參閱[管理 IAM 政策](#)。

## 入門

IAM 政策必須授予或拒絕使用一或多個 Amazon EC2 動作的許可。政策中還必須指定可用於動作的資源，可為所有資源，或在某些案例中為特定的資源。政策也可以包含您套用到資源的條件。

Amazon EC2 部分支援資源層級的許可。這表示針對某些 EC2 API 動作，您無法指定使用者允許使用該動作的資源。相反的，您必須允許使用者使用該動作的所有資源。

任務	主題
了解政策的基本結構	<a href="#">政策語法</a>
在政策內定義動作	<a href="#">Amazon EC2 動作</a>
在政策內定義特定資源	<a href="#">適用於 Amazon EC2 的 Amazon Resource Name (ARN)</a>
套用資源的使用條件	<a href="#">Amazon EC2 的條件金鑰</a>
使用 Amazon EC2 適用的資源層級可用許可	<a href="#">適用於 Amazon EC2 的動作、資源及條件金鑰</a>
測試您的政策	<a href="#">檢查使用者是否擁有必要的許可</a>
產生 IAM 政策	<a href="#">根據存取活動產生政策</a>
CLI 或軟體開發套件的範例政策	<a href="#">使用 AWS CLI 或 AWS SDK 的範例原則</a>
Amazon EC2 主控台的範例政策	<a href="#">在 Amazon EC2 主控台中進行操作的範例政策。</a>

## 向使用者、群組和角色授予許可

以下是一些 AWS 受管理政策的範例，在符合您的需求時可供使用：

- PowerUserAccess
- ReadOnlyAccess
- AmazonEC2FullAccess
- AmazonEC2ReadOnlyAccess

如需詳細資訊，請參閱 [the section called “AWS 受管理政策”](#)。

若要提供存取權，請新增權限至您的使用者、群組或角色：

- 使用者和群組位於 AWS IAM Identity Center：

建立權限合集。請按照 AWS IAM Identity Center 使用者指南 中的 [建立權限合集](#) 說明進行操作。

- 透過身分提供者在 IAM 中管理的使用者：

建立聯合身分的角色。請按照 IAM 使用者指南 的 [為第三方身分提供者 \(聯合\) 建立角色](#) 中的指示進行操作。

- IAM 使用者：

- 建立您的使用者可擔任的角色。請按照 IAM 使用者指南 的 [為 IAM 使用者建立角色](#) 中的指示進行操作。

- (不建議) 將政策直接附加至使用者，或將使用者新增至使用者群組。請遵循 IAM 使用者指南的 [新增許可到使用者 \(主控台\)](#) 中的指示。

## 政策結構

下列主題說明 IAM 政策的結構。

### 內容

- [政策語法](#)
- [Amazon EC2 動作](#)
- [Amazon EC2 API 動作支援的資源層級許可](#)
- [適用於 Amazon EC2 的 Amazon Resource Name \(ARN\)](#)
- [Amazon EC2 的條件金鑰](#)
- [檢查使用者是否擁有必要的許可](#)

### 政策語法

IAM 政策為包含一或多個陳述式的 JSON 文件。每個陳述式的結構如下所示。

```
{
 "Statement": [{
 "Effect": "effect",
 "Action": "action",
 "Resource": "arn",
```

```
 "Condition":{
 "condition":{
 "key":"value"
 }
 }
]
}
```

陳述式由各種元素組成：

- **Effect (效果)**：效果 可以是 Allow 或 Deny。根據預設，使用者沒有使用資源和 API 動作的許可，因此所有請求均會遭到拒絕。明確允許覆寫預設值。明確拒絕覆寫任何允許。
- **Action (動作)**：動作 是您授予或拒絕許可的特定 API 動作。若要了解指定動作，請參閱[Amazon EC2 動作](#)。
- **Resource (資源)**：受動作影響的資源。有些 Amazon EC2 API 動作可讓您在政策中包含可由動作建立或修改的特定資源。您可以使用 Amazon Resource Name (ARN) 來指定資源，或是使用萬用字元 (\*) 來指定陳述式套用到所有資源。如需詳細資訊，請參閱 [Amazon EC2 API 動作支援的資源層級許可](#)。
- **Condition (條件)**：條件為選擇性。您可以使用它們來控制何時政策開始生效。如需指定 Amazon EC2 條件的詳細資訊，請參閱 [Amazon EC2 的條件金鑰](#)。

如需這些欄位的詳細資訊，請參閱《IAM 使用者指南》中的 [IAM JSON 政策元素參考](#)。如需 Amazon EC2 的範例 IAM 政策陳述式，請參閱 [使用 AWS CLI 或 AWS SDK 的範例原則](#)。

## Amazon EC2 動作

在 IAM 政策陳述式中，您可以從任何支援 IAM 的服務指定任何 API 動作。針對 Amazon EC2，請在 API 動作名稱使用下列字首：ec2:。例如：ec2:RunInstances 和 ec2:CreateImage。

若要在單一陳述式中指定多個動作，請用逗號分隔，如下所示：

```
"Action": ["ec2:action1", "ec2:action2"]
```

您也可以使用萬用字元指定多個動作。例如，您可以指定名稱開頭有「Describe」文字的所有動作，如下所示：

```
"Action": "ec2:Describe*"
```

**Note**

目前，Amazon EC2 Describe\* API 動作並不支援資源層級許可。如需 Amazon EC2 資源層級許可的相關詳細資訊，請參閱 [適用於 Amazon EC2 的 IAM 政策](#)。

若要指定所有的 Amazon EC2 API 動作，請使用 \* 萬用字元，如下所示：

```
"Action": "ec2:*"
```

若要查看 Amazon EC2 動作的清單，請參閱服務授權參考中的 [Amazon EC2 定義的動作](#)。

### Amazon EC2 API 動作支援的資源層級許可

資源層級許可能夠讓您指定使用者可執行動作的資源。Amazon EC2 支援部分的資源層級許可。這表示針對特定 Amazon EC2 動作，您可以根據應滿足的條件，或允許使用者使用特定的資源，來控制使用者何時可以使用那些動作。例如，您可以授予使用者啟動執行個體的許可，但僅限特定類型，且僅能藉由特定的 AMI 來達成。

若要在 IAM 政策陳述式中指定資源，請使用它的 Amazon Resource Name (ARN)。如需指定 ARN 值的詳細資訊，請參閱 [適用於 Amazon EC2 的 Amazon Resource Name \(ARN\)](#)。如果 API 動作不支援個別 ARN，您必須使用萬用字元 (\*) 來指定該動作可以影響所有資源。

若要查看識別哪些 Amazon EC2 API 動作支援資源層級許可的表格，以及您可以在政策中使用的 ARN 和條件金鑰，請參閱 [動作、資源與條件金鑰 Amazon EC2](#)。

您可以在用於 Amazon EC2 API 動作的 IAM 政策中，套用以標籤為基礎的資源層級許可。這可讓您更有效地控制使用者可以建立、修改或使用的資源。如需詳細資訊，請參閱 [在建立期間授予標籤資源的許可](#)。

### 適用於 Amazon EC2 的 Amazon Resource Name (ARN)

每個 IAM 政策陳述式都會套用到您使用其 ARN 指定的資源。

ARN 一般的語法如下所示：

```
arn:aws:[service]:[region]:[account-id]:resourceType/resourcePath
```

### 服務

服務 (例如，ec2)。

## region

資源的區域 (例如, us-east-1)。

## account-id

AWS 帳號 ID, 不含連字號 (例如, 123456789012)。

## resourceType

資源類型 (例如, instance)。

## resourcePath

識別資源的路徑。您可以在路徑中使用 \* 萬用字元。

例如, 您可以在陳述式中使用特定執行個體 (i-1234567890abcdef0) 的 ARN 來加以指定, 如下所示。

```
"Resource": "arn:aws:ec2:us-east-1:123456789012:instance/i-1234567890abcdef0"
```

您也可以使用 \* 萬用字元指定所有屬於特定帳戶的執行個體, 如下所示。

```
"Resource": "arn:aws:ec2:us-east-1:123456789012:instance/*"
```

您也可以使用 \* 萬用字元指定屬於特定帳戶的所有 Amazon EC2 資源, 如下所示。

```
"Resource": "arn:aws:ec2:us-east-1:123456789012:*"
```

若要指定所有資源, 或如果特定的 API 動作不支援 ARN, 請在 Resource 元素中使用 \* 萬用字元, 如下所示。

```
"Resource": "*"
```

許多 Amazon EC2 API 動作都涉及多個資源。例如, AttachVolume 會將 Amazon EBS 磁碟區連接至執行個體, 所以使用者必須具備該磁碟區與執行個體的使用許可。若要在單一陳述式中指定多項資源, 請用逗號分隔其 ARN, 如下所示。

```
"Resource": ["arn1", "arn2"]
```

如需 Amazon EC2 資源的 ARN 清單, 請參閱 [由 Amazon EC2 定義的資源類型](#)。



## Amazon EC2 的條件金鑰

在政策陳述式中，您可以選擇性的指定生效時發揮控制效果的條件。每個條件都包含一或多個索引鍵/值對。條件鍵不區分大小寫。我們已定義 AWS 全域條件金鑰，以及其他服務特定條件金鑰。

如需 Amazon EC2 的服務特定條件金鑰清單，請參閱 [Amazon EC2 的條件金鑰](#)。Amazon EC2 也實作了 AWS 全域條件金鑰。如需詳細資訊，請參閱 IAM 使用者指南中的 [可用於所有請求的資訊](#)。

若要在 IAM 政策中使用條件金鑰，請使用 Condition 陳述式。例如，下列政策會授與使用者新增及移除任何安全群組之傳入和傳出規則的許可。它會使用 ec2:Vpc 條件金鑰，來指定這些動作只能在特定 VPC 中的安全群組上執行。

```
{
 "Version": "2012-10-17",
 "Statement": [{
 "Effect": "Allow",
 "Action": [
 "ec2:AuthorizeSecurityGroupIngress",
 "ec2:AuthorizeSecurityGroupEgress",
 "ec2:RevokeSecurityGroupIngress",
 "ec2:RevokeSecurityGroupEgress"],
 "Resource": "arn:aws:ec2:region:account:security-group/*",
 "Condition": {
 "StringEquals": {
 "ec2:Vpc": "arn:aws:ec2:region:account:vpc/vpc-11223344556677889"
 }
 }
]
}
```

若您指定多個條件，或在單一條件中指定多個索引鍵，我們會使用邏輯 AND 操作加以評估。若您針對單一索引鍵使用多個值指定單一條件，我們會使用邏輯 OR 操作評估條件。若要授予許可，必須符合所有的條件。

您也可以指定條件時使用預留位置。如需詳細資訊，請參閱《IAM 使用者指南》中的 [IAM 政策元素：變數和標籤](#)。

### Important

許多條件鍵均為資源專屬，而且部分 API 動作使用多個資源。若您撰寫的政策具備條件鍵，請使用陳述式的 Resource 元素來指定套用該條件鍵的資源。若未這麼做，該政策可能會導致使

用者完全無法執行該動作，因為未套用條件金鑰之資源的條件檢查會失敗。若您不希望指定資源，或者若已撰寫政策的 Action 元素來納入多個 API 動作，則您必須使用 `...IfExists` 條件類型，以確保未使用該條件金鑰的資源，忽略該條件金鑰。如需詳細資訊，請參閱 [... IfExists IAM 使用者指南中的條件](#)。

所有 Amazon EC2 操作都支援 `aws:RequestedRegion` 和 `ec2:Region` 條件索引鍵。如需詳細資訊，請參閱 [範例：限制特定區域的存取](#)。

### `ec2:SourceInstanceARN` 條件金鑰

針對發出請求的執行個體，`ec2:SourceInstanceARN` 條件金鑰可用於指定其 ARN 的條件。這是 AWS 全域條件金鑰，不是服務特定的。如需政策範例，請參閱 [Amazon EC2：將磁碟區與 EC2 執行個體連接或分開](#) 和 [範例：允許特定執行個體檢視其他 AWS 服務中的資源](#)。`ec2:SourceInstanceARN` 金鑰無法用來當做變數來填入陳述式內 Resource 元素的 ARN。

如需 Amazon EC2 的範例政策陳述式，請參閱 [使用 AWS CLI 或 AWS SDK 的範例原則](#)。

### `ec2:Attribute` 條件金鑰

`ec2:Attribute` 條件金鑰可用於按資源屬性篩選存取的條件。條件索引鍵僅支援原始資料類型 (例如字串或整數) 的屬性，或只支援 Value 屬性 (例如「描述」或「[ModifyImage屬性 API](#)」動作的 `ImdsSupport物件`) 的複雜 [AttributeValue](#) 物件。

#### Important

條件索引鍵不能用於具有多個屬性的複雜物件，例如 A [ModifyImageattribute API](#) 動作的 `LaunchPermission物件`。

例如，下列原則會使用 `ec2:Attribute/Description` 條件索引鍵，依「[ModifyImage屬性 API](#)」動作的複雜「描述」物件篩選存取。條件金鑰僅允許將映像描述修改為 `Production` 或 `Development` 的請求。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "ec2:ModifyImageAttribute",
 "Resource": "arn:aws:ec2:us-east-1::image/ami-*",
```

```

 "Condition": {
 "StringEquals": {
 "ec2:Attribute/Description": [
 "Production",
 "Development"
]
 }
 }
]
}

```

下列範例原則會使用 `ec2:Attribute` 條件索引鍵，依 `Attribute API` 動作的原始屬 `ModifyImage` 性內容篩選存取。條件金鑰會拒絕嘗試修改映像描述的所有請求。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Deny",
 "Action": "ec2:ModifyImageAttribute",
 "Resource": "arn:aws:ec2:us-east-1::image/ami-*",
 "Condition": {
 "StringEquals": {
 "ec2:Attribute": "Description"
 }
 }
 }
]
}

```

### **ec2:ResourceID** 條件索引鍵

使用以下 `ec2:ResourceID` 條件索引鍵搭配指定的 API 動作時，條件索引鍵值會用於指定由 API 動作建立之產生的資源。`ec2:ResourceID` 條件索引鍵不能用於指定 API 請求中指定的來源資源。如果使用下列其中一項 `ec2:ResourceID` 條件索引鍵搭配指定的 API，您必須一律指定萬用字元 (\*)。如果指定不同的值，則該條件在執行期期間一律會解析為 \*。例如，若要搭配 `CopyImageAPI` 使用 `ec2:ImageId` 條件索引鍵，您必須依照下列方式指定條件索引鍵：

```

{
 "Version": "2012-10-17",
 "Statement": [

```

```

{
 "Effect": "Allow",
 "Action": "ec2:CopyImage",
 "Resource": "arn:aws:ec2:us-east-1::image/ami-*",
 "Condition": {
 "StringEquals": {
 "ec2:ImageID": "*"
 }
 }
}
]
}

```

條件鍵	API 動作			
ec2:DhcpOptionsID	<ul style="list-style-type: none"> <li>CreateDhcp選項</li> </ul>			
ec2:ImageID	<ul style="list-style-type: none"> <li>CopyImage</li> <li>CreateImage</li> <li>ImportImage</li> <li>RegisterImage</li> </ul>			
ec2:InstanceID	<ul style="list-style-type: none"> <li>RunInstances</li> <li>ImportInstance</li> </ul>			
ec2:InternetGatewayID	<ul style="list-style-type: none"> <li>CreateInternet閘道</li> </ul>			

條件鍵	API 動作			
ec2:NetworkACLID	<ul style="list-style-type: none"> <li>CreateNetwork+韌帶</li> </ul>			
ec2:NetworkInterfaceID	<ul style="list-style-type: none"> <li>CreateNetwork介面</li> </ul>			
ec2:PlacementGroupName	<ul style="list-style-type: none"> <li>CreatePlacement集團</li> </ul>			
ec2:RouteTableID	<ul style="list-style-type: none"> <li>CreateRoute表</li> </ul>			
ec2:SecurityGroupID	<ul style="list-style-type: none"> <li>CreateSecurity集團</li> </ul>			
ec2:SnapshotID	<ul style="list-style-type: none"> <li>CopySnapshot</li> <li>CreateSnapshot</li> <li>CreateSnapshots</li> <li>ImportSnapshots</li> </ul>			

條件鍵	API 動作			
ec2:SubnetID	<ul style="list-style-type: none"> <li>CreateSubnet</li> </ul>			
ec2:VolumeID	<ul style="list-style-type: none"> <li>CreateVolume</li> <li>ImportVolume</li> </ul>			
ec2:VpcID	<ul style="list-style-type: none"> <li>CreateVpc</li> </ul>			
ec2:VpcPeeringConnectionID	<ul style="list-style-type: none"> <li>CreateVpcPeeringConnection</li> </ul>			

我們建議您避免使用 ec2:*Resource*ID 條件索引鍵搭配這些 API 動作。相反地，如果需要根據特定資源 ID 篩選存取權，建議您按如下方式使用 Resource 政策元素：

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "ec2:CopyImage",
 "Resource": "arn:aws:ec2:us-east-1:image/ami-01234567890abcdef"
 }
]
}
```

## 檢查使用者是否擁有必要的許可

建立 IAM 政策之後，我們建議您檢查該政策是否授予使用者許可，允許他們使用所需要的特定 API 動作和資源，再讓政策生效。

首先，建立一個用於測試的使用者，然後將建立的 IAM 政策連接到該測試使用者。接著，以測試使用者的身分提出請求。

若您要測試的 Amazon EC2 動作會建立或修改資源，您應使用 DryRun 參數進行請求 (或使用 `--dry-run` 選項執行 AWS CLI 命令)。在此案例中，呼叫會完成授權檢查，但不會完成操作。例如，您可以檢查使用者是否可以終止特定執行個體，而無須實際終止它。若測試使用者具有必要的許可，請求會傳回 `DryRunOperation`。否則，會傳回 `UnauthorizedOperation`。

如果政策未授予使用者預期的許可，或授予過多許可，您可以視需要調整政策並重新測試，直到您取得所要的結果。

### Important

政策變更的散佈可能需要幾分鐘時間才能生效。因此，我們建議您等待五分鐘，然後再測試您的政策更新。

如果授權檢查失敗，請求將傳回包含診斷資訊的編碼訊息。您可使用 `DecodeAuthorizationMessage` 動作將訊息解碼。如需詳細資訊，請參閱 AWS Security Token Service API 參考中的訊 [DecodeAuthorization](#) 息和命令參考中的 [解碼授權訊息](#)。AWS CLI

## 在建立期間授予標籤資源的許可

有些資源建立 Amazon EC2 API 動作可讓您在建立資源時指定標籤。您可以使用資源標籤來實作以屬性為基礎的控制 (ABAC)。如需詳細資訊，請參閱 [標記您的資源](#) 及 [使用資源標籤控制對 EC2 資源的存取](#)。

使用者若要在建立時標記資源，他們必須具備建立資源動作 (如 `ec2:RunInstances` 或 `ec2:CreateVolume`) 的使用許可。若標籤於資源建立動作指定，Amazon 會針對 `ec2:CreateTags` 動作執行其他授權，以確認使用者具備建立標籤的許可。因此，使用者必須同時具備使用 `ec2:CreateTags` 動作的明確許可。

在 `ec2:CreateTags` 動作的 IAM 政策定義中，搭配 `ec2:CreateAction` 條件金鑰使用 `Condition` 元素，將標記許可給與建立資源的動作。

下列範例示範一個政策，其允許使用者啟動執行個體，並在啟動期間將標籤套用至執行個體與磁碟區。使用者沒有標記現有資源的權限 (他們不能直接呼叫 `ec2:CreateTags` 動作)。

```
{
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:RunInstances"
],
 "Resource": "*"
 },
 {
 "Effect": "Allow",
 "Action": [
 "ec2:CreateTags"
],
 "Resource": "arn:aws:ec2:region:account:*/*",
 "Condition": {
 "StringEquals": {
 "ec2:CreateAction" : "RunInstances"
 }
 }
 }
]
}
```

同樣的，下列政策允許使用者建立磁碟區，並在磁碟區建立期間將標籤套用至該磁碟區。使用者沒有標記現有資源的權限 (他們不能直接呼叫 `ec2:CreateTags` 動作)。

```
{
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:CreateVolume"
],
 "Resource": "*"
 },
 {
 "Effect": "Allow",
 "Action": [
 "ec2:CreateTags"
]
 }
]
}
```



```
],
 "Resource": "arn:aws:ec2:region:account:*/*",
 "Condition": {
 "StringEquals": {
 "ec2:CreateAction" : "CreateVolume"
 }
 }
 }
]
```

只有在資源建立動作中套用了標籤時，才評估 `ec2:CreateTags` 動作。因此，在沒有標記條件的情況下，若請求中未指定標籤，則具備資源建立許可的使用者不需要使用 `ec2:CreateTags` 動作的許可。然而，若該使用者試圖建立具有標籤的資源卻未具備使用 `ec2:CreateTags` 動作的許可，則該請求會失敗。

如果在啟動範本中提供標籤，也會評估 `ec2:CreateTags` 動作。如需政策範例，請參閱 [啟動範本中的標籤](#)。

### 控制對特定標籤的存取

您可以在 IAM 政策的 `Condition` 元素中使用其他條件，來控制可套用至資源的標籤金鑰索和值。

下列條件金鑰機可與前一節中的範例搭配使用：

- `aws:RequestTag`：表示請求中必須存在特定標籤金鑰或標籤金鑰與值。請求內亦可指定其他標籤。
  - 搭配 `StringEquals` 條件運算子使用，以強制結合特定標籤金鑰與值，例如強制執行標籤 `cost-center=cc123`：

```
"StringEquals": { "aws:RequestTag/cost-center": "cc123" }
```

- 搭配 `StringLike` 條件運算子使用，以在請求中強制執行特定標籤金鑰，例如強制執行標籤金鑰 `purpose`：

```
"StringLike": { "aws:RequestTag/purpose": "*" }
```

- `aws:TagKeys`：強制執行請求中使用的標籤金鑰。
  - 搭配 `ForAllValues` 修飾詞使用，若請求內提供特定標籤金鑰，將強制加以執行 (若請求內指定標籤，則僅允許特定標籤金鑰，不允許其他標籤)。例如，允許標籤金鑰 `environment` 或 `cost-center`：

```
"ForAllValues:StringEquals": { "aws:TagKeys": ["environment","cost-center"] }
```

- 搭配 ForAnyValue 修飾詞使用，以強制要求請求內至少具有一個指定的標籤金鑰。例如，請求內必須出現至少下列標籤金鑰 environment 或 webserver 之一：

```
"ForAnyValue:StringEquals": { "aws:TagKeys": ["environment","webserver"] }
```

這些條件鍵可套用於支援標記的資源建立動作，以及 ec2:CreateTags 和 ec2>DeleteTags 動作。若要了解 Amazon EC2 API 動作是否支援標記，請參閱 [Amazon EC2 的動作、資源和條件金鑰](#)。

若要強制使用者在建立資源時指定標籤，您必須在資源建立動作內，搭配 aws:RequestTag 修飾詞使用 aws:TagKeys 條件金鑰或 ForAnyValue 條件金鑰。若使用者未針對資源建立動作指定標籤，則不會評估 ec2:CreateTags 動作。

以條件而言，條件金鑰不區分大小寫，而條件值會區分大小寫。因此，欲強制標籤鍵區分大小寫，請使用 aws:TagKeys 條件索引鍵，其中標籤鍵指定為條件值。

如需 IAM 政策的範例，請參閱 [使用 AWS CLI 或 AWS SDK 的範例原則](#)。如需多值條件的詳細資訊，請參閱 IAM 使用者指南中的 [建立可測試多個金鑰值的條件](#)。

## 使用資源標籤控制對 EC2 資源的存取

當您建立授予使用者使用 EC2 資源之許可的 IAM 政策時，您可以在政策的 Condition 元素中包含標籤資訊，以根據標籤控制存取。這稱為以屬性為基礎的存取控制 (ABAC)。ABAC 對於使用者可以修改、使用或刪除哪些資源提供了更佳的控制。如需詳細資訊，請參閱 [AWS 的 ABAC 是什麼？](#)

例如：您可以建立一個政策，允許使用者終止執行個體，但如果執行個體具有標籤 environment=production，則拒絕該動作。若要這樣做，您可以使用 aws:ResourceTag 條件金鑰，根據連接至資源的標籤允許或拒絕存取資源。

```
"StringEquals": { "aws:ResourceTag/environment": "production" }
```

若要了解 Amazon EC2 API 動作是否支援使用 aws:ResourceTag 條件金鑰控制存取，請參閱 [Amazon EC2 的動作、資源和條件金鑰](#)。由於 Describe 動作不支援資源層級許可，您必須在不同的陳述式中指定它們，無需條件。

如需 IAM 政策的範例，請參閱 [使用 AWS CLI 或 AWS SDK 的範例原則](#)。

如果您允許或拒絕使用者根據標籤存取資源，請務必考慮明確拒絕使用者將這些標籤新增至相同資源或從中移除的能力。否則，使用者可能透過修改標籤來避開您的限制，並取得資源的存取。

## 使用 AWS CLI 或 AWS SDK 的範例原則

您必須使用 IAM 政策授予使用者對於 Amazon EC2 所需的許可。以下範例顯示您可以用來控制使用者具有之 Amazon EC2 許可的政策陳述式。這些原則是針對使用 AWS CLI 或 AWS SDK 發出的要求而設計的。如需詳細資訊，請參閱《IAM 使用者指南》中的[建立 IAM 政策](#)。如需可在 Amazon EC2 主控台中使用的政策範例，請參閱[在 Amazon EC2 主控台中進行操作的範例政策](#)。如需特定 Amazon VPC 的 IAM 政策範例，請參閱 [Identity and Access Management for Amazon VPC](#)。

在下列範例中，將每個#####取代為您自己的資訊。

### 範例

- [範例：唯讀存取](#)
- [範例：限制特定區域的存取](#)
- [使用執行個體](#)
- [啟動執行個體 \(RunInstances\)](#)
- [使用 競價型執行個體](#)
- [範例：使用 預留執行個體](#)
- [範例：標籤資源](#)
- [範例：使用 IAM 角色](#)
- [範例：使用路由表](#)
- [範例：允許特定執行個體檢視其他 AWS 服務中的資源](#)
- [範例：使用啟動範本](#)
- [使用執行個體中繼資料](#)
- [使用 Amazon EBS 磁碟區和快照](#)

### 範例：唯讀存取

下列政策可授予使用者使用名稱開頭為 Describe 之所有 Amazon EC2 API 動作的許可。Resource 元素使用萬用字元，表示使用者可以為這些 API 動作指定所有資源。如果 API 動作不支援資源層級許可，也需要使用 \* 萬用字元。如需可與 Amazon EC2 API 動作搭配使用的 ARN 詳細資訊，請參閱 [Amazon EC2 的動作、資源與條件金鑰](#)。

因為根據預設會拒絕使用者使用 API 動作的許可，所以使用者沒有對資源執行任何動作的許可 (除非其他陳述式授予他們該許可)。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "ec2:Describe*",
 "Resource": "*"
 }
]
}
```

#### 範例：限制特定區域的存取

下列政策拒絕使用所有 Amazon EC2 API 動作的使用者許可，除非該區域為歐洲 (法蘭克福)。它使用全域條件金鑰 `aws:RequestedRegion`，所有 Amazon EC2 API 動作都支援它。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Deny",
 "Action": "ec2:*",
 "Resource": "*",
 "Condition": {
 "StringNotEquals": {
 "aws:RequestedRegion": "eu-central-1"
 }
 }
 }
]
}
```

或者，您可以使用條件金鑰 `ec2:Region`，它為 Amazon EC2 獨有並受所有 Amazon EC2 API 動作支援。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
```

```
 "Effect": "Deny",
 "Action": "ec2:*",
 "Resource": "*",
 "Condition": {
 "StringNotEquals": {
 "ec2:Region": "eu-central-1"
 }
 }
 }
]
```

## 使用執行個體

### 範例

- [範例：描述、啟動、停止、開始及終止所有執行個體](#)
- [範例：描述所有執行個體，並只停止、啟動及終止特定執行個體](#)

### 範例：描述、啟動、停止、開始及終止所有執行個體

下列政策可授予使用者使用在 Action 元素中指定之 API 動作的許可。Resource 元素使用 \* 萬用字元，表示使用者可以為這些 API 動作指定所有資源。如果 API 動作不支援資源層級許可，也需要使用 \* 萬用字元。如需可與 Amazon EC2 API 動作搭配使用的 ARN 詳細資訊，請參閱 [Amazon EC2 的動作、資源與條件金鑰](#)。

因為根據預設會拒絕使用者使用 API 動作的許可，所以使用者沒有使用任何其他 API 動作的許可 (除非其他陳述式授予他們該許可)。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:DescribeInstances",
 "ec2:DescribeImages",
 "ec2:DescribeKeyPairs",
 "ec2:DescribeSecurityGroups",
 "ec2:DescribeAvailabilityZones",
 "ec2:RunInstances",
 "ec2:TerminateInstances",
```

```

 "ec2:StopInstances",
 "ec2:StartInstances"
],
 "Resource": "*"
}
]
}

```

**範例：**描述所有執行個體，並只停止、啟動及終止特定執行個體

下列政策允許使用者描述所有執行個體、只開始及停止執行個體 i-1234567890abcdef0 和 i-0598c7d356eba48d7，並只終止美國東部 (維吉尼亞北部) 區域 (us-east-1) 中資源標籤為 "purpose=test" 的執行個體。

第一個陳述式的 Resource 元素使用 \* 萬用字元，表示使用者可以為該動作指定所有資源；在本例中，他們可以列出所有執行個體。如果 API 動作不支援資源層級許可 (在本例中為 ec2:DescribeInstances)，也需要使用 \* 萬用字元。如需可與 Amazon EC2 API 動作搭配使用的 ARN 詳細資訊，請參閱 [Amazon EC2 的動作、資源與條件金鑰](#)。

第二個陳述式的 StopInstances 和 StartInstances 動作使用資源層級許可。特定的執行個體在 Resource 元素中以其 ARN 表示。

第三個陳述式可讓使用者終止美國東部 (維吉尼亞北部) 區域 (us-east-1) 中屬於指定 AWS 帳戶的所有執行個體，但僅限執行個體具有標籤的執行個體 "purpose=test"。Condition 元素可限定政策陳述式生效的條件。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "ec2:DescribeInstances",
 "Resource": "*"
 },
 {
 "Effect": "Allow",
 "Action": [
 "ec2:StopInstances",
 "ec2:StartInstances"
],
 "Resource": [
 "arn:aws:ec2:us-east-1:account-id:instance/i-1234567890abcdef0",

```

```
 "arn:aws:ec2:us-east-1:account-id:instance/i-0598c7d356eba48d7"
]
},
{
 "Effect": "Allow",
 "Action": "ec2:TerminateInstances",
 "Resource": "arn:aws:ec2:us-east-1:account-id:instance/*",
 "Condition": {
 "StringEquals": {
 "aws:ResourceTag/purpose": "test"
 }
 }
}
]
}
```

## 啟動執行個體 (RunInstances)

[RunInstances](#) API 動作會啟動一或多個隨需執行個體或一或多個 Spot 執行個體。RunInstances 需要 AMI 並創建一個實例。使用者可以在請求中指定金鑰對和安全性群組。啟動至 VPC 需要子網，並會建立網路介面。從由 Amazon EBS 支援的 AMI 啟動會建立磁碟區。因此，使用者必須具有使用這些 Amazon EC2 資源的許可。您可以建立需要使用者在 RunInstances 上指定選用參數，或限制使用者使用特定參數值的政策陳述式。

如需啟動執行個體所需的資源層級許可詳細資訊，請參閱 [Amazon EC2 的動作、資源及條件金鑰](#)。

根據預設，使用者不具描述、開始、停止或終止所產生執行個體的許可。將管理所產生執行個體之許可授予使用者的一個方式，就是為每個執行個體建立特定標籤，再建立讓他們管理具有該標籤之執行個體的陳述式。如需詳細資訊，請參閱 [使用執行個體](#)。

## 資源

- [AMI](#)
- [執行個體類型](#)
- [子網](#)
- [EBS 磁碟區](#)
- [標籤](#)
- [啟動範本中的標籤](#)
- [Elastic GPU](#)

- [啟動範本](#)

## AMI

下列政策只允許使用者使用指定的 AMI `ami-9e1670f7` 和 `ami-45cf5c3c` 啟動執行個體。使用者無法使用其他 AMI 啟動執行個體 (除非其他陳述式授予使用者該許可)。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "ec2:RunInstances",
 "Resource": [
 "arn:aws:ec2:region::image/ami-9e1670f7",
 "arn:aws:ec2:region::image/ami-45cf5c3c",
 "arn:aws:ec2:region:account-id:instance/*",
 "arn:aws:ec2:region:account-id:volume/*",
 "arn:aws:ec2:region:account-id:key-pair/*",
 "arn:aws:ec2:region:account-id:security-group/*",
 "arn:aws:ec2:region:account-id:subnet/*",
 "arn:aws:ec2:region:account-id:network-interface/*"
]
 }
]
}
```

或者，下列政策允許使用者從 Amazon 或某些受信任和已驗證的合作夥伴擁有的所有 AMI 啟動執行個體。第一個陳述式的 Condition 元素會測試 `ec2:Owner` 是否為 `amazon`。使用者無法使用其他 AMI 啟動執行個體 (除非其他陳述式授予使用者該許可)。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "ec2:RunInstances",
 "Resource": [
 "arn:aws:ec2:region::image/ami-*"
],
 "Condition": {
 "StringEquals": {
```



```

 "ec2:Owner": "amazon"
 }
}
},
{
 "Effect": "Allow",
 "Action": "ec2:RunInstances",
 "Resource": [
 "arn:aws:ec2:region:account-id:instance/*",
 "arn:aws:ec2:region:account-id:subnet/*",
 "arn:aws:ec2:region:account-id:volume/*",
 "arn:aws:ec2:region:account-id:network-interface/*",
 "arn:aws:ec2:region:account-id:key-pair/*",
 "arn:aws:ec2:region:account-id:security-group/*"
]
}
]
}
}

```

## 執行個體類型

下列政策只允許使用者使用 `t2.micro` 或 `t2.small` 執行個體類型啟動執行個體，您可能會為了控制成本而這樣做。因為第一個陳述式的 `Condition` 元素會測試 `ec2:InstanceType` 是否為 `t2.micro` 或 `t2.small`，所以使用者無法啟動更大的執行個體。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "ec2:RunInstances",
 "Resource": [
 "arn:aws:ec2:region:account-id:instance/*"
],
 "Condition": {
 "StringEquals": {
 "ec2:InstanceType": ["t2.micro", "t2.small"]
 }
 }
 },
 {
 "Effect": "Allow",
 "Action": "ec2:RunInstances",

```

```

 "Resource": [
 "arn:aws:ec2:region::image/ami-*",
 "arn:aws:ec2:region:account-id:subnet/*",
 "arn:aws:ec2:region:account-id:network-interface/*",
 "arn:aws:ec2:region:account-id:volume/*",
 "arn:aws:ec2:region:account-id:key-pair/*",
 "arn:aws:ec2:region:account-id:security-group/*"
]
 }
]
}

```

或者，您可以建立政策來拒絕使用者啟動 t2.micro 和 t2.small 執行個體類型以外之任何執行個體的許可。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Deny",
 "Action": "ec2:RunInstances",
 "Resource": [
 "arn:aws:ec2:region:account-id:instance/*"
],
 "Condition": {
 "StringNotEquals": {
 "ec2:InstanceType": ["t2.micro", "t2.small"]
 }
 }
 },
 {
 "Effect": "Allow",
 "Action": "ec2:RunInstances",
 "Resource": [
 "arn:aws:ec2:region::image/ami-*",
 "arn:aws:ec2:region:account-id:network-interface/*",
 "arn:aws:ec2:region:account-id:instance/*",
 "arn:aws:ec2:region:account-id:subnet/*",
 "arn:aws:ec2:region:account-id:volume/*",
 "arn:aws:ec2:region:account-id:key-pair/*",
 "arn:aws:ec2:region:account-id:security-group/*"
]
 }
]
}

```

```
]
}
```

## 子網

下列政策只允許使用者使用指定的子網 `subnet-12345678` 啟動執行個體。群組無法將執行個體啟動至任何其他子網 (除非其他陳述式授予使用者該許可)。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "ec2:RunInstances",
 "Resource": [
 "arn:aws:ec2:region:account-id:subnet/subnet-12345678",
 "arn:aws:ec2:region:account-id:network-interface/*",
 "arn:aws:ec2:region:account-id:instance/*",
 "arn:aws:ec2:region:account-id:volume/*",
 "arn:aws:ec2:region::image/ami-*",
 "arn:aws:ec2:region:account-id:key-pair/*",
 "arn:aws:ec2:region:account-id:security-group/*"
]
 }
]
}
```

或者，您可以建立拒絕使用者將執行個體啟動至任何其他子網之許可的政策。陳述式執行此作業的方式是拒絕建立網路介面的許可，但指定子網 `subnet-12345678` 的介面除外。此拒絕會覆寫為允許將執行個體啟動至其他子網所建立的任何其他政策。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Deny",
 "Action": "ec2:RunInstances",
 "Resource": [
 "arn:aws:ec2:region:account-id:network-interface/*"
],
 "Condition": {
 "ArnNotEquals": {
```

```

 "ec2:Subnet": "arn:aws:ec2:region:account-id:subnet/subnet-12345678"
 }
}
},
{
 "Effect": "Allow",
 "Action": "ec2:RunInstances",
 "Resource": [
 "arn:aws:ec2:region::image/ami-*",
 "arn:aws:ec2:region:account-id:network-interface/*",
 "arn:aws:ec2:region:account-id:instance/*",
 "arn:aws:ec2:region:account-id:subnet/*",
 "arn:aws:ec2:region:account-id:volume/*",
 "arn:aws:ec2:region:account-id:key-pair/*",
 "arn:aws:ec2:region:account-id:security-group/*"
]
}
]
}
}

```

## EBS 磁碟區

下列政策只允許使用者在執行個體的 EBS 磁碟區已加密時啟動執行個體。使用者必須從使用加密快照建立的 AMI 啟動執行個體，以確保根磁碟區經過加密。使用者在啟動期間連接至執行個體的任何其他磁碟區也必須經過加密。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "ec2:RunInstances",
 "Resource": [
 "arn:aws:ec2:*:*:volume/*"
],
 "Condition": {
 "Bool": {
 "ec2:Encrypted": "true"
 }
 }
 },
 {
 "Effect": "Allow",

```

```

 "Action": "ec2:RunInstances",
 "Resource": [
 "arn:aws:ec2:*:*:image/ami-*",
 "arn:aws:ec2:*:*:network-interface/*",
 "arn:aws:ec2:*:*:instance/*",
 "arn:aws:ec2:*:*:subnet/*",
 "arn:aws:ec2:*:*:key-pair/*",
 "arn:aws:ec2:*:*:security-group/*"
]
 }
]
}

```

## 標籤

### 建立時為執行個體加上標籤

下列政策允許使用者在建立期間啟動執行個體並為執行個體套用標籤。針對套用標籤的資源建立動作，使用者必須具有使用 `CreateTags` 動作的許可。第二個陳述式使用 `ec2:CreateAction` 條件鍵限制使用者在 `RunInstances` 的條件下才可建立標籤，且僅限為執行個體建立。使用者無法為現有資源套用標籤，也無法使用 `RunInstances` 請求為磁碟區套用標籤。

如需詳細資訊，請參閱 [在建立期間授予標籤資源的許可](#)。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:RunInstances"
],
 "Resource": "*"
 },
 {
 "Effect": "Allow",
 "Action": [
 "ec2:CreateTags"
],
 "Resource": "arn:aws:ec2:us-east-1:account-id:instance/*",
 "Condition": {
 "StringEquals": {
 "ec2:CreateAction" : "RunInstances"
 }
 }
 }
]
}

```

```

 }
 }
}
]
}

```

使用特定標籤建立時為執行個體和磁碟區加上標籤

下列政策包含 `aws:RequestTag` 條件鍵，需要使用者使用標籤 `RunInstances` 和 `environment=production` 為 `purpose=webserver` 建立的任何執行個體和磁碟區套用標籤。如果使用者未傳遞這些特定標籤，或完全未指定標籤，請求會失敗。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:RunInstances"
],
 "Resource": [
 "arn:aws:ec2:region::image/*",
 "arn:aws:ec2:region:account-id:subnet/*",
 "arn:aws:ec2:region:account-id:network-interface/*",
 "arn:aws:ec2:region:account-id:security-group/*",
 "arn:aws:ec2:region:account-id:key-pair/*"
]
 },
 {
 "Effect": "Allow",
 "Action": [
 "ec2:RunInstances"
],
 "Resource": [
 "arn:aws:ec2:region:account-id:volume/*",
 "arn:aws:ec2:region:account-id:instance/*"
],
 "Condition": {
 "StringEquals": {
 "aws:RequestTag/environment": "production" ,
 "aws:RequestTag/purpose": "webserver"
 }
 }
 }
]
}

```

```

 },
 {
 "Effect": "Allow",
 "Action": [
 "ec2:CreateTags"
],
 "Resource": "arn:aws:ec2:region:account-id:*/*",
 "Condition": {
 "StringEquals": {
 "ec2:CreateAction" : "RunInstances"
 }
 }
 }
]
}

```

使用至少一個特定標籤建立時為執行個體和磁碟區加上標籤

下列政策在 ForAnyValue 條件的部分使用 `aws:TagKeys` 修飾詞，表示至少必須在請求中指定一個標籤，而且它必須包含 `environment` 鍵或 `webserver` 鍵。此標籤必須同時套用至執行個體和磁碟區。使用者可以在請求中指定任何標籤值。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:RunInstances"
],
 "Resource": [
 "arn:aws:ec2:region::image/*",
 "arn:aws:ec2:region:account-id:subnet/*",
 "arn:aws:ec2:region:account-id:network-interface/*",
 "arn:aws:ec2:region:account-id:security-group/*",
 "arn:aws:ec2:region:account-id:key-pair/*"
]
 },
 {
 "Effect": "Allow",
 "Action": [
 "ec2:RunInstances"
],

```

```

 "Resource": [
 "arn:aws:ec2:region:account-id:volume/*",
 "arn:aws:ec2:region:account-id:instance/*"
],
 "Condition": {
 "ForAnyValue:StringEquals": {
 "aws:TagKeys": ["environment","webserver"]
 }
 }
 },
 {
 "Effect": "Allow",
 "Action": [
 "ec2:CreateTags"
],
 "Resource": "arn:aws:ec2:region:account-id:*/*",
 "Condition": {
 "StringEquals": {
 "ec2:CreateAction" : "RunInstances"
 }
 }
 }
]
}

```

如果執行個體在建立時被標籤，則必須使用特定標籤來加以標籤

在下列政策中，使用者不需要在請求中指定標籤，但若需指定，標籤必須是 purpose=test。不允許其他標籤。使用者可以套用標籤至 RunInstances 請求中任何可套用標籤的資源。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:RunInstances"
],
 "Resource": "*"
 },
 {
 "Effect": "Allow",
 "Action": [

```



```

 "ec2:CreateTags"
],
 "Resource": "arn:aws:ec2:region:account-id:*/*",
 "Condition": {
 "StringEquals": {
 "aws:RequestTag/purpose": "test",
 "ec2:CreateAction" : "RunInstances"
 },
 "ForAllValues:StringEquals": {
 "aws:TagKeys": "purpose"
 }
 }
}
]
}

```

### 禁止任何人在創建時調用標籤 RunInstances

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "AllowRun",
 "Effect": "Allow",
 "Action": [
 "ec2:RunInstances"
],
 "Resource": [
 "arn:aws:ec2:us-east-1:image/*",
 "arn:aws:ec2:us-east-1:*:subnet/*",
 "arn:aws:ec2:us-east-1:*:network-interface/*",
 "arn:aws:ec2:us-east-1:*:security-group/*",
 "arn:aws:ec2:us-east-1:*:key-pair/*",
 "arn:aws:ec2:us-east-1:*:volume/*",
 "arn:aws:ec2:us-east-1:*:instance/*",
 "arn:aws:ec2:us-east-1:*:spot-instances-request/*"
]
 },
 {
 "Sid": "VisualEditor0",
 "Effect": "Deny",
 "Action": "ec2:CreateTags",
 "Resource": "*"
 }
]
}

```

```

 }
]
}

```

僅允許的特定標籤 `spot-instances-request`。不一致的數字 2 會意外在此發揮作用。在一般情況下，不指定任何標籤會導致「未驗證」狀態。在的情況下 `spot-instances-request`，如果沒有標籤，則不會評估此原則，因此非標 `spot-instances-request` 籤 Spot 執行時要求將會成功。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "AllowRun",
 "Effect": "Allow",
 "Action": [
 "ec2:RunInstances"
],
 "Resource": [
 "arn:aws:ec2:us-east-1::image/*",
 "arn:aws:ec2:us-east-1:*:subnet/*",
 "arn:aws:ec2:us-east-1:*:network-interface/*",
 "arn:aws:ec2:us-east-1:*:security-group/*",
 "arn:aws:ec2:us-east-1:*:key-pair/*",
 "arn:aws:ec2:us-east-1:*:volume/*",
 "arn:aws:ec2:us-east-1:*:instance/*",
]
 },
 {
 "Sid": "VisualEditor0",
 "Effect": "Allow",
 "Action": "ec2:RunInstances",
 "Resource": "arn:aws:ec2:us-east-1:*:spot-instances-request/*",
 "Condition": {
 "StringEquals": {
 "aws:RequestTag/environment": "production"
 }
 }
 }
]
}

```

## 啟動範本中的標籤

在下列範例中，使用者可以啟動執行個體，但前提是他們要使用特定的啟動範本 (lt-09477bcd97b0d310e)。ec2:IsLaunchTemplateResource 條件鍵可防止使用者覆寫在啟動範本中指定的任何資源。陳述式的第二部分允許使用者在建立時為執行個體套用標籤 – 如果在啟動範本中指定執行個體的標籤，則需要陳述式的這個部分。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "ec2:RunInstances",
 "Resource": "*",
 "Condition": {
 "ArnLike": {
 "ec2:LaunchTemplate": "arn:aws:ec2:region:account-id:launch-template/lt-09477bcd97b0d310e"
 },
 "Bool": {
 "ec2:IsLaunchTemplateResource": "true"
 }
 }
 },
 {
 "Effect": "Allow",
 "Action": [
 "ec2:CreateTags"
],
 "Resource": "arn:aws:ec2:region:account-id:instance/*",
 "Condition": {
 "StringEquals": {
 "ec2:CreateAction" : "RunInstances"
 }
 }
 }
]
}
```

## Elastic GPU

在下列政策中，使用者可以啟動執行個體，並指定要連接至執行個體的 Elastic GPU。使用者可以在任何區域中啟動執行個體，但他們只可在 us-east-2 區域中於啟動期間連接彈性 GPU。

ec2:ElasticGpuType 條件索引鍵可確保執行個體使用 eg1.medium 或 eg1.large 彈性 GPU 類型。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:RunInstances"
],
 "Resource": [
 "arn:aws:ec2:*:account-id:elastic-gpu/*"
],
 "Condition": {
 "StringEquals": {
 "ec2:Region": "us-east-2",
 "ec2:ElasticGpuType": [
 "eg1.medium",
 "eg1.large"
]
 }
 }
 },
 {
 "Effect": "Allow",
 "Action": "ec2:RunInstances",
 "Resource": [
 "arn:aws:ec2:*::image/ami-*",
 "arn:aws:ec2:*:account-id:network-interface/*",
 "arn:aws:ec2:*:account-id:instance/*",
 "arn:aws:ec2:*:account-id:subnet/*",
 "arn:aws:ec2:*:account-id:volume/*",
 "arn:aws:ec2:*:account-id:key-pair/*",
 "arn:aws:ec2:*:account-id:security-group/*"
]
 }
]
}
```

## 啟動範本

在下列範例中，使用者可以啟動執行個體，但前提是他們要使用特定的啟動範本 (lt-09477bcd97b0d310e)。使用者可以透過在 RunInstances 動作中指定參數，來覆寫啟動範本中的任何參數。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "ec2:RunInstances",
 "Resource": "*",
 "Condition": {
 "ArnLike": {
 "ec2:LaunchTemplate": "arn:aws:ec2:region:account-id:launch-template/lt-09477bcd97b0d310e"
 }
 }
 }
]
}
```

在此範例中，使用者只有在使用啟動範本時才能啟動執行個體。此政策會使用 ec2:IsLaunchTemplateResource 條件索引鍵，以防止使用者覆寫啟動範本中預先存在的任何 ARN。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "ec2:RunInstances",
 "Resource": "*",
 "Condition": {
 "ArnLike": {
 "ec2:LaunchTemplate": "arn:aws:ec2:region:account-id:launch-template/*"
 },
 "Bool": {
 "ec2:IsLaunchTemplateResource": "true"
 }
 }
 }
]
}
```

```

 }
]
}

```

下列政策範例允許使用者啟動執行個體，但前提是他們要使用啟動範本。使用者無法覆寫請求中的子網和網路介面參數；這些參數只能在啟動範本中予以指定。陳述式的第一部分會使用 [NotResource](#) 元素來允許除子網路和網路介面以外的所有其他資源。陳述式的第二部分允許子網和網路介面資源，但前提是它們來自啟動範本。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "ec2:RunInstances",
 "NotResource": ["arn:aws:ec2:region:account-id:subnet/*",
 "arn:aws:ec2:region:account-id:network-interface/*"],
 "Condition": {
 "ArnLike": {
 "ec2:LaunchTemplate": "arn:aws:ec2:region:account-id:launch-template/*"
 }
 }
 },
 {
 "Effect": "Allow",
 "Action": "ec2:RunInstances",
 "Resource": ["arn:aws:ec2:region:account-id:subnet/*",
 "arn:aws:ec2:region:account-id:network-interface/*"],
 "Condition": {
 "ArnLike": {
 "ec2:LaunchTemplate": "arn:aws:ec2:region:account-id:launch-template/*"
 },
 "Bool": {
 "ec2:IsLaunchTemplateResource": "true"
 }
 }
 }
]
}

```

下列範例只允許使用者在使用啟動範本且該啟動範本具有標籤 `Purpose=Webserver` 時，啟動執行個體。使用者無法覆寫 `RunInstances` 動作中的任何啟動範本參數。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "ec2:RunInstances",
 "NotResource": "arn:aws:ec2:region:account-id:launch-template/*",
 "Condition": {
 "ArnLike": {
 "ec2:LaunchTemplate": "arn:aws:ec2:region:account-id:launch-template/*"
 },
 "Bool": {
 "ec2:IsLaunchTemplateResource": "true"
 }
 }
 },
 {
 "Effect": "Allow",
 "Action": "ec2:RunInstances",
 "Resource": "arn:aws:ec2:region:account-id:launch-template/*",
 "Condition": {
 "StringEquals": {
 "aws:ResourceTag/Purpose": "Webservers"
 }
 }
 }
]
}
```

## 使用 競價型執行個體

您可以使用此 RunInstances 動作建立競價型執行個體請求，並在建立時標記競價型執行個體請求。要為其指定的資源 RunInstances 為 spot-instances-request。

spot-instances-request 資源會在 IAM 政策中進行評估，如下所示：

- 如果您在建立時未標記競價型執行個體請求，Amazon EC2 不會評估 RunInstances 陳述式中的 spot-instances-request 資源。
- 如果您在建立時標記競價型執行個體請求，Amazon EC2 會評估 RunInstances 陳述式中的 spot-instances-request 資源。

因此，對於 `spot-instances-request` 資源，下列規則適用於 IAM 政策：

- 如果您使 `RunInstances` 用建立競價型執行個體請求，且不打算在建立時標記競價型執行個體請求，則不需要明確允許 `spot-instances-request` 資源；呼叫將會成功。
- 如果您使 `RunInstances` 用建立競價型執行個體請求並打算在建立時標記競價型執行個體請求，則必須在 `RunInstances allow` 陳述式中包含 `spot-instances-request` 資源，否則呼叫將會失敗。
- 如果您使用 `RunInstances` 建立競價型執行個體請求並打算在建立時標記競價型執行個體請求，則必須在 `CreateTags allow` 陳述式中指定 `spot-instances-request` 資源或 \* 萬用字元，否則呼叫將會失敗。

您可以使用 `RunInstances` 或請求 `Spot` 執行個體 `RequestSpotInstances`。下列 IAM 政策範例僅適用於請求 `Spot` 執行個體使用 `RunInstances`。

範例：請求 `Spot` 執行個體 `RunInstances`

下列原則允許使用者使用動作來請求 `Spot` 執 `RunInstances` 行個體。由 `RunInstances` 建立的 `spot-instances-request` 資源請求 `Spot` 執行個體。

#### Note

若要用 `RunInstances` 來建立競價型執行個體請求，如果您不打算在建立時標記 `Spot` 執行個體請求，則可以從 `Resource` 清單 `spot-instances-request` 中省略。這是因為如果競價型執行個體請求未在建立時標記，Amazon EC2 不會評估 `RunInstances` 陳述式中的 `spot-instances-request` 資源。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "AllowRun",
 "Effect": "Allow",
 "Action": [
 "ec2:RunInstances"
],
 "Resource": [
 "arn:aws:ec2:us-east-1:image/*",
 "arn:aws:ec2:us-east-1:*:subnet/*",
 "arn:aws:ec2:us-east-1:*:network-interface/*",
```



```

 "arn:aws:ec2:us-east-1:*:security-group/*",
 "arn:aws:ec2:us-east-1:*:key-pair/*",
 "arn:aws:ec2:us-east-1:*:volume/*",
 "arn:aws:ec2:us-east-1:*:instance/*",
 "arn:aws:ec2:us-east-1:*:spot-instances-request/*"
]
}
]
}

```

### ⚠ Warning

不支援 — 範例：拒絕使用者請求 Spot 執行個體使用的權限 RunInstances spot-instances-request 資源不支援下列政策。

下列政策旨在提供使用者啟動 隨需執行個體 的許可，但拒絕使用者請求 競價型執行個體 的許可。由 RunInstances 建立的 spot-instances-request 資源是請求 Spot 執行個體的資源。第二個陳述式是為了拒絕 spot-instances-request 資源的 RunInstances 動作。但是，不支援此條件，因為如果競價型執行個體請求在建立時未標記，Amazon EC2 不會評估 RunInstances 陳述式中的 spot-instances-request 資源。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "AllowRun",
 "Effect": "Allow",
 "Action": [
 "ec2:RunInstances"
],
 "Resource": [
 "arn:aws:ec2:us-east-1::image/*",
 "arn:aws:ec2:us-east-1:*:subnet/*",
 "arn:aws:ec2:us-east-1:*:network-interface/*",
 "arn:aws:ec2:us-east-1:*:security-group/*",
 "arn:aws:ec2:us-east-1:*:key-pair/*",
 "arn:aws:ec2:us-east-1:*:volume/*",
 "arn:aws:ec2:us-east-1:*:instance/*"
]
 },
 {
 "Sid": "DenySpotInstancesRequests - NOT SUPPORTED - DO NOT USE!",
 "Effect": "Deny",

```

```

 "Action": "ec2:RunInstances",
 "Resource": "arn:aws:ec2:us-east-1:*:spot-instances-request/*"
 }
]
}

```

### 範例：在建立時標記 Spot 執行個體請求

下列政策可讓使用者標記執行個體啟動期間建立的所有資源。第一個語句允 RunInstances 許創建列出的資源。由 RunInstances 建立的 spot-instances-request 資源是請求 Spot 執行個體的資源。第二個陳述式提供 \* 萬用字元，允許在執行個體啟動時建立所有資源時加上標記。

#### Note

如果您在建立時標記競價型執行個體請求，Amazon EC2 會評估 RunInstances 陳述式中的 spot-instances-request 資源。因此，您必須明確允許 RunInstances 動作的 spot-instances-request 資源，否則呼叫將失敗。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "AllowRun",
 "Effect": "Allow",
 "Action": [
 "ec2:RunInstances"
],
 "Resource": [
 "arn:aws:ec2:us-east-1::image/*",
 "arn:aws:ec2:us-east-1:*:subnet/*",
 "arn:aws:ec2:us-east-1:*:network-interface/*",
 "arn:aws:ec2:us-east-1:*:security-group/*",
 "arn:aws:ec2:us-east-1:*:key-pair/*",
 "arn:aws:ec2:us-east-1:*:volume/*",
 "arn:aws:ec2:us-east-1:*:instance/*",
 "arn:aws:ec2:us-east-1:*:spot-instances-request/*"
]
 },
 {

```

```

 "Sid": "TagResources",
 "Effect": "Allow",
 "Action": "ec2:CreateTags",
 "Resource": "*"
 }
]
}

```

### 範例：拒絕在建立時標記 Spot 執行個體請求

下列政策拒絕使用者標記執行個體啟動期間建立的資源的許可。

第一個語句允 RunInstances 許創建列出的資源。由 RunInstances 建立的 spot-instances-request 資源是請求 Spot 執行個體的資源。第二個陳述式會提供 \* 萬用字元，以拒絕在執行個體啟動時建立所有標記的資源。如果 spot-instances-request 或任何其他資源在創建時被標記，則 RunInstances 調用將失敗。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "AllowRun",
 "Effect": "Allow",
 "Action": [
 "ec2:RunInstances"
],
 "Resource": [
 "arn:aws:ec2:us-east-1::image/*",
 "arn:aws:ec2:us-east-1:*:subnet/*",
 "arn:aws:ec2:us-east-1:*:network-interface/*",
 "arn:aws:ec2:us-east-1:*:security-group/*",
 "arn:aws:ec2:us-east-1:*:key-pair/*",
 "arn:aws:ec2:us-east-1:*:volume/*",
 "arn:aws:ec2:us-east-1:*:instance/*",
 "arn:aws:ec2:us-east-1:*:spot-instances-request/*"
]
 },
 {
 "Sid": "DenyTagResources",
 "Effect": "Deny",
 "Action": "ec2:CreateTags",
 "Resource": "*"
 }
]
}

```

```
]
}
```

### Warning

不支援 - 範例：只有系統指派特定標籤時，才允許建立 Spot 執行個體請求 `spot-instances-request` 資源不支援下列政策。

以下政策旨在僅在請求使 `RunInstances` 用特定標籤標記時，才授予建立競價型執行個體請求的權限。

第一個語句允 `RunInstances` 許創建列出的資源。

第二個陳述式向使用者授予許可，以便僅在請求具有 `environment=production` 標籤時才建立 Spot 執行個體請求。如果此條件套用至由建立的其他資源 `RunInstances`，則不指定任何標籤會導致 `Unauthenticated` 誤。但是，如果沒有為競價型執行個體請求指定標籤，Amazon EC2 不會評估 `RunInstances` 陳述式中的 `spot-instances-request` 資源，這會導致由 `RunInstances` 建立未標記的 Spot 執行個體請求。

請注意，指定其他標籤會 `environment=production` 導致 `Unauthenticated` 錯誤，因為如果使用者標記競價型執行個體請求，Amazon EC2 會評估 `RunInstances` 陳述式中的 `spot-instances-request` 資源。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "AllowRun",
 "Effect": "Allow",
 "Action": [
 "ec2:RunInstances"
],
 "Resource": [
 "arn:aws:ec2:us-east-1::image/*",
 "arn:aws:ec2:us-east-1::*:subnet/*",
 "arn:aws:ec2:us-east-1::*:network-interface/*",
 "arn:aws:ec2:us-east-1::*:security-group/*",
 "arn:aws:ec2:us-east-1::*:key-pair/*",
 "arn:aws:ec2:us-east-1::*:volume/*",
 "arn:aws:ec2:us-east-1::*:instance/*"
]
 }
],
}
```

```

 "Sid": "RequestSpotInstancesOnlyIfTagIs_environment=production - NOT
 SUPPORTED - DO NOT USE!",
 "Effect": "Allow",
 "Action": "ec2:RunInstances",
 "Resource": "arn:aws:ec2:us-east-1:*:spot-instances-request/*",
 "Condition": {
 "StringEquals": {
 "aws:RequestTag/environment": "production"
 }
 }
 },
 {
 "Sid": "TagResources",
 "Effect": "Allow",
 "Action": "ec2:CreateTags",
 "Resource": "*"
 }
]
}

```

**範例：**如果系統指派特定標籤，則拒絕建立 Spot 執行個體請求

如果請求標記為，下列政策會拒絕 RunInstances 建立競價型執行個體請求的權限。environment=production

第一個語句允 RunInstances 許創建列出的資源。

如果請求具有 environment=production 標籤，第二個陳述式會拒絕使用者建立 Spot 執行個體請求的許可。指定 environment=production 為標籤會導致 Unauthenticated 錯誤。指定其他標籤或不指定標籤將導致建立 Spot 執行個體請求。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "AllowRun",
 "Effect": "Allow",
 "Action": [
 "ec2:RunInstances"
],
 }
],
}

```

```

 "Resource": [
 "arn:aws:ec2:us-east-1:image/*",
 "arn:aws:ec2:us-east-1:*:subnet/*",
 "arn:aws:ec2:us-east-1:*:network-interface/*",
 "arn:aws:ec2:us-east-1:*:security-group/*",
 "arn:aws:ec2:us-east-1:*:key-pair/*",
 "arn:aws:ec2:us-east-1:*:volume/*",
 "arn:aws:ec2:us-east-1:*:instance/*",
 "arn:aws:ec2:us-east-1:*:spot-instances-request/*"
]
 },
 {
 "Sid": "DenySpotInstancesRequests",
 "Effect": "Deny",
 "Action": "ec2:RunInstances",
 "Resource": "arn:aws:ec2:us-east-1:*:spot-instances-request/*",
 "Condition": {
 "StringEquals": {
 "aws:RequestTag/environment": "production"
 }
 }
 },
 {
 "Sid": "TagResources",
 "Effect": "Allow",
 "Action": "ec2:CreateTags",
 "Resource": "*"
 }
]
}

```

### 範例：使用 預留執行個體

下列政策可提供使用者檢視、修改及購買您帳戶中 預留執行個體 的許可。

您無法針對個別的 預留執行個體 設定資源層級的權限。此政策表示使用者可存取帳戶中的所有 預留執行個體。

Resource 元素使用 \* 萬用字元，表示使用者可以為此動作指定所有資源；在本例中，他們可以列出及修改帳戶中的所有預留執行個體。他們也可以使用帳戶登入資料購買 預留執行個體。如果 API 動作不支援資源層級許可，也需要使用 \* 萬用字元。

```
{
```

```

"Version": "2012-10-17",
"Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:DescribeReservedInstances",
 "ec2:ModifyReservedInstances",
 "ec2:PurchaseReservedInstancesOffering",
 "ec2:DescribeAvailabilityZones",
 "ec2:DescribeReservedInstancesOfferings"
],
 "Resource": "*"
 }
]
}

```

用以允許使用者檢視及修改您帳戶中的預留執行個體，但不允許購買新的預留執行個體。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:DescribeReservedInstances",
 "ec2:ModifyReservedInstances",
 "ec2:DescribeAvailabilityZones"
],
 "Resource": "*"
 }
]
}

```

### 範例：標籤資源

下列政策只允許使用者在標籤包含 `CreateTags` 鍵和 `environment` 值時，使用 `production` 動作將標籤套用至執行個體。不允許其他標籤，並且使用者無法將標籤用於任何其他資源類型。

```

{
 "Version": "2012-10-17",
 "Statement": [

```

```

 {
 "Effect": "Allow",
 "Action": [
 "ec2:CreateTags"
],
 "Resource": "arn:aws:ec2:region:account-id:instance/*",
 "Condition": {
 "StringEquals": {
 "aws:RequestTag/environment": "production"
 }
 }
 }
]
}

```

下列政策允許使用者套用標籤至任何已有 `owner` 金鑰和使用者名稱值之標籤的可套用標籤資源。此外，使用者也必須在請求中使用 `anycompany:environment-type` 鍵和 `test` 值或 `prod` 值指定標籤。使用者可以在請求中指定其他標籤。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:CreateTags"
],
 "Resource": "arn:aws:ec2:region:account-id:*/*",
 "Condition": {
 "StringEquals": {
 "aws:RequestTag/anycompany:environment-type": ["test", "prod"],
 "aws:ResourceTag/owner": "${aws:username}"
 }
 }
 }
]
}

```

您可以建立允許使用者為資源刪除特定標籤的 IAM 政策。例如，下列政策允許使用者在請求中指定的標籤鍵為 `environment` 或 `cost-center` 時，刪除磁碟區的標籤。您可以為標籤指定任何值，但標籤鍵必須符合指定的任一鍵。



**Note**

如果您刪除資源，也會刪除與該資源相關聯的所有標籤。使用者不需要使用 `ec2:DeleteTags` 動作的許可，也能刪除具有標籤的資源；他們只需要執行刪除動作的許可。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "ec2:DeleteTags",
 "Resource": "arn:aws:ec2:us-east-1:account-id:volume/*",
 "Condition": {
 "ForAllValues:StringEquals": {
 "aws:TagKeys": ["environment", "cost-center"]
 }
 }
 }
]
}
```

此政策只允許使用者刪除任何資源上的 `environment=prod` 標籤，而且僅限資源已套用 `owner` 金鑰和使用者名稱值之標籤的情況。使用者無法刪除資源的任何其他標籤。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:DeleteTags"
],
 "Resource": "arn:aws:ec2:region:account-id:*/*",
 "Condition": {
 "StringEquals": {
 "aws:RequestTag/environment": "prod",
 "aws:ResourceTag/owner": "${aws:username}"
 },
 "ForAllValues:StringEquals": {
```

```
 "aws:TagKeys": ["environment"]
 }
}
]
```

### 範例：使用 IAM 角色

下列政策允許使用者對具有標籤 `department=test` 的執行個體連接、取代及分離 IAM 角色。取代或分離 IAM 角色需要關聯 ID，因此政策也會授予使用者使用 `ec2:DescribeIamInstanceProfileAssociations` 動作的許可。

使用者必須具有使用 `iam:PassRole` 動作的許可，才能將角色傳遞給執行個體。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:AssociateIamInstanceProfile",
 "ec2:ReplaceIamInstanceProfileAssociation",
 "ec2:DisassociateIamInstanceProfile"
],
 "Resource": "arn:aws:ec2:us-east-1:account-id:instance/*",
 "Condition": {
 "StringEquals": {
 "aws:ResourceTag/department": "test"
 }
 }
 },
 {
 "Effect": "Allow",
 "Action": "ec2:DescribeIamInstanceProfileAssociations",
 "Resource": "*"
 },
 {
 "Effect": "Allow",
 "Action": "iam:PassRole",
 "Resource": "arn:aws:iam::account-id:role/DevTeam*"
 }
]
}
```

```
}
```

下列政策允許使用者對任何執行個體連接或取代 IAM 角色。使用者只能連接或取代名稱開頭為 `TestRole-` 的 IAM 角色。針對 `iam:PassRole` 動作，請確定您指定 IAM 角色的名稱，而不是執行個體描述檔 (如果名稱不同)。如需詳細資訊，請參閱 [執行個體描述檔](#)。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:AssociateIamInstanceProfile",
 "ec2:ReplaceIamInstanceProfileAssociation"
],
 "Resource": "*"
 },
 {
 "Effect": "Allow",
 "Action": "ec2:DescribeIamInstanceProfileAssociations",
 "Resource": "*"
 },
 {
 "Effect": "Allow",
 "Action": "iam:PassRole",
 "Resource": "arn:aws:iam::account-id:role/TestRole-*"
 }
]
}
```

#### 範例：使用路由表

下列政策允許使用者新增、移除及取代只與 VPC `vpc-ec43eb89` 相關聯之路由表的路由。若要針對 `ec2:Vpc` 條件鍵指定 VPC，您必須指定 VPC 的完整 ARN。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:DeleteRoute",
```

```

 "ec2:CreateRoute",
 "ec2:ReplaceRoute"
],
 "Resource": [
 "arn:aws:ec2:region:account-id:route-table/*"
],
 "Condition": {
 "StringEquals": {
 "ec2:Vpc": "arn:aws:ec2:region:account-id:vpc/vpc-ec43eb89"
 }
 }
}
]
}

```

範例：允許特定執行個體檢視其他 AWS 服務中的資源

下列是您可連接至 IAM 角色的政策範例。此原則可讓執行個體檢視各種 AWS 服務中的資源。它使用 `ec2:SourceInstanceARN` 條件鍵指定發出請求的執行個體必須是執行個體 `i-093452212644b0dd6`。如果同一個 IAM 角色與另一個執行個體相關聯，其他執行個體就無法執行這些動作。

`ec2:SourceInstanceARN` 金鑰是 AWS 全域條件金鑰，因此可用於其他服務動作，而不僅僅是 Amazon EC2。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:DescribeVolumes",
 "s3:ListAllMyBuckets",
 "dynamodb:ListTables",
 "rds:DescribeDBInstances"
],
 "Resource": [
 "*"
],
 "Condition": {
 "ArnEquals": {
 "ec2:SourceInstanceARN": "arn:aws:ec2:region:account-id:instance/i-093452212644b0dd6"
 }
 }
 }
]
}

```

```

 }
 }
}
]
}

```

### 範例：使用啟動範本

下列政策允許使用者建立啟動範本版本並修改啟動範本，但僅限特定啟動範本 (lt-09477bcd97b0d3abc)。使用者無法使用其他啟動範本。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Action": [
 "ec2:CreateLaunchTemplateVersion",
 "ec2:ModifyLaunchTemplate"
],
 "Effect": "Allow",
 "Resource": "arn:aws:ec2:region:account-id:launch-template/lt-09477bcd97b0d3abc"
 }
]
}

```

下列政策允許使用者刪除任何啟動範本和啟動範本版本，但前提是啟動範本具有標籤 Purpose=Testing。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Action": [
 "ec2>DeleteLaunchTemplate",
 "ec2>DeleteLaunchTemplateVersions"
],
 "Effect": "Allow",
 "Resource": "arn:aws:ec2:region:account-id:launch-template/*",
 "Condition": {
 "StringEquals": {
 "aws:ResourceTag/Purpose": "Testing"
 }
 }
 }
]
}

```

```
 }
 }
]
}
```

## 使用執行個體中繼資料

下列政策可確保使用者只能使用 執行個體中繼資料服務第 2 版 (IMDSv2) 擷取 [執行個體中繼資料](#)。您可以將以下四個政策合併為一個包含四個陳述式的政策。當合併為一個政策時，您可以將政策用作為服務控制政策 (SCP)。它同樣可用作為套用至現有 IAM 政策 (取消和限制現有許可) 的拒絕政策，或者用作為跨帳戶、組織單位 (OU) 或整個組織全域套用的 SCP。

### Note

下列中 RunInstances 繼資料選項原則必須與授予主體權限的原則搭配使用，以啟動執行個體 RunInstances。如果主參與者沒有 RunInstances 權限，就無法啟動執行個體。如需詳細資訊，請參閱[使用執行個體](#)和[啟動執行個體 \(RunInstances\)](#)中的政策。

### Important

如果您使用 Auto Scaling 群組，而且需要在所有新執行個體上要求使用 IMDSv2，您的 Auto Scaling 群組必須使用啟動範本。

當 Auto Scaling 群組使用啟動範本時，會在建立新的 Auto Scaling 群組時，檢查 IAM 主體的 ec2:RunInstances 許可。當現有的 Auto Scaling 群組更新為使用新的啟動範本或新版本的啟動範本時，也會檢查這些許可。

只有在建立或更新使用啟動範本的 Auto Scaling 群組時，才會對 RunInstances 檢查 IAM 主體使用 IMDSv1 的限制。對於設為使用 Latest or Default 啟動範本的 Auto Scaling 群組，建立新版本的啟動範本時，不會檢查許可。對於要檢查的許可，您必須將 Auto Scaling 群組設為使用特定版本的啟動範本。

若要在 Auto Scaling 群組啟動的執行個體上強制使用 IMDSv2，需要以下額外步驟：

1. 藉由為建立的新委託人使用服務控制政策 (SCP) 或 IAM 許可界限，對組織中的所有帳戶停用啟動組態。對於具有 Auto Scaling 群組許可的 IAM 主體，使用此條件金鑰更新其相關的政策。若要停用啟動組態，請使用 "autoscaling:LaunchConfigurationName" 條件金鑰 (其值指定為 null)，建立或修改相關的 SCP、許可界限或 IAM 政策。
2. 對於新的啟動範本，請在啟動範本中設定執行個體中繼資料選項。對於現有的啟動範本，請建立新版本的啟動範本，並在新版本中設定執行個體中繼資料選項。

3. 在授予任何委託人許可使用啟動範本的政策中，藉由指定 "autoscaling:LaunchTemplateVersionSpecified": "true" 來限制 \$latest 和 \$default 的關聯。藉由限制使用特定版本的啟動範本，可以確保使用已設定執行個體中繼資料選項的版本來啟動新執行個體。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling API 參考中的 [LaunchTemplate規格](#)，特別是Version參數。
4. 對於使用啟動組態的 Auto Scaling 群組，將啟動組態取代為啟動範本。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南中的 [將啟動組態取代為啟動範本](#)。
5. 對於使用啟動範本的 Auto Scaling 群組，請確保其將新啟動範本與設定的執行個體中繼資料選項搭配使用，或將新版本的目前啟動範本與設定的執行個體中繼資料選項搭配使用。若要取得更多資訊，請參閱《指AWS CLI 令參考》 [update-auto-scaling-group](#) 中的。

## 範例

- [需要使用 IMDSv2](#)
- [拒絕選擇退出 IMDSv2](#)
- [指定最大跳轉限制](#)
- [限制可修改執行個體中繼資料選項的人員](#)
- [需要從 IMDSv2 擷取角色登入資料](#)

## 需要使用 IMDSv2

下列原則指定您無法呼叫 RunInstances API，除非執行個體也選擇加入要求使用 IMDSv2 (以表示)。"ec2:MetadataHttpTokens": "required" 如果您未指定執行個體需要 ImDSv2，則在呼叫 API 時會收到 UnauthorizedOperation 錯誤訊息。RunInstances

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "RequireImdsV2",
 "Effect": "Deny",
 "Action": "ec2:RunInstances",
 "Resource": "arn:aws:ec2:*:*:instance/*",
 "Condition": {
 "StringNotEquals": {
 "ec2:MetadataHttpTokens": "required"
 }
 }
 }
]
}
```

```

 }
 }
]
}

```

## 拒絕選擇退出 IMDSv2

下列政策規定您無法呼叫 `ModifyInstanceMetadataOptions` API，也無法允許 IMDSv1 或 IMDSv2 選項。如果您呼叫 `ModifyInstanceMetadataOptions` API，則必須將 `HttpTokens` 屬性設定為 `required`。

```

{
 "Version": "2012-10-17",
 "Statement": [{
 "Sid": "DenyIMDSv1HttpTokensModification",
 "Effect": "Deny",
 "Action": "ec2:ModifyInstanceMetadataOptions",
 "Resource": "arn:aws:ec2:*:*:instance/*",
 "Condition": {
 "StringNotEquals": {
 "ec2:Attribute/HttpTokens": "required"
 },
 "Null": {
 "ec2:Attribute/HttpTokens": false
 }
 }
 }]
}

```

## 指定最大跳轉限制

下列原則指定除非您同時指定躍點限制，且躍點限制不能超過 3，否則您無法呼叫 `RunInstances` API。如果你沒有這樣做，你會得到一個 `UnauthorizedOperation` 錯誤，當你調用 `RunInstances` API。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "MaxImdsHopLimit",
 "Effect": "Deny",
 "Action": "ec2:RunInstances",

```



```

 "Resource": "arn:aws:ec2:*:*:instance/*",
 "Condition": {
 "NumericGreaterThan": {
 "ec2:MetadataHttpPutResponseHopLimit": "3"
 }
 }
 }
]
}

```

### 限制可修改執行個體中繼資料選項的人員

下列政策只允許具有角色 `ec2-imsd-admins` 的使用者對執行個體中繼資料選項進行變更。如果 `ec2-imsd-admins` 角色以外的任何主體嘗試呼叫 `ModifyInstanceMetadataOptions` API，就會收到 `UnauthorizedOperation` 錯誤訊息。這個陳述式可用來控制 `ModifyInstanceMetadataOptions` API 的使用；目前 `ModifyInstanceMetadataOptions` API 沒有細微的存取控制 (條件)。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "AllowOnlyImsdAdminsToModifySettings",
 "Effect": "Deny",
 "Action": "ec2:ModifyInstanceMetadataOptions",
 "Resource": "*",
 "Condition": {
 "StringNotLike": {
 "aws:PrincipalARN": "arn:aws:iam:*:*:role/ec2-imsd-admins"
 }
 }
 }
]
}

```

### 需要從 IMDSv2 擷取角色登入資料

下列政策指出，如果此政策套用至角色，且 EC2 服務假定該角色，並將產生的登入資料用於簽署請求，則必須由從 IMDSv2 擷取的 EC2 角色登入資料簽署該請求。否則，其所有 API 呼叫將得到 `UnauthorizedOperation` 錯誤。一般情況下，可套用此陳述式/政策，因為如果請求未由 EC2 角色登入資料簽署，則不具任何作用。

```

{

```

```
"Version": "2012-10-17",
"Statement": [
 {
 "Sid": "RequireAllEc2RolesToUseV2",
 "Effect": "Deny",
 "Action": "*",
 "Resource": "*",
 "Condition": {
 "NumericLessThan": {
 "ec2:RoleDelivery": "2.0"
 }
 }
 }
]
```

## 使用 Amazon EBS 磁碟區和快照

如需使用 Amazon EBS 磁碟區和快照的範例政策，請參閱 Amazon EBS 以[身分識別為基礎的政策範例](#)。

在 Amazon EC2 主控台中進行操作的範例政策。

您必須使用 IAM 政策授予使用者對於 Amazon EC2 所需的許可。您可以利用 IAM 政策來授予使用者許可，讓使用者檢視和使用 Amazon EC2 主控台內的特定資源。您可以使用上一節中的範例原則；不過，這些原則是針對使用 AWS CLI 或 AWS SDK 發出的要求而設計的。如需詳細資訊，請參閱《IAM 使用者指南》中的[使用 AWS CLI 或 AWS SDK 的範例原則](#) 和[建立 IAM 政策](#)。

主控台會針對其功能使用其他的 API 動作，所以這些政策可能不會如預期般運作。例如，使用者的權限如果只能使用 DescribeVolumes API 動作，則當該使用者試著在主控台中檢視磁碟區時，就可能會遇到錯誤。本節中所示範的政策，可讓使用者使用主控台的特定部分。如需為 Amazon EC2 主控台建立政策的其他資訊，請參閱下列 AWS 安全部落格文章：[授與使用者在 Amazon EC2 主控台中工作的權限](#)。

### Tip

在主控台中執行工作時，為協助找出所需的 API 動作，您可以使用像是 AWS CloudTrail 的服務。如需詳細資訊，請參閱《[AWS CloudTrail 使用者指南](#)》。如果政策並未授予權限來建立或修改特定的資源，主控台會顯示編碼的訊息，其中包含診斷資訊。[您可以使用的「DecodeAuthorization 訊息 API」動作來解碼郵件 AWS STS，或使用 AWS CLI](#)

## 範例

- [範例：唯讀存取](#)
- [範例：使用 EC2 啟動執行個體精靈](#)
- [範例：使用安全群組](#)
- [範例：使用彈性 IP 地址](#)
- [範例：使用預留執行個體](#)

### 範例：唯讀存取

若要允許使用者檢視 Amazon EC2 主控台中的所有資源，您可以使用與下列範例相同的政策：[範例：唯讀存取](#)。除非有另一個陳述式授予使用者此種權限，否則使用者無法對其他資源執行任何動作，也無法建立新資源。

### 檢視執行個體、AMI 和快照

或者，您可以提供對資源子集的唯一讀存取權限。若要這麼做，請針對每項資源，使用特定的 `ec2:Describe` 動作，來取代 `ec2:Describe` API 動作中的 \* 萬用字元。下列的政策可讓使用者檢視 Amazon EC2 主控台中的所有執行個體、AMI 和快照。`ec2:DescribeTags` 動作可讓使用者檢視公用 AMI。主控台需要標記資訊來顯示公用 AMI；不過，您可以移除這項動作，來讓使用者只檢視私有 AMI。

```
{
 "Version": "2012-10-17",
 "Statement": [{
 "Effect": "Allow",
 "Action": [
 "ec2:DescribeInstances",
 "ec2:DescribeImages",
 "ec2:DescribeTags",
 "ec2:DescribeSnapshots"
],
 "Resource": "*"
 }
]
```

**Note**

Amazon EC2 `ec2:Describe*` API 動作不支援資源層級的許可，因此您無法控制使用者可以在主控台中檢視哪些個別資源。因此，在上述的陳述式中，`Resource` 元素必須包含 \* 萬用字元。如需可與 Amazon EC2 API 動作搭配使用的 ARN 詳細資訊，請參閱 [Amazon EC2 的動作、資源與條件金鑰](#)。

## 檢視執行個體和 CloudWatch 指標

下列政策允許使用者在 Amazon EC2 主控台中檢視執行個體，以及在「執行個體」頁面的「監控」索引標籤中檢視 CloudWatch 警示和指標。Amazon EC2 主控台使用 CloudWatch API 顯示警示和指標，因此您必須授予使用者使用 `cloudwatch:DescribeAlarms`、`cloudwatch:DescribeAlarmsForMetric`、`cloudwatch:ListMetrics` 和 `cloudwatch:GetMetricData` 動作的權限。

```
{
 "Version": "2012-10-17",
 "Statement": [{
 "Effect": "Allow",
 "Action": [
 "ec2:DescribeInstances",
 "ec2:DescribeInstanceTypes",
 "cloudwatch:DescribeAlarms",
 "cloudwatch:DescribeAlarmsForMetric",
 "cloudwatch:ListMetrics",
 "cloudwatch:GetMetricStatistics",
 "cloudwatch:GetMetricData"
],
 "Resource": "*"
 }
]
```

## 範例：使用 EC2 啟動執行個體精靈

Amazon EC2 啟動執行個體精靈是一個內含選項的畫面，可用來設定和啟動執行個體。政策必須包含使用 API 動作的權限，來讓使用者能夠使用精靈的選項。如果政策未包含使用這些動作的權限，可能無法適當地載入精靈中的某些項目，使用者就無法完成啟動。

## 基本啟動執行個體精靈存取權

若要順利地完成啟動，必須授予使用者權限，來使用 `ec2:RunInstances` API 動作和至少下列的 API 動作：

- `ec2:DescribeImages`：檢視和選取 AMI。
- `ec2:DescribeInstanceTypes`：檢視和選取執行個體類型。
- `ec2:DescribeVpcs`：檢視可用的網路選項。
- `ec2:DescribeSubnets`：檢視所選擇 VPC 的所有可用子網。
- `ec2:DescribeSecurityGroups` 或 `ec2:CreateSecurityGroup`：檢視和選取現有的安全群組，或建立新的安全群組。
- `ec2:DescribeKeyPairs` 或 `ec2:CreateKeyPair`：選取現有的金鑰對，或建立新的金鑰對。
- `ec2:AuthorizeSecurityGroupIngress`：新增傳入規則。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:DescribeInstances",
 "ec2:DescribeImages",
 "ec2:DescribeInstanceTypes",
 "ec2:DescribeKeyPairs",
 "ec2:DescribeVpcs",
 "ec2:DescribeSubnets",
 "ec2:DescribeSecurityGroups",
 "ec2:CreateSecurityGroup",
 "ec2:AuthorizeSecurityGroupIngress",
 "ec2:CreateKeyPair"
],
 "Resource": "*"
 },
 {
 "Effect": "Allow",
 "Action": "ec2:RunInstances",
 "Resource": "*"
 }
]
}
```

您可以在政策中加入 API 動作，來為使用者提供更多選項，例如：

- `ec2:DescribeAvailabilityZones`：檢視並選取特定的可用區域。
- `ec2:DescribeNetworkInterfaces`：針對選取的子網，檢視並選取現有的網路介面。
- 若要在 VPC 安全群組中新增外傳規則，必須授予使用者權限來使用 `ec2:AuthorizeSecurityGroupEgress` API 動作。若要修改或刪除現有的規則，必須授予使用者權限，以使用相關的 `ec2:RevokeSecurityGroup*` API 動作。
- `ec2:CreateTags`：標記由 `RunInstances` 所建立的資源。如需詳細資訊，請參閱 [在建立期間授予標籤資源的許可](#)。如果使用者不具有使用此動作的許可，而且試著在 啟動執行個體精靈的標記頁面上使用標籤，則啟動會失敗。

#### Important

啟動執行個體時指定 Name (名稱) 可建立標籤，並且需要 `ec2:CreateTags` 動作。請小心授與使用者使用 `ec2:CreateTags` 動作的許可，因為這樣做會限制您使用 `aws:ResourceTag` 條件金鑰來限制使用者使用其他資源的能力。如果您授與使用者使用 `ec2:CreateTags` 動作的許可，他們可以變更資源的標籤，以略過這些限制。如需詳細資訊，請參閱 [使用資源標籤控制對 EC2 資源的存取](#)。

- 若要在選取 AMI 時使用 Systems Manager 參數，您必須在政策中新增 `ssm:DescribeParameters` 和 `ssm:GetParameters`。`ssm:DescribeParameters` 會授予使用者檢視和選取 Systems Manager 參數的許可。`ssm:GetParameters` 會授與使用者取得 Systems Manager 參數值的許可。您也可以限制對特定 Systems Manager 參數的存取。如需詳細資訊，請參閱本節稍後的 `Restrict access to specific Systems Manager parameters` (限制特定 Systems Manager 參數的存取)。

目前，Amazon EC2 `Describe*` API 動作並不支援資源層級的許可，因此您無法限制使用者可以在啟動執行個體精靈中檢視哪些個別資源。不過，您可以對 `ec2:RunInstances` API 動作套用資源層級的權限，來限制使用者可以使用哪些資源來啟動執行個體。如果使用者選取了自己無使用授權的選項，啟動會失敗。

`Restrict access to a specific instance type, subnet, and Region` (限制對特定執行個體類型、子網和區域的存取)

下列政策可讓使用者使用 Amazon 所擁有的 AMI，來啟動 `t2.micro` 執行個體，而且只啟動至指定的子網 (`subnet-1a2b3c4d`)。使用者只能在 `sa-east-1` 區域中啟動。如果使用者選取了不同的區域，或是在啟動執行個體精靈中選取了不同的執行個體類型、AMI 或子網路，啟動會失敗。

第一個陳述式授與使用者許可來檢視啟動執行個體精靈中的選項，或建立新的選項。如上列的範例所解釋。第二個陳述式授予使用者許可，來針對 `ec2:RunInstances` 動作使用網路介面、磁碟區、金鑰對、安全群組和子網資源，如果要將執行個體啟動到 VPC，就必須執行此項動作。如需關於使用 `ec2:RunInstances` 動作的詳細資訊，請參閱 [啟動執行個體 \(RunInstances\)](#)。第三和第四個陳述式分別授予使用者使用執行個體和 AMI 資源的許可，但前提是執行個體為 `t2.micro` 執行個體，並且 AMI 為 Amazon 或某些受信任和已驗證的合作夥伴所有。

```
{
 "Version": "2012-10-17",
 "Statement": [{
 "Effect": "Allow",
 "Action": [
 "ec2:DescribeInstances",
 "ec2:DescribeImages",
 "ec2:DescribeInstanceTypes",
 "ec2:DescribeKeyPairs",
 "ec2:CreateKeyPair",
 "ec2:DescribeVpcs",
 "ec2:DescribeSubnets",
 "ec2:DescribeSecurityGroups",
 "ec2:CreateSecurityGroup",
 "ec2:AuthorizeSecurityGroupIngress"
],
 "Resource": "*"
 },
 {
 "Effect": "Allow",
 "Action": "ec2:RunInstances",
 "Resource": [
 "arn:aws:ec2:sa-east-1:111122223333:network-interface/*",
 "arn:aws:ec2:sa-east-1:111122223333:volume/*",
 "arn:aws:ec2:sa-east-1:111122223333:key-pair/*",
 "arn:aws:ec2:sa-east-1:111122223333:security-group/*",
 "arn:aws:ec2:sa-east-1:111122223333:subnet/subnet-1a2b3c4d"
]
 },
 {
 "Effect": "Allow",
 "Action": "ec2:RunInstances",
 "Resource": [
 "arn:aws:ec2:sa-east-1:111122223333:instance/*"
],
 "Condition": {
```

```

 "StringEquals": {
 "ec2:InstanceType": "t2.micro"
 }
 },
 {
 "Effect": "Allow",
 "Action": "ec2:RunInstances",
 "Resource": [
 "arn:aws:ec2:sa-east-1::image/ami-*"
],
 "Condition": {
 "StringEquals": {
 "ec2:Owner": "amazon"
 }
 }
 }
]
}

```

Restrict access to specific Systems Manager parameters (限制對特定 Systems Manager 參數的存取)

下列政策授與使用具有特定名稱之 Systems Manager 參數的存取權。

第一個陳述式會授與使用者許可，以在啟動執行個體精靈中選取 AMI 時檢視 Systems Manager 參數。

第二個陳述式會授與使用者只使用命名為 prod-\* 參數的許可。

```

{
 "Version": "2012-10-17",
 "Statement": [{
 "Effect": "Allow",
 "Action": [
 "ssm:DescribeParameters"
],
 "Resource": "*"
 },
 {
 "Effect": "Allow",
 "Action": [
 "ssm:GetParameters"
],
 "Resource": "arn:aws:ssm:us-east-2:123456123:parameter/prod-*"
 }
]
}

```



```
]
}
```

## 範例：使用安全群組

### 檢視安全群組，以及新增和移除規則

下列的政策會授予使用者許可，讓使用者可在 Amazon EC2 主控台中檢視安全群組，並針對具有 Department=Test 標籤的現有安全群組新增和移除傳入與傳出規則，以及列出和修改規則說明。

在第一個陳述式中，ec2:DescribeTags 動作可讓使用者在主控台中檢視標籤，如此就能更容易找出使用者獲得允許修改的安全群組。

```
{
 "Version": "2012-10-17",
 "Statement": [{
 "Effect": "Allow",
 "Action": [
 "ec2:DescribeSecurityGroups",
 "ec2:DescribeSecurityGroupRules",
 "ec2:DescribeTags"
],
 "Resource": "*"
 },
 {
 "Effect": "Allow",
 "Action": [
 "ec2:AuthorizeSecurityGroupIngress",
 "ec2:RevokeSecurityGroupIngress",
 "ec2:AuthorizeSecurityGroupEgress",
 "ec2:RevokeSecurityGroupEgress",
 "ec2:ModifySecurityGroupRules",
 "ec2:UpdateSecurityGroupRuleDescriptionsIngress",
 "ec2:UpdateSecurityGroupRuleDescriptionsEgress"
],
 "Resource": [
 "arn:aws:ec2:region:111122223333:security-group/*"
],
 "Condition": {
 "StringEquals": {
 "aws:ResourceTag/Department": "Test"
 }
 }
 }
}
```

```
 },
 {
 "Effect": "Allow",
 "Action": [
 "ec2:ModifySecurityGroupRules"
],
 "Resource": [
 "arn:aws:ec2:region:111122223333:security-group-rule/*"
]
 }
]}
```

### 使用 Create Security Group (建立安全群組) 對話方塊

您可以建立政策，允許使用者在 Amazon EC2 主控台中，使用 Create Security Group (建立安全群組) 對話方塊。若要使用此對話方塊，必須授予使用者權限，來至少使用下列的 API 動作：

- `ec2:CreateSecurityGroup`：建立新的安全群組。
- `ec2:DescribeVpcs`：在 VPC 清單中，檢視現有 VPC 的清單。

使用者可利用這些權限來順利建立新的安全群組，但無法新增任何規則到安全群組。若要在 Create Security Group (建立安全群組) 對話方塊中使用規則，您可以將下列的 API 動作加入政策：

- `ec2:AuthorizeSecurityGroupIngress`：新增傳入規則。
- `ec2:AuthorizeSecurityGroupEgress`：將外傳規則加入 VPC 安全群組。
- `ec2:RevokeSecurityGroupIngress`：修改或刪除現有的傳入規則。這是一項實用的動作，可讓使用者在主控台中使用 Copy to new (複製到新的) 功能。這項功能會開啟 Create Security Group (建立安全群組) 對話方塊，並使用與所選取安全群組相同的規則，來填入規則。
- `ec2:RevokeSecurityGroupEgress`：修改或刪除 VPC 安全群組的外傳規則。這是一項實用的動作，可讓使用者修改或刪除預設的外傳規則 (此規則允許所有外傳流量)。
- `ec2>DeleteSecurityGroup`：適用於無法儲存個別規則的情況。主控台會先建立安全群組，然後再加入指定的規則。如果規則無效，動作會失敗，而主控台會試著刪除安全群組。使用者仍在使用 Create Security Group (建立安全群組) 對話方塊，因此可以修正無效的規則，然後再次試著建立安全群組。此 API 動作並非必要，但如果使用者並未獲得使用該動作的權限，並試著建立包含無效規則的安全群組，則會建立未包含任何規則的安全群組，使用者必須在之後加入規則。
- `ec2:UpdateSecurityGroupRuleDescriptionsIngress`：新增或更新傳入 (輸入) 安全群組規則的描述。

- `ec2:UpdateSecurityGroupRuleDescriptionsEgress` : 新增或更新傳出 (輸出) 安全群組規則的描述。
- `ec2:ModifySecurityGroupRules` : 修改安全群組規則。
- `ec2:DescribeSecurityGroupRules` : 列出安全群組規則。

下列政策授予使用者許可來使用 `Create Security Group` (建立安全群組) 對話方塊，並針對和指定 VPC (`vpc-1a2b3c4d`) 具有關聯的安全群組，建立其傳入和傳出規則。使用者可以建立 VPC 的安全群組，但無法將任何規則新增至這些安全群組。同樣地，使用者也無法將任何的規則，加入與 VPC `vpc-1a2b3c4d` 無關聯的現有安全群組。使用者也會獲得權限，來檢視主控台中的所有安全群組。如此，使用者就能更輕鬆地找出可以加入傳入規則的安全群組。此政策也會授予使用者權限，來刪除和 VPC `vpc-1a2b3c4d` 具有關聯的安全群組。

```
{
 "Version": "2012-10-17",
 "Statement": [{
 "Effect": "Allow",
 "Action": [
 "ec2:DescribeSecurityGroups",
 "ec2:CreateSecurityGroup",
 "ec2:DescribeVpcs"
],
 "Resource": "*"
 },
 {
 "Effect": "Allow",
 "Action": [
 "ec2>DeleteSecurityGroup",
 "ec2:AuthorizeSecurityGroupIngress",
 "ec2:AuthorizeSecurityGroupEgress"
],
 "Resource": "arn:aws:ec2:region:111122223333:security-group/*",
 "Condition": {
 "ArnEquals": {
 "ec2:Vpc": "arn:aws:ec2:region:111122223333:vpc/vpc-1a2b3c4d"
 }
 }
 }
]
}
```

## 範例：使用彈性 IP 地址

若要讓使用者在 Amazon EC2 主控台中檢視彈性 IP 地址，您必須授予使用者許可來使用 `ec2:DescribeAddresses` 動作。

若要讓使用者使用彈性 IP 地址，您可以將下列的動作加入政策。

- `ec2:AllocateAddress`：配置彈性 IP 地址。
- `ec2:ReleaseAddress`：釋出彈性 IP 地址。
- `ec2:AssociateAddress`：建立彈性 IP 地址與執行個體或網路介面的關聯。
- `ec2:DescribeNetworkInterfaces` 和 `ec2:DescribeInstances`：使用 Associate address (關聯地址) 畫面。此畫面會顯示可以和彈性 IP 地址建立關聯的執行個體或網路介面。
- `ec2:DisassociateAddress`：取消彈性 IP 地址與執行個體或網路介面的關聯。

下列政策可讓使用者檢視和配置彈性 IP 地址，並建立這些地址與執行個體的關聯。使用者不能建立彈性 IP 地址與網路介面的關聯、取消彈性 IP 地址的關聯，也不能釋出這些地址。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:DescribeAddresses",
 "ec2:AllocateAddress",
 "ec2:DescribeInstances",
 "ec2:AssociateAddress"
],
 "Resource": "*"
 }
]
}
```

## 範例：使用 預留執行個體

以下政策允許使用者檢視和修改帳戶中的預留執行個體，以及在 AWS Management Console 中購買新的預留執行個體。

此政策允許使用者檢視帳戶中的所有預留執行個體，以及帳戶中的所有隨需執行個體。您無法針對個別的預留執行個體設定資源層級的許可。

```
{
 "Version": "2012-10-17",
 "Statement": [{
 "Effect": "Allow",
 "Action": [
 "ec2:DescribeReservedInstances",
 "ec2:ModifyReservedInstances",
 "ec2:PurchaseReservedInstancesOffering",
 "ec2:DescribeInstances",
 "ec2:DescribeInstanceTypes",
 "ec2:DescribeAvailabilityZones",
 "ec2:DescribeReservedInstancesOfferings"
],
 "Resource": "*"
 }
]
```

若要確保 Amazon EC2 主控台能夠針對您可在其中購買預留執行個體的可用區域顯示相關的資訊，則必須執行 `ec2:DescribeAvailabilityZones` 動作。`ec2:DescribeInstances` 動作則非必要，但可確保使用者能夠檢視帳戶中的執行個體，並購買保留，以配合目前的規格。

您可以調整 API 動作來限制使用者的存取，例如移除 `ec2:DescribeInstances` 和 `ec2:DescribeAvailabilityZones` 代表使用者具有唯讀存取權限。

## AWS Amazon EC2 的受管政策

若要新增使用者、群組和角色的權限，使用 AWS 受管理的原則比自己撰寫原則更容易。建立 [IAM 客戶受管政策](#) 需要時間和專業知識，而受管政策可為您的團隊提供其所需的許可。若要快速開始使用，您可以使用我們的 AWS 受管政策。這些政策涵蓋常見使用案例，並可在您的 AWS 帳戶中使用。如需 AWS 受管政策的詳細資訊，請參閱 IAM 使用者指南中的 [AWS 受管政策](#)。

AWS 服務會維護和更新 AWS 受管理的策略。您無法變更 AWS 受管理原則中的權限。服務偶爾會在 AWS 受管政策中新增其他許可以支援新功能。此類型的更新會影響已連接政策的所有身分識別 (使用者、群組和角色)。當新功能啟動或新操作可用時，服務很可能會更新 AWS 受管政策。服務不會從 AWS 受管理的政策移除權限，因此政策更新不會破壞您現有的權限。

此外，還 AWS 支援跨多個服務之工作職能的受管理原則。例如，`ReadOnly` 存取 AWS 管理原則會提供所有 AWS 服務和資源的唯讀存取權。當服務啟動新功能時，會為新作業和資源新 AWS 增唯讀權限。如需任務職能政策的清單和說明，請參閱 IAM 使用者指南中 [有關任務職能的 AWS 受管政策](#)。

## AWS 受管理的策略：AmazonEC2FullAccess

您可以將 AmazonEC2FullAccess 政策連接到 IAM 身分。此政策授與允許完全存取 Amazon EC2 的許可。

若要檢視此原則的權限，請參閱AWS 受管理[AmazonEC2FullAccess](#)的策略參考中的。

## AWS 受管理的策略：AmazonEC2ReadOnlyAccess

您可以將 AmazonEC2ReadOnlyAccess 政策連接到 IAM 身分。此政策授予允許對 Amazon EC2 進行唯讀存取的許可。

若要檢視此原則的權限，請參閱AWS 受管理[AmazonEC2ReadOnlyAccess](#)的策略參考中的。

## AWS 受管理的策略：AWSEC2CapacityReservationFleetRolePolicy

此政策連接至名為 AWSServiceRoleForEC2CapacityReservationFleet 的服務連結角色，可讓容量保留功能代表您建立、修改和取消容量保留。如需詳細資訊，請參閱[容量保留機群的服務連結角色](#)。

若要檢視此原則的權限，請參閱AWS 受管理[AWSEC2CapacityReservationFleetRolePolicy](#)的策略參考中的。

## AWS 受管理的策略：AWSEC2FleetServiceRolePolicy

此政策連接到名為 AWSServiceRoleForEC2Fleet 的服務連結角色，允許 EC2 Fleet 代表您請求、啟動、終止和標記執行個體。如需詳細資訊，請參閱 [EC2 Fleet 的服務連結角色](#)。

若要檢視此原則的權限，請參閱AWS 受管理[AWSEC2FleetServiceRolePolicy](#)的策略參考中的。

## AWS 受管理的策略：AWSEC2SpotFleetServiceRolePolicy

此政策連接到名為 AWSServiceRoleForEC2SpotFleet 的服務連結角色，允許 Spot Fleet 代表您啟動和管理執行個體。如需詳細資訊，請參閱 [Spot Fleet 的服務連結角色](#)。

若要檢視此原則的權限，請參閱AWS 受管理[AWSEC2SpotFleetServiceRolePolicy](#)的策略參考中的。

## AWS 受管理的策略：AWSEC2SpotServiceRolePolicy

此政策連接到名為 AWSServiceRoleForEC2Spot 的服務連結角色，允許 Amazon EC2 代表您啟動和管理 Spot 執行個體。如需詳細資訊，請參閱 [Spot 執行個體請求的服務連結角色](#)。

若要檢視此原則的權限，請參閱AWS 受管理[AWSEC2SpotServiceRolePolicy](#)的策略參考中的。

## AWS 受管理的策略：AWSEC2VssSnapshotPolicy

您可以將此受管政策附加到用於 Amazon EC2 Windows 執行個體的 IAM 執行個體設定檔角色。該政策授予許可，以允許 Amazon EC2 代表您建立和管理 VSS 快照。

若要檢視此原則的權限，請參閱AWS 受管理[AWSEC2VssSnapshotPolicy](#)的策略參考中的。

## AWS 受管理的策略：EC2FastLaunchFullAccess

您可以將EC2FastLaunchFullAccess政策附加到執行個體設定檔或其他 IAM 角色。此政策授予 EC2 快速啟動動作和目標許可的完整存取權，如下所示。

### 許可詳細資訊

- EC2 快速啟動 — 授予管理存取權，以便該角色可以啟用或停用 EC2 快速啟動，並說明 EC2 快速啟動映像。
- Amazon EC2 — 授與 Amazon EC2 的存取權限 RunInstances，CreateTags 並描述驗證資源許可所需的動作。
- IAM — 授予存取權以取得和使用名稱包含的執行個體設定檔，ec2fastlaunch以建立 EC2FastLaunchServiceRolePolicy服務連結角色。

若要檢視此原則的權限，請參閱AWS 受管理[EC2FastLaunchFullAccess](#)的策略參考中的。

## AWS 受管理的策略：EC2FastLaunchServiceRolePolicy

此政策附加到名為的服務連結角色，可AWSServiceRoleForEC2FastLaunch讓 Amazon EC2 建立和管理一組預先佈建的快照，以減少從啟用 EC2 快速啟動的 AMI 啟動執行個體所需的時間。如需詳細資訊，請參閱 [the section called “服務連結角色”](#)。

若要檢視此原則的權限，請參閱AWS 受管理[EC2FastLaunchServiceRolePolicy](#)的策略參考中的。

## AWS 受管理的策略：Ec2InstanceConnectEndpoint

此政策附加至名為的服務連結角色，AWSServiceRoleForEC2InstanceConnect以允許 EC2 執行個體 Connect 端點代表您執行動作。如需詳細資訊，請參閱 [EC2 Instance Connect Endpoint 的服務連結角色](#)。

若要檢視此原則的權限，請參閱AWS 受管理[Ec2InstanceConnectEndpoint](#)的策略參考中的。

## Amazon EC2 更新受 AWS 管政策

檢視有關 Amazon EC2 AWS 受管政策更新的詳細資訊，因為此服務開始追蹤這些變更。

變更	描述	日期
<a href="#">EC2FastLaunchFullAccess</a> – 新政策	Amazon EC2 新增此政策以從執行個體執行與 EC2 快速啟動功能相關的 API 動作。該政策可以附加到從啟用 EC2 快速啟動的 AMI 啟動的執行個體的執行個體設定檔。	2024 年 5 月 14 日
<a href="#">AWSEC2VssSnapshotPolicy</a> – 新政策	Amazon EC2 新增了 AWSEC2VssSnapshotPolicy 政策，其中包含建立和新增標籤至 Amazon 機器映像 (AMI) 和 EBS 快照的許可。	2024年3月28日
<a href="#">EC2FastLaunchServiceRolePolicy</a> – 新政策	Amazon EC2 新增了 EC2 快速啟動功能，透過建立一組預先佈建的快照，讓 Windows AMI 能夠更快地啟動執行個體。	2021 年 11 月 26 日
Amazon EC2 開始追蹤變更	Amazon EC2 開始追蹤其 AWS 受管政策的變更	2021 年 3 月 1 日

## Amazon EC2 的 IAM 的角色

應用程式必須使用 AWS 認證簽署其 API 要求。因此，若您是應用程式開發人員，對於在 EC2 執行個體上運作的應用程式，您會需要其登入資料的管理策略。例如，您可安全將 AWS 登入資料分配給執行個體，讓在這些執行個體上運作的應用程式能夠使用此登入資料來簽署請求，同時避免其他使用者取得您的登入資料。但是，要將憑證安全地分配到每個執行個體，尤其是代表您 AWS 建立的執行個體 (例如 Spot 執行個體或 Auto Scaling 群組中的執行個體) 是一件困難的事。輪換認證時，您也必須能夠更新每個執行個體上的 AWS 認證。



**Note**

對於 Amazon EC2 工作負載，建議您使用下述方法擷取工作階段憑證。這些憑證應該可讓您的工作負載發起 AWS API 請求，而無需使用 `sts:AssumeRole` 擔任已與執行個體相關聯的相同角色。除非您需要為屬性型存取控制 (ABAC) 傳遞工作階段標籤，或需要傳遞工作階段政策以進一步限制角色權限，否則此類角色擔任呼叫是不必要的，因為它們會建立一組新的相同的臨時角色工作階段憑證。

如果您的工作負載使用角色來擔任它自己，則必須建立信任政策，明確允許該角色擔任它自己。如果您不建立信任政策，會得到 `AccessDenied` 錯誤。如需詳細資訊，請參閱《IAM 使用者指南》中的[修改角色信任政策](#)。

我們設計的 IAM 角色，讓您的應用程式能夠從執行個體安全提出 API 請求，您無須管理應用程式使用的安全登入資料。您可以使用 IAM 角色委派許可發出 API 請求，而不是建立和分發 AWS 登入資料，如下所示：

1. 建立 IAM 角色。
2. 定義哪些帳戶或 AWS 服務可以擔任該角色。
3. 定義應用程式在角色擔任後可使用的 API 動作與資源。
4. 啟動執行個體時指定該角色，或將該角色連接至現有執行個體。
5. 讓應用程式擷取一組暫時登入資料並加以使用。

例如，若執行於您執行個體上的應用程式需要使用 Amazon S3 內的儲存貯體，您可使用 IAM 角色為其授予許可。您可建立 JSON 格式的政策，藉此指定 IAM 角色的許可。這些政策類似於您為使用者建立的政策。若您變更角色，此變更會傳播至所有執行個體。

**Note**

Amazon EC2 IAM 角色登入資料不受該角色中設定的最長工作階段持續時間限制。如需詳細資訊，請參閱《IAM 使用者指南》中的[使用 IAM 角色](#)。

建立 IAM 角色時，請與最低權限 IAM 政策建立關聯，以限制對應用程式所需之特定 API 呼叫的存取。對於 Windows 至 Windows 通訊，請使用妥善定義且妥善記載的 Windows 群組和角色，授與 Windows 執行個體之間的應用程式層級存取權。群組和角色可讓客戶定義最低權限應用程式和 NTFS 資料夾層級許可，以限制對應用程式特定需求的存取。

雖然您只能將一個 IAM 角色連接到一個執行個體中，但您可以將相同的角色連接到多個執行個體中。如需建立 IAM 角色的詳細資訊，請參閱IAM 使用者指南中的[角色](#)。

您可將資源層級許可套用至 IAM 政策，藉此控制使用者連接、取代或分離執行個體的 IAM 角色的能力。如需詳細資訊，請參閱 [Amazon EC2 API 動作支援的資源層級許可](#) 及下列範例：[範例：使用 IAM 角色](#)。

## 內容

- [執行個體描述檔](#)
- [從執行個體中繼資料中擷取安全登入資料](#)
- [授予 IAM 使用者將 IAM 角色傳遞至執行個體的許可](#)
- [使用 IAM 角色](#)

## 執行個體描述檔

Amazon EC2 會使用執行個體描述檔做為 IAM 角色的容器。使用 IAM 主控台建立 IAM 角色時，主控台會自動建立執行個體描述檔，並將其命名為與對應角色相同的名稱。若您使用 Amazon EC2 主控台來啟動具備 IAM 角色的執行個體，或使用該主控台將 IAM 角色連接至執行個體，您須根據執行個體描述檔名稱清單來選擇角色。

如果您使用 AWS CLI、API 或 AWS SDK 建立角色，則可以將角色和執行個體設定檔建立為個別動作，名稱可能不同。如果您接著使用 AWS CLI、API 或 AWS SDK 啟動具有 IAM 角色的執行個體，或將 IAM 角色附加至執行個體，請指定執行個體設定檔名稱。

執行個體描述檔內僅有一個 IAM 角色。此限制無法提高。

如需詳細資訊，請參閱IAM 使用者指南中的[執行個體描述檔](#)。

## 從執行個體中繼資料中擷取安全登入資料

執行個體上的應用程式會從執行個體中繼資料項目 `iam/security-credentials/role-name`，擷取角色提供的安全登入資料。如此即授予該應用程式許可，能夠執行您在建立角色與安全登入資料的關聯時所定義的角色動作與資源。這些安全登入資料為暫時的，我們會定期更換。舊的登入資料過期前至少五分鐘，我們就會提供新的登入資料。

**⚠ Warning**

若您使用的服務會運用具備 IAM 角色的執行個體中繼資料，請確認該服務代您發出 HTTP 呼叫時不會暴露您的登入資料。可能會暴露登入資料的服務類型包括 HTTP 代理、HTML/CSS 驗證器服務和支援納入 XML 的 XML 處理器。

下列命令會擷取名為 s3access 之 IAM 角色的安全登入資料。

## Linux

## IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/iam/security-credentials/s3access
```

## IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/iam/security-credentials/s3access
```

## Windows

## IMDSv2

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/meta-data/iam/security-credentials/s3access
```

## IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/iam/security-credentials/s3access
```

下列為範例輸出。

```
{
 "Code" : "Success",
 "LastUpdated" : "2012-04-26T16:39:16Z",
 "Type" : "AWS-HMAC",
 "AccessKeyId" : "ASIAIOSFODNN7EXAMPLE",
 "SecretAccessKey" : "wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY",
 "Token" : "token",
 "Expiration" : "2017-05-17T15:09:54Z"
}
```

對於在執行個體上執行的應用程式 AWS CLI、和 Windows 專用工具 PowerShell 命令，您不必明確取得臨時安全登入資料 — AWS SDK 和 Windows Tools PowerShell 會自動從 EC2 執行個體中繼資料服務取得認證並加以使用。AWS CLI 欲使用暫時安全登入資料在執行個體外部進行呼叫 (如為了測試 IAM 政策)，您必須提供存取金鑰、私有金鑰和工作階段字符。如需詳細資訊，請參閱 IAM 使用者指南中的[使用臨時安全登入資料申請 AWS 資源存取權](#)。

如需執行個體中繼資料的詳細資訊，請參閱[使用執行個體中繼資料](#)。如需執行個體中繼資料 IP 地址的相關資訊，請參閱[擷取執行個體中繼資料](#)。

## 授予 IAM 使用者將 IAM 角色傳遞至執行個體的許可

若要讓使用者能夠啟動具備 IAM 角色的執行個體，或者連接或取代現有執行個體的 IAM 角色，您必須授予使用者許可才能使用下列 API 動作：

- iam:PassRole
- ec2:AssociateIamInstanceProfile
- ec2:ReplaceIamInstanceProfileAssociation

例如，使用 AWS CLI，下列 IAM 政策授予使用者許可以啟動具備 IAM 角色的執行個體，或讓其能夠連接或取代現有執行個體的 IAM 角色。

### Note

如果希望政策將使用者存取權限授予給所有角色，請在政策中將資源指定為 \*。但請考慮將[最低權限](#)原則當作最佳實務。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:RunInstances",
 "ec2:AssociateIamInstanceProfile",
 "ec2:ReplaceIamInstanceProfileAssociation"
],
 "Resource": "*"
 },
 {
 "Effect": "Allow",
 "Action": "iam:PassRole",
 "Resource": "arn:aws:iam::123456789012:role/DevTeam*"
 }
]
}
```

若要授予使用者許可以啟動具備 IAM 角色的執行個體，或讓其能夠使用 Amazon EC2 主控台為現有執行個體連接或取代 IAM 角色，除了他們可能需要的任何其他許可之外，您還必須授予他們使用 `iam:ListInstanceProfiles`、`iam:PassRole`、`ec2:AssociateIamInstanceProfile` 和 `ec2:ReplaceIamInstanceProfileAssociation` 的許可。如需取得範例政策，請參閱「[在 Amazon EC2 主控台中進行操作的範例政策](#)」。

## 使用 IAM 角色

您可建立 IAM 角色，並在啟動期間或之後將其連接至執行個體。您亦可取代或分離執行個體的 IAM 角色。

### 內容

- [建立 IAM 角色](#)
- [使用 IAM 角色啟動執行個體](#)
- [將 IAM 角色連接至執行個體](#)
- [取代 IAM 角色](#)
- [分離 IAM 角色](#)
- [根據存取活動產生 IAM 角色的政策](#)

## 建立 IAM 角色

您必須建立 IAM 角色，之後才能啟動具備該角色的執行個體，或將該角色連接至執行個體。

### Console

使用 IAM 主控台建立 IAM 角色

1. 在以下網址開啟 IAM 主控台：<https://console.aws.amazon.com/iam/>。
2. 在導覽窗格中，選擇角色，然後選擇建立角色。
3. 在選取信任的實體頁面上，選擇 AWS 服務，然後選取 EC2 使用案例。選擇下一步。
4. 在新增許可頁面上，選取可授予執行個體存取所需資源的政策。選擇下一步。
5. 在命名、檢閱和建立頁面上，輸入角色的名稱和描述。或者，將標籤新增至角色。選擇建立角色。

### Command line

下列範例利用允許角色使用 Amazon S3 儲存貯體的政策來建立 IAM 角色。

建立 IAM 角色和執行個體設定檔 (AWS CLI)

1. 建立下列信任政策並將其儲存為名為 `ec2-role-trust-policy.json` 的文字檔案。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Principal": { "Service": "ec2.amazonaws.com" },
 "Action": "sts:AssumeRole"
 }
]
}
```

2. 建立 `s3access` 角色，並指定您已使用 [create-role](#) 命令建立的信任政策。

```
aws iam create-role \
 --role-name s3access \
 --assume-role-policy-document file://ec2-role-trust-policy.json
```

## 回應範例

```
{
 "Role": {
 "AssumeRolePolicyDocument": {
 "Version": "2012-10-17",
 "Statement": [
 {
 "Action": "sts:AssumeRole",
 "Effect": "Allow",
 "Principal": {
 "Service": "ec2.amazonaws.com"
 }
 }
]
 },
 "RoleId": "AROAIIZKPBKS2LEXAMPLE",
 "CreateDate": "2013-12-12T23:46:37.247Z",
 "RoleName": "s3access",
 "Path": "/",
 "Arn": "arn:aws:iam::123456789012:role/s3access"
 }
}
```

3. 建立存取政策並將其儲存為名為 `ec2-role-access-policy.json` 的文字檔案。例如，下列政策將 Amazon S3 的管理許可授予執行個體上運作的應用程式。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": ["s3:*"],
 "Resource": ["*"]
 }
]
}
```

4. 使用 [put-role-policy](#) 命令將存取政策連接至角色。

```
aws iam put-role-policy \
 --role-name s3access \
 --policy-name ec2-role-access-policy \
 --policy-document file://ec2-role-access-policy.json
```

```
--policy-name S3-Permissions \
--policy-document file://ec2-role-access-policy.json
```

5. 使用 [create-instance-profile](#) 命令建立名為 s3access-profile 的執行個體設定檔。

```
aws iam create-instance-profile --instance-profile-name s3access-profile
```

#### 回應範例

```
{
 "InstanceProfile": {
 "InstanceProfileId": "AIPAJTLPJLEGREXAMPLE",
 "Roles": [],
 "CreateDate": "2013-12-12T23:53:34.093Z",
 "InstanceProfileName": "s3access-profile",
 "Path": "/",
 "Arn": "arn:aws:iam::123456789012:instance-profile/s3access-profile"
 }
}
```

6. 將 s3access 角色新增至 s3access-profile 執行個體設定檔。

```
aws iam add-role-to-instance-profile \
 --instance-profile-name s3access-profile \
 --role-name s3access
```

或者，您可以使用下列 AWS Tools for Windows PowerShell 指令：

- [New-IAMRole](#)
- [域名註冊 RolePolicy](#)
- [新 IAM InstanceProfile](#)

#### 使用 IAM 角色啟動執行個體

建立 IAM 角色後，即可啟動執行個體，並在啟動期間將該角色與執行個體建立關聯。



### ⚠ Important

建立 IAM 角色後，可能需要幾秒鐘的時間來傳播許可。若您第一次嘗試啟動具備角色的執行個體失敗，請等待數秒鐘後再試一次。如需詳細資訊，請參閱《IAM 使用者指南》中的 [IAM 角色故障診斷](#)。

## New console

使用 IAM 角色啟動執行個體 (主控台)

1. 請遵循該程序來[啟動執行個體](#)。
2. 展開 Advanced details (進階詳細資訊)，在 IAM instance profile (IAM 執行個體設定檔) 中選取您建立的 IAM 角色。

### **i** Note

IAM instance profile (IAM 執行個體設定檔) 清單會顯示您在建立 IAM 角色時所建立的執行個體設定檔名稱。若您已使用主控台建立 IAM 角色，則會為您建立執行個體描述檔，且其名稱會與該角色相同。如果您使用 AWS CLI、API 或 AWS SDK 建立 IAM 角色，可能會以不同的方式命名您的執行個體設定檔。

3. 設定執行個體所需的任何其他詳細資訊，或接受預設值，然後選取一個金鑰對。如需有關啟動執行個體精靈中欄位的資訊，請參閱 [使用定義的參數啟動執行個體](#)。
4. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。
5. 如果您在應用程式中使用 Amazon EC2 API 動作，請擷取執行個體上提供的 AWS 安全登入資料，並使用它們來簽署請求。AWS SDK 會為您執行此動作。

## IMDSV2

對於 Linux 執行個體，請參閱下列範例：

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H
"X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
meta-data/iam/security-credentials/role_name
```

對於 Windows 執行個體，請參閱下列範例：

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/meta-data/iam/security-credentials/role_name
```

## IMDSv1

對於 Linux 執行個體，請參閱下列範例：

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/iam/security-credentials/role_name
```

對於 Windows 執行個體，請參閱下列範例：

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/iam/security-credentials/role_name
```

## Old console

使用 IAM 角色啟動執行個體 (主控台)

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在儀表板上，選擇啟動執行個體。
3. 選取 AMI 和執行個體類型，然後選取 Next: Configure Instance Details (下一步：設定執行個體詳細資訊)。
4. 在 Configure Instance Details (設定執行個體詳細資訊) 頁面上，針對 IAM role (IAM 角色)，請選取您建立的 IAM 角色。

### Note

IAM role (IAM 角色) 清單會顯示您在建立 IAM 角色時所建立的執行個體描述檔名稱。若您已使用主控台建立 IAM 角色，則會為您建立執行個體描述檔，且其名稱會與該角

色相同。如果您使用 AWS CLI、API 或 AWS SDK 建立 IAM 角色，可能會以不同的方式命名您的執行個體設定檔。

5. 設定其他詳細資訊，然後按照指示完成剩下的精靈，或選擇 Review and Launch (檢閱和啟動) 來接受預設設定並直接前往 Review Instance Launch (檢閱執行個體的啟動) 頁面。
6. 檢閱設定，然後選擇 Launch (啟動)，選擇金鑰對並啟動您的執行個體。
7. 如果您在應用程式中使用 Amazon EC2 API 動作，請擷取執行個體上提供的 AWS 安全登入資料，並使用它們來簽署請求。AWS SDK 會為您執行此動作。

## IMDSV2

對於 Linux 執行個體，請參閱下列範例：

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/iam/security-credentials/role_name
```

對於 Windows 執行個體，請參閱下列範例：

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/meta-data/iam/security-credentials/role_name
```

## IMDSv1

對於 Linux 執行個體，請參閱下列範例：

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/iam/security-credentials/role_name
```

對於 Windows 執行個體，請參閱下列範例：

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/iam/security-credentials/role_name
```

## Command line

您可以使用在 AWS CLI 啟動期間將角色與執行個體相關聯。您必須在命令中指定執行個體描述檔。

### 使用 IAM 角色啟動執行個體 (AWS CLI)

1. 使用 [run-instances](#) 命令來啟動使用執行個體描述檔的執行個體。下列範例說明如何啟動具備執行個體描述檔的執行個體。

```
aws ec2 run-instances \
 --image-id ami-11aa22bb \
 --iam-instance-profile Name="s3access-profile" \
 --key-name my-key-pair \
 --security-groups my-security-group \
 --subnet-id subnet-1a2b3c4d
```

或者，使用視窗 [New-EC2Instance](#) 工具 PowerShell 指令。

2. 如果您在應用程式中使用 Amazon EC2 API 動作，請擷取執行個體上提供的 AWS 安全登入資料，並使用它們來簽署請求。AWS SDK 會為您執行此動作。

```
curl http://169.254.169.254/latest/meta-data/iam/security-credentials/role_name
```

## 將 IAM 角色連接至執行個體

若要將 IAM 角色連接至無角色的執行個體，則執行個體可以處於 stopped 或 running 狀態。

## Console

### 將 IAM 角色連接至執行個體 (主控台)

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，選取 動作、安全性、修改 IAM 角色。
4. 選取要連接至執行個體的 IAM 角色，然後選取 Save (儲存)。

## Command line

將 IAM 角色連接至執行個體 (AWS CLI)

1. 您可視需要描述執行個體來取得欲連接角色的執行個體 ID。

```
aws ec2 describe-instances
```

2. 使用 [associate-iam-instance-profile](#) 命令，透過指定執行個體描述檔將 IAM 角色連接至執行個體。您可使用執行個體描述檔的 Amazon Resource Name (ARN) 或其名稱。

```
aws ec2 associate-iam-instance-profile \
 --instance-id i-1234567890abcdef0 \
 --iam-instance-profile Name="TestRole-1"
```

### 回應範例

```
{
 "IamInstanceProfileAssociation": {
 "InstanceId": "i-1234567890abcdef0",
 "State": "associating",
 "AssociationId": "iip-assoc-0dbd8529a48294120",
 "IamInstanceProfile": {
 "Id": "AIPAJLNLDX3AMYZWNYYAY",
 "Arn": "arn:aws:iam::123456789012:instance-profile/TestRole-1"
 }
 }
}
```

或者，您也可以使用下列適用於 Windows PowerShell 指令的工具：

- [Get-EC2Instance](#)
- [Register-EC2IamInstanceProfile](#)

## 取代 IAM 角色

若要取代已擁有連接 IAM 角色之執行個體上的 IAM 角色，則執行個體必須處於 running 狀態。如果您想要變更執行個體上的 IAM 角色，但不中斷現有的角色，您可以這麼做。例如，您可以這麼做確保執行個體上應用程式執行的 API 動作不會中斷。

## Console

### 取代執行個體的 IAM 角色

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，選取 動作、安全性、修改 IAM 角色。
4. 選取要連接至執行個體的 IAM 角色，然後選取 Save (儲存)。

## Command line

### 取代執行個體的 IAM 角色 (AWS CLI)

1. 您可視需要描述您的 IAM 執行個體描述檔關聯，以取得 IAM 執行個體描述檔欲取代的關聯 ID。

```
aws ec2 describe-iam-instance-profile-associations
```

2. 使用 [replace-iam-instance-profile-association](#) 命令，指定現有執行個體描述檔的關聯 ID，並指定取代後的執行個體描述檔 ARN 或名稱，藉此取代 IAM 執行個體描述檔。

```
aws ec2 replace-iam-instance-profile-association \
 --association-id iip-assoc-0044d817db6c0a4ba \
 --iam-instance-profile Name="TestRole-2"
```

### 回應範例

```
{
 "IamInstanceProfileAssociation": {
 "InstanceId": "i-087711ddaf98f9489",
 "State": "associating",
 "AssociationId": "iip-assoc-09654be48e33b91e0",
 "IamInstanceProfile": {
 "Id": "AIPAJCJEDKX7QYHWYK7GS",
 "Arn": "arn:aws:iam::123456789012:instance-profile/TestRole-2"
 }
 }
}
```

或者，您也可以使用下列適用於 Windows PowerShell 指令的工具：

- [Get-EC2IamInstanceProfileAssociation](#)
- [Set-EC2IamInstanceProfileAssociation](#)

## 分離 IAM 角色

您可從運作中或停止的執行個體分離 IAM 角色。

### Console

#### 分離 IAM 角色和執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，選取 動作、安全性、修改 IAM 角色。
4. 針對 IAM role (IAM 角色)，選擇 No IAM Role (無 IAM 角色)。選擇 Save (儲存)。
5. 在確認對話方塊中，輸入 Detach (分離)，然後選擇 Detach (分離)。

### Command line

#### 分離 IAM 角色和執行個體 (AWS CLI)

1. 您可視需要使用 [describe-iam-instance-profile-associations](#) 來描述您的 IAM 執行個體描述檔關聯，並取得 IAM 執行個體描述檔欲分離的關聯 ID。

```
aws ec2 describe-iam-instance-profile-associations
```

#### 回應範例

```
{
 "IamInstanceProfileAssociations": [
 {
 "InstanceId": "i-088ce778fbfeb4361",
 "State": "associated",
 "AssociationId": "iip-assoc-0044d817db6c0a4ba",
 "IamInstanceProfile": {
 "Id": "AIPAJEDNCAA64SSD265D6",
 "Arn": "arn:aws:iam::123456789012:instance-profile/TestRole-2"
 }
 }
]
}
```

```

 }
 }
]
}

```

2. 使用 [disassociate-iam-instance-profile](#) 命令，透過關聯 ID 分離 IAM 執行個體描述檔。

```
aws ec2 disassociate-iam-instance-profile --association-id iip-
assoc-0044d817db6c0a4ba
```

### 回應範例

```
{
 "IamInstanceProfileAssociation": {
 "InstanceId": "i-087711ddaf98f9489",
 "State": "disassociating",
 "AssociationId": "iip-assoc-0044d817db6c0a4ba",
 "IamInstanceProfile": {
 "Id": "AIPAJEDNCAA64SSD265D6",
 "Arn": "arn:aws:iam::123456789012:instance-profile/TestRole-2"
 }
 }
}
```

或者，您也可以使用下列適用於 Windows PowerShell 指令的工具：

- [Get-EC2IamInstanceProfileAssociation](#)
- [Unregister-EC2IamInstanceProfile](#)

### 根據存取活動產生 IAM 角色的政策

當您第一次為應用程式建立 IAM 角色時，有時可能會授予超出所需的許可。在生產環境中啟動應用程式之前，您可以根據 IAM 角色的存取活動產生 IAM 政策。IAM Access Analyzer 會檢閱您的 AWS CloudTrail 記錄檔，並產生政策範本，其中包含指定日期範圍內角色已使用的許可。您可以使用範本建立具有精細許可的受管政策，然後將其連接至 IAM 角色。如此一來，您只會針對特定使用案例授與角色與 AWS 資源互動所需的權限。這有助於您符合[授予最低權限](#)的最佳實務。若要進一步了解，請參閱 IAM 使用者指南中的[根據存取活動產生政策](#)。



## 使用介面 VPC 端點存取 Amazon EC2

您可以在 VPC 和 Amazon EC2 之間建立私有連線，藉此改善 VPC 的安全狀態。您可以像在 VPC 中一樣存取 Amazon EC2，而無需使用網際網路閘道、NAT 裝置、VPN 連線或 AWS Direct Connect 連線。VPC 中的執行個體無需公有 IP 地址，即可存取 Amazon EC2。

如需詳細資訊，請參閱[AWS PrivateLink 指南 AWS PrivateLink 中的 AWS 服務 透過存取](#)。

### 目錄

- [建立介面 VPC 端點](#)
- [建立端點政策](#)

## 建立介面 VPC 端點

使用以下服務名稱為 Amazon EC2 建立介面端點：

- `com.amazonaws.region.ec2` – 為 Amazon EC2 API 動作建立端點。

如需詳細資訊，請參閱[AWS PrivateLink 指南中的 AWS 服務 使用介面 VPC 端點](#)存取。

## 建立端點政策

端點政策為 IAM 資源，您可將其連接至您的介面端點。預設端點政策可允許透過介面端點完整存取 Amazon EC2 API。若要控制 VPC 對 Amazon EC2 API 的存取權限，請將自訂端點政策連接至介面端點。

端點政策會指定以下資訊：

- 可執行動作的主體。
- 可執行的動作。
- 可供執行動作的資源。

### Important

將非預設政策套用至 Amazon EC2 的介面 VPC 端點時，某些失敗的 API 請求 (例如失敗者) 可能不會記錄到 Amazon AWS CloudTrail 或 Amazon。RequestLimitExceeded CloudWatch

如需詳細資訊，請參閱《AWS PrivateLink 指南》中的[使用端點政策控制對服務的存取](#)。

下列範例顯示 VPC 端點政策，該政策拒絕建立未加密磁碟區或啟動具有未加密磁碟區之執行個體的許可。範例政策也會授與執行所有其他 Amazon EC2 動作的許可。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Action": "ec2:*",
 "Effect": "Allow",
 "Resource": "*",
 "Principal": "*"
 },
 {
 "Action": [
 "ec2:CreateVolume"
],
 "Effect": "Deny",
 "Resource": "*",
 "Principal": "*",
 "Condition": {
 "Bool": {
 "ec2:Encrypted": "false"
 }
 }
 },
 {
 "Action": [
 "ec2:RunInstances"
],
 "Effect": "Deny",
 "Resource": "*",
 "Principal": "*",
 "Condition": {
 "Bool": {
 "ec2:Encrypted": "false"
 }
 }
 }
]
}
```

# Amazon EC2 視窗執行個體的更新管理

建議您定期修補、更新和保護 EC2 執行個體上的作業系統與應用程式。您可以使用 [AWS Systems Manager Patch Manager](#) 會針對作業系統及應用程式自動化安裝安全性相關更新的程序。

對於 Auto Scaling 群組中的 EC2 執行個體，您可以使用 [AWS-PatchAsgInstance](#) 執行手冊協助避免正在進行修補的執行個體遭到取代。或者，您也可以使用任何自動更新服務或建議的程序，以安裝應用程式廠商提供的更新。

## 資源

- AL2023 — 在 Amazon 使用者指南中 [更新 AL2023](#)。
- AL2 — 在 Amazon Linux 2 使用者指南中管理您的 Amazon Linux 2 [執行個體上的軟體](#)。
- 視窗實例 — [the section called “更新管理”](#)。

# Windows 執行個體的安全性最佳做法

我們建議您遵循 Windows 執行個體的這些安全性最佳做法。

## 目錄

- [高階安全性最佳做法](#)
- [更新管理](#)
- [組態管理](#)
- [變更管理](#)
- [Amazon EC2 視窗執行個體的稽核和責任制](#)

## 高階安全性最佳做法

您應該遵守下列適用於 Windows 執行個體的高階安全性最佳做法：

- 最少存取權 — 僅授予對受信任且預期的系統和位置的存取權。這適用於所有 Microsoft 產品，例如 Active Directory、Microsoft 商務生產力伺服器，以及基礎設施服務，例如遠端桌面服務，反向代理伺服器、IIS Web 伺服器。使用 Amazon EC2 執行個體安全群組、網路存取控制清單 (ACL) 和 Amazon VPC 公有/私有子網路等 AWS 功能，將架構中多個位置的安全性分層。在 Windows 執行個體中，客戶可以使用 Windows 防火牆在部署中進一步分層 defense-in-depth 策略。僅安裝系統依設

計運作所需的作業系統元件和應用程式。將基礎設施服務 (例如 IIS) 設定為在服務帳戶下執行，或使用應用程式集區身分這類功能，在本機和遠端存取整個基礎設施中的資源。

- **最小權限** — 決定執行個體和帳戶執行其功能所需的最低權限集。限制這些伺服器和使用者的使用者只允許這些已定義的許可。使用角色型存取控制這類技術來減少管理帳戶的表面區域，並建立最受限的角色來完成任務。使用作業系統功能 (例如 NTFS 內的加密檔案系統 (EFS))，來加密敏感的敏感資料，並控制應用程式和使用者對它的存取。
- **組態管理** — 建立基準伺服器組態，其中包含 up-to-date 安全性修補程式和主機型保護套件，其中包括防毒、反惡意程式碼、入侵偵測/預防，以及檔案完整性監控。根據目前記錄的基準來評定每部伺服器，以識別並標示任何偏差。確保每部伺服器都已設定為產生並安全地存放適當的日誌和稽核資料。
- **變更管理** — 建立程序以控制伺服器組態基準的變更，並朝著完全自動化的變更流程進行。此外，運用 Windows PowerShell DSC 的「剛夠管理」(JEA)，將系統管理存取限制在最低限度的必要功能。
- **修補程式管理** — 實作定期修補、更新及保護 EC2 執行個體上作業系統和應用程式的程序。
- **稽核日誌** — 稽核 Amazon EC2 執行個體的存取和所有變更，以驗證伺服器完整性，並確保僅進行授權的變更。利用 [IIS 的增強型記錄](#) 等功能來增強預設記錄功能。AWS VPC 流程記錄等功能也可用 AWS CloudTrail 於稽核網路存取，包括分別允許/拒絕的要求和 API 呼叫。

## 更新管理

為了確保在 Amazon EC2 上執行 Windows 伺服器時獲得最佳結果，我們建議您實作下列最佳實務：

- [Configure Windows Update](#)
- [Update drivers](#)
- [Use the latest Windows AMIs](#)
- [Test performance before migration](#)
- [Update launch agents](#)
- 安裝更新之後，請重新啟動 Windows 執行個體。如需詳細資訊，請參閱 [重新啟動您的執行個體](#)。

如需如何將 Windows 執行個體升級或遷移至較新版 Windows Server 的資訊，請參閱 [將 Amazon EC2 Windows 執行個體升級至較新版本的 Windows Server](#)。

### 設定視窗更新

根據預設，從 AWS Windows 伺服器 AMI 啟動的執行個體不會透過 Windows 更新接收更新。

## 更新 Windows 驅動程式

在所有 Windows EC2 執行個體上維護最新的驅動程式，以確保在機群中套用最新的問題修正和效能增強。視您的執行個體類型而定，您應該更新 AWS PV、Amazon ENA 和 AWS NVMe 驅動程式。

- 使用 [SNS 主題](#) 接收新驅動程式的版本更新。
- 使用 AWS Systems Manager 自動化手冊 [AWSSupport- 輕鬆地 Upgrade Windows AWS Drivers](#) 將更新套用到您的執行個體。

## 使用最新的視窗 AMI 啟動執行個體

AWS 每個月都會發佈新的 Windows AMI，其中包含最新的作業系統修補程式、驅動程式和啟動代理程式。在啟動新執行個體或建構自己的自訂映像時，您應運用最新的 AMI。

- 若要檢視視窗 AMI 的每個發行版本的更新，請參閱視 AWS [AWS 窗 AMI 版本歷史記錄](#)。
- 若要建構最新的可用 AMI，請參閱 [使用 Systems Manager 參數存放區查詢最新的 Windows AMI](#)。
- 如需有關可用來啟動資料庫執行個體的專用 Windows AMI 的詳細資訊，以及相容性強化使用案例，請參閱 [Windows AMI 參考中的專門化 AWS Windows AMI](#)。

## 在遷移前測試系統/應用程式效能

將企業應用程式移轉至 AWS 可能涉及許多變數和組態。一律對 EC2 解決方案執行效能測試以確保：

- 已正確配置執行個體類型，包括執行個體大小、增強型聯網和租用 (共用或專用)。
- 執行個體拓撲適用於工作負載，並在必要時運用高效能功能，例如專用租用、置放群組、執行個體存放區磁碟區、裸機。

## 更新啟動代理程式

更新至最新的 EC2Launch v2 代理程式，以確保您的機群中套用最新的增強功能。如需詳細資訊，請參閱 [the section called “遷移”](#)。

如果您有混合機群，或想要繼續使用 EC2Launch (Windows Server 2016 和 2019) 或 EC2 Config (僅限舊版作業系統) 代理程式，則請更新至各自代理程式的最新版本。

下列 Windows Server 版本和啟動代理程式組合支援自動更新。您可以在 Amazon EC2 啟動代理程式下的 [SSM 快速設定主機管理](#) 主控台中選擇自動更新。

Windows 版本	EC2Launch v1	EC2Launch v2
2016	✓	✓
2019	✓	✓
2022		✓

- 如需更新至 EC2Launch v2 的詳細資訊，請參閱。[the section called “安裝”](#)
- 如需手動更新 EC2Config 的資訊，請參閱。[the section called “安裝 EC2Config”](#)
- 如需手動更新 EC2Launch 的資訊，請參閱。[the section called “安裝 EC2Launch”](#)

## 組態管理

Amazon Machine Image (AMI) 為 Amazon EC2 執行個體提供初始配置，其中包括 Windows 作業系統和選用的客戶特定自訂，例如應用程式和安全控制。建立包含自訂安全性設定基準的 AMI 目錄，以確保所有 Windows 執行個體都以標準安全性控制項啟動。安全基準可以轉換為 AMI、在 EC2 執行個體啟動時動態啟動，或封裝為產品，以透過 AWS Service Catalog 產品組合進行統一發佈。如需保護 AMI 的詳細資訊，請參閱[建置 AMI 的最佳實務](#)。

每個 Amazon EC2 執行個體都應該遵守組織安全標準。不要安裝任何不需要的 Windows 角色和功能，並安裝軟體以防範惡意程式碼 (防毒、反惡意軟體、漏洞防護)、監控主機完整性，以及執行入侵偵測。設定安全軟體以監控和維護作業系統安全設定、保護重要作業系統檔案的完整性，以及在偏離安全基準時發出提醒。考慮實作由 Microsoft、網際網路安全中心 (CIS) 或國家標準技術研究所 (NIST) 所發佈的建議安全組態基準。考慮使用其他適用於特定應用程式伺服器的 Microsoft 工具，例如 [Best Practice Analyzer for SQL Server](#)。

AWS 客戶也可以執行 Amazon Inspector 評估，以改善 Amazon EC2 執行個體上部署之應用程式的安全性和合規性。Amazon Inspector 會自動評定應用程式是否有漏洞或偏離最佳實務，並包含一個知識庫，其中有數百個映射至常見安全合規性標準 (例如 PCI DSS) 和漏洞定義的規則。內建規則的範例包括檢查是否已啟用遠端根登入，或是否已安裝易受攻擊的軟體版本。這些規則由 AWS 安全研究人員定期更新。

保護 Windows 執行個體時，我們建議您實作 Active Directory Domain Services，為分散式位置啟用可擴展、安全且可管理的基礎設施。此外，從 Amazon EC2 主控台啟動執行個體之後，或使用 Amazon EC2 佈建工具 (例如 AWS CloudFormation)，在發生組態偏移時，最好使用原生作業系統功能 (例如 [Microsoft Windows PowerShell DSC](#)) 來維護組態狀態。

## 變更管理

在啟動時將初始安全基準套用至 Amazon EC2 執行個體之後，系統會控制持續的 Amazon EC2 變更以維護虛擬機器的安全。建立變更管理程序，以授權並納入 AWS 資源 (例如安全群組、路由表和網路 ACL) 的變更，以及作業系統和應用程式組態 (例如 Windows 或應用程式修補、軟體升級或組態檔更新)。

AWS 提供數種工具來協助管理 AWS 資源的變更 AWS CloudTrail AWS Config AWS CloudFormation，包括系統中心作業管理員和系統中心虛擬機器管理員的管理組件 AWS OpsWorks，以及管理組件。AWS Elastic Beanstalk 請注意，Microsoft 每週二 (有時甚至每天) 發布 Windows 補丁，並在 Microsoft 發布修補程序後五天 AWS 內 AWS 更新所有 Windows AMI 管理。因此，請務必持續修補所有基準 AMI、使用最新的 AMI ID 更新 AWS CloudFormation 範本和 Auto Scaling 群組組態，並實作工具以自動執行執行的執行個體修補程式管理。

Microsoft 提供了數個選項，用於管理 Windows 作業系統和應用程式變更。例如，SCCM 提供環境修改的完整生命週期涵蓋範圍。選取處理商務需求的工具，並控制變更如何影響應用程式 SLA、容量、安全性和災難復原程序。避免手動變更，而是利用自動化組態管理軟體或命令列工具 (例如 EC2 Run Command 或 Windows) PowerShell 來實作可重複的指令碼變更程序。為了協助滿足此需求，請將具有增強記錄功能的堡壘主機用於與 Windows 執行個體的所有互動，以確保自動記錄所有事件和任務。

## Amazon EC2 視窗執行個體的稽核和責任制

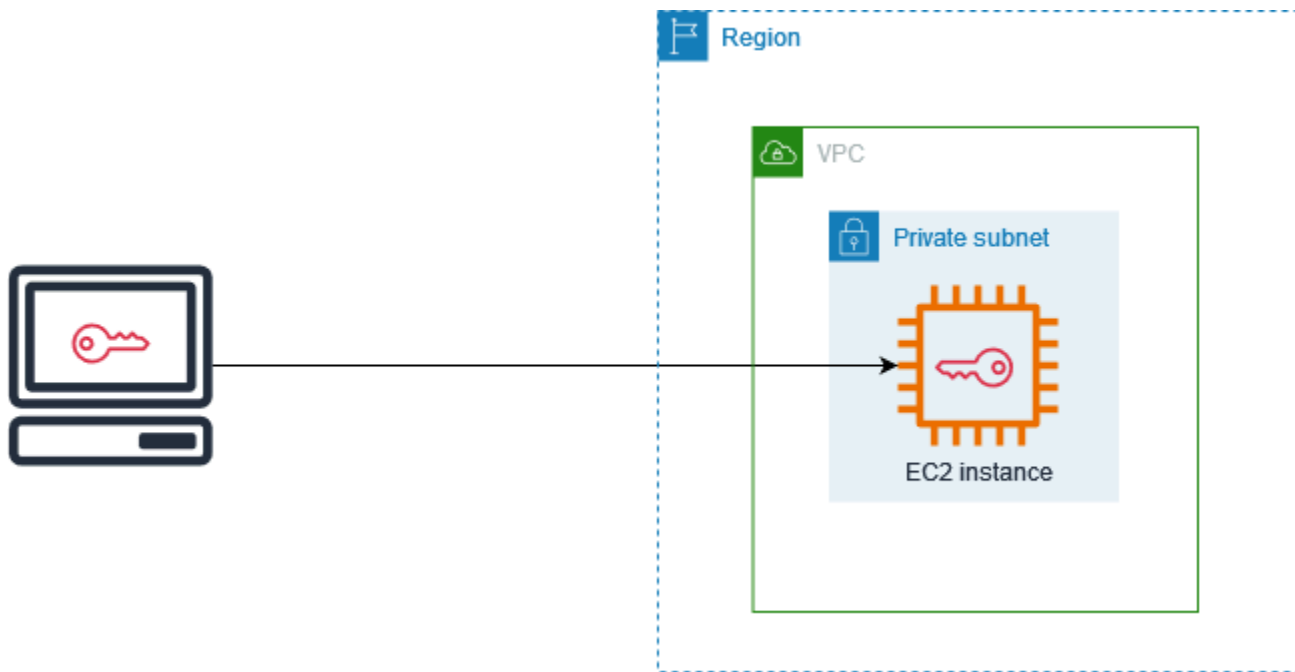
AWS CloudTrail、AWS Config，並 AWS Config 規則 提供稽核與變更追蹤功能，以便稽核 AWS 資源變更。將 Windows 事件日誌設定為將本機日誌檔傳送至集中式日誌管理系統，來保留日誌資料以進行安全性和操作行為分析。Microsoft System Center Operations Manager (SCOM) 可彙總部署到 Windows 執行個體之 Microsoft 應用程式的相關資訊，並根據應用程式角色和服務套用預先設定和自訂的規則集。System Center 管理套件建置在 SCOM 上，以提供應用程式特定的監控和設定指引。這些 [管理組件](#) 支援視窗伺服器作用中目錄、SharePoint 伺服器 2013 年、交換伺服器 2013 年、Lync 伺服器 2013 年、SQL 伺服器 2014 年，以及更多的伺服器和技術。

除了 Microsoft 系統管理工具之外，客戶還可以使用 Amazon CloudWatch 監控執行個體 CPU 使用率、磁碟效能、網路 I/O，以及執行主機和執行個體狀態檢查。EC2Config、EC2Launch 和 EC2Launch v2 啟動代理程式可讓您存取 Windows 執行個體的其他進階功能。例如，他們可以將 Windows 系統、安全性、應用程式和網際網路資訊服務 (IIS) 記錄匯出到 CloudWatch 日誌，然後可與 Amazon CloudWatch 指標和警示整合。客戶也可以建立將 Windows 效能計數器匯出至 Amazon CloudWatch 自訂指標的指令碼。

## Amazon EC2 金鑰配對和 Amazon EC2 執行個體

金鑰對 (由公有金鑰和私有金鑰組成) 是一組安全登入資料，您可以在連線至 Amazon EC2 執行個體時用來證明身分。對於 Linux 執行個體，私密金鑰可讓您安全使用 SSH 存取執行個體。對於 Windows 執行個體，需要私密金鑰才能解密管理員密碼，然後您可以使用該密碼連線至執行個體。

Amazon EC2 會將公開金鑰存放在您的執行個體上，並存放私密金鑰，如下圖所示。請務必將私密金鑰儲存在安全的地方，因為擁有您私密金鑰的任何人都可以連線到使用該 key pair 的執行個體。



啟動執行個體時，您可以指定 [key pair](#)，以便使用需要金 key pair 的方法連線至執行個體。根據您管理安全性的方式，您可以為所有執行個體指定相同的金鑰對，也可以指定不同的金鑰對。

對於 Linux 執行個體，當您的執行個體首次啟動時，您在啟動時指定的公開金鑰會放置在 Linux 執行個體中的項目中 `~/.ssh/authorized_keys`。當您使用 SSH 連線到 Linux 執行個體時，若要登入，則必須指定與公有金鑰相對應的私有金鑰。

如需連線至 EC2 執行個體的詳細資訊，請參閱 [Connect 至您的 EC2 執行個體](#)。

### ⚠ Important

由於 Amazon EC2 不會保留您私有金鑰的複本，因此，如果私有金鑰遺失了，就無法恢復。但是，仍然有一種方法可以連線到遺失私有金鑰的執行個體。如需詳細資訊，請參閱 [我的私有金鑰遺失。如何連線到我的 Linux 執行個體？](#)。



作為金鑰配對的替代方案，您可[AWS Systems Manager Session Manager](#)以使用互動式按一下瀏覽器殼層或 AWS Command Line Interface (AWS CLI) 連線至執行個體。

## 目錄

- [為您的 Amazon EC2 執行個體建立 key pair](#)
- [標記金鑰對](#)
- [描述您的金鑰配對](#)
- [刪除您的金鑰對](#)
- [在 Linux 執行個體上新增或移除公開金鑰](#)
- [驗證金鑰對的指紋](#)

## 為您的 Amazon EC2 執行個體建立 key pair

您可以使用 Amazon EC2 建立金鑰配對，也可以使用第三方工具建立金鑰配對，然後將其匯入 Amazon EC2。

Amazon EC2 支援適用於 Linux 和視窗執行個體的 2048 位元 SSH-2 RSA 金鑰。Amazon EC2 也支援適用於 Linux 執行個體的 ED25519 金鑰。

如需在建立 key pair 之後使用 SSH 連線至 Linux 執行個體的步驟，請參閱[the section called “連接至您的 Linux 執行個體”](#)。

如需在建立 key pair 之後，使用 RDP 連線至 Windows 執行個體的步驟，請參閱[the section called “連線至 Windows 執行個體”](#)。

## 目錄

- [使用 Amazon EC2 建立金鑰對](#)
- [使用建立 key pair AWS CloudFormation](#)
- [使用第三方工具來建立金鑰對，然後將公有金鑰匯入 Amazon EC2](#)

## 使用 Amazon EC2 建立金鑰對

使用 Amazon EC2 建立金鑰對時，公有金鑰會儲存在 Amazon EC2 中，而您可儲存私有金鑰。

每個區域最多可以建立 5,000 個金鑰配對。若要請求增加，請建立支援案例。如需詳細資訊，請參閱 AWS Support 使用者指南中的[建立支援案例](#)。

## Console

### 使用 Amazon EC2 建立金鑰對

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 請在導覽窗格的 Network & Security (網路與安全性) 下方，選擇 Key Pairs (金鑰對)。
3. 選擇 Create key pair (建立金鑰對)。
4. 對於 Name (名稱)，輸入金鑰對的描述性名稱。Amazon EC2 會將公有金鑰與您指定為金鑰名稱的名稱相關聯。金鑰名稱最多可包含 255 個 ASCII 字元。它不能包括前置或結尾空格。
5. 選取適合您作業系統的 key pair 類型：

(Linux 執行個體) 針對金鑰配對類型，請選擇 RSA 或 ED25519。

(視窗執行個體) 針對金鑰配對類型，選擇 RSA。視窗執行個體不支援 ED25519 金鑰。

6. 針對 Private key file format (私有金鑰檔案格式)，選擇私有金鑰的儲存格式。若要將私有金鑰儲存為可與 OpenSSH 搭配使用的格式，請選擇 pem。若要將私有金鑰儲存為可與 PuTTY 搭配使用的格式，請選擇 ppk。
7. 若要向公有金鑰新增標籤，請選擇 Add tag (新增標籤)，然後輸入標籤的金鑰和值。針對每個標籤，重複執行上述動作。
8. 選擇 Create key pair (建立金鑰對)。
9. 您的瀏覽器會自動下載私有金鑰檔案。基礎檔案名稱是您所指定的金鑰對名稱，而副檔名是根據您選擇的檔案格式所決定。將私有金鑰檔案存放在安全的地方。

#### Important

這是您儲存私有金鑰檔案的唯一機會。

10. (Linux 執行個體) 如果您打算在 macOS 或 Linux 電腦上使用安全殼層用戶端連線到 Linux 執行個體，請使用下列指令來設定私密金鑰檔案的權限，以便只有您可以讀取它。

```
chmod 400 key-pair-name.pem
```

如果您未設定這些許可，則無法使用此金鑰對來連接至您的執行個體。如需詳細資訊，請參閱 [錯誤：未受保護的私有金鑰檔案](#)。

## AWS CLI

### 使用 Amazon EC2 建立金鑰對

1. 按如下說明，使用 [create-key-pair](#) 命令，產生金鑰對並將私有金鑰儲存到 .pem 檔案中。

對於 `--key-name`，指定公有金鑰的名稱。名稱最多為 255 個 ASCII 字元。

對於 `--key-type`，指定 `rsa` 或 `ed25519`。如果您未包含 `--key-type` 參數，預設會建立 `rsa` 金鑰。請注意，Windows 執行個體不支援 ED25519 金鑰。

對於 `--key-format`，指定 `pem` 或 `ppk`。如果您未包含 `--key-format` 參數，預設會建立 `pem` 檔案。

`--query "KeyMaterial"` 將私有金鑰材料列印到輸出。

`--output text > my-key-pair.pem` 會將私有金鑰資料儲存在有特定副檔名的檔案中。副檔名可為 `.pem` 或 `.ppk`。私有金鑰的名稱可以與公有金鑰名稱不同，但為了方便使用，請使用相同的名稱。

```
aws ec2 create-key-pair \
 --key-name my-key-pair \
 --key-type rsa \
 --key-format pem \
 --query "KeyMaterial" \
 --output text > my-key-pair.pem
```

2. (Linux 執行個體) 如果您打算在 macOS 或 Linux 電腦上使用安全殼層用戶端連線到 Linux 執行個體，請使用下列指令來設定私密金鑰檔案的權限，以便只有您可以讀取它。

```
chmod 400 key-pair-name.pem
```

如果您未設定這些許可，則無法使用此金鑰對來連接至您的執行個體。如需詳細資訊，請參閱 [錯誤：未受保護的私有金鑰檔案](#)。

## PowerShell

### 使用 Amazon EC2 建立金鑰對

使用如下 [New-EC2KeyPair](#) AWS Tools for Windows PowerShell 命令來生成密鑰並將其保存到 .pem 或 .ppk 文件中。

對於 `-KeyName`，指定公有金鑰的名稱。名稱最多為 255 個 ASCII 字元。

對於 `-KeyType`，指定 `rsa` 或 `ed25519`。如果您未包含 `-KeyType` 參數，預設會建立 `rsa` 金鑰。請注意，Windows 執行個體不支援 `ED25519` 金鑰。

對於 `-KeyFormat`，指定 `pem` 或 `ppk`。如果您未包含 `-KeyFormat` 參數，預設會建立 `pem` 檔案。

`KeyMaterial` 將私有金鑰材料列印到輸出。

`Out-File -Encoding ascii -FilePath C:\path\my-key-pair.pem` 會將私有金鑰資料儲存在有特定副檔名的檔案中。副檔名可為 `.pem` 或 `.ppk`。私有金鑰的名稱可以與公有金鑰名稱不同，但為了方便使用，請使用相同的名稱。

```
PS C:\> (New-EC2KeyPair -KeyName "my-key-pair" -KeyType "rsa" -KeyFormat "pem").KeyMaterial | Out-File -Encoding ascii -FilePath C:\path\my-key-pair.pem
```

## 使用建立 key pair AWS CloudFormation

使用建立新 key pair 時 AWS CloudFormation，私密金鑰會儲存至 AWS Systems Manager 參數存放區。參數名稱具有以下格式：

```
/ec2/keypair/key_pair_id
```

如需詳細資訊，請參閱《AWS Systems Manager 使用者指南》中的 [AWS Systems Manager 參數存放區](#)。

若要使用建立 key pair AWS CloudFormation

1. 在範本中指定 [AWS::EC2::KeyPair](#) 資源。

```
Resources:
 NewKeyPair:
 Type: 'AWS::EC2::KeyPair'
 Properties:
 KeyName: new-key-pair
```

2. 按如下說明，使用 [describe-key-pairs](#) 命令以取得金鑰對 ID。

```
aws ec2 describe-key-pairs --filters Name=key-name,Values=new-key-pair --query KeyPairs[*].KeyPairId --output text
```

下列為範例輸出。

```
key-05abb699beEXAMPLE
```

- 按如下說明，使用 [get-parameter](#) 命令以取得金鑰參數，並將金鑰材料儲存在 `.pem` 檔案中。

```
aws ssm get-parameter --name /ec2/keypair/key-05abb699beEXAMPLE --with-decryption --query Parameter.Value --output text > new-key-pair.pem
```

## 所需的 IAM 許可

若 AWS CloudFormation 要啟用以代表您管理參數存放區參數，您 AWS CloudFormation 或您的使用者所擔任的 IAM 角色必須具有下列權限：

- `ssm:PutParameter` – 授予許可，以建立私有金鑰材料的參數。
- `ssm>DeleteParameter` - 准許刪除用來存放私有金鑰材料的參數。無論金鑰對是否是由 AWS CloudFormation 匯入或建立，都需要此許可。

AWS CloudFormation 刪除由堆疊建立或匯入的 key pair 時，AWS CloudFormation 會執行權限檢查以判斷您是否具有刪除參數的權限，即使只在建立 key pair 時建立參數，而不是在匯入 key pair 時建立參數。AWS CloudFormation 使用與您帳戶中任何參數不相符的製造參數名稱來測試所需權限。因此，您可能會在 `AccessDeniedException` 錯誤訊息中看到虛構參數名稱。

## 使用第三方工具來建立金鑰對，然後將公有金鑰匯入 Amazon EC2

### Linux 執行個體

您可以不使用 Amazon EC2 來建立金鑰對，而改用第三方工具來建立 RSA 或 ED25519 金鑰對，然後再將公有金鑰匯入 Amazon EC2。

### 金鑰對的需求

- 支援的類型：RSA 和 ED25519。Amazon EC2 不接受 DSA 金鑰。
- 支援的格式：

- OpenSSH 公開金鑰格式 (格式為 `~/.ssh/authorized_keys`)。如果您在使用 EC2 Instance Connect API 時以 SSH 連線，則還支援 SSH2 格式。
- SSH 私有金鑰檔案格式必須是 PEM 或 PPK
- (僅 RSA) Base64 編碼的 DER 格式
- (僅 RSA) 如 [RFC 4716](#) 中所指定的 SSH 公有金鑰檔案格式
- 支援的長度：1024、2048 和 4096。如果您在使用 EC2 Instance Connect API 時以 SSH 連線，則支援的長度為 2048 和 4096。

### 使用第三方工具來產生金鑰對

1. 使用您自選的第三方工具來產生金鑰對 例如：您可以使用 `ssh-keygen` (標準 OpenSSH 安裝隨附的工具)。或者，Java、Ruby、Python 和其他多種程式設計語言提供了標準程式庫，您可以使用這些程式庫來產生 RSA 或 ED25519 金鑰對。

#### Important

私有金鑰必須是 PEM 或 PPK 格式。例如：使用 `ssh-keygen -m PEM` 來產生 PEM 格式的 OpenSSH 金鑰。

2. 將公有金鑰儲存到本機檔案。例如，`~/.ssh/my-key-pair.pub`。此檔案的副檔名不重要。
3. 將私有金鑰儲存到具有 `.pem` 或 `.ppk` 副檔名的本機檔案。例如 `~/.ssh/my-key-pair.pem` 或 `~/.ssh/my-key-pair.ppk`。

#### Important

將私有金鑰檔案存放在安全的地方。您在啟動執行個體時需要提供公有金鑰的名稱，且每次連線至執行個體時也需要提供對應的私有金鑰。

### Windows 執行個體

您可以不使用 Amazon EC2 來建立金鑰對，而改用第三方工具來產生 RSA 金鑰對，然後再將公有金鑰匯入 Amazon EC2。

### 金鑰對的需求

- 支援的類型：RSA。Amazon EC2 不接受 DSA 金鑰。

**Note**

Windows 執行個體不支援 ED25519 金鑰。

- 支援的格式：
  - 公開金鑰格式
  - SSH 私有金鑰檔案格式必須是 PEM 或 PPK
  - (僅 RSA) Base64 編碼的 DER 格式
  - (僅 RSA) 如 [RFC 4716](#) 中所指定的 SSH 公有金鑰檔案格式
- 支援的長度：1024、2048 和 4096。

### 使用第三方工具來產生金鑰對

1. 使用您自選的第三方工具來產生金鑰對 例如：您可以使用 ssh-keygen (標準 OpenSSH 安裝隨附的工具)。或者，Java、Ruby、Python 和其他多種程式設計語言提供了標準程式庫，您可以使用這些程式庫來產生 RSA 金鑰對。

**Important**

私有金鑰必須是 PEM 或 PPK 格式。例如：使用 `ssh-keygen -m PEM` 來產生 PEM 格式的 OpenSSH 金鑰。

2. 將公有金鑰儲存到本機檔案。例如，`C:\keys\my-key-pair.pub`。此檔案的副檔名不重要。
3. 將私有金鑰儲存到具有 `.pem` 或 `.ppk` 副檔名的本機檔案。例如 `C:\keys\my-key-pair.pem` 或 `C:\keys\my-key-pair.ppk`。此檔案的副檔名很重要，因為從 EC2 主控台連線到 Windows 執行個體時，只 `.pem` 能選取檔案。

**Important**

將私有金鑰檔案存放在安全的地方。您在啟動執行個體時需要提供公有金鑰的名稱，且每次連線至執行個體時也需要提供對應的私有金鑰。

建立金鑰對之後，請使用以下其中一個方法，將您的公有金鑰匯入 Amazon EC2。

## Console

將公有金鑰匯入至 Amazon EC2

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Key Pairs (金鑰對)。
3. 選擇 Import key pair (匯入金鑰對)。
4. 在 Name (名稱) 處，輸入公有金鑰的描述性名稱。名稱最多可包含 255 個 ASCII 字元。它不能包括前置或結尾空格。

### Note

當您從 EC2 主控台連線到執行個體時，主控台會為您的私有金鑰檔案名稱建議此名稱。

5. 選取 Browse (瀏覽) 以導覽並選取您的公有金鑰，或將公有金鑰的內容貼到 Public key contents (公有金鑰內容) 欄位。
6. 選擇 Import key pair (匯入金鑰對)。
7. 確認您匯入的公開金鑰出現在金鑰對清單中。

## AWS CLI

將公有金鑰匯入至 Amazon EC2

使用 [import-key-pair](#) AWS CLI 命令。

確認是否已成功匯入金鑰對

使用 [describe-key-pairs](#) AWS CLI 命令。

## PowerShell

將公有金鑰匯入至 Amazon EC2

使用 [Import-EC2KeyPair](#) AWS Tools for Windows PowerShell 命令。

確認是否已成功匯入金鑰對

使用 [Get-EC2KeyPair](#) AWS Tools for Windows PowerShell 命令。



## 標記金鑰對

為了協助分類和管理您使用 Amazon EC2 建立或匯入到 Amazon EC2 的金鑰配對，您可以使用自訂中繼資料標記它們。如需有關標籤運作方式的詳細資訊，請參閱[標記您的 Amazon EC2 資源](#)。

### Console

檢視、新增或刪除 key pair 的標籤

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Key Pairs (金鑰對)。
3. 選取公有金鑰，然後選取 Actions (動作)、Manage tags (管理標籤)。
4. Manage tags (管理標籤) 頁面會顯示指派給公有金鑰的任何標籤。
  - 若要加入標籤，請選擇 Add tag (新增標籤)，然後輸入標籤索引鍵和值。每個金鑰最多可新增 50 個標籤。如需詳細資訊，請參閱 [標籤限制](#)。
  - 若要移除標籤，請選擇標籤旁的 Remove (移除) 以移除。
5. 選擇 Save (儲存)。

### AWS CLI

檢視金鑰配對的標籤

使用 [describe-tags](#) AWS CLI 命令。下列範例描述所有公有金鑰的標籤。

```
aws ec2 describe-tags --filters "Name=resource-type,Values=key-pair"
```

```
{
 "Tags": [
 {
 "Key": "Environment",
 "ResourceId": "key-0123456789EXAMPLE",
 "ResourceType": "key-pair",
 "Value": "Production"
 },
 {
 "Key": "Environment",
 "ResourceId": "key-9876543210EXAMPLE",
```

```
 "ResourceType": "key-pair",
 "Value": "Production"
]}
}
```

描述 key pair 的標籤

使用 [describe-key-pairs](#) AWS CLI 命令。

```
aws ec2 describe-key-pairs --key-pair-ids key-0123456789EXAMPLE
```

```
{
 "KeyPairs": [
 {
 "KeyName": "MyKeyPair",
 "KeyFingerprint":
"1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f",
 "KeyPairId": "key-0123456789EXAMPLE",
 "Tags": [
 {
 "Key": "Environment",
 "Value": "Production"
 }
]
 }
]
}
```

標記 key pair

使用 [create-tags](#) AWS CLI 命令。在下列範例中，會使用 Key=Cost-Center 和 Value=CC-123 來標記公有金鑰。

```
aws ec2 create-tags --resources key-0123456789EXAMPLE --tags Key=Cost-Center,Value=CC-123
```

從金鑰對刪除標籤

使用 [delete-tags](#) AWS CLI 命令。如需範例，請參閱 AWS CLI 命令參考中的 [範例](#)。

PowerShell

若要檢視金鑰配對的標籤

使用 [Get-EC2Tag](#) 命令。

描述 key pair 的標籤

使用 [Get-EC2KeyPair](#) 命令。

標記 key pair

使用 [New-EC2Tag](#) 命令。

從金鑰對刪除標籤

使用 [Remove-EC2Tag](#) 命令。

## 描述您的金鑰配對

您可以說明存放在 Amazon EC2 中的金鑰配對。您也可以擷取公有金鑰材料，並辨識啟動時指定的公有金鑰。

主題

- [描述您的金鑰配對](#)
- [擷取公有金鑰資料](#)
- [識別啟動時指定的公有金鑰](#)

## 描述您的金鑰配對

您可以查看存放於 Amazon EC2 中公有金鑰的以下資訊：公有金鑰名稱、ID、金鑰類型、指紋、公有金鑰資料、Amazon EC2 建立金鑰的日期和時間 (UTC 時區) (若金鑰是由第三方工具所建立，則為該金鑰匯入 Amazon EC2 的日期及時間)，以及任何與公有金鑰相關聯的標籤。

您可以使用 Amazon EC2 主控台或 AWS CLI 檢視有關公開金鑰的資訊。

Console

檢視您的公有金鑰相關資訊

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽器中，選擇 Key Pairs (金鑰對)。

3. 您可以在金鑰對表中檢視每個公有金鑰的相關資訊。

**Key pairs (23)** [Info](#)

Filter key pairs

<input type="checkbox"/>	Name	Type	Created	Fingerprint	ID
<input type="checkbox"/>		ed25519	2021/08/05 10:06 GMT+2	xeDxC7/IVRZ8mFlzsKidfQ2FcfWig4C3...	key-
<input type="checkbox"/>		rsa	2020/05/13 17:16 GMT+2	ed:71:62:da:a4:d1:f6:47:61:4b:d1:a7:2...	key-

4. 若要檢視公有金鑰標籤，請勾選金鑰旁的核取方塊，然後選取 Actions (動作)、Manage tags (管理標籤)。

## AWS CLI

### 描述公有金鑰

使用 [describe-key-pairs](#) 命令，並指定 `--key-names` 參數。

```
aws ec2 describe-key-pairs --key-names key-pair-name
```

### 範例輸出

```
{
 "KeyPairs": [
 {
 "KeyPairId": "key-0123456789example",
 "KeyFingerprint":
"1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f",
 "KeyName": "key-pair-name",
 "KeyType": "rsa",
 "Tags": [],
 "CreateTime": "2022-04-28T11:37:26.000Z"
 }
]
}
```

您也可以不要 `--key-names`，而是指定 `--key-pair-ids` 參數來識別公有金鑰。

```
aws ec2 describe-key-pairs --key-pair-ids key-0123456789example
```

若要查看輸出中的公有金鑰資料，您必須指定 `--include-public-key` 參數。

```
aws ec2 describe-key-pairs --key-names key-pair-name --include-public-key
```

範例輸出：在輸出中，PublicKey 欄位會包含公有金鑰資料。

```
{
 "KeyPairs": [
 {
 "KeyPairId": "key-0123456789example",
 "KeyFingerprint":
"1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f",
 "KeyName": "key-pair-name",
 "KeyType": "rsa",
 "Tags": [],
 "PublicKey": "ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIij7azlDjVHAsSxgcpCRZ3oWnTm0nAFM64y9jd22ioI/ my-key-pair",
 "CreateTime": "2022-04-28T11:37:26.000Z"
 }
]
}
```

## 擷取公有金鑰資料

您可以使用多種不同的方法來存取公有金鑰資料。您可以從本機電腦上相符的私密金鑰、使用公開金鑰啟動的執行個體中繼資料，或使用describe-key-pairs AWS CLI 指令擷取公開金鑰資料。對於 Linux 執行個體，也可以從執行個體上的authorized\_keys檔案擷取公開金鑰材料。

使用下列其中一種方法擷取公有金鑰材料。

### Linux 執行個體

#### From the private key

從私有金鑰中擷取公有金鑰材料

在本機的 Linux 或 macOS 電腦上，您可以使用 ssh-keygen 命令來擷取金鑰對的公有金鑰。指定已將私有金鑰 (.pem 檔案) 下載至其中的路徑。

```
ssh-keygen -y -f /path_to_key_pair/my-key-pair.pem
```

此命令會傳回公有金鑰，如下列範例所示。

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQAClKsfkNkuSevGj3eYhCe53pcjqP3maAhDFcvBS706V
hz2ItxCih+PnDSUaw+WNQn/mZphTk/a/gU8jEzo0WbkM4yxyb/wB96xbiFveSFJuOp/d6RJhJ0I0iBXr
lsLnBItnctkiJ7FbtXJMXLvwwJryDUi1BMTjYtwB+QhYXUM0zce5Pjz5/i8SeJtjnV3iAoG/cQk+0FzZ
qaeJAAHco+CY/5WrUBkrHmFJr6HcXkvJdWPkYQS3xqC0+FmUZofz221CBt5IMucxXPkX4rWi+z7wB3Rb
BQoQzd8v7yeb70z1PnW0yN0qFU0XA246RA8QFYiCNYwI3f05p6KLxEXAMPLE
```

如果此命令失敗，請執行下列命令，來確保您已在私有金鑰對檔案中變更許可，以便只有您能夠檢視它。

```
chmod 400 key-pair-name.pem
```

## From the instance metadata

您可以使用執行個體中繼資料服務第 2 版或執行個體中繼資料服務第 1 版，從執行個體中繼資料擷取公有金鑰。

### Note

如果變更用於連接到執行個體的金鑰對，Amazon EC2 不會更新執行個體中繼資料以顯示新的公有金鑰。執行個體中繼資料會繼續顯示啟動執行個體時指定之金鑰對的公有金鑰。

從執行個體中繼資料擷取公有金鑰

從執行個體使用下列其中一個命令。

### IMDSV2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-
aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-
data/public-keys/0/openssh-key
```

### IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
```

範例輸出

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQAClKsfkNkuSevGj3eYhCe53pcjqP3maAhDFcvBS706V
hz2ItxCih+PnDSUaw+WNQn/mZphTk/a/gU8jEzo0WbkM4yxyb/wB96xbiFveSFJuOp/d6RJhJ0I0iBXr
```

```
lsLnBItnctckiJ7FbtXJMXLvwwJryDUiLBMTjYtwB+QhYXUM0zce5Pjz5/i8SeJtjnV3iAoG/cQk+0FzZ
qaeJAAHco+CY/5WrUBkrHmFJr6HcXkvJdWPkYQS3xqC0+FmUZofz221CBt5IMucxXPkX4rWi+z7wB3Rb
BQoQzd8v7yeb70z1PnW0yN0qFU0XA246RA8QFYiCNYwI3f05p6KLxEXAMPLE key-pair-name
```

如需執行個體中繼資料的詳細資訊，請參閱[擷取執行個體中繼資料](#)。

From the instance

若您在啟動 Linux 執行個體時指定金鑰對，當執行個體第一次啟動時，公有金鑰內容會放置在 `~/.ssh/authorized_keys` 內項目的執行個體上。

從執行個體擷取公有金鑰材料

1. [連線到您的執行個體](#)。
2. 在終端機視窗中，使用您最愛的文字編輯器 (例如 `authorized_keys` 或 `vim`) 開啟 `nano` 檔案。

```
[ec2-user ~]$ nano ~/.ssh/authorized_keys
```

`authorized_keys` 檔案開啟，會顯示公有金鑰，後面為金鑰對的名稱。下列是名為 *key-pair-name* 之金鑰對的輸入內容範例。

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQCLKsfkNkuSevGj3eYhCe53pcjqP3maAhDFcvBS706V
hz2ItxCih+PnDSUaw+WNQn/mZphTk/a/gU8jEzo0WbkM4yxyb/wB96xbiFveSFJu0p/d6RJhJ0I0iBXr
lsLnBItnctckiJ7FbtXJMXLvwwJryDUiLBMTjYtwB+QhYXUM0zce5Pjz5/i8SeJtjnV3iAoG/cQk+0FzZ
qaeJAAHco+CY/5WrUBkrHmFJr6HcXkvJdWPkYQS3xqC0+FmUZofz221CBt5IMucxXPkX4rWi+z7wB3Rb
BQoQzd8v7yeb70z1PnW0yN0qFU0XA246RA8QFYiCNYwI3f05p6KLxEXAMPLE key-pair-name
```

From describe-key-pairs

使用 **describe-key-pairs** AWS CLI 命令擷取公有金鑰

使用 [describe-key-pairs](#) 命令，然後指定 `--key-names` 參數來識別公有金鑰。若要在輸出中包含公有金鑰資料，請指定 `--include-public-key` 參數。

```
aws ec2 describe-key-pairs --key-names key-pair-name --include-public-key
```

範例輸出：在輸出中，`PublicKey` 欄位會包含公有金鑰資料。

```
{
```

```

"KeyPairs": [
 {
 "KeyPairId": "key-0123456789example",
 "KeyFingerprint":
"1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f",
 "KeyName": "key-pair-name",
 "KeyType": "rsa",
 "Tags": [],
 "PublicKey": "ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIIj7az1DjVHAsSxgcpCRZ3oWnTm0nAFM64y9jd22ioI/ my-key-pair",
 "CreateTime": "2022-04-28T11:37:26.000Z"
 }
]
}

```

您也可以不要 `--key-names`，而是指定 `--key-pair-ids` 參數來識別公有金鑰。

```
aws ec2 describe-key-pairs --key-pair-ids key-0123456789example --include-public-key
```

## Windows 執行個體

### From the private key

從私有金鑰中擷取公有金鑰材料

在本機的 Windows 電腦上，您可以使用 PuTTYgen 來擷取金鑰對的公有金鑰。

啟動 PuTTYgen 並選擇 Load (載入)。選取 `.ppk` 或 `.pem` 私有金鑰檔案。PuTTYgen 會在 Public key for pasting into OpenSSH authorized\_keys file (貼到 OpenSSH authorized\_keys 檔案的公有金鑰) 下顯示公有金鑰。您也可以選擇 Save public key (儲存公有金鑰)、指定檔案名稱、儲存檔案，然後開啟檔案，以檢視公有金鑰。

### From the instance metadata

您可以使用執行個體中繼資料服務第 2 版或執行個體中繼資料服務第 1 版，從執行個體中繼資料擷取公有金鑰。

#### Note

如果變更用於連接到執行個體的金鑰對，Amazon EC2 不會更新執行個體中繼資料以顯示新的公有金鑰。執行個體中繼資料會繼續顯示啟動執行個體時指定之金鑰對的公有金鑰。



從執行個體中繼資料擷取公有金鑰

從執行個體使用下列其中一個命令。

### IMDSV2

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
```

### IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
```

### 範例輸出

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQCLKsfkNkuSevGj3eYhCe53pcjqP3maAhDFcvBS706Vhz2ItxCih+PnDSUaw+WNQn/mZphTk/a/gU8jEzo0WbkM4yxyb/wB96xbiFveSFJuOp/d6RJhJOI0iBXrLsLnBITntckiJ7FbtXJMXLvvwJryDUiLBMTjYtwB+QhYXUM0zce5Pjz5/i8SeJtjnV3iAoG/cQk+0FzZqaeJAAHco+CY/5WrUBkrHmFJr6HcXkvJdWpkYQS3xqC0+FmUZofz221CBt5IMucxXPkX4rWi+z7wB3RbBQoQzd8v7yeb70z1PnW0yN0qFU0XA246RA8QFYiCNYwI3f05p6KLxEXAMPLE key-pair-name
```

如需執行個體中繼資料的詳細資訊，請參閱[擷取執行個體中繼資料](#)。

From describe-key-pairs

使用 **describe-key-pairs** AWS CLI 命令擷取公有金鑰

使用 [describe-key-pairs](#) 命令，然後指定 `--key-names` 參數來識別公有金鑰。若要在輸出中包含公有金鑰資料，請指定 `--include-public-key` 參數。

```
aws ec2 describe-key-pairs --key-names key-pair-name --include-public-key
```

範例輸出：在輸出中，PublicKey 欄位會包含公有金鑰資料。

```
{
 "KeyPairs": [
 {
```

```
 "KeyPairId": "key-0123456789example",
 "KeyFingerprint":
"1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f",
 "KeyName": "key-pair-name",
 "KeyType": "rsa",
 "Tags": [],
 "PublicKey": "ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIIj7azlDjVHAsSxgcpCRZ3oWnTm0nAFM64y9jd22ioI/ my-key-pair",
 "CreateTime": "2022-04-28T11:37:26.000Z"
 }
]
```

您也可以不要 `--key-names`，而是指定 `--key-pair-ids` 參數來識別公有金鑰。

```
aws ec2 describe-key-pairs --key-pair-ids key-0123456789example --include-public-key
```

## 識別啟動時指定的公有金鑰

如果您在啟動執行個體時指定了公有金鑰，執行個體會記錄公有金鑰名稱。

### 識別啟動時指定的公有金鑰

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取您的執行個體。
3. 在詳細資訊標籤中，執行個體詳細資訊下的啟動時指派的金鑰對欄位會顯示您在啟動執行個體時指定的公有金鑰名稱。

#### Note

即使您變更執行個體上的公有金鑰，或新增公有金鑰，啟動時指派的金鑰對欄位的值也不會變更。

## 刪除您的金鑰對

您可以刪除 key pair，這會移除存放在 Amazon EC2 中的公開金鑰。刪除 key pair 並不會刪除相符的私密金鑰。

使用下列方法刪除公有金鑰時，只會刪除您在[建立](#)或[匯入](#)金鑰對時儲存在 Amazon EC2 中的公有金鑰。無論在啟動執行個體時或啟動之後，移除公有金鑰都不會從您新增金鑰的任何執行個體中移除公有金鑰。它也不會刪除您本機電腦上的私有金鑰。只要您仍然擁有私有金鑰 (.pem) 檔案，即可繼續使用從 Amazon EC2 刪除的公有金鑰來連線至已啟動的執行個體。

### Important

如果使用 Auto Scaling 群組 (例如，在 Elastic Beanstalk 環境中)，請確保在相關聯的啟動範本或啟動組態中並未指定您要刪除的公有金鑰。如果 Amazon EC2 Auto Scaling 偵測到運作狀態不良的執行個體，則會啟動替代執行個體。不過，如果找不到公有金鑰，執行個體啟動就會失敗。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南中的[啟動範本](#)。

## Console

刪除 Amazon EC2 上的公有金鑰

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Key Pairs (金鑰對)。
3. 選取要刪除的金鑰對，然後選取 Actions (動作)、Delete (刪除)。
4. 在確認欄位中，輸入 Delete，然後選擇 Delete (刪除)。

## AWS CLI

刪除 Amazon EC2 上的公有金鑰

使用 [delete-key-pair](#) AWS CLI 命令。

## PowerShell

刪除 Amazon EC2 上的公有金鑰

使用 [Remove-EC2KeyPair](#) AWS Tools for Windows PowerShell 命令。

## 在 Linux 執行個體上新增或移除公開金鑰

如果遺失私密金鑰，您將無法存取任何使用 key pair 的執行個體。如需使用與您在啟動時指定之金鑰組不同的 key pair 連線至執行個體的詳細資訊，請參閱[我遺失私密金鑰](#)。

當您啟動執行個體時，您可[指定金鑰對](#)。若您在啟動時指定金鑰對，當執行個體第一次啟動時，公有金鑰材料會放置在 `~/.ssh/authorized_keys` 內項目的 Linux 執行個體上。

您可以變更改用來存取執行個體預設系統帳戶的金鑰對，方法是在執行個體上新增公有金鑰，或是取代執行個體上的公有金鑰 (刪除現有公有金鑰並在執行個體上新增一個)。您也可以從執行個體中移除所有公有金鑰。若要新增或取代金鑰對，您必須能夠連線至您的執行個體。

您可能基於下列原因新增或取代 key pair：

- 如果組織中的使用者需要使用不同的金鑰對來存取系統使用者，您可以將該共有金鑰新增到執行個體。
- 如果某些人擁有私有金鑰的複本 (.pem 檔案)，而您希望阻止他們連線到您的執行個體 (例如，如果他們已離開您的組織)，則您可刪除執行個體上的公有金鑰，並以新的金鑰取代。
- 如果您從執行個體中建立 Linux AMI，則公有金鑰材料會從執行個體複製到 AMI。如果從 AMI 中啟動執行個體，則新的執行個體會包含來自原始執行個體的公有金鑰。若要防止擁有私有金鑰的某些人連線到新執行個體，請從原始執行個體中移除公有金鑰，然後再建立 AMI。

使用下列程序修改預設使用者的 key pair，例如 `ec2-user`。如需將使用者新增至執行個體的相關資訊，請參閱執行個體上作業系統的說明文件。

### 新增或替換金鑰對

1. 使用 [Amazon EC2 主控台](#) 或 [第三方工具](#) 來建立新的金鑰對。
2. 從新的金鑰對擷取公有金鑰。如需詳細資訊，請參閱 [擷取公有金鑰資料](#)。
3. 使用現有的私有金鑰，[連線到執行個體](#)。
4. 使用您自選的文字編輯器，在執行個體上開啟 `.ssh/authorized_keys` 檔案。貼上新金鑰對的公有金鑰資訊 (在現有公有金鑰資訊的下方)。儲存檔案。
5. 中斷與執行個體的連線，然後測試是否能使用新的私有金鑰檔案來連線到執行個體。
6. (選用) 如果要替換現有的金鑰對，請連線到執行個體，然後從 `.ssh/authorized_keys` 檔案，刪除原始金鑰對的公有金鑰資訊。

#### Important

如果使用 Auto Scaling 群組，請確保在啟動範本或啟動組態中並未指定您要取代的金鑰對。如果 Amazon EC2 Auto Scaling 偵測到運作狀態不良的執行個體，則會啟動替代執行個體。不

過，如果找不到金鑰對，執行個體啟動就會失敗。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南中的[啟動範本](#)。

若要從執行個體中移除公有金鑰

1. [連線到您的執行個體](#)。
2. 使用您自選的文字編輯器，在執行個體上開啟 `.ssh/authorized_keys` 檔案。刪除公有金鑰資訊，然後儲存檔案。

#### Warning

從執行個體中移除所有公有金鑰，並從執行個體中斷連線後，除非 AMI 提供另一種登入方式，否則無法再次連線。

## 驗證金鑰對的指紋

若要驗證金鑰對的指紋，請將 Amazon EC2 主控台金鑰對頁面上顯示的指紋，或由 [describe-key-pairs](#) 命令傳回的指紋，與您使用本機電腦上的私密金鑰產生的指紋進行比較。這些指紋應該相符。

當 Amazon EC2 計算指紋時，Amazon EC2 可能會使用 `=` 字元向指紋附加填補。其他工具，例如 `ssh-keygen`，可能會忽略此填補。

如果您嘗試驗證 Linux EC2 執行個體的指紋，而不是 key pair 的指紋，請參閱[取得執行個體指紋](#)。

## 如何計算指紋

Amazon EC2 使用不同的雜湊函數來計算 RSA 和 ED25519 金鑰對的指紋。此外，對於 RSA 金鑰對，Amazon EC2 使用不同的雜湊函數來計算指紋，具體取決於金鑰對是由 Amazon EC2 建立或是匯入至 Amazon EC2。

下表列出了針對由 Amazon EC2 建立和匯入至 Amazon EC2 的 RSA 和 ED25519 金鑰對，用來計算其指紋的雜湊函數。

(Linux 實例) 用於計算指紋的雜湊函數

金鑰對來源	RSA 金鑰配對 (視窗和 Linux)	ED25519 金鑰配對
由 Amazon EC2 建立	SHA-1	SHA-256

金鑰對來源	RSA 金鑰配對 (視窗和 Linux)	ED25519 金鑰配對
匯入至 Amazon EC2	MD5 <sup>1</sup>	SHA-256

<sup>1</sup> 如果您將公有 RSA 金鑰匯入至 Amazon EC2，則會使用 MD5 雜湊函數來計算指紋。無論您如何建立金鑰對 (例如透過使用第三方工具，或從使用 Amazon EC2 建立的現有私有金鑰產生新的公有金鑰)，都是如此。

## 在不同區域中使用相同的金鑰對

如果您打算使用相同的 key pair 連線至不同的執行個體 AWS 區域，則必須將公開金鑰匯入您要使用該金鑰的所有區域。如果您使用 Amazon EC2 來建立金鑰對，則可以 [擷取公有金鑰資料](#)，以便將公有金鑰匯入至其他區域。

### Note

- 如果您使用 Amazon EC2 建立 RSA 金鑰對，然後使用 Amazon EC2 私有金鑰產生公有金鑰，則匯入的公有金鑰將具有與原始公有金鑰不同的指紋。這是因為使用 Amazon EC2 建立的原始 RSA 金鑰的指紋是使用 SHA-1 雜湊函數計算，而匯入的 RSA 金鑰的指紋則使用 MD5 雜湊函數計算。
- 對於 ED25519 金鑰對，因為使用相同的 SHA-256 雜湊函數來計算指紋，所以無論是由 Amazon EC2 建立或是匯入至 Amazon EC2，指紋均相同。

## 從私有金鑰產生指紋

使用下列命令之一從本機電腦上的私有金鑰產生指紋。

如果您使用的是 Windows 本機電腦，您可以使用 Windows Subsystem for Linux (WSL) 來執行下列命令。使用 [Windows 10 安裝指南](#) 中的指示來安裝 WSL 和 Linux 發行版本。指示中的範例是安裝 Linux 的 Ubuntu 發行版本，但您可以安裝任何發行版本。您將會收到提示，要求您重新啟動電腦以讓變更生效。

- 如果是使用 Amazon EC2 建立金鑰對

如下列範例所示，使用 OpenSSL 工具來產生指紋。

對於 RSA 金鑰對：

```
openssl pkcs8 -in path_to_private_key -inform PEM -outform DER -topk8 -nocrypt |
openssl sha1 -c
```

(Linux 執行個體) 針對 ED25519 金鑰配對：

```
ssh-keygen -l -f path_to_private_key
```

- (僅限 RSA 金鑰對) 如果您已將公有金鑰匯入 Amazon EC2

無論您如何建立金鑰對 (例如透過使用第三方工具，或從使用 Amazon EC2 建立的現有私有金鑰產生新的公有金鑰)，您都可以遵循此程序

如下列範例所示，使用 OpenSSL 工具來產生指紋。

```
openssl rsa -in path_to_private_key -pubout -outform DER | openssl md5 -c
```

- 如果您已使用 OpenSSH 7.8 或更新版本建立 OpenSSH 金鑰對，並將公有金鑰匯入至 Amazon EC2

如下列範例所示，使用 ssh-keygen 來產生指紋。

對於 RSA 金鑰對：

```
ssh-keygen -ef path_to_private_key -m PEM | openssl rsa -RSAPublicKey_in -outform DER
| openssl md5 -c
```

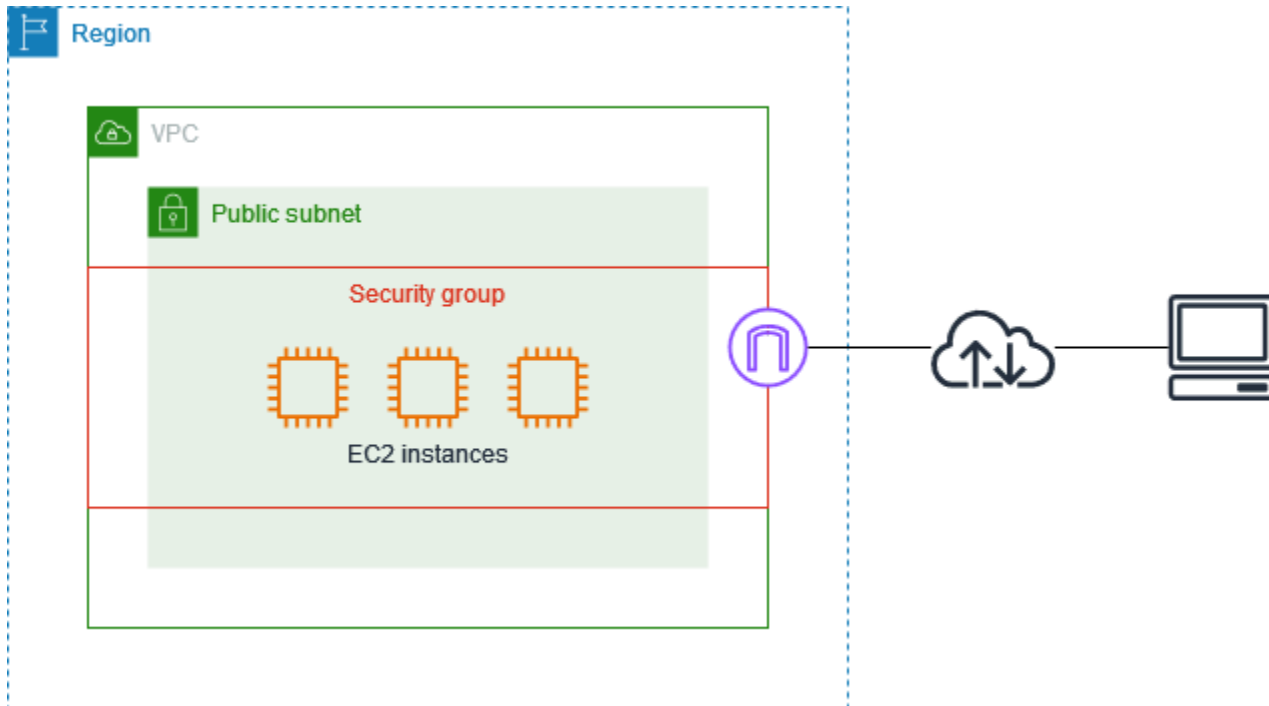
(Linux 執行個體) 針對 ED25519 金鑰配對：

```
ssh-keygen -l -f path_to_private_key
```

## 適用於 Amazon EC2 執行個體的亞馬遜 EC2 安全群組

安全群組會做為您 EC2 執行個體的虛擬防火牆，控制傳入及傳出流量。傳入規則會控制傳入至您的執行個體的流量，以及傳出規則會控制從您的執行個體傳出的流量。啟動執行個體時，您會為其指派一個或多個安全群組。如果您未指定安全群組，Amazon EC2 便會使用 VPC 的預設安全群組。您可在各個安全群組中新增規則，允許流量往返於建立關聯的執行個體。您可隨時修改安全群組規則。新的、修改過的規則會自動套用至與安全群組建立關聯的所有執行個體。當 Amazon EC2 決定是否要允許流量觸達執行個體時，我們會評估所有與執行個體建立關聯之安全群組的所有規則。

下圖顯示具有子網路、網際網路閘道和安全群組的 VPC。子網路包含 EC2 執行個體。安全性群組會指派給執行個體。僅有安全群組規則允許的流量才能到達執行個體。例如，如果安全性群組包含允許來自網路的 SSH 流量的規則，您就可以使用 SSH 從電腦連線到執行個體。如果安全性群組包含允許來自指派給該群組之資源的所有流量的規則，則每個執行個體都可以接收從其他執行個體傳送的任何流量。



在您啟動執行個體之後，您可變更其安全群組。安全群組與網路介面關聯。變更執行個體的安全群組也會變更與主要網路介面 (eth0) 關聯的安全群組。如需詳細資訊，請參閱 [變更執行個體的安全群組](#)。您也可以變更與任何其他網路介面關聯的安全群組。如需詳細資訊，請參閱 [修改網路介面屬性](#)。

安全是 AWS 與您之間共同的責任。如需詳細資訊，請參閱 [Amazon EC2 中的安全](#)。AWS 提供安全群組做為保護執行個體安全性的工具之一，因此您需要設定這些群組以符合安全性需求。若您有安全群組無法完全滿足的需求，您可以在使用安全群組的同時在您的任何執行個體上維持您自己的防火牆。

使用安全群組無需額外收費。

## 目錄

- [安全群組規則](#)
- [安全群組連線追蹤](#)
- [預設和自訂安全性群組](#)
- [使用安全群組](#)
- [不同使用案例的安全群組規則](#)



## 安全群組規則

安全群組的規則可控制允許觸達與安全群組相關聯之執行個體的對內流量。規則也會控制允許離開的對外流量。

以下為安全群組規則的特性：

- 根據預設，安全群組會包含允許所有傳出流量的規則。您可以刪除這些規則。請注意，根據預設 Amazon EC2 會封鎖連接埠 25 上的流量。如需詳細資訊，請參閱 [使用通訊埠 25 傳送的電子郵件限制](#)。
- 安全群組規則一律為許可制。您無法建立拒絕存取的規則。
- 安全群組規則可讓您根據通訊協定和連接埠號碼篩選流量。
- 安全群組是具有狀態的群組—若您從執行個體傳送請求，該請求的回應流量將允許流入，與對內安全群組規則無關。針對 VPC 安全群組，這也表示獲得允許之對內流量的回應也將允許流出，與對外規則無關。如需詳細資訊，請參閱 [安全群組連線追蹤](#)。
- 您可以隨時新增和移除規則。您的變更會自動套用到與安全群組關聯的執行個體。

有些規則變更的效果可取決於追蹤流量的方式。如需詳細資訊，請參閱 [安全群組連線追蹤](#)。

- 當您將多個安全群組與執行個體建立關聯時，每個安全群組的規則都會有效彙總並建立一組規則。Amazon EC2 會使用這一組規則判斷是否要允許存取。

您可以將多個安全性群組指派給執行個體。因此，執行個體可以有數百個適用的規則。這可能會在您存取執行個體時產生問題。但建議您盡可能緊縮您的規則。

### Note

安全群組無法封鎖傳送至 Route 53 解析器的 DNS 請求，有時也稱為「VPC+2 IP 位址」（請參閱[什麼是 Amazon Route 53 Resolver？](#) 在 Amazon Route 53 開發人員指南中）或「AmazonProvidedDNS」（請參閱 Amazon Virtual Private Cloud 用戶指南中的使用 [DHCP 選項集](#)）。如果您想要透過 Route 53 Resolver 篩選 DNS 請求，則可以啟用 Route 53 Resolver DNS Firewall（請參閱《Amazon Route 53 開發人員指南》中的 [Route 53 Resolver DNS Firewall](#)）。

針對每個規則，您指定下列項目：

- 名稱：安全群組的名稱（例如，「my-security-group」）。

名稱的長度上限為 255 個字元。允許的字元為 a-z、A-Z、0-9、空格鍵和 `._-:/()#,@[]+=;{}!$*`。當名稱尾隨空格時，我們會裁切空格並儲存名稱。例如，如果您輸入「測試安全群組」做為名稱，我們會將其儲存為「測試安全群組」。

- 通訊協定：要允許的通訊協定。最常見的通訊協定為 6 (TCP)、17 (UDP) 和 1 (ICMP)。
- 連接埠範圍：適用於 TCP、UDP 或自訂通訊協定，要允許的連接埠範圍。您可以指定單一連接埠號碼 (例如，22)，或是連接埠號碼的範圍 (例如，7000-8000)。
- ICMP 類型及代碼：適用於 ICMP，為 ICMP 的類型及代碼。例如，使用類型 8 代表「ICMP Echo 請求」，輸入 128 則代表「ICMPv6 Echo 請求」。
- 來源或目的地：允許流量的來源 (傳入規則) 或目的地 (傳出規則)。請指定下列其中一項：
  - 單一 IPv4 地址。您必須使用 /32 的字首長度。例如 203.0.113.1/32。
  - 單一 IPv6 地址。您必須使用 /128 的字首長度。例如 2001:db8:1234:1a00::123/128。
  - IPv4 地址範圍，以 CIDR 區塊標記法表示。例如 203.0.113.0/24。
  - IPv6 地址範圍，以 CIDR 區塊標記法表示。例如 2001:db8:1234:1a00::/64。
  - 字首清單的 ID。例如 p1-1234abc1234abc123。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的[字首清單](#)。
  - 安全群組的 ID (此處係指指定的安全群組)。例如，當前安全群組、來自同一 VPC 的安全群組或對等 VPC 的安全群組。這會根據與指定安全群組相關聯資源的私有 IP 地址來允許流量。這不會將來自指定安全群組的規則新增至當前安全群組。
- (Optional) Description ((選用) 描述)：您可以為規則新增描述，這可協助您在稍後更容易識別它。描述的長度最高可達 255 個字元。允許的字元為 a-z、A-Z、0-9、空格鍵和 `._-:/()#,@[]+=;{}!$*`。

當您建立安全性群組規則時，AWS 會為規則指派唯一 ID。當您使用 API 或 CLI 修改或刪除規則時，可以使用規則的 ID。

當您將安全群組指定為規則的來源或目標時，規則會影響所有與安全群組相關聯的執行個體。傳入流量會根據與來源安全群組相關聯之執行個體的私有 IP 地址允許 (而非公有 IP 或彈性 IP 地址)。如需有關 IP 地址的詳細資訊，請參閱 [Amazon EC2 執行個體 IP 定址](#)。如果您的安全群組規則是參考同一 VPC 或對等 VPC 中遭刪除之安全群組的規則，或是參考已刪除 VPC 對等互連連線的對等 VPC 中安全群組的規則，則該規則標記為過時。如需詳細資訊，請參閱 Amazon VPC Peering Guide 中的[使用過時安全群組規則](#)。

若特定連接埠有超過一個規則，Amazon EC2 會套用最寬鬆的規則。例如，如果您有一個規則允許從 IP 位址 203.0.113.1 存取 TCP 連接埠 22 (SSH)，以及另一個允許所有人存取 TCP 連接埠 22 的規則，則每個人都可以存取 TCP 連接埠 22。

當您新增、更新或移除規則時，變更會自動套用至與安全群組建立關聯的所有執行個體。

## 安全群組連線追蹤

您的安全群組使用連線追蹤來追蹤流入和流出執行個體流量的資訊。規則會根據流量的連線狀態套用，以判斷該允許或拒絕該流量。使用這種方法，安全群組是可設定狀態的。這表示傳入流量的回應可從執行個體傳出，與對外安全群組規則無關，反之亦然。

舉例來說，假設您從家用電腦中對執行個體啟動一個命令 (例如 netcat 或類似的命令)，並且您的傳入安全群組規則允許 ICMP 流量。會追蹤連線的相關資訊 (包括連接埠資訊)。來自執行個體對命令的回應流量不會作為新的請求追蹤，而是作為已建立的連線，並且將允許流出執行個體，即使您的傳出安全群組規則限制傳出 ICMP 流量。

針對 TCP、UDP 或 ICMP 之外的通訊協定，只會追蹤 IP 地址和通訊協定號碼。若您的執行個體傳送流量到另一個主機，且該主機也在 600 秒內將相同類型的流量傳送至您的執行個體，無論傳入安全群組規則為何，執行個體的安全群組都會接受該流量。安全群組會接受該流量，因為將其視為原始流量的回應流量。

變更安全群組規則時，其跟蹤的連線不會立即中斷。安全群組將繼續允許封包，直到現有連線逾時為止。若要確保無論追蹤狀態為何，流量皆會立即中斷，或所有流量都受到防火牆規則的限制，您可以為子網使用網路 ACL。網路 ACL 為無狀態，因此不會自動允許回應流量。新增阻止任一方向流量的網路 ACL 會破壞現有連線。如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的[網路 ACL](#)。

### Note

安全組對進出 Route 53 解析器的 DNS 流量沒有影響，有時也稱為「VPC+2 IP 地址」（請參閱[什麼是 Amazon Route 53 Resolver？](#) 在 Amazon Route 53 開發人員指南中）或「AmazonProvidedDNS」（請參閱 Amazon Virtual Private Cloud 用戶指南中的使用[DHCP 選項集](#)）。如果您想要透過 Route 53 Resolver 篩選 DNS 請求，則可以啟用 Route 53 Resolver DNS Firewall (請參閱《Amazon Route 53 開發人員指南》中的[Route 53 Resolver DNS Firewall](#))。

## 未追蹤的連線

並非所有的流量流程都會追蹤。[如果安全群組規則允許所有流量使用 TCP 或 UDP 流程 \(0.0.0.0/0 或 :: /0\)，並且在另一個方向有對應規則允許所有回應流量 \(0.0.0.0/0 或 :: /0\) 任何連接埠 \(0-65535\)，則不會追蹤該流量流程，除非它屬於自動追蹤連線的一部分。](#)接著會依據允許回應流量的傳入或傳出規則允許未追蹤流的回應流量，而非依據追蹤資訊。

若啟用未受追蹤之流量流程的規則遭到移除或修改，則該流程會立即中斷。例如，如果您有一個開啟的 (0.0.0.0/0) 輸出規則，並移除允許所有 (0.0.0.0/0) 輸入 SSH (TCP 連接埠 22) 流量至執行個體的規則 (或進行修改以不再允許連線)，則您現有的 SSH 連線會立即中斷與執行個體的連線。該連線之前未被追蹤，因此該變更會中斷連線。另一方面，如果您的輸入規則範圍較窄，最初允許 SSH 連線 (表示連線已被追蹤)，但會將規則變更為不再允許來自目前 SSH 用戶端地址的新連線，則現有的 SSH 連線將不會因為其遭到追蹤而中斷。

## 自動追蹤連線

即使安全性群組組態不需要追蹤，透過下列項目建立的連線也會自動追蹤：

- 輸出限定網際網路閘道
- Global Accelerator 加速器
- NAT 閘道
- Network Firewall 防火牆端點
- Network Load Balancer
- AWS PrivateLink (介面 VPC 端端點)
- AWS Lambda (超平面彈性網路介面)

## 連線追蹤限額

Amazon EC2 會定義每個執行個體可追蹤的連線數目上限。達到最大值之後，任何傳送或接收的封包都會捨棄，因為無法建立新的連線。發生這種情況時，傳送及接收封包的應用程式無法正確通訊。使用 `conntrack_allowance_available` 網路效能指標來判斷該執行個體類型仍可使用的追蹤連線數目。

若要判斷是否因為執行個體的網路流量超過可追蹤的連線數目上限而捨棄封包，請使用 `conntrack_allowance_exceeded` 網路效能指標。如需更多詳細資訊，請參閱 [監控 EC2 執行個體的網路效能](#)。

使用 Elastic Load Balancing 時，如果超過每個執行個體可追蹤的連線數目上限，建議您使用負載平衡器調整執行個體數目，或使用負載平衡器調整執行個體大小。

## 連線追蹤效能考量

非對稱路由是流量透過一個網路介面進入執行個體，並透過不同的網路介面離開，可降低執行個體在追蹤流程時所能達到的尖峰效能。

若要在為您的安全性群組啟用連線追蹤時維持最佳效能，我們建議您進行下列組態：

- 如果可能，請避免使用非對稱的路由拓撲。
- 而不是使用安全群組進行篩選，而是使用網路 ACL。
- 如果您必須將安全群組與連線追蹤搭配使用，請盡可能設定最短的連線逾時。

如需 Nitro 系統效能調整的詳細資訊，請參閱[對於性能調整的硝基系統考量](#)。

## 閒置連線追蹤逾時

安全群組會追蹤每個建立的連線，以確保傳回封包如預期交付。每個執行個體可追蹤的連線數目上限。處於閒置狀態的連線可能會導致連線追蹤耗盡，並造成不追蹤連線及捨棄封包。您可以在彈性網路介面上設定閒置連線追蹤的逾時。

### Note

此功能僅適用於在 [AWS Nitro 系統上建置的執行個體](#)。

可設定的逾時有三種：

- 已建立的 TCP 逾時：處於已建立狀態的閒置 TCP 連線逾時 (以秒為單位)。最小值：60 秒。最大值：432000 秒 (5 天)。預設：432000 秒。建議使用：432000 秒以下。
- UDP 逾時：只出現單一方向或單一請求回應交易流量的閒置 UDP 流量逾時 (以秒為單位)。最小值：30 秒。最大值：60 秒。預設：30 秒。
- UDP 串流逾時：分類為出現多個請求回應交易串流的閒置 UDP 流量逾時 (以秒為單位)。最小值：60 秒。最大值：180 秒 (3 分鐘) 預設值：180 秒。

您可修改下列任何情況的預設逾時：

- 如果您使用 [Amazon EC2 網路效能指標監控追蹤的連線](#)，`contrack_allowance_exceeded` 和 `contrack_allowance_available` 可用指標可讓您監控丟棄的封包和追蹤連線使用率，以透過縱向擴展或橫向擴展動作來主動管理 EC2 執行個體容量，以協助在丟棄封包之前滿足網路連線需求。如果您在 EC2 執行個體上觀察到 `contrack_allowance_exceeded`，將已建立的 TCP 逾時設定得較低可能對您有所幫助，可解決因不正確的用戶端或網路中間盒造成的過時 TCP/UDP 工作階段。

- 一般而言，負載平衡器或防火牆已建立的 TCP 閒置逾時範圍為 60 到 90 分鐘。如果您執行的工作負載預期會處理來自網路防火牆等設備的大量連線 (10 萬以上)，建議您在 EC2 網路介面上設定相似的逾時。
- 如果您執行的工作負載使用非對稱路由拓撲，建議您將 TCP 已建立的閒置逾時設定為 60 秒。
- 如果您執行的工作負載具有大量連線 (例如 DNS、SIP、SNMP、Syslog、Radius 和其他主要使用 UDP 來處理請求的服務)，則將「UDP 串流」逾時設為 60 秒，可為現有容量提供更高的規模/效能，並防止發生灰色故障。
- 系統會追蹤所有經過網路負載平衡器 (NLB) 和彈性負載平衡器 (ELB) 的 TCP/UDP 連線。TCP 流量的閒置逾時值為 350 秒，UDP 流量為 120 秒，且會隨介面層級逾時值而異。您可在網路介面層級設定逾時，讓逾時的彈性大於 ELB/NLB 的預設值。

執行下列操作時，您可以選擇設定連線追蹤逾時：

- [建立網路介面](#)
- [修改網路介面屬性](#)
- [啟動 EC2 執行個體](#)
- [建立 EC2 執行個體啟動範本](#)

## 範例

在下列範例中，安全群組具備允許 TCP 和 ICMP 流量的傳入規則，以及允許所有傳出流量的對外規則。

### 傳入

通訊協定類型	連接埠號碼	來源
TCP	22 (SSH)	203.0.113.1/32
TCP	80 (HTTP)	0.0.0.0/0
TCP	80 (HTTP)	::/0
ICMP	全部	0.0.0.0/0

## 傳出

通訊協定類型	連接埠號碼	目的地
全部	全部	0.0.0.0/0
全部	全部	::/0

對於執行個體或網路介面的直接網路連線，追蹤行為如下：

- 會追蹤連接埠 22 (SSH) 上傳入和傳出的 TCP 流量，因為傳入規則僅允許來自 203.0.113.1/32 的流量，而非所有 IP 地址 (0.0.0.0/0)。
- 連接埠 80 (HTTP) 上傳入和傳出的 TCP 流量則不會受到追蹤，因為傳入和傳出規則允許來自所有 IP 地址的流量。
- ICMP 流量一律會受到追蹤。

若您移除 IPv4 流量的傳出規則，則所有傳入和傳出的 IPv4 流量都會受到追蹤，包括位於連接埠 80 (HTTP) 上的流量。如果您移除 IPv6 流量的傳出規則，IPv6 流量也會受到相同影響。

## 預設和自訂安全性群組

您的 AWS 帳戶會自動為每個區域中的預設 VPC 設定一個預設安全性群組。若您並未在啟動執行個體時指定安全群組，則執行個體會自動與 VPC 的預設安全群組相關聯。若您不希望執行個體使用預設安全群組，您可以建立自訂安全群組，並在啟動執行個體時指定安全群組。

### 目錄

- [預設安全群組](#)
- [自訂安全群組](#)

## 預設安全群組

每個 VPC 都隨附一個預設安全群組。建議您為特定執行個體或執行個體群組建立安全群組，而非使用預設安全群組。不過，如果您在啟動執行個體時未指定安全群組，系統會將該執行個體與 VPC 的預設安全群組建立關聯。

預設安全群組的名稱是「default」。以下為預設安全群組規則的預設規則。

## 傳入

來源	通訊協定	連接埠範圍	描述
<code>sg-1234567890abcde</code> <code>f0</code>	全部	全部	允許來自指派給此安全群組的所有資源的傳入流量。來源為此安全群組的 ID。

## 傳出

目的地	通訊協定	連接埠範圍	描述
0.0.0.0/0	全部	全部	允許所有傳出 IPv4 流量。
::/0	全部	全部	允許所有傳出 IPv6 流量。只有在 VPC 有相關聯的 IPv6 CIDR 區塊時，才會新增此規則。

## 預設安全群組基本概念

- 您可以變更預設安全群組的規則。
- 您無法刪除預設安全群組。若您嘗試刪除預設安全群組，系統會傳回下列錯誤代碼：`Client.CannotDelete`。

## 自訂安全群組

您可以建立多個安全群組，反映執行個體擔任的不同角色。例如：Web 伺服器或資料庫伺服器。

當您建立安全群組時，您必須提供名稱和描述。安全群組的名稱和描述最高可達 255 個字元，並且限制使用下列字元：

a-z、A-Z、0-9、空格鍵，以及 `._-:/()#,@[]+=&{}!$*`

安全群組名稱的開頭不能是下列字元：`sg-`。安全群組名稱對於 VPC 必須是唯一的。

以下是您建立之安全群組的預設規則：

- 不允許對內流量
- 允許所有對外流量



在您建立安全群組之後，您可以變更其對內規則，反映您希望觸達相關聯執行個體的對內流量類型。您也可以變更其傳出規則。

如需您可以新增至安全群組之規則的詳細資訊，請參閱[不同使用案例的安全群組規則](#)。

## 使用安全群組

您可以在啟動執行個體時，為執行個體指派安全群組。當您新增或移除規則時，這些變更會自動套用到所有您已指派該安全群組的執行個體。如需詳細資訊，請參閱[將安全群組指派至執行個體](#)。

在您啟動執行個體之後，您可變更其安全群組。如需詳細資訊，請參閱[變更執行個體的安全群組](#)。

您可以使用 Amazon EC2 主控台與命令列工具建立、檢視、更新和刪除安全群組和安全群組規則。

### 工作

- [建立安全群組](#)
- [複製安全群組](#)
- [檢視您的安全群組](#)
- [將規則新增至安全群組](#)
- [更新安全群組規則](#)
- [從安全群組刪除規則](#)
- [刪除安全群組](#)
- [將安全群組指派至執行個體](#)
- [變更執行個體的安全群組](#)

## 建立安全群組

雖然您可以針對您的執行個體使用預設安全群組，但您可能會希望建立您自己的群組，以反映執行個體在您系統上所扮演的不同角色。

根據預設，新的安全群組一開始只有允許流量離開執行個體的傳出規則。您必須新增規則啟用任何傳入流量，或是限制傳出流量。

安全群組只能在建立該群組的 VPC 中使用。

## Console

### 建立安全群組

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Security Groups (安全群組)。
3. 選擇 Create Security Group (建立安全群組)。
4. 在 Basic details (基本詳細資訊) 區段中，執行下列動作。
  - a. 輸入安全性群組的描述性名稱和簡短描述。建立安全群組之後，就無法再度編輯。名稱和描述最多可包含 255 個字元。有效字元為 a-z、A-Z、0-9、空格鍵和 `._-:/()#,@[]+=&:{}!$*`。
  - b. 對於 VPC，請選擇 VPC。
5. 您可以立即新增安全群組規則，也可以稍後再新增。如需詳細資訊，請參閱 [將規則新增至安全群組](#)。
6. 您可以立即新增標籤，也可以稍後再新增。若要新增標籤，請選擇 Add new tag (新增標籤)，然後輸入標籤的索引鍵和值。
7. 選擇 Create Security Group (建立安全群組)。

## Command line

### 建立安全群組

請使用下列其中一個命令：

- [create-security-group](#) (AWS CLI)
- [New-EC2SecurityGroup](#) (AWS Tools for Windows PowerShell)

## 複製安全群組

您可以建立現有安全性群組的複本，以建立新的安全性群組。當您複製安全性群組時，系統會使用與原始安全性群組相同的輸入和輸出規則來建立複本。如果原始安全性群組位於 VPC 中，除非您指定其他安全性群組，否則系統會在同一個 VPC 中建立複本。

複本會收到新的唯一安全性群組 ID，而您必須為它命名。您也可以新增描述。

您無法將安全性群組從一個區域複製到另一個區域。

您可以使用 Amazon EC2 主控台建立您的安全群組的複本。

### 複製安全性群組

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Security Groups (安全群組)。
3. 選取要複製的安全性群組，然後選取 Actions (動作)、Copy to new security group (複製到新的安全性群組)。
4. 指定名稱和選擇性描述，並視需要變更 VPC 和安全群組規則。
5. 選擇 Create (建立)。

### 檢視您的安全群組

您可以使用下列其中一種方法來檢視安全性群組的相關資訊。

#### Console

##### 檢視您的安全性群組

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Security Groups (安全群組)。
3. 系統會列出您的安全性群組。若要檢視包括其輸入和輸出規則等特定安全群組詳細資訊，請在 Security group ID (安全性群組 ID) 欄中選擇其 ID。

#### Command line

##### 檢視您的安全性群組

請使用下列其中一個命令。

- [describe-security-groups](#) (AWS CLI)
- [describe-security-group-rules](#) (AWS CLI)
- [Get-EC2SecurityGroup](#) (AWS Tools for Windows PowerShell)

#### Amazon EC2 Global View

您可以使用 Amazon EC2 全域檢視，在啟用 AWS 帳戶的所有區域中檢視安全群組。如需詳細資訊，請參閱 [Amazon EC2 全域檢視](#)。

## 將規則新增至安全群組

當您為安全群組新增規則時，新規則會自動套用到所有與安全群組相關聯的任何執行個體。規則在套用之前可能會有短暫的延遲。如需詳細資訊，請參閱 [不同使用案例的安全群組規則](#) 及 [安全群組規則](#)。

### Console

#### 將傳入規則新增至安全群組

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇安全群組。
3. 選取安全群組，然後選取 Actions (動作)、Edit inbound rules (編輯傳入規則)。
4. 針對每個規則，選擇 Add rule (新增規則) 並執行下列動作。
  - a. 在 Type (類型) 中，選擇要允許的通訊協定類型。
    - 對於 [自訂 TCP] 或 [自訂 UDP]，您必須輸入要允許的連接埠範圍。例如 0-99。
    - 對於「自訂 ICMP」，您必須從「通訊協定」中選擇 ICMP 類型。連接埠範圍已為您設定。例如，若要允許 ping 命令，請在 Protocol (通訊協定) 中選擇 Echo Request (回聲請求)。
    - 如果是任何其他類型，則系統會自動為您設定通訊協定和連接埠範圍。
  - b. 對於 Source (來源)，請執行下列其中一項動作來允許流量。
    - 選擇 Custom (自訂)，然後以 CIDR 標記法、CIDR 區塊、其他安全群組或字首清單輸入 IP 地址。
    - 選擇 Anywhere (隨處) 以允許指定通訊協定的所有流量可到達您的執行個體。此選項會自動新增 0.0.0.0/0 IPv4 CIDR 區塊作為來源。如果您的安全性群組位於已啟用 IPv6 的 VPC 中，則此選項會自動為 ::/0 IPv6 CIDR 區塊新增規則。
  - c. (選用) 針對 Description (描述)，指定規則的簡短描述。

#### Warning

如果選擇 Anywhere (隨處)，則可讓所有 IPv4 和 IPv6 地址使用指定的通訊協定存取您的執行個體。如果您要新增連接埠 22 (SSH) 或 3389 (RDP) 的規則，則應只授權特定 IP 地址或特定範圍的地址來存取您的執行個體。

- 選擇 My IP (我的 IP) 以僅允許來自本機電腦的公有 IPv4 位址輸入流量。

5. 選擇 Preview changes (預覽變更) 及 Save rules (儲存規則)。

將輸出規則新增至安全性群組

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇安全群組。
3. 選取安全群組，然後選取 Action (動作)、Edit outbound rules (編輯傳出規則)。
4. 針對每個規則，選擇 Add rule (新增規則) 並執行下列動作。
  - a. 在 Type (類型) 中，選擇要允許的通訊協定類型。
    - 對於 [自訂 TCP] 或 [自訂 UDP]，您必須輸入要允許的連接埠範圍。例如 0-99。
    - 對於「自訂 ICMP」，您必須從「通訊協定」中選擇 ICMP 類型。連接埠範圍已為您設定。
    - 如果是任何其他類型，則系統會自動設定通訊協定和連接埠範圍。
  - b. 針對 Destination (目的地)，請執行下列其中一項動作。
    - 選擇自訂，然後以 CIDR 標記法、CIDR 區塊或其他允許輸入流量的安全群組或字首清單輸入 IP 地址。
    - 選擇 Anywhere (任何地方) 以允許所有 IP 地址的輸出流量。此選項會自動新增 0.0.0.0/0 IPv4 CIDR 區塊作為目的地。

如果您的安全性群組位於已啟用 IPv6 的 VPC 中，則此選項會自動為 ::/0 IPv6 CIDR 區塊新增規則。
    - 選擇 My IP (我的 IP)，以僅允許本機電腦公有 IPv4 地址的輸出流量。
  - c. (選用) 針對 Description (描述)，指定規則的簡短描述。
5. 選擇 Preview changes (預覽變更)、Confirm (確認)。

## Command line

將規則新增至安全性群組

請使用下列其中一個命令。

- [authorize-security-group-ingress](#) (AWS CLI)
- [Grant-EC2SecurityGroupIngress](#) (AWS Tools for Windows PowerShell)

## 新增一或多個輸入規則至安全群組

請使用下列其中一個命令。

- [authorize-security-group-egress](#) (AWS CLI)
- [Grant-EC2SecurityGroupEgress](#) (AWS Tools for Windows PowerShell)

## 更新安全群組規則

您可以使用下列其中一種方法來更新安全性群組規則。更新的規則會自動套用至與安全群組相關聯的任何執行個體。

### Console

當您使用主控台修改現有安全群組規則的通訊協定、連接埠範圍，或來源或目標時，主控台會刪除現有規則並為您建立新的規則。

#### 更新安全群組規則

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇安全群組。
3. 選取安全群組。
4. 選擇 Actions (動作)，然後選擇 Edit inbound rules (編輯傳入規則) 以更新傳入流量規則，或者選擇 Actions (動作)，然後選擇 Edit outbound rules (編輯傳出規則) 以更新傳出流量規則。
5. 視需求更新規則。
6. 選擇 Preview changes (預覽變更)、Confirm (確認)。

#### 標記安全群組規則

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇安全群組。
3. 選取安全群組。
4. 在 Inbound rules (傳入規則) 或 Outbound rules (傳出規則) 索引標籤上，選取規則的核取方塊，然後選取 Manage tags (管理標籤)。
5. Manage tags (管理標籤) 頁面會顯示指派給規則的任何標籤。若要新增標籤，請選擇 Add tag (新增標籤)，然後輸入標籤金鑰和值。若要移除標籤，請在您要移除的標籤旁邊選擇 Remove (移除)。

## 6. 選擇 Save changes (儲存變更)。

### Command line

您無法使用 Amazon EC2 API 或命令列工具修改現有規則的通訊協定、連接埠範圍，或來源或目標。相反的，您必須刪除現有規則並新增新的規則。但是，您可以更新現有規則的描述。

#### 更新規則

請使用下列其中一個命令。

- [modify-security-group-rules](#) (AWS CLI)

#### 更新現有輸入規則的描述

請使用下列其中一個命令。

- [update-security-group-rule-descriptions-ingress](#) (AWS CLI)
- [Update-EC2SecurityGroupRuleIngressDescription](#) (AWS Tools for Windows PowerShell)

#### 更新現有輸出規則的描述

請使用下列其中一個命令。

- [update-security-group-rule-descriptions-egress](#) (AWS CLI)
- [Update-EC2SecurityGroupRuleEgressDescription](#) (AWS Tools for Windows PowerShell)

#### 標記安全群組規則

請使用下列其中一個命令。

- [create-tags](#) (AWS CLI)
- [New-EC2Tag](#) (AWS Tools for Windows PowerShell)

### 從安全群組刪除規則

當您從安全群組刪除規則時，變更會自動套用到所有與安全群組相關聯的任何執行個體。

您可以使用下列其中一種方法從安全性群組刪除規則。

## Console

### 刪除安全群組規則

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Security Groups (安全群組)。
3. 選取要更新的安全性群組，選取 Actions (動作)，然後選取 Edit inbound rules (編輯輸入規則) 以移除輸入規則，或選取 Edit outbound rules (編輯輸出規則) 以移除輸出規則。
4. 選擇規則右側的 刪除 按鈕以刪除。
5. 選擇儲存規則。或者，選擇預覽變更，檢閱變更，然後選擇確認。

## Command line

### 從安全群組中移除一或多個輸入規則

請使用下列其中一個命令。

- [revoke-security-group-ingress](#) (AWS CLI)
- [Revoke-EC2SecurityGroupIngress](#) (AWS Tools for Windows PowerShell)

### 從安全性群組移除一或多個輸出規則

請使用下列其中一個命令。

- [revoke-security-group-egress](#) (AWS CLI)
- [Revoke-EC2SecurityGroupEgress](#) (AWS Tools for Windows PowerShell)

## 刪除安全群組

您無法刪除已和執行個體建立關聯的安全群組。您無法刪除預設安全群組。您無法刪除相同 VPC 中另一個安全群組中規則所參考的安全群組。若您的安全群組已由自身的規則參考，您必須先刪除規則，才能刪除安全群組。

## Console

### 刪除安全群組

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。



2. 在導覽窗格中，選擇安全群組。
3. 選取安全群組，然後選擇動作、刪除安全群組。
4. 出現確認提示時，請選擇刪除。

## Command line

### 刪除安全群組

請使用下列其中一個命令。

- [delete-security-group](#) (AWS CLI)
- [Remove-EC2SecurityGroup](#) (AWS Tools for Windows PowerShell)

## 將安全群組指派至執行個體

您可以在啟動執行個體時將一或多個安全群組指派給執行個體。您也可以在此啟動範本中指定一或多個安全群組。安全群組會指派給使用啟動範本啟動的所有執行個體。

- 若要在啟動執行個體時，為執行個體指派安全群組，請參閱 [使用定義的參數啟動執行個體的 Network settings \(網路設定\)](#) (新主控台) 或 [步驟 6：設定安全群組](#) (舊主控台)。
- 若要在啟動範本中指定安全群組，請參閱 [從參數建立啟動範本](#) 的 [Network settings \(網路設定\)](#)。

## 變更執行個體的安全群組

啟動執行個體後，您可以透過新增或移除安全群組來變更其安全群組。

### 要求

- 執行個體必須處於 `running` 或 `stopped` 狀態。
- 安全群組是特定於 VPC 的。您可以將安全群組指派給您為其建立安全群組的 VPC 中啟動的一個或多個執行個體。

## Console

### 變更執行個體的安全群組

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。

2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選取 動作、安全性、變更安全群組。
4. 對於關聯的安全群組，請從清單中選取安全群組，然後選取新增安全群組。

若要移除關聯的安全群組，請為該安全群組選擇移除。

5. 選擇 Save (儲存)。

## Command line

### 變更執行個體的安全群組

請使用下列其中一個命令。

- [modify-instance-attribute](#) (AWS CLI)
- [Edit-EC2InstanceAttribute](#) (AWS Tools for Windows PowerShell)

## 不同使用案例的安全群組規則

您可以建立安全群組並新增可反映與該安全群組相關聯之執行個體角色的規則。例如，設定為 Web 伺服器的執行個體需要允許輸入 HTTP 和 HTTPS 存取的安全性群組規則。同樣，資料庫執行個體需要允許存取資料庫類型的規則，例如 MySQL 透過連接埠 3306 的存取。

以下為您可以針對特定類型存取，為安全群組新增之規則類型的範例。

### 範例

- [Web 伺服器規則](#)
- [資料庫伺服器規則](#)
- [從您電腦連線到執行個體的規則](#)
- [從執行個體使用相同安全群組連線到執行個體的規則](#)
- [Ping/ICMP 的規則](#)
- [DNS 伺服器規則](#)
- [Amazon EFS 規則](#)
- [Elastic Load Balancing 規則](#)
- [VPC 對等規則](#)

## Web 伺服器規則

以下對內規則會允許來自任何 IP 地址的 HTTP 和 HTTPS 存取。若您的 VPC 已啟用 IPv6，您可以新增規則，控制來自 IPv6 地址的對內 HTTP 和 HTTPS 流量。

通訊協定類型	通訊協定號碼	連接埠	來源 IP	備註
TCP	6	80 (HTTP)	0.0.0.0/0	允許來自任何 IPv4 地址的對內 HTTP 存取
TCP	6	443 (HTTPS)	0.0.0.0/0	允許來自任何 IPv4 地址的對內 HTTPS 存取
TCP	6	80 (HTTP)	::/0	允許來自任何 IPv6 地址的傳入 HTTP 存取
TCP	6	443 (HTTPS)	::/0	允許來自任何 IPv6 地址的傳入 HTTPS 存取

## 資料庫伺服器規則

以下對內規則是您可以為資料庫存取新增之規則的範例，取決於您在執行個體上執行的資料庫類型為何。如需 Amazon RDS 執行個體的詳細資訊，請參閱 [Amazon RDS 使用者指南](#)。

針對來源 IP，請指定下列其中一項：

- 您本機網路中的特定 IP 地址或 IP 地址範圍 (CIDR 區塊標記法)
- 存取資料庫之執行個體群組的安全群組 ID

通訊協定類型	通訊協定號碼	連接埠	備註
TCP	6	1433 (MS SQL)	在 Amazon RDS 執行個體上存取 Microsoft SQL Server 資料庫的預設連接埠 (範例)

通訊協定類型	通訊協定號碼	連接埠	備註
TCP	6	3306 (MySQL/Aurora)	在 Amazon RDS 執行個體上存取 MySQL 或 Aurora 資料庫的預設連接埠 (範例)
TCP	6	5439 (Redshift)	存取 Amazon Redshift 叢集資料庫的預設連接埠。
TCP	6	5432 (PostgreSQL)	在 Amazon RDS 執行個體上存取 PostgreSQL 資料庫的預設連接埠 (範例)
TCP	6	1521 (Oracle)	在 Amazon RDS 執行個體上存取 Oracle 資料庫的預設連接埠 (範例)

您可以選擇性限制來自資料庫伺服器的輸出流量。例如，您可能想允許存取網際網路以取得軟體更新，但限制所有其他類型的流量。您必須先移除允許所有對外流量的預設對外規則。

通訊協定類型	通訊協定號碼	連接埠	目標 IP	備註
TCP	6	80 (HTTP)	0.0.0.0/0	允許前往任何 IPv4 地址的對外 HTTP 存取
TCP	6	443 (HTTPS)	0.0.0.0/0	允許前往任何 IPv4 地址的對外 HTTPS 存取
TCP	6	80 (HTTP)	::/0	(僅限啟用 IPv6 的 VPC) 允許前往任何 IPv6 地址的對外 HTTP 存取
TCP	6	443 (HTTPS)	::/0	(僅限啟用 IPv6 的 VPC) 允許前往任何 IPv6 地址的對外 HTTPS 存取

## 從您電腦連線到執行個體的規則

若要連線到您的執行個體，您的安全群組必須具有允許 SSH 存取 (適用於 Linux 執行個體) 或 RDP 存取 (適用於 Windows 執行個體) 的對內規則。

通訊協定類型	通訊協定號碼	連接埠	來源 IP
TCP	6	22 (SSH)	您電腦的公有 IPv4 地址，或是您本機網路中的 IP 地址範圍。若您的 VPC 已啟用 IPv6，且您的執行個體具有 IPv6 地址，您可以輸入 IPv6 地址或範圍。
TCP	6	3389 (RDP)	您電腦的公有 IPv4 地址，或是您本機網路中的 IP 地址範圍。若您的 VPC 已啟用 IPv6，且您的執行個體具有 IPv6 地址，您可以輸入 IPv6 地址或範圍。

## 從執行個體使用相同安全群組連線到執行個體的規則

若要允許與相同安全群組相關聯的執行個體彼此之間互相通訊，您必須為此明確新增規則。

### Note

如果您將路由設定為透過中間設備來轉遞不同子網中兩個執行個體之間的流量，則您必須確保兩個執行個體的安全群組均允許流量在執行個體之間流動。每個執行個體的安全群組都必須參考另一個執行個體的私有 IP 地址，或是包含其他執行個體之子網的 CIDR 範圍作為來源。如果您參考另一個執行個體的安全群組作為來源，這不會允許流量在執行個體之間流動。

下表說明可讓相關聯的執行個體互相通訊的安全群組傳入規則。規則允許所有類型的流量。

通訊協定類型	通訊協定號碼	連接埠	來源 IP
-1 (全部)	-1 (全部)	-1 (全部)	安全群組的 ID，或是包含其他執行個體之子網的 CIDR 範圍 (請參閱備註)。

## Ping/ICMP 的規則

ping 命令是一種 ICMP 流量。若要 ping 您的執行個體，您必須新增下列任一對內 ICMP 規則。

Type	通訊協定	來源		
自訂 ICMP - IPv4	回聲請求	您電腦的公有 IPv4 地址、特定 IPv4 地址，或任何位置的 IPv4 或 IPv6 地址。		
所有 ICMP - IPv4	IPv4 ICMP (1)	您電腦的公有 IPv4 地址、特定 IPv4 地址，或任何位置的 IPv4 或 IPv6 地址。		

若要使用 ping6 命令 ping 您執行個體的 IPv6 地址，您必須新增下列對內 ICMPv6 規則。

Type	通訊協定	來源		
所有 ICMP - IPv6	IPv6 ICMP (58)	您電腦的 IPv6 地址、特定 IPv4 地址，或任何位置的 IPv4 或 IPv6 地址。		

## DNS 伺服器規則

若您已將您的 EC2 執行個體設為 DNS 伺服器，您必須確保 TCP 和 UDP 流量能透過連接埠 53 觸達您的 DNS 伺服器。

針對來源 IP，請指定下列其中一項：

- 網路中的特定 IP 地址或 IP 地址範圍 (CIDR 區塊標記法)
- 您網路中需要存取 DNS 伺服器之執行個體集的安全群組 ID

通訊協定類型	通訊協定號碼	連接埠
TCP	6	53
UDP	17	53

## Amazon EFS 規則

若您正在搭配 Amazon EC2 執行個體使用 Amazon EFS 檔案系統，您與 Amazon EFS 掛載目標建立關聯的安全群組必須允許透過 NFS 通訊協定的流量。

通訊協定類型	通訊協定號碼	連接埠	來源 IP	備註
TCP	6	2049 (NFS)	安全群組的 ID	允許來自與此安全群組相關聯之資源 (包括掛載目標) 的對內 NFS 存取。

若要在您的 Amazon EC2 執行個體上掛載 Amazon EFS 檔案系統，您必須連線到您的執行個體。因此，與您的執行個體相關聯的安全群組必須具備允許來自您本機電腦或本機網路對內 SSH 的規則。

通訊協定類型	通訊協定號碼	連接埠	來源 IP	備註
TCP	6	22 (SSH)	您本機電腦的 IP 地址範圍，或是您網路的 IP 地	允許來自您本機電腦的對內 SSH 存取。

通訊協定類型	通訊協定號碼	連接埠	來源 IP 地址範圍 (CIDR 區塊標記法)。	備註
--------	--------	-----	--------------------------	----

## Elastic Load Balancing 規則

若您正在使用負載平衡器，與您負載平衡器相關聯的安全群組必須具備允許和您的執行個體或目標通訊的規則。如需詳細資訊，請參閱《Classic Load Balancer 使用者指南》中的[設定 Classic Load Balancer 的安全群組](#)，以及 Application Load Balancer 使用者指南中的[Application Load Balancer 的安全群組](#)。

## VPC 對等規則

您可以更新您 VPC 安全群組的傳入和傳出規則，以參考互連 VPC 中的安全群組。執行此作業，可允許流量傳入和傳出與互連 VPC 中參考之安全群組相關聯的執行個體。如需如何設定 VPC 對等的安全群組的詳細資訊，請參閱[更新您的安全群組以參考對等 VPC 群組](#)。

## NitroTPM

Nitro Trusted Platform Module (NitroTPM) 是由 [AWS Nitro System](#) 提供的虛擬裝置，符合 [TPM 2.0 規格](#)。它可安全存放用於對執行個體執行身分驗證的成品 (如密碼、憑證或加密金鑰)。NitroTPM 可產生金鑰，並將其用於密碼編譯功能 (例如雜湊、簽署、加密和解密)。

NitroTPM 會採用測量開機過程，在這個過程中，開機載入器和作業系統會建立各開機二進位的密碼編譯雜湊，並將其與 NitroTPM 內部平台組態註冊 (PCR) 中的過往值結合。透過測量開機，您可從 NitroTPM 獲得簽署的 PCR 值，然後使用這些值向遠端實體證明執行個體開機軟體的完整性。這稱為遠端證明。

透過 NitRotTPM，可以使用特定 PCR 值標記金鑰和秘密，這樣便可確保，如果 PCR 值和執行個體完整性發生變化，就再也無法存取這些金鑰和秘密。這種特殊形式的條件式存取被稱為密封和開封。作業系統技術 (例如) 可以使用 NitroTPM 來密封磁碟機解密金鑰 [BitLocker](#)，以便只有在作業系統已正確開機且處於已知良好狀態時，才能解密磁碟機。

若要使用 NitroTPM，您必須選取已針對 NitroTPM 支援設定的 [Amazon 電腦映像](#) (AMI)，然後使用 AMI 啟動在硝基系統上建置的執行個體。AWS 您可以選取一個 Amazon 預先建置的 AMI，或自行建立一個。



## 成本

使用 NitroTPM 無需額外付費。只需為您使用的基礎資源支付費用。

## 主題

- [考量事項](#)
- [啟動時啟用的先決條件](#)
- [建立 Linux AMI 以取得 NitroTPM 支援](#)
- [驗證 AMI 是否已啟用 NitroTPM](#)
- [啟用或停用執行個體上的 NitroTPM](#)
- [擷取執行個體的公開背書金鑰](#)

## 考量事項

使用 NitroTPM 時需要注意以下事項：

- BitLocker 使用 NitroTPM 金鑰加密的磁碟區只能用於原始執行個體。
- [Amazon EBS 快照](#) 中不包含 NitroTPM 狀態。
- [VM Import/Export](#) 映像中不包含 NitroTPM 狀態。
- 在建立 AMI 時，將 `tpm-support` 參數的值指定為 `v2.0` 即可啟用 NitroTPM 支援。使用 AMI 啟動執行個體後，便無法再修改執行個體的屬性。具有 NitroTPM 的執行個體不支援 [ModifyInstance](#) 屬性 API。
- 您只能在 Amazon EC2 主控台上使用 AWS CLI 和而非使用 [RegisterImage](#) API 來建立具有 NitroTPM 設定的 AMI。
- Outposts 上不支援 NitroTPM。
- 本機區域或 Wavelength 區域皆不支援 NitroTPM。

## 啟動時啟用的先決條件

若要在啟用 NitroTPM 的情況下啟動執行個體，必須符合下列先決條件。

### Linux 執行個體

#### AMI

需要已啟用 NitroTPM 的 AMI。

目前，沒有具有已啟用 NitroTPM 的 Amazon Linux AMI。若要使用支援的 AMI，必須在自有的 Linux AMI 上執行多個設定步驟。如需詳細資訊，請參閱 [建立 Linux AMI 以取得 NitroTPM 支援](#)。

## 作業系統

AMI 必須包含具有 TPM 2.0 Command Response Buffer (CRB) 驅動程式的作業系統。大多數目前的作業系統，例如 Amazon Linux 2，都包含 TPM 2.0 CRB 驅動程式。

## UEFI 開機模式

NitroTPM 要求執行個體在 UEFI 開機模式下執行，這會要求 AMI 必須設定 UEFI 開機模式。如需詳細資訊，請參閱 [UEFI 安全開機](#)。

## Windows 執行個體

### AMI

需要已啟用 NitroTPM 的 AMI。

以下 Windows AMI 已預先設定為使用 Microsoft 金鑰來啟用 NitroTPM 和 UEFI Secure Boot：

- TPM-Windows\_Server-2022-English-Core-Base
- TPM-Windows\_Server-2022-English-Full-Base
- TPM-Windows\_Server-2022-English-Full-SQL\_2022\_Enterprise
- TPM-Windows\_Server-2022-English-Full-SQL\_2022\_Standard
- TPM-Windows\_Server-2019-English-Core-Base
- TPM-Windows\_Server-2019-English-Full-Base
- TPM-Windows\_Server-2019-English-Full-SQL\_2019\_Enterprise
- TPM-Windows\_Server-2019-English-Full-SQL\_2019\_Standard
- TPM-Windows\_Server-2016-English-Core-Base
- TPM-Windows\_Server-2016-English-Full-Base

目前，我們不支援使用 [import-image](#) 命令來匯入具有 NitroTPM 的 Windows。

## 作業系統

AMI 必須包含具有 TPM 2.0 Command Response Buffer (CRB) 驅動程式的作業系統。大多數目前的操作系統，例如 TPM-視窗服務器 2022-英文全基礎，包含 TPM 2.0 CRB 驅動程序。

## UEFI 開機模式

NitroTPM 要求執行個體在 UEFI 開機模式下執行，這會要求 AMI 必須設定 UEFI 開機模式。如需詳細資訊，請參閱 [UEFI 安全開機](#)。

## 執行個體類型

您必須使用下列其中一種虛擬化執行個體類型：

- 一般用途:M5, M5a, M5ad, M5dN, M5n, M5n, M6a, M6i, M6id, M6idn, M6 英寸, M7 一, M7 一, M7-柔性, T3, T3A
- 計算最佳化：C5、C5 安培、C5ad、C5d、C5n、C6i、C6id、C6 英寸、C7 安培、C7 I-彈性
- 內存優化：R5，R5，R5 阿，R5 B，R5 D，R5 DN，R5 N，R6 A，R6 I，R6 英寸，R6 英寸，R7 一，R7 I，R7 Iz，U7-12 TB，U7 英寸，X2TB，EDN, 二次, Z1D
- 儲存最佳化：D3、D3 安裝、i3en、i4i
- 加速運算：G4dN、G5、G6、GR6、第一次、二次
- 高效能運算：HPC6a、高效能

### Note

不支援重力型執行個體、Xen 執行個體、Mac 執行個體和裸機執行個體。

## 建立 Linux AMI 以取得 NitroTPM 支援

必須在註冊 AMI 時設定 Linux AMI 以取得 NitroTPM 支援。之後便無法再設定 NitroTPM 支援。

如需預先設定為 NitroTPM 支援的視窗 AMI 清單，請參閱 [啟動時啟用的先決條件](#)

若要註冊 Linux AMI 以取得硝基分配管理支援

1. 使用您所需的 Linux AMI 啟動臨時執行個體。
2. 執行個體到達狀running態後，建立執行個體根磁碟區的快照。
3. 註冊新的 AMI。使用 [註冊影像指令](#)。在 `--tpm-support`，請指定 `v2.0`。在 `--boot-mode`，請指定 `uefi`。並使用您在上一個步驟中建立的快照指定根磁碟區的區塊裝置對應。

```
aws ec2 register-image \
```

```
--name my-image \
--boot-mode uefi \
--architecture x86_64 \
--root-device-name /dev/xvda \
--block-device-mappings DeviceName=/dev/xvda,Ebs={SnapshotId=snapshot_id} \
--tpm-support v2.0
```

預期的輸出結果

```
{
 "ImageId": "ami-0123456789example"
}
```

4. 如果不再需要，請終止您在步驟 1 中啟動的暫存執行個體。

## 驗證 AMI 是否已啟用 NitroTPM

您可以使用 `describe-images` 或 `describe-image-attributes` 來驗證 AMI 是否已啟用 NitroTPM。

使用 **`describe-images`** 驗證 AMI 是否已啟用 NitroTPM

使用 [describe-images](#) 命令並指定 AMI ID。

```
aws ec2 describe-images --image-ids ami-0123456789example
```

若 AMI 已啟用 NitroTPM，則輸出中會顯示 `"TpmSupport": "v2.0"`。

```
{
 "Images": [
 {
 ...
 "BootMode": "uefi",
 ...
 "TpmSupport": "v2.0"
 }
]
}
```

使用 **`describe-image-attribute`** 驗證 AMI 是否已啟用 NitroTPM

使用 [describe-image-attribute](#) 命令，並將 `attribute` 參數值指定為 `tpmSupport`。

#### Note

您必須是 AMI 擁有者才能呼叫 `describe-image-attribute`。

```
aws ec2 describe-image-attribute \
 --region us-east-1 \
 --image-id ami-0123456789example \
 --attribute tpmSupport
```

若 AMI 已啟用 NitroTPM，則 `TpmSupport` 的值為 `"v2.0"`。請注意，`describe-image-attribute` 僅會傳回請求中指定的屬性。

```
{
 "ImageId": "ami-0123456789example",
 "TpmSupport": {
 "Value": "v2.0"
 }
}
```

## 啟用或停用執行個體上的 NitroTPM

從已啟用 NitroTPM 支援的 AMI 啟動執行個體時，會啟動執行個體，同時也會啟用 NitroTPM。您可以設定執行個體來停止使用 NitroTPM。您可以驗證執行個體是否已啟用 NitroTPM。

### 主題

- [啟動執行個體並啟用 NitroTPM](#)
- [停用執行個體的 NitroTPM](#)
- [驗證是否可存取執行個體內的 NitroTPM](#)

### 啟動執行個體並啟用 NitroTPM

在滿足[先決條件](#)的情況下啟動執行個體時，NitroTPM 會在執行個體上自動啟用。您僅可在執行個體啟動時啟用執行個體的 NitroTPM。如需有關啟動執行個體的詳細資訊，請參閱[啟動您的執行個體](#)。

## 停用執行個體的 NitroTPM

啟動執行個體並啟用其 NitroTPM 後，便無法停用該執行個體的 NitroTPM。但是，您可以使用以下工具，停用執行個體上的 TPM 2.0 裝置驅動程式，從而將作業系統設定為停止使用 NitroTPM：

- [Linux 實例] 使用 TPM-工具。
- [視窗執行個體] 使用 TPM 管理主控台 tpm.msc。

如需有關停用裝置驅動程式的詳細資訊，請參閱適用於您作業系統的相關文件。

## 驗證是否可存取執行個體內的 NitroTPM

若要確認執行個體是否已啟用 NitroTPM 支援，請使用 AWS CLI

使用 [describe-instances](#) AWS CLI 命令並指定執行個體 ID。目前，Amazon EC2 主控台不會顯示 TpmSupport 欄位。

```
aws ec2 describe-instances --instance-ids i-0123456789example
```

若執行個體上已啟用 NitroTPM，則輸出中會顯示 "TpmSupport": "v2.0"。

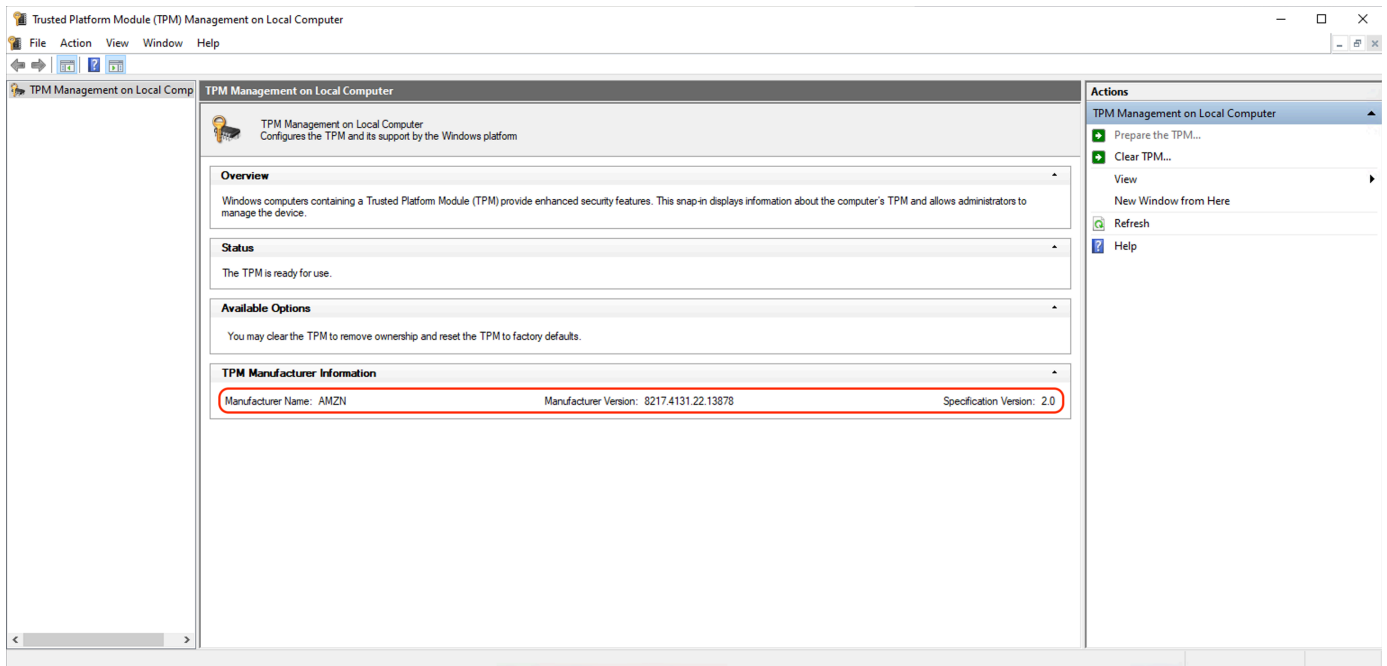
```
"Instances": {
 "InstanceId": "0123456789example",
 "InstanceType": "c5.large",
 ...
 "BootMode": "uefi",
 "TpmSupport": "v2.0"
 ...
}
```

(視窗執行個體) 確認是否可在 Amazon EC2 視窗執行個體內部存取 NitRTP

1. [連接至 EC2 Windows 執行個體。](#)
2. 在執行個體上，執行 tpm.msc 程式。

系統將會開啟 TPM Management on Local Computer (本機電腦上的 TPM 管理) 視窗。

3. 查看 TPM Manufacturer Information (TPM 製造商資訊) 欄位。此欄位中包含製造商的名稱和執行個體上的 NitroTPM 版本。



## 擷取執行個體的公開背書金鑰

您可以隨時使用安全地擷取執行個體的公開背書金鑰。AWS CLI

### 擷取執行個體的公開背書金鑰

使用 [get-instance-tpm-ek-pub](#) AWS CLI 指令。

#### 範例 1

下列範例命令會以例如tpmt格式取得rsa-2048公開認可金鑰。i-01234567890abcdef

```
$ aws ec2 get-instance-tpm-ek-pub \
--instance-id i-01234567890abcdef \
--key-format tpmt \
--key-type rsa-2048
```

以下是範例輸出。

```
{
 "InstanceId": "i-01234567890abcdef",
 "KeyFormat": "tpmt",
 "KeyType": "rsa-2048",
 "KeyValue": "AAEACwADALIAIINx12dEhLEXAMPLEUa11yT9UtduB1ILZPKh2hszFGmqAAYAgABDA"
```

```

EXAMPLEAAABA0iRd7WmgtdGNoV1h/AxmW+CXExblG8pEUfNm0L0LiYnEXAMPLERqApiFa/UhvEYqN4
Z7jKMD/usbhsQaAB1gKA5RmzuhSazHQkax7EXAMPLEzDthlS7HNGuYn5eG7qnJndRcakS+iNxT8Hvf
0S1ZtNuItMs+Yp4S06aU28MT/JZk0KsXIdMerY3GdWbNQz9AvYbMEXAMPLEPyHfzgv00QTTJVGDxh
vxtXC0u9GYf0crbjEXAMPLEd4YTbWdDdg0KWF9fjzDytJSDhrLA0UctNzHPCd/9215zEXAMPLE0IFA
Ss50C0/802c17W2pMSVHvCCa91YCiAfxH/vYKovAAE="
}

```

## 範例 2

下列範例命令會以例如der格式取得rsa-2048公開認可金鑰。i-01234567890abcdef

```

$ aws ec2 get-instance-tpm-ek-pub \
--instance-id i-01234567890abcdef \
--key-format der \
--key-type rsa-2048

```

以下是範例輸出。

```

{
 "InstanceId": "i-01234567890abcdef",
 "KeyFormat": "der",
 "KeyType": "rsa-2048",
 "KeyValue": "MIIBIjANBgEXAMPLEew0BAQEFAAOCAQ8AMIIBCgKCAQEA6JF3taEXAMPLEXWH8DGZb4
JcTFuUbykRR82bQs4uJifaKS0v5NGoEXAMPLEG8Ri03hnuMowP+6xuGxBoAHWAoD1Gb06FJrMdEXAMP
LEnYUHVm02GVLsc0a5ifl4buqcmd1FqxqRL6I3FPwe9/REXAMPLE0yz5inhI7ppTbwxP81mQ4qxch0x6
tjcZ1Zs1DP0EXAMPLERUYLQ/Id/0BU7RBNM1UZ0PGG/G1cI670Zh/Rytu0dx9iEXAMPLEtZ0N2A4pYX
1+PMPK01I0GssA5Ry03Mc8J3/3aXn0D2/ASRQ4gUBKznQLT/zTZEXAMPLEJUe8IJr2VgKIB/Ef+9gqi
8AAQIDAQAB"
}

```

## 視窗執行個體的認證保護

AWS 硝基系統支援亞馬遜彈性運算雲端 (Amazon EC2) Windows 執行個體的登入資料保護。Credential Guard 是基於 Windows 虛擬化的安全性 (VBS) 功能，可讓您建立隔離環境以保護諸如 Windows 使用者憑證和程式碼完整性強制執行等安全性資產，而這些超出了 Windows 核心保護範圍。當您執行 EC2 Windows 執行個體時，登入資料保護會使用 AWS Nitro 系統來保護 Windows 登入憑證，避免從作業系統記憶體中擷取。

### 目錄

- [必要條件](#)



- [啟動支援的執行個體](#)
- [禁用內存完整性](#)
- [開啟認證保全](#)
- [確認認證保全正在執行](#)

## 必要條件

Windows 執行個體必須滿足以下先決條件，才能使用 Credential Guard：

### Amazon Machine Images (AMI)

必須預先設定 AMI，才能啟用 NitroTPM 和 UEFI Secure Boot。如需支援 AMI 的詳細資訊，請參閱 [the section called “必要條件”](#)。

### 記憶體完整性

不支援記憶體完整性 (也稱為受 Hypervisor 保護的程式碼完整性 (HVCI)) 或 Hypervisor 強制執行的程式碼完整性。開啟 Credential Guard 之前，您必須確定此功能已停用。如需詳細資訊，請參閱 [禁用內存完整性](#)。

### 執行個體類型

下列執行個體類型的所有大小均支援 Credential

Guard：C5、C5d、C5n、C6i、C6id、C6in、M5、M5d、M5dn、M5n、M5zn、M6i、M6id、M6idn、M

#### Note

雖然 NitroTPM 有一些共同的必要執行個體類型，但執行個體類型必須是上述其中一種，才能支援 Credential Guard。

## 啟動支援的執行個體

您可以使用 Amazon EC2 主控台或 AWS Command Line Interface (AWS CLI) 啟動可支援登入資料防護的執行個體。您將需要一個相容 AMI ID 來啟動執行個體，該 ID 對於每個 AWS 區域都是唯一的。

#### Tip

可以使用下列連結在 Amazon EC2 主控台中探索並啟動具有相容 Amazon 提供之 AMI 的執行個體：

[https://console.aws.amazon.com/ec2/v2/home?#Images:visibility=public-images;v=3;search=:TPM-Windows\\_Server;ownerAlias=amazon](https://console.aws.amazon.com/ec2/v2/home?#Images:visibility=public-images;v=3;search=:TPM-Windows_Server;ownerAlias=amazon)

## Amazon EC2 console

使用 Amazon EC2 主控台來啟動執行個體

請依照下列步驟[啟動執行個體](#)，指定支援的執行個體類型和預先設定的 Windows AMI。

## AWS CLI

使用啟動執行個體 AWS CLI

使用 [run-instances](#) 命令，利用支援的執行個體類型和預先設定的 Windows AMI 來啟動執行個體。

```
aws ec2 run-instances \
 --image-id resolve:ssm:/aws/service/ami-windows-latest/TPM-Windows_Server-2022-English-Full-Base \
 --instance-type c6i.large \
 --region us-east-1 \
 --subnet-id subnet-id \
 --key-name key-name
```

## PowerShell

使用啟動執行個體 AWS Tools for PowerShell

使用 [New-EC2Instance](#) 命令，利用支援的執行個體類型和預先設定的 Windows AMI 來啟動執行個體。

```
New-EC2Instance \
 -ImageId resolve:ssm:/aws/service/ami-windows-latest/TPM-Windows_Server-2022-English-Full-Base \
 -InstanceType c6i.large \
 -Region us-east-1 \
 -SubnetId subnet-id \
 -KeyName key-name
```

## 禁用內存完整性

在支援的情況下，您可以使用本機群組政策編輯器停用記憶體完整性。下列指引適用於虛擬化型程式碼完整性保護下的每個組態設定：

- 已啟用但未鎖定：將設定修改為已停用以停用記憶體完整性。
- 已透過 UEFI 鎖定啟用：已透過 UEFI 鎖定啟用記憶體完整性。透過 UEFI 鎖定啟用記憶體完整性後，就無法停用記憶體完整性。建議您建立停用記憶體完整性的新執行個體，並終止未使用且不受支援的執行個體。

### 使用本機群組政策編輯器停用記憶體完整性

1. 使用遠端桌面通訊協定 (RDP) 以擁有管理員權限的使用者帳戶身分連線至執行個體。如需詳細資訊，請參閱 [the section called “使用 RDP 用戶端 Connect 至您的 Windows 執行個體”](#)。
2. 開啟「開始」選單並搜尋 **cmd** 以啟動命令提示。
3. 執行以下命令以開啟本機群組政策編輯器：`gpedit.msc`
4. 在本機群組政策編輯器中，選擇電腦設定、管理範本、系統、Device Guard。
5. 選取開啟基於虛擬化的安全性，然後選取編輯政策設定。
6. 開啟虛擬化型程式碼完整性保護的設定下拉式清單，選擇已停用，然後選擇套用。
7. 重新啟動執行個體以套用變更。

## 開啟認證保全

啟動具有受支援執行個體類型和相容 AMI 的 Windows 執行個體，並確認已停用記憶體完整性之後，可以開啟 Credential Guard。


### Important

需要管理員權限執行以下步驟來開啟 Credential Guard。

### 開啟 Credential Guard

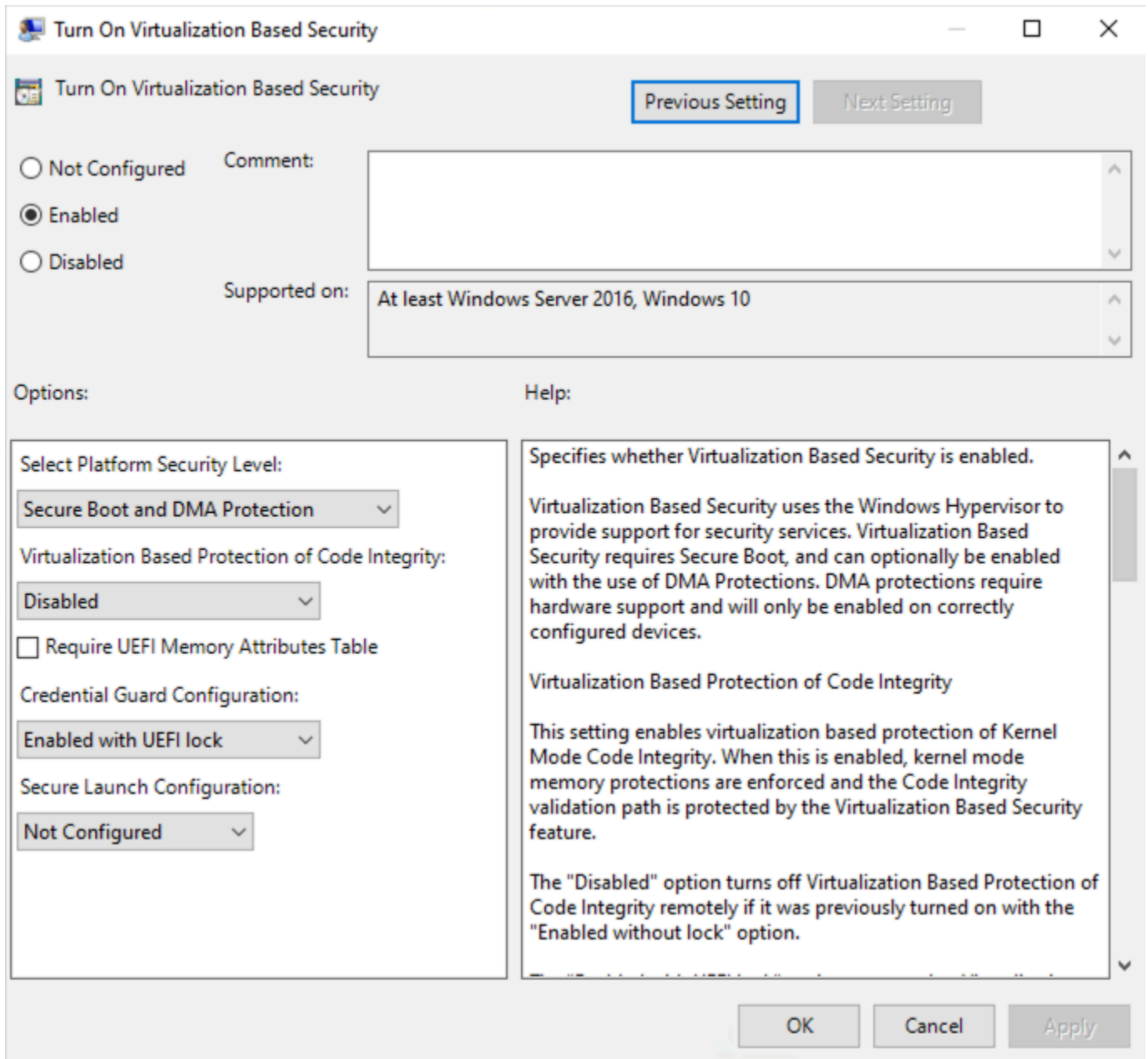
1. 使用遠端桌面通訊協定 (RDP) 以擁有管理員權限的使用者帳戶身分連線至執行個體。如需詳細資訊，請參閱 [the section called “使用 RDP 用戶端 Connect 至您的 Windows 執行個體”](#)。
2. 開啟「開始」選單並搜尋 **cmd** 以啟動命令提示。

3. 執行以下命令以開啟本機群組政策編輯器：`gpedit.msc`
4. 在本機群組政策編輯器中，選擇電腦設定、管理範本、系統、Device Guard。
5. 選取開啟基於虛擬化的安全性，然後選取編輯政策設定。
6. 在開啟基於虛擬化的安全性選單中選擇已啟用。
7. 針對選取平台安全性層級，選擇安全開機和 DMA 保護。
8. 對於 [認證保全組態]，選擇 [透過 UEFI 鎖定啟用]。

 Note

其餘的政策設定不需要來啟用 Credential Guard，而且可以保留為未設定。

下圖顯示如先前所述配置的 VBS 設定：



9. 重新啟動執行個體以套用設定。

## 確認認證保全正在執行

可以使用 Microsoft System Information (Msinfo32.exe) 工具來確認 Credential Guard 是否正在執行。

**⚠ Important**

您必須先重新啟動執行個體，才能完成套用啟用 Credential Guard 所需的政策設定。

**驗證 Credential Guard 正在執行**

1. 使用遠端桌面通訊協定 (RDP) 連線至執行個體。如需詳細資訊，請參閱 [the section called “使用 RDP 用戶端 Connect 至您的 Windows 執行個體”](#)。
2. 在執行個體的 RDP 工作階段中，開啟「開始」選單並搜尋 **cmd** 以啟動命令提示。
3. 透過執行以下命令，開啟 System Information : `msinfo32.exe`
4. Microsoft System Information 工具會列出 VBS 組態的詳細資訊。在基於虛擬化的安全性服務旁邊，確認 Credential Guard 顯示為正在執行。

下圖顯示 VBS 正在執行，如前所述：

Virtualization-based security	Running
Virtualization-based security Required Security Properties	Base Virtualization Support, Secure Boot, DMA Protection
Virtualization-based security Available Security Properties	Base Virtualization Support, Secure Boot, DMA Protection, UEFI Code Readonly, Mode Based Execution Control
Virtualization-based security Services Configured	Credential Guard
Virtualization-based security Services Running	Credential Guard

# Amazon EC2 執行個體的儲存選項

Amazon EC2 為您的執行個體提供彈性、符合成本效益的 easy-to-use 資料儲存選項。每種選項都有獨特的效能與耐用性組合。這些儲存選項可獨立或組合使用，因應您的要求。

## [Amazon EBS](#)

Amazon EBS 提供可從執行個體連接和分離的耐久區塊層級儲存體磁碟區。您可以將多個 EBS 磁碟區連接至同一個執行個體。EBS 磁碟區的存留與其相關執行個體的生命週期無關。您可以加密 EBS 磁碟區。若要保留資料的備份副本，您可以從 EBS 磁碟區建立快照。快照會存放在 Amazon S3。您可以從快照建立 EBS 磁碟區。

## [執行個體存放區](#)

執行個體存放區提供執行個體的暫時區塊層級儲存。執行個體儲存體磁碟區的數量、大小和類型取決於執行個體類型和執行個體大小。執行個體存放區磁碟區上的資料只會在相關聯執行個體的生命週期期間持續保留；如果您讓執行個體停止、休眠或終止，則會遺失執行個體存放區磁碟區上的任何資料。

## [Amazon EFS](#)(僅限 Linux 執行個體)

Amazon EFS 提供可擴展的檔案儲存體，以搭配 Amazon EC2 使用。您可以建立 EFS 檔案系統，並設定執行個體來掛載檔案系統。您可以使用 EFS 檔案系統做為多個執行個體上執行之工作負載和應用程式的共同資料來源。

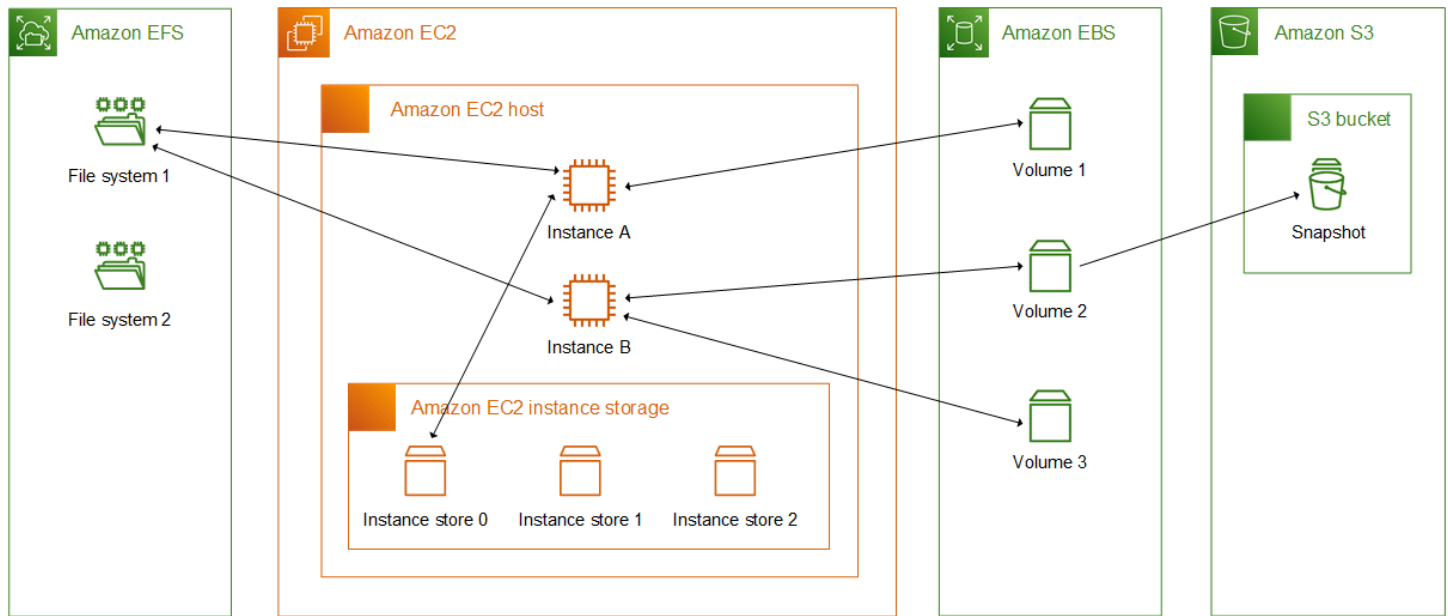
## [Amazon S3](#)

Amazon S3 提供對可靠且價廉之資料儲存體基礎設施的存取。其設計可以簡化 Web 規模運算，讓您存放和擷取任意數量的資料，且隨時可從 Amazon EC2 或 Web 上的任意位置執行。例如，您可以使用 Amazon S3 來存放資料和應用程式的備份複本。Amazon EC2 使用 Amazon S3 來存放 EBS 快照和執行個體後端 AMI。

## [Amazon FSx](#)

透過 Amazon FSx，您可以在雲端啟動、執行和擴展功能豐富的高效能檔案系統。Amazon FSx 是一項全受管服務，支援各種工作負載。您可以在這些廣泛使用的檔案系統之間進行選擇：光碟、NetApp ONTAP、OpenZFS 和視窗檔案伺服器。

下圖顯示這些儲存體選項和執行個體之間的關係。



## 儲存定價

開啟 [\[AWS 定價\]](#)，捲動至 [\[AWS 產品的定價\]](#)，然後選取 [\[儲存 選擇儲存產品\]](#)，開啟其定價頁面。

## 將 Amazon EBS 與亞 Amazon EC2 搭配使用

Amazon Elastic Block Store (Amazon EBS) 提供可擴展的高效能區塊儲存資源，可與 Amazon Elastic Compute Cloud (Amazon EC2) 執行個體搭配使用。使用 Amazon EBS，您可以建立和管理下列區塊儲存資源：

- 亞馬遜 EBS 磁碟區 — 這些是您連接到 Amazon EC2 執行個體的儲存磁碟區。將磁碟區連接至執行個體後，您可以使用與使用區塊儲存相同的方式來使用磁碟區。執行個體可與磁碟區互動，就像與本機磁碟機一樣。
- Amazon EBS 快照 — 這些是 Amazon EBS 磁碟區的 point-in-time 備份，可獨立於磁碟區本身保留。您可以建立快照以備份 Amazon EBS 磁碟區上的資料。然後，您可以隨時從這些快照還原新磁碟區。

您可以在啟動期間建立 Amazon EBS 磁碟區並將其連接到執行個體，也可以在啟動後隨時建立 EBS 磁碟區並將其連接到執行個體。您也可以在建後隨時從磁碟區建立快照。

如需使用磁碟區和快照的詳細資訊，請參閱 [Amazon EBS 使用者指南](#)。

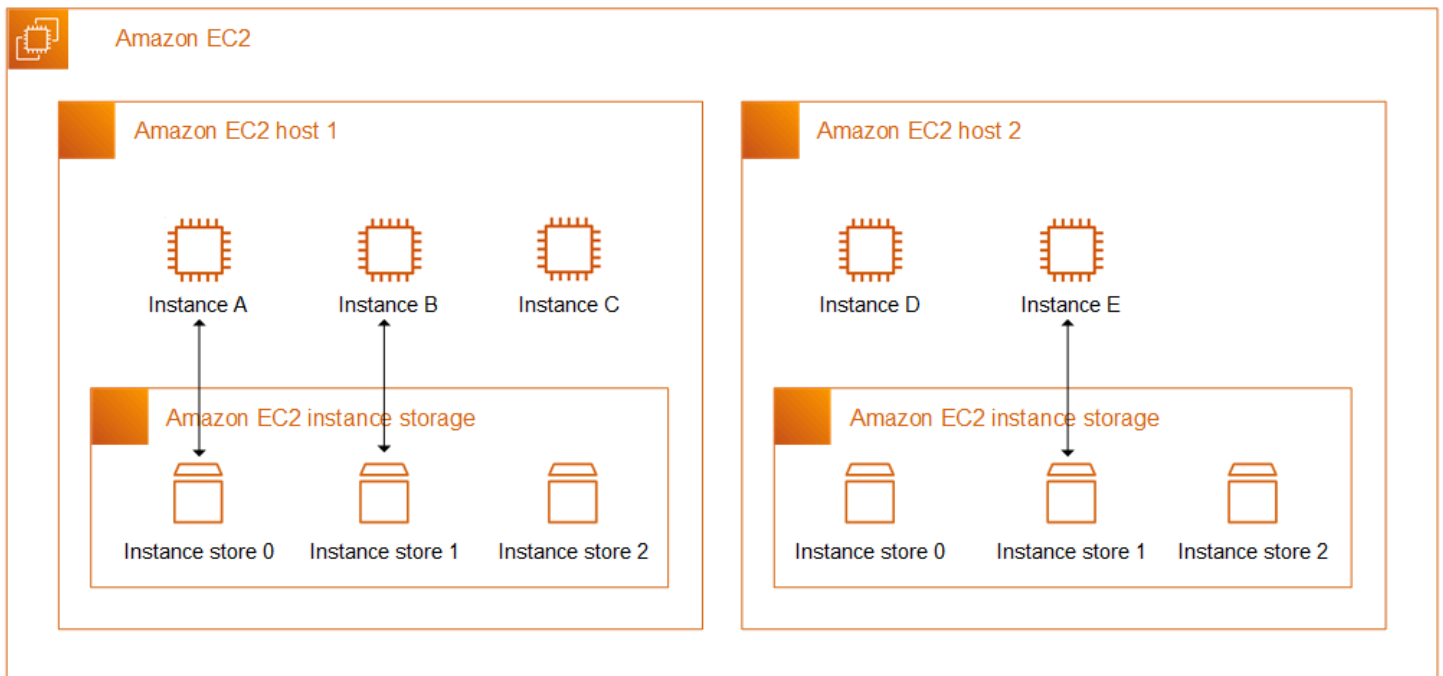


# Amazon EC2 執行個體存放區

執行個體存放區為執行個體提供暫時的區塊層級儲存空間。這個儲存空間位於實際連接到主機電腦的磁碟上。執行個體儲存體非常適合暫時儲存不斷變更的資訊，例如緩衝區、快取、臨時資料及其他暫存的內容。也可用來存放複製到整個執行個體機群的暫時性資料，例如網頁伺服器的負載平衡集區。

執行個體存放區包含顯示為區塊型儲存設備的一個或多個執行個體存放磁碟區。執行個體儲存體的大小和可用裝置的數量，會隨執行個體類型和執行個體大小而有不同。如需詳細資訊，請參閱 [執行個體儲存體磁碟區](#)。

執行個體存放磁碟區的虛擬裝置為 ephemeral[0-23]。支援一個執行個體存放磁碟區的執行個體類型具有 ephemeral0。支援兩個以上執行個體儲存體磁碟區的執行個體類型具有 ephemeral0、ephemeral1 等。



## 執行個體儲存體定價

執行個體的使用費包含了執行個體存放磁碟區。

## 目錄

- [執行個體儲存體磁碟區和資料存留期](#)
- [執行個體儲存體磁碟區](#)
- [將執行個體存放區磁碟區新增至 EC2 執行個體](#)
- [SSD 執行個體存放區磁碟區](#)

- [Linux 執行個體執行個體儲存交換磁碟](#)
- [優化 Linux 實例上實例儲存卷的磁盤性能](#)

## 執行個體儲存體磁碟區和資料存留期

執行個體儲存體磁碟區的數量、大小和類型取決於執行個體類型和執行個體大小。如需詳細資訊，請參閱 [執行個體儲存體磁碟區](#)。

執行個體儲存體磁碟區只會在執行個體啟動時連接。啟動後即無法連接執行個體儲存體磁碟區。您不能將執行個體儲存體磁碟區與某個執行個體中斷連接，然後再連接到不同的執行個體。

執行個體儲存體磁碟區只存在於所連接執行個體的存留期。您無法將執行個體儲存體磁碟區設定為保留超過其關聯執行個體的存留期。

即使將執行個體重新開機，執行個體儲存體磁碟區上的資料仍會存在。但如果停止、休眠或終止執行個體，則資料不會存在。執行個體停止、休眠或終止時，會以密碼編譯方式清除執行個體儲存體磁碟區的每個區塊。

因此，如果是重要的長期資料，請不要只使用執行個體儲存體磁碟區。如果您需要將儲存在執行個體儲存體磁碟區上的資料保留超過執行個體的存留期，必須手動將該資料複製到更持久的儲存體，例如 Amazon EBS 磁碟區、Amazon S3 儲存貯體或 Amazon EFS 檔案系統。

部分事件可能會導致您的資料在執行個體的整個存留期中未持續存在。下表指出在特定事件期間，虛擬化和裸機執行個體的執行個體儲存體磁碟區上的資料是否持續存在。

事件	您的資料會發生什麼事？
使用者啟動的執行個體生命週期事件	
<a href="#">執行個體已重新啟動</a>	數據仍然存在
<a href="#">執行個體已停止</a>	數據不會持續存在
<a href="#">執行個體進入休眠狀態</a>	數據不會持續存在
<a href="#">執行個體已終止</a>	數據不會持續存在
<a href="#">執行個體類型已變更</a>	資料不會持續存在 *
<a href="#">從執行個體建立 EBS 支援的 AMI</a>	數據不會保留在創建的 AMI 中 **

事件	您的資料會發生什麼事？
<a href="#">從執行個體 (Linux 執行個體) 建立執行個體支援的 AMI</a>	數據仍然存在於上傳到 Amazon S3 的 AMI 捆綁包中 ***
使用者啟動的作業系統事件	
啟動關閉	數據不會持續 +
啟動重新啟動	數據仍然存在
AWS 排程事件	
<a href="#">實例停止</a>	數據不會持續存在
<a href="#">實例重新啟</a>	數據仍然存在
<a href="#">系統重啟</a>	數據仍然存在
<a href="#">實例淘汰</a>	數據不會持續存在
意外事件	
<a href="#">簡化的自動恢復</a>	數據不會持續存在
<a href="#">CloudWatch 基於動作的復原</a>	數據不會持續存在
基礎磁碟失敗	故障磁碟上的資料不會持續存在
電源故障	重新啟動後數據仍然存在

\* 如果新執行個體類型支援執行個體儲存體，執行個體會具有新執行個體類型支援的執行個體儲存體磁碟區數量，但資料不會傳輸至新執行個體。如果新執行個體類型不支援執行個體儲存體，則執行個體不會取得執行個體儲存體磁碟區。

\*\* 資料不包含在 EBS 支援的 AMI 中，且不包含在連接至該 AMI 啟動之執行個體的執行個體儲存體磁碟區中。

\*\*\* 資料包含在上傳至 Amazon S3 的 AMI 套件中。從該 AMI 啟動執行個體時，執行個體會取得 AMI 中綁定的執行個體儲存體，以及建立 AMI 時包含的資料。

† 終止保護和停止保護不會防止執行個體因透過執行個體上的作業系統關機，而停止或終止。儲存在執行個體儲存體磁碟區上的資料不會保留在執行個體停止和終止事件中。

## 執行個體儲存體磁碟區

執行個體儲存體磁碟區的數量、大小和類型取決於執行個體類型和執行個體大小。部分執行個體類型 (如 M6、C6 和 R6) 不支援執行個體儲存體磁碟區，其他執行個體類型 (如 M5d、C6gd 和 R6gd) 則支援執行個體儲存體磁碟區。您無法將超過執行個體類型所支援的執行個體儲存體磁碟區數量，連接至執行個體。對於支援執行個體儲存體磁碟區的執行個體類型，執行個體儲存體磁碟區的數量和大小會依執行個體大小而異。例如，m5d.large 支援 1 個 75 GB 的執行個體儲存體磁碟區，而 m5d.24xlarge 支援 4 個 900 GB 的執行個體儲存體磁碟區。

對於具有 NVMe 執行個體儲存體磁碟區的執行個體類型，所有支援的執行個體儲存體磁碟區都會在啟動時自動連接至執行個體。對於具有非 NVMe 執行個體儲存體磁碟區的執行個體類型 (例如 C1、C3、M1、M2、R3、D2、H1、I2、X1 和 X1e)，您必須手動為要在啟動時連接的執行個體儲存體磁碟區指定區塊裝置對映。執行個體啟動後，您必須先[格式化並掛載連接的執行個體儲存體磁碟區](#)，才能使用它們。啟動執行個體之後，即無法連接執行個體儲存體磁碟區。

部分執行個體類型使用 NVMe 或 SATA 型固態硬碟 (SSD)，其他執行個體類型則使用 SATA 型硬碟 (HDD)。SSD 提供極低延遲的高隨機 I/O 效能，但您不希望資料在執行個體終止時仍持續存在，或是可以善用容錯架構。如需詳細資訊，請參閱 [SSD 執行個體存放區磁碟區](#)。

NVMe 執行個體存放區磁碟區和某些 HDD 執行個體存放區磁碟區上的資料會靜態加密。如需詳細資訊，請參閱 [Amazon EC2 中的資料保護](#)。

### 可用執行個體儲存體磁碟區

Amazon EC2 執行個體類型指南針對每個受支援的執行個體類型提供執行個體存放區磁碟區的數量、大小、類型和效能優化。如需詳細資訊，請參閱下列內容：

- [例項存放區規格 — 一般用途](#)
- [執行個體存放區規格 — 最佳化](#)
- [執行個體存放區規格 — 記憶體](#)
- [執行個體存放區規格 — 儲存優](#)
- [執行個體商店規格 — 加速運算](#)
- [執行個體存放區規格 — 高效能](#)
- [執行個體存放區規格 — 上一代](#)

## 若要使用擷取執行個體儲存磁碟區資訊 AWS CLI

您可以使用 `description` [-執行個體類型 AWS CLI 指令](#) 來顯示執行個體類型的相關資訊，例如其執行個體儲存磁碟區。下列範例顯示具有執行個體存放區磁碟區之所有 R5 執行個體的執行個體儲存體總大小。

```
aws ec2 describe-instance-types \
 --filters "Name=instance-type,Values=r5*" "Name=instance-storage-
supported,Values=true" \
 --query "InstanceTypes[].[InstanceType, InstanceStorageInfo.TotalSizeInGB]" \
 --output table
```

### 範例輸出

```

| DescribeInstanceTypes |
+-----+-----+
| r5ad.24xlarge | 3600 |
| r5ad.12xlarge | 1800 |
| r5dn.8xlarge | 1200 |
| r5ad.8xlarge | 1200 |
| r5ad.large | 75 |
| r5d.4xlarge | 600 |
| . . . |
| r5dn.2xlarge | 300 |
| r5d.12xlarge | 1800 |
+-----+-----+
```

下列範例顯示指定執行個體類型的完整執行個體儲存體詳細資訊。

```
aws ec2 describe-instance-types \
 --filters "Name=instance-type,Values=r5d.4xlarge" \
 --query "InstanceTypes[].InstanceStorageInfo"
```

範例輸出顯示此執行個體類型有兩個 300 GB NVMe SSD 磁碟區，總共有 600 GB 的執行個體儲存體。

```
[
 {
 "TotalSizeInGB": 600,
 "Disks": [
 {
```

```
 "SizeInGB": 300,
 "Count": 2,
 "Type": "ssd"
 }
],
 "NvmeSupport": "required"
}
]
```

## 將執行個體存放區磁碟區新增至 EC2 執行個體

對於具有 NVMe 執行個體儲存體磁碟區的執行個體類型，所有支援的執行個體儲存體磁碟區都會在啟動時自動連接至執行個體。執行個體啟動時，系統會自動列舉這些執行個體儲存體磁碟區並指派裝置名稱。

對於具有非 NVMe 執行個體儲存磁碟區的執行個體類型 (例如 C1、C3、M1、M2、R3、D2、H1、I2、X1 和 X1e)，您必須手動為要在啟動時連接的執行個體儲存磁碟區指定區塊裝置對映。您可以在執行個體啟動要求或用來啟動執行個體的 AMI 中，指定區塊型裝置映射。區塊型裝置映射包含裝置名稱和對應的磁碟區。如需更多資訊，請參閱[區塊型儲存設備對應](#)

### Important

您只能在啟動執行個體時，將執行個體儲存體磁碟區連接至執行個體。啟動執行個體之後，即無法將執行個體存放磁碟區連結到執行個體。

在啟動執行個體之後，您必須先確定執行個體的執行個體存放磁碟區已經過格式化並掛載，才能使用這些磁碟區。以執行個體存放區為後端的執行個體，其根磁碟區會自動掛載。

### 根磁碟區的考量

區塊型設備映射一律會指定執行個體的根磁碟區。根磁碟區一律會自動掛載。

**Linux 執行個體** — 根磁碟區可以是 Amazon EBS 磁碟區或執行個體存放區磁碟區。執行個體如果具有根磁碟區的執行個體存放磁碟區，則此磁碟區的大小會隨 AMI 而有不同，但其上限為 10 GB。如需詳細資訊，請參閱[根設備儲存](#)。

**視窗執行個體** — 根磁碟區必須是 Amazon EBS 磁碟區。根磁碟區不支援執行個體存放區。

### 目錄

- [將執行個體存放區的磁碟區新增至 AMI](#)

- [將非 NVMe 執行個體儲存體磁碟區新增至執行個體](#)
- [讓執行個體上的執行個體存放區的磁碟區變成可用](#)

## 將執行個體存放區的磁碟區新增至 AMI

您可以建立 AMI，此 AMI 具備包含執行個體存放磁碟區的區塊型設備映射。

如果您啟動執行個體，其使用指定執行個體儲存體磁碟區區塊型裝置映射的 AMI，支援非 NVMe 執行個體儲存體磁碟區，則執行個體包含這些執行個體儲存體磁碟區。如果 AMI 中執行個體儲存體磁碟區區塊型裝置映射的數量，超過執行個體可用的執行個體儲存體磁碟區數量，請忽略額外的執行個體儲存體磁碟區區塊型裝置映射。

如果您啟動執行個體，其使用指定執行個體儲存體磁碟區區塊型裝置映射的 AMI，支援 NVMe 執行個體儲存體磁碟區，則會忽略執行個體儲存體磁碟區區塊型裝置映射。支援 NVMe 執行個體儲存體磁碟區的執行個體會取得所有支援的執行個體儲存體磁碟區，無論執行個體啟動要求和 AMI 中指定的區塊型裝置映射為何。

### 考量事項

- 如果是 M3 執行個體，在執行個體的區塊型設備映射中指定執行個體存放區，而非 AMI。Amazon EC2 可能會忽略 AMI 中的執行個體儲存體磁碟區區塊型裝置映射。
- 啟動執行個體時，您可以略過在 AMI 區塊型設備映射中指定的非 NVMe 執行個體存放磁碟區，或新增執行個體存放磁碟區。

### New console

若要使用主控台來將執行個體存放磁碟區加入以 Amazon EBS 為後端的 AMI

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取該執行個體。
3. 選擇 Actions (動作)、Image and templates (映像和範本)、Create image (建立映像)。
4. 在 建立影像 頁面，為您的影像輸入有意義的名稱和說明。
5. 針對每個要新增的執行個體存放區的磁碟區，選取 Add New Volume (新增新磁碟區)、從 Volume Type (磁碟區類型) 選取一個執行個體存放區的磁碟區，並從 Device (裝置) 選取一個裝置名稱。(如需詳細資訊，請參閱「[Amazon EC2 實例上的設備名稱](#)」。) 可用執行個體存放磁碟區的數量，會視執行個體類型而定。如果是具備 NVMe 執行個體存放磁碟區的執行個體，這些磁碟區的裝置映射取決於作業系統列舉這些磁碟區的順序。

## 6. 選擇 Create image (建立映像)。

### AWS CLI

使用命令列來將執行個體存放磁碟區加入 AMI

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [create-image](#) 或 [register-image](#) (AWS CLI)
- [New-EC2Image](#)和 [Register-EC2Image](#) ( AWS Tools for Windows PowerShell )

### 將非 NVMe 執行個體儲存體磁碟區新增至執行個體

啟動支援非 NVMe 執行個體儲存體磁碟區的執行個體時，您必須為要連接的執行個體儲存體磁碟區，指定區塊型裝置映射。必須在執行個體啟動要求或用於啟動執行個體的 AMI 中，指定區塊型裝置映射。

如果 AMI 包含執行個體儲存體磁碟區的區塊型裝置映射，您不需要在執行個體啟動要求中指定區塊型裝置映射，除非您需要的執行個體儲存體磁碟區數量超過 AMI 中所包含的數量。

如果 AMI 不包括執行個體儲存體磁碟區的區塊型裝置映射，則必須在執行個體啟動請求中指定區塊型裝置映射。

### 考量事項

- 如果是 M3 執行個體，即使未在執行個體的區塊型設備映射中指定這些執行個體，您可能也會獲得執行個體存放磁碟區。

若要在執行個體啟動要求中指定區塊型裝置映射，請使用下列任一方法。

### Amazon EC2 console

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在儀表中，選擇啟動執行個體。
3. 在 Application and OS Images (應用程式和 OS 映像) 區段中，選取要使用的 AMI。
4. 在設定儲存體區段中，執行個體儲存體磁碟區區段會列出可連結至該執行個體的執行個體儲存體磁碟區。可用執行個體存放磁碟區的數量，會視執行個體類型而定。
5. 對於要連接的每個執行個體儲存體磁碟區，針對裝置名稱，請選取要使用的裝置名稱。



6. 根據需要進行剩餘的執行個體設定，然後選擇啟動執行個體。

## Command line

您可以使用下列任一命令，搭配對應的選項。

- `--block-device-mappings` 搭配 [run-instances](#) (AWS CLI)
- `-BlockDeviceMapping` 與 [New-EC2Instance](#) ( AWS Tools for Windows PowerShell )

## 讓執行個體上的執行個體存放區的磁碟區變成可用

啟動具有連接執行個體儲存體磁碟區的執行個體之後，您必須先掛接磁碟區，才能存取這些磁碟區。

### Note

許多執行個體存放磁碟區會預先格式化為 ext3 檔案系統。支援 TRIM 指示、採用 SSD 的執行個體存放磁碟區，不會預先格式化為任何檔案系統。不過，在啟動執行個體之後，您可以將磁碟區格式化為自選的檔案系統。如需詳細資訊，請參閱 [執行個體存放區磁碟區 TRIM 支援](#)。如果是 Windows 執行個體，系統會將執行個體儲存體磁碟區重新格式化為 NTFS 檔案系統。

## Linux 執行個體

您可以依照下列程序所述檢視和掛接執行個體儲存磁碟區。

在 Linux 上讓執行個體存放磁碟區變成可用

1. 使用 SSH 用戶端連線到執行個體。如需詳細資訊，請參閱 [連接至您的 Linux 執行個體](#)。
2. 使用 `df -h` 指令來檢視磁碟區是否已格式化和掛載。

```
$ df -h
Filesystem Size Used Avail Use% Mounted on
devtmpfs 3.8G 72K 3.8G 1% /dev
tmpfs 3.8G 0 3.8G 0% /dev/shm
/dev/nvme0n1p1 7.9G 1.2G 6.6G 15% /
```

3. 使用 `lsblk` 來檢視是否有任何磁碟區已在啟動時映射，但尚未格式化和掛載。

```
$ lsblk
```

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
nvme0n1	259:1	0	8G	0	disk	
##nvme0n1p1	259:2	0	8G	0	part	/
##nvme0n1p128	259:3	0	1M	0	part	
nvme1n1	259:0	0	69.9G	0	disk	

4. 若要只針對映射的執行個體存放磁碟區來進行格式化和掛載，請執行下列動作：

a. 使用 `mkfs` 指令來在裝置上建立檔案系統。

```
$ sudo mkfs -t xfs /dev/nvme1n1
```

b. 使用 `mkdir` 指令來建立用來掛載裝置的目錄。

```
$ sudo mkdir /data
```

c. 使用 `mount` 指令，將裝置掛載到新建立的目錄上。

```
$ sudo mount /dev/nvme1n1 /data
```

## Windows 執行個體

您也可以使用 Windows 磁碟管理來檢視執行個體儲存磁碟區。如需詳細資訊，請參閱 [使用磁碟管理列出磁碟](#)。

### 手動掛載執行個體存放區磁碟區

1. 選擇開始，輸入電腦管理，然後按 Enter。
2. 在左側面板中，選擇磁碟管理。
3. 如果系統提示您初始化磁碟區，請選取要初始化的磁碟區，根據您的使用案例選取所需的磁碟分割類型，然後選取 確定。
4. 在磁碟區清單中，以滑鼠右鍵按一下要裝載的磁碟區，然後選擇新增簡單磁碟區。
5. 在精靈上，選擇下一步。
6. 在「指定磁碟區大小」畫面上，選擇下一步以使用最大磁碟區大小。或者，選擇介於最小和最大磁碟空間之間的磁碟區大小。
7. 在「指派磁碟機代號或路徑」畫面上，執行下列其中一項作業，然後選擇下一步。
  - 若要使用磁碟機代號掛接磁碟區，請選擇指派下列磁碟機代號，然後選擇要使用的磁碟機代號。

- 若要將磁碟區掛接為資料夾，請在下列空白 NTFS 資料夾中選取掛接，然後選取瀏覽以建立或選取要使用的資料夾。
  - 若要在沒有磁碟機代號或路徑的情況下掛接磁碟區，請選擇不要指派磁碟機代號或磁碟機路徑。
8. 在「格式化磁碟分割」畫面上，指定是否要格式化磁碟區。如果您選擇格式化磁碟區，請選擇所需的檔案系統和單位大小，然後指定磁碟區標籤。
  9. 依序選擇下一步，完成。

如需有關如何在重新開機後自動掛接連接的磁碟區的指示，請參閱 Amazon EBS 使用者指南中的[重新開機後自動掛接連接的磁碟區](#)。

## SSD 執行個體存放區磁碟區

如同其他的執行個體存放磁碟區，您必須在啟動執行個體時，對應執行個體的 SSD 執行個體存放磁碟區。SSD 執行個體磁碟區上的資料，只會存在於其相關執行個體的生命週期期間。如需詳細資訊，請參閱[將執行個體存放區磁碟區新增至 EC2 執行個體](#)。

## NVMe SSD 磁碟區

某些執行個體提供非揮發性記憶體儲存裝置 (NVMe) 固態硬碟 (SSD) 執行個體存放區磁碟區。關於每種執行個體類型所支援的執行個體存放磁碟區類型，詳細資訊請參閱[執行個體儲存體磁碟區](#)。

NVMe 執行個體儲存體上的資料會以執行個體上的硬體模組中實作的 XTS-AES-256 區塊編碼器來加密。加密金鑰是以硬體模組來產生，且對每個 NVMe 執行個體儲存體設備而言是唯一的。所有加密金鑰會在執行個體停止或終止時銷毀，且無法復原。您無法停用此加密，也無法提供您自己的加密金鑰。

## Linux 執行個體

若要存取 NVMe 磁碟區，必須安裝 NVMe 驅動程式。下列 AMI 符合此需求：

- AL2023
- Amazon Linux 2
- Amazon Linux AMI 2018.03 及更新版本
- Ubuntu 14.04 或更新版本 (帶 linux-aws 核心)

### Note

AWS 以重力為基礎的執行個體類型需要 Ubuntu 18.04 或更新版本搭配核心 linux-aws

- Red Hat Enterprise Linux 7.4 或更新版本
- SUSE Linux Enterprise Server 12 SP2 或更新版本
- CentOS 7.4.1708 或更新版本
- FreeBSD 11.1 或更新版本
- Debian GNU/Linux 9 或更新版本
  
- Bottlerocket

在連線到執行個體之後，您可以使用 `lspci` 指令，來列出 NVMe 裝置。下列是 `i3.8xlarge` 執行個體的輸出範例，此執行個體支援四種 NVMe 裝置。

```
[ec2-user ~]$ lspci
00:00.0 Host bridge: Intel Corporation 440FX - 82441FX PMC [Natoma] (rev 02)
00:01.0 ISA bridge: Intel Corporation 82371SB PIIX3 ISA [Natoma/Triton II]
00:01.1 IDE interface: Intel Corporation 82371SB PIIX3 IDE [Natoma/Triton II]
00:01.3 Bridge: Intel Corporation 82371AB/EB/MB PIIX4 ACPI (rev 01)
00:02.0 VGA compatible controller: Cirrus Logic GD 5446
00:03.0 Ethernet controller: Device 1d0f:ec20
00:17.0 Non-Volatile memory controller: Device 1d0f:cd01
00:18.0 Non-Volatile memory controller: Device 1d0f:cd01
00:19.0 Non-Volatile memory controller: Device 1d0f:cd01
00:1a.0 Non-Volatile memory controller: Device 1d0f:cd01
00:1f.0 Unassigned class [ff80]: XenSource, Inc. Xen Platform Device (rev 01)
```

如果使用支援的作業系統，但是未顯示 NVMe 裝置，請利用下列命令來確認已載入 NVMe 模組。

- Amazon Linux, Amazon Linux 2, Ubuntu 14/16, Red Hat Enterprise Linux, SUSE Linux Enterprise Server, CentOS 7

```
$ lsmod | grep nvme
nvme 48813 0
```

- Ubuntu 18

```
$ cat /lib/modules/$(uname -r)/modules.builtin | grep nvme
s/nvme/host/nvme-core.ko
kernel/drivers/nvme/host/nvme.ko
kernel/drivers/nvmem/nvmem_core.ko
```

NVMe 磁碟區符合 NVMe 1.0e 規格。您可以對 NVMe 磁碟區使用 NVMe 指令。如果使用 Amazon Linux，您可以使用 `nvme-cli` 命令，從儲存庫安裝 `yum install` 套件。使用其他支援的 Linux 版本時，如果映像中未提供，您可以下載 `nvme-cli` 套件。

## Windows 執行個體

適用於下列作業系統的最新 AWS Windows AMI 包含 AWS NVMe 驅動程式，可用來與 SSD 執行個體儲存磁碟區互動，這些磁碟區會公開為 NVMe 區塊裝置，以獲得更好的效能：

- Windows Server 2022
- Windows Server 2019
- Windows Server 2016
- Windows Server 2012 R2

連線到執行個體之後，您可以在 Disk Manager 中，確認是否顯示 NVMe 磁碟區。在任務列上開啟 Windows 標誌的內容 (按右鍵) 選單，然後選擇 Disk Management (磁碟管理)。

Amazon 提供的 AWS 視窗 AMI 包括 AWS NVMe 驅動程序。如果您沒有使用最新的 AWS 視窗 AMI，您可以[安裝目前的 AWS NVMe 驅動程式](#)。

## 非 NVMe SSD 磁碟區

下列執行個體支援使用非 NVMe 固態硬碟以提供高隨機 I/O 效能的執行個體儲存磁碟區：

C3、I2、M3、R3 和 X1。關於每種執行個體類型所支援的執行個體存放磁碟區，詳細資訊請參閱[執行個體儲存體磁碟區](#)。

## SSD 型執行個體儲存體磁碟區 I/O 效能

隨著執行個體的 SSD 執行個體存放區磁碟區越來越滿，可達到的寫入 IOPS 次數將下降。這是因為 SSD 控制器必須執行額外的操作，去尋找可用的空間、重寫現有資料，以及清除未使用的空間以供重寫。這個收集廢棄項目的程序會造成 SSD 的內部寫入放大，也就是 SSD 寫入操作與使用者寫入操作的比例放大。假如寫入操作不是 4,096 位元組的倍數或未符合 4,096 位元組的邊界，效能下降的幅度會更明顯。因為當您寫入少量位元組或未符合的位元組時，SSD 控制器必須讀取週圍的資料並將結果存放到新的位置，此模式會造成寫入放大大幅增加，提高延遲，因而使 I/O 效能巨幅下降。

SSD 控制器利用多項策略來減少寫入放大的影響。其中一項策略，就是在 SSD 執行個體儲存體內保留空間，讓控制器能更有效管理寫入操作可用的空間。此策略稱為過度佈建。提供給執行個體的 SSD 型執行個體儲存磁碟區沒有任何保留空間以進行過度佈建。為了減少寫入放大，我們建議您將 10% 的磁

碟區保留未分割，以便 SSD 控制器可以將其用於過度佈建。這樣做雖然會減少可用的儲存空間，但可提高效能，即使磁碟即將用完所有容量。

例如儲存支援 TRIM 的磁碟區，您可以在不再需要寫入的資料時，使用 TRIM 指令來通知 SSD 控制器。這會讓控制器有更多的可用空間能夠使用，可減少寫入放大現象，並提升效能。如需詳細資訊，請參閱 [執行個體存放區磁碟區 TRIM 支援](#)。

## 執行個體存放區磁碟區 TRIM 支援

某些執行個體類型支援具有 TRIM 的 SSD 磁碟區。如需詳細資訊，請參閱 [執行個體儲存體磁碟區](#)。

### Note

(僅限視窗執行個體) 執行視窗伺服器 2012 R2 的執行個體支援從 AWS 光伏驅動程式 7.3.0 版起的 TRIM。執行較早版本 Windows Server 的執行個體不支援 TRIM。

支援 TRIM 的執行個體存放磁碟區，會在分配給執行個體之前完全停用刪除通知 (執行 trim 指令)。在執行個體啟動時，這些磁碟區不會格式化為任何檔案系統，因此您必須先將這些磁碟區格式化之後，才能掛載和使用這些磁碟區。若要更快速地存取這些磁碟區，應在將這些磁碟區格式化時跳過 TRIM 操作。

(Windows 執行個體) 若要在初始格式化期間暫時停用 TRIM 支援，請使用 `fsutil behavior set DisableDeleteNotify 1` 指令。格式化完成後，使用重新啟用 `fsutil behavior set DisableDeleteNotify 0` TRIM 支援。

使用支援 TRIM 的執行個體存放磁碟區時，當不再需要已寫入的資料時，可以利用 TRIM 指令來通知 SSD 控制器。這會讓控制器有更多的可用空間能夠使用，可減少寫入放大現象，並提升效能。在 Linux 執行個體上，使用 `fstrim` 指令來啟用定期 TRIM。在 Windows 執行個體上，使用 `fsutil behavior set DisableDeleteNotify 0` 指令確保在正常作業期間啟用 TRIM 支援。

## Linux 執行個體執行個體儲存交換磁碟

### Note

本主題僅適用於 Linux 執行個體。

當系統需要比實體配置更多的記憶體時，可以使用 Linux 的置換空間。啟用置換空間時，Linux 系統可以將不常使用的記憶體分頁，從實體記憶體換到置換空間 (專用磁區或是現有檔案系統中的調換檔)，以釋出空間，給需要高速存取的記憶體分頁使用。

### Note

使用置換空間來做為記憶體分頁時，其速度和效率並不如使用 RAM。如果工作負載會定期將分頁記憶體換到置換空間中，您應考慮轉而使用較大的執行個體類型和更多的 RAM。如需詳細資訊，請參閱 [變更執行個體類型](#)。

c1.medium 和 m1.small 執行個體類型可用的實體記憶體數量有限，且啟動時會獲得 900 MB 置換磁碟區做為 Linux AMIs 的虛擬記憶體。雖然 Linux 核心將此置換空間視為根設備上的分割區，但該空間實際上為獨立的執行個體存放磁碟區，而不論根設備磁碟區是何種類型。

Amazon Linux 會自動啟用和使用此置換空間，但 AMI 可能會需要執行一些其他的步驟才能辨識和使用此置換空間。若要查看執行個體是否正在使用置換空間，您可以使用 `swapon -s` 命令。

```
[ec2-user ~]$ swapon -s
```

Filename	Type	Size	Used	Priority
/dev/xvda3	partition	917500	0	-1

上述的執行個體連結和啟用了 900 MiB 的智慧磁碟區。如果使用此指令未列出置換磁碟區，可能需要啟用該設備的置換空間。使用 `lsblk` 命令來檢查可用的磁碟。

```
[ec2-user ~]$ lsblk
```

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
xvda1	202:1	0	8G	0	disk	/
xvda3	202:3	0	896M	0	disk	

在此處，置換磁碟區 `xvda3` 可供執行個體使用，但是並未啟用 (請注意，`MOUNTPOINT` 欄位為空白)。您可以利用 `swapon` 命令來啟用置換磁碟區。

### Note

您必須在 `/dev/` 命令所列出的裝置名稱前方，加上 `lsblk`。裝置可能需有不同的名稱，例如 `sda3`、`sde3` 或 `xvde3`。在下列的指令中使用系統的裝置名稱。

```
[ec2-user ~]$ sudo swapon /dev/xvda3
```

現在置換空間應該會顯示 `lsblk` 和 `swapon -s` 輸出中。

```
[ec2-user ~]$ lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
xvda1 202:1 0 8G 0 disk /
xvda3 202:3 0 896M 0 disk [SWAP]
[ec2-user ~]$ swapon -s
Filename Type Size Used Priority
/dev/xvda3 partition 917500 0 -1
```

您也需要編輯 `/etc/fstab` 檔案，讓此一置換空間在每次系統開機時自動啟用。

```
[ec2-user ~]$ sudo vim /etc/fstab
```

將下列的指令行附加到 `/etc/fstab` 檔案之後 (使用系統的置換裝置名稱)：

```
/dev/xvda3 none swap sw 0 0
```

將執行個體存放磁碟區做為置換空間使用

任何執行個體存放磁碟區皆可做為置換空間使用。例如，`m3.medium` 執行個體類型包含適合做為置換空間的 4 GB SSD 執行個體存放磁碟區。如果執行個體存放磁碟區的容量大上許多 (例如 350 GB)，您可以考慮將磁碟區分割出一個較小的 4 到 8 GB 置換磁區，其他的部分則做為資料磁碟區。

### Note

此程序僅適用於支援執行個體儲存體的執行個體類型。如需支援的執行個體類型清單，請參閱 [執行個體儲存體磁碟區](#)。

1. 列出連結至執行個體的區塊型儲存設備，以取得執行個體存放磁碟區的裝置名稱。

```
[ec2-user ~]$ lsblk -p
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
/dev/xvdb 202:16 0 4G 0 disk /media/ephemeral0
/dev/xvda1 202:1 0 8G 0 disk /
```



在此範例中，執行個體存放磁碟區為 `/dev/xvdb`。因為這是 Amazon Linux 執行個體，因此執行個體存放磁碟區會經過格式化並掛載於 `/media/ephemeral0`；不是所有的 Linux 作業系統都會自動執行這項動作。

2. (選用) 如果執行個體存放磁碟區已掛載 (會在 `MOUNTPOINT` 命令輸出中列出 `lsblk`)，請利用下列指令來卸載該磁碟區。

```
[ec2-user ~]$ sudo umount /dev/xvdb
```

3. 使用 `mkswap` 命令，在裝置上設定 Linux 置換區。

```
[ec2-user ~]$ sudo mkswap /dev/xvdb
mkswap: /dev/xvdb: warning: wiping old ext3 signature.
Setting up swap space version 1, size = 4188668 KiB
no label, UUID=b4f63d28-67ed-46f0-b5e5-6928319e620b
```

4. 啟用新的置換空間。

```
[ec2-user ~]$ sudo swapon /dev/xvdb
```

5. 確認新的置換空間正在使用。

```
[ec2-user ~]$ swapon -s
Filename Type Size Used Priority
/dev/xvdb partition 4188668 0 -1
```

6. 編輯 `/etc/fstab` 檔案，讓此一置換空間在每次系統開機時自動啟用。

```
[ec2-user ~]$ sudo vim /etc/fstab
```

如果 `/etc/fstab` 檔案具有 `/dev/xvdb` (或 `/dev/sdb`) 的項目，請修改為和下列行相同；如果該檔案不具有此裝置的項目，請將下列行附加於 `/etc/fstab` 檔案之後 (讓系統使用置換裝置的名稱)：

```
/dev/xvdb none swap sw 0 0
```

**⚠ Important**

當執行個體停止或休眠時，執行個體存放磁碟區上的資料會遺失；這包括在 [Step 3](#) 中所建立的執行個體存放區置換空間格式。如果停止和重新啟動執行個體，而此執行個體已設定使用執行個體存放區置換空間，則您必須在新的執行個體存放磁碟區上，重複執行 [Step 1](#) 到 [Step 5](#)。

## 優化 Linux 實例上實例存儲卷的磁盤性能

**📘 Note**

本主題僅適用於 Linux 執行個體。

由於 Amazon EC2 將磁碟虛擬化的方式，在某些執行個體存放磁碟區上，第一次寫入任意位置的動作，其執行速度會比後續的寫入動作慢。對大多數應用程式而言，在執行個體的整個生命週期內分攤此效能減損是可以接受的。不過，如果要求高度磁碟效能，建議在正式生產使用之前，藉由寫入每個磁碟位置一次，來將磁碟初始化。

**📘 Note**

某些執行個體類型具備直連式固態硬碟 (SSD) 和 TRIM 支援，可在啟動時提供最高效能，不需初始化。關於適用於每種執行個體類型的執行個體存放區，詳細資訊請參閱 [執行個體儲存體磁碟區](#)。

如果需要在延遲或傳輸量方面獲得更高的彈性，建議使用 Amazon EBS。

若要將執行個體存放磁碟區初始化，請根據要初始化的存放區 (例如 `dd` 或 `/dev/sdb`)，使用下列 `dev/nvme1n1` 指令來進行。

**📘 Note**

請務必在執行此指令之前卸載磁碟。

初始化作業可能需要很長的時間來完成 (如果是超大型的執行個體，約需要 8 小時)。

若要將執行個體存放磁碟區初始化，請針對

m1.large、m1.xlarge、c1.xlarge、m2.xlarge、m2.2xlarge 和 m2.4xlarge 執行個體類型使用下列指令：

```
dd if=/dev/zero of=/dev/sdb bs=1M
dd if=/dev/zero of=/dev/sdc bs=1M
dd if=/dev/zero of=/dev/sdd bs=1M
dd if=/dev/zero of=/dev/sde bs=1M
```

若要在所有的執行個體存放磁碟區上同時進行初始化，請使用下列指令：

```
dd if=/dev/zero bs=1M|tee /dev/sdb|tee /dev/sdc|tee /dev/sde > /dev/sdd
```

設定 RAID 的磁碟這項動作，會藉由寫入所有磁碟位置來將這些磁碟初始化。設定軟體式 RAID 時，請務必變更最低重建速度：

```
echo $((30*1024)) > /proc/sys/dev/raid/speed_limit_min
```

## 檔案儲存

雲端檔案儲存是一種在雲端存放資料的方式，提供伺服器 and 應用程式透過共享檔案系統存取資料。這種相容性使得雲端檔案儲存非常適合用於倚賴共享檔案系統的工作負載，而且提供簡易的整合，無須程式碼變更。

目前存在許多檔案儲存解決方案，從運算執行個體上的單一節點檔案伺服器使用區塊儲存做為沒有延展性的基礎，或是保護資料的冗餘性很少，到 do-it-yourself 叢集解決方案，到完全管理的解決方案。以下內容介紹一些由 Amazon EC2 執行個體使 AWS 用所提供的儲存服務。

### 目錄

- [搭配使用 Amazon S3 與 Amazon EC2](#)
- [將 Amazon EFS 搭配 Linux 執行個體](#)
- [使用 Amazon FSx 搭配 Amazon EC2](#)
- [搭配 Amazon EC2 使用亞馬遜檔案快取](#)

## 搭配使用 Amazon S3 與 Amazon EC2

Amazon Simple Storage Service (Amazon S3) 是一項物件儲存服務，提供領先業界的可擴展性、資料可用性、安全性和效能。您可以使用 Amazon S3 存放和擷取各種使用案例的任意數量資料，例如資料湖、網站、備份和大數據分析、從 Amazon EC2 執行個體或網際網路上的任何地方。如需詳細資訊，請參閱[什麼是 Amazon S3？](#)

物件是存放在 Amazon S3 中的基本實體。Amazon S3 中存放的每個物件都包含在儲存貯體中。儲存貯體在最高層級組織 Amazon S3 命名空間，並識別負責該儲存體的帳戶。Amazon S3 儲存貯體與網際網路網域名稱類似。儲存貯體內存放的物件都有一個唯一的金鑰值，可用 URL 擷取。例如，含有 /photos/mygarden.jpg 金鑰值的物件存放在 DOC-EXAMPLE-BUCKET1 儲存貯體中，則可以使用 URL `https://DOC-EXAMPLE-BUCKET1.s3.amazonaws.com/photos/mygarden.jpg` 定址。如需詳細資訊，請參閱[Amazon S3 的運作方式](#)。

### 使用範例

既然 Amazon S3 有儲存上的優勢，您可能會決定用這項服務來存放檔案和資料集，以供 EC2 執行個體使用。有幾種方法可在 Amazon S3 與執行個體之間移動資料。除了底下討論的範例之外，您也能從電腦或執行個體使用其他人所編寫的多種工具來存取 Amazon S3 中的資料。其中一些常見工具的討論請參閱 AWS 論壇。

如果您有許可，您可用下列其中一種方式在 Amazon S3 和執行個體之間複製檔案。

GET or wget (Linux)

#### Note

此方法僅適用於公有物件。如果物件非公有，您會收到 ERROR 403: Forbidden 訊息。如果收到此錯誤，則必須使用 Amazon S3 主控台、AWS API AWS CLI、AWS 開發套件或 AWS Tools for Windows PowerShell，而且您必須擁有必要的許可。如需詳細資訊，請參閱 Amazon S3 使用者指南中的[Amazon S3 中的身分和存取管理及下載物件](#)。

wget 公用程式是一種 HTTP 和 FTP 用戶端，可讓您從 Amazon S3 下載公有物件。其預設安裝在 Amazon Linux 和多數其他的發行版本上，也可供 Windows 下載。若要下載 Amazon S3 物件，請用下列命令並更換要下載之物件的 URL。

```
[ec2-user ~]$ wget https://my_bucket.s3.amazonaws.com/path-to-file
```

## AWS Tools for Windows PowerShell (Windows)

Windows 執行個體的優點為使用圖形瀏覽器，可讓您用來直接存取 Amazon S3 主控台；不過，在編寫指令碼時，Windows 使用者也可使用 [AWS Tools for Windows PowerShell](#) 在 Amazon S3 之間移動物件。

請用下列命令將 Amazon S3 物件複製至 Windows 執行個體。

```
PS C:\> Copy-S3Object -BucketName my_bucket -Key path-to-file -
LocalFile my_copied_file.ext
```

## AWS CLI (Linux and Windows)

AWS Command Line Interface (AWS CLI) 是用來管理您的 AWS 服務的統一工具。AWS CLI 可讓使用者進行自我身分驗證，以及從 Amazon S3 下載受限制的項目，還有上傳項目。如需詳細資訊，包括如何安裝與設定工具，請參閱 [AWS Command Line Interface 詳細資訊頁面](#)。

aws s3 cp 命令結果類似以下命令：Unix cp。可讓您從 Amazon S3 複製檔案到執行個體，從執行個體複製檔案到 Amazon S3，以及從某一 Amazon S3 位置複製檔案到其他位置。

請用下列命令從 Amazon S3 將物件複製至執行個體。

```
aws s3 cp s3://my_bucket/my_folder/my_file.ext my_copied_file.ext
```

請用下列命令從執行個體將物件複製回 Amazon S3。

```
aws s3 cp my_copied_file.ext s3://my_bucket/my_folder/my_file.ext
```

aws s3 sync 命令可將整個 Amazon S3 儲存貯體同步至本機目錄位置。這對於下載資料集以及將本機副本與遠端設定— up-to-date 起保留很有幫助。如果您擁有 Amazon S3 儲存貯體的適當許可，還能將命令中的來源與目的地位置對調，於完成後將本機目錄備份推送至雲端。

請用下列命令將整個 Amazon S3 儲存貯體下載至執行個體上的本機目錄。

```
aws s3 sync s3://remote_S3_bucket local_directory
```

## Amazon S3 API

如果您是開發人員，可以使用 API 存取 Amazon S3 中的資料。您可以使用此 API 協助開發應用程式，並將其與其他 API 和 SDK 整合。如需詳細資訊，請參閱 [Amazon S3 使用者指南中的使用 AWS 開發套件的 Amazon S3 程式碼範例](#)。

## 將 Amazon EFS 搭配 Linux 執行個體

### Note

Windows 執行個體上不支援 Amazon EFS。

Amazon EFS 提供可擴展的檔案儲存體，以搭配 Amazon EC2 使用。您可以使用 EFS 檔案系統做為多個執行個體上執行之工作負載和應用程式的共同資料來源。如需詳細資訊，請參閱 [Amazon Elastic File System 產品頁面](#)。

本教學說明如何在執行個體啟動期間使用 Amazon EFS 快速建立精靈建立和連接 Amazon EFS 檔案系統。如需有關如何使用 Amazon EFS 主控台建立檔案系統的教學課程，請參閱《Amazon Elastic File System 使用者指南》中的 [Amazon Elastic File System 入門](#)。

### Note

當您使用 EFS Quick Create 建立 EFS 檔案系統時，會使用下列服務建議的設定建立檔案系統：

- 啟用 [自動備份](#)。
- 在所選 VPC [中的每個預設子網路中掛載目標](#)。
- [一般用途效能模式](#)。
- [大量批量輸送量模式](#)。
- 使用 Amazon EFS [的預設金鑰啟用靜態資料加密](#) (aws/elasticfilesystem)。
- 透過 30 天的政策 [啟用 Amazon EFS 生命週期管理功能](#)。

### 任務

- [使用 Amazon EFS Quick Create 建立 EFS 檔案系統](#)
- [測試 EFS 檔案系統](#)
- [刪除 EFS 檔案系統](#)

## 使用 Amazon EFS Quick Create 建立 EFS 檔案系統

您可以建立 EFS 檔案系統，並在使用 Amazon EC2 [啟動執行個體精靈](#) 的 Amazon EFS Quick Create 功能啟動您的執行個體時，將其掛載到您的執行個體。

### 使用 Amazon EFS Quick Create 建立 EFS 檔案系統

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 選擇啟動執行個體。
3. (選用) 在 Name and tags (名稱和標籤) 下，針對 Name (名稱)，輸入用於識別執行個體的名稱。
4. 在 Application and OS Images (Amazon Machine Image) (應用程式和作業系統映像 (Amazon Machine Image)) 下，請選取 Linux 作業系統，然後針對 Amazon Machine Image (AMI)，選取 Linux AMI。
5. 在 Instance type (執行個體類型) 下，針對 Instance type (執行個體類型)，選取執行個體類型或保留預設值。
6. 在 Key pair (login) (金鑰對 (登入)) 下，針對 Key pair name (金鑰對名稱)，選擇現有的金鑰對或建立新的金鑰對。
7. 在 Network settings (網路設定) 下，選取 Edit (編輯) (在右側)，然後針對 Subnet (子網)，選取一個子網。

#### Note

您必須先選取子網，才能新增 EFS 檔案系統。

8. 在 Configure storage (設定儲存) 下，選擇 Edit (編輯) (在右下角)，然後執行下列動作：
  - a. 針對檔案系統，請確定已選取 EFS，然後選擇 [建立新的共用檔案系統]。
  - b. 針對檔案系統名稱，輸入 Amazon EFS 檔案系統的名稱，然後選擇 [建立檔案系統]。
  - c. 對於「掛載點」，請指定自訂掛載點或保留預設掛載點。
  - d. 若要啟用檔案系統的存取權，請選取 Automatically create and attach security groups (自動建立和連接安全群組)。選取此核取方塊後，會自動建立下列安全群組，並將其附加至檔案系統的執行個體和掛載目標：
    - 執行個體安全性群組 — 包含允許透過 NFS 2049 連接埠流量的輸出規則，但不包含輸入規則。
    - 檔案系統掛載目標安全群組 - 包含允許來自執行個體安全群組的 NFS 2049 連接埠流量的傳入規則 (如上所述)，以及允許透過 NFS 2049 連接埠流量的傳出規則。

**Note**

或者，您可以手動建立並連接安全性群組。如果您要手動建立並連接安全群組，請清除 Automatically create and attach the required security groups (自動建立並連接必要的安全群組)。

- e. 若要在執行個體啟動時自動掛載共用檔案系統，請選取 Automatically mount shared file system by attaching required user data script (透過連接所需的使用者指令碼自動掛載共用檔案系統)。若要檢視自動產生的使用者資料，請展開 Advanced details (進階詳細資訊)，然後向下捲動至 User data (使用者資料)。

**Note**

如果您在選取此核取方塊之前加入了使用者資料，原始使用者資料將被自動產生的使用者資料覆寫。

9. 視需要進行任何其他執行個體組態設定。
10. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。如需詳細資訊，請參閱 [使用新的啟動執行個體精靈啟動執行個體](#)。

## 測試 EFS 檔案系統

您可以連線至執行個體，並確認將檔案系統掛載至您指定的目錄 (例如 /mnt/efs)。

### 確認檔案系統已掛載

1. 連線到您的執行個體。如需詳細資訊，請參閱 [連接至您的 Linux 執行個體](#)。
2. 從執行個體的終端機視窗，執行 `df -T` 命令，確認 EFS 檔案系統已掛載。

```
$ df -T
Filesystem Type 1K-blocks Used Available Use% Mounted
on
/dev/xvda1 ext4 8123812 1949800 6073764 25% /
devtmpfs devtmpfs 4078468 56 4078412 1% /dev
tmpfs tmpfs 4089312 0 4089312 0% /dev/shm
efs-dns nfs4 9007199254740992 0 9007199254740992 0% /mnt/efs
```



請注意，檔案系統名稱 (在範例輸出中顯示為 *efs-dns*) 的格式如下。

```
file-system-id.efs.aws-region.amazonaws.com:/
```

3. (選用) 在檔案系統中從執行個體建立檔案，然後確認您可以從另一個執行個體檢視檔案。
  - a. 從執行個體中，執行下列命令來建立檔案。

```
$ sudo touch /mnt/efs/test-file.txt
```

- b. 從另一個執行個體中，執行下列命令來檢視檔案。

```
$ ls /mnt/efs
test-file.txt
```

## 刪除 EFS 檔案系統

若您不再需要檔案系統，可以刪除它。

### 刪除檔案系統

1. 開啟 Amazon Elastic File System 主控台，網址為 <https://console.aws.amazon.com/efs/>。
2. 選取要刪除的檔案系統。
3. 選擇 Actions (動作)、Delete file system (刪除檔案系統)。
4. 出現確認提示時，請輸入檔案系統 ID，然後選擇 Delete file system (刪除檔案系統)。

## 使用 Amazon FSx 搭配 Amazon EC2

Amazon FSx 系列服務使您可以輕鬆啟動、執行和擴展由常用商業和開放原始碼檔案系統提供支援的共用儲存體。您可以使用新的啟動執行個體精靈，在啟動時自動將以下類型的 Amazon FSx 檔案系統連接到您的 Amazon EC2 執行個體：

- 適用於 NetApp ONTAP 的 NetApp Amazon FSx 在 AWS 雲端提供全受管共用儲存，並具備 ONTAP 熱門的資料存取和管理功能。
- Amazon FSx for OpenZFS 可提供由常用 OpenZFS 檔案系統支援的全受管共用儲存體，此儲存體具備成本效益。

### Note

- 此功能僅在新的啟動執行個體精靈中可用。如需詳細資訊，請參閱 [使用新的啟動執行個體精靈啟動執行個體](#)
- Amazon FSx for Windows File Server 和 Amazon FSx for Lustre 檔案系統無法在啟動時掛載。啟動後，必須手動掛載這些檔案系統。

您可以選擇掛載先前建立的現有檔案系統，也可以建立新的檔案系統，以便在啟動時掛載到執行個體。

### 主題

- [安全群組和使用者資料指令碼](#)
- [在啟動時掛載 Amazon FSx 檔案系統](#)

## 安全群組和使用者資料指令碼

當您使用啟動執行個體精靈將 Amazon FSx 檔案系統掛載到執行個體時，您可以選擇是否自動建立並連接啟用存取檔案系統所需的安全群組，以及是否自動包含掛載檔案系統所需的使用者資料指令碼並將其設定為可供使用。

### 主題

- [安全群組](#)
- [使用者資料指令碼](#)

## 安全群組

如果您選擇自動建立啟用存取檔案系統所需的安全群組，啟動執行個體精靈將建立並連接兩個安全群組：一個安全群組連接到執行個體，另一個連接到檔案系統。有關安全群組要求的詳細資訊，請參閱《[適用於 Amazon VPC ONTAP 檔案系統存取控制的 FSx](#)》和《[適用於 Amazon VPC 的 OpenZFS 檔案系統存取控制的 FSx](#)》。

我們會將標籤 Name=instance-sg-1 新增至建立並附加至執行個體的安全群組。標籤中的值會在每次啟動執行個體精靈為 Amazon FSx 檔案系統建立安全群組時自動遞增。

安全群組包含下列輸出規則，但不包含輸入規則。

## 傳出規則

通訊協定類型	連接埠號碼	目的地
UDP	111	#####
UDP	二零零一至二零零三年	#####
UDP	4049	#####
UDP	2049	#####
UDP	635	#####
UDP	4045 - 4046	#####
TCP	4049	#####
TCP	635	#####
TCP	2049	#####
TCP	111	#####
TCP	4045 - 4046	#####
TCP	二零零一至二零零三年	#####
全部	全部	#####

建立並附加至檔案系統的安全群組會以 Name=fsx-sg-1 標記。標籤中的值會在每次啟動執行個體精靈為 Amazon FSx 檔案系統建立安全群組時自動遞增。

安全群組包括以下規則。

## 傳入規則

通訊協定類型	連接埠號碼	來源
UDP	2049	#####
UDP	二零零一至二零零三年	#####
UDP	4049	#####
UDP	111	#####
UDP	635	#####
UDP	4045 - 4046	#####
TCP	4045 - 4046	#####
TCP	635	#####
TCP	2049	#####
TCP	4049	#####
TCP	二零零一至二零零三年	#####
TCP	111	#####

## 傳出規則

通訊協定類型	連接埠號碼	目的地
全部	全部	0.0.0.0/0

## 使用者資料指令碼

如果您選擇自動連接使用者資料指令碼，啟動執行個體精靈會將以下使用者資料新增到執行個體中。此指令碼將安裝必要的套件、掛載檔案系統並更新您的執行個體設定，以便在執行個體重新啟動時，自動重新掛載檔案系統。

```
#cloud-config
package_update: true
package_upgrade: true
runcmd:
- yum install -y nfs-utils
- apt-get -y install nfs-common
- svm_id_1=svm_id
- file_system_id_1=file_system_id
- vol_path_1=/vol1
- fsx_mount_point_1=/mnt/fsx/fs1
- mkdir -p "${fsx_mount_point_1}"
- if [-z "$svm_id_1"]; then printf "\n${file_system_id_1}.fsx.eu-
north-1.amazonaws.com:${vol_path_1} ${fsx_mount_point_1} nfs4
nfsvers=4.1,rsz=1048576,wsz=1048576,hard,timeo=600,retrans=2,noresvport,_netdev
0 0\n" >> /etc/fstab; else printf "\n${svm_id_1}.${file_system_id_1}.fsx.eu-
north-1.amazonaws.com:${vol_path_1} ${fsx_mount_point_1} nfs4
nfsvers=4.1,rsz=1048576,wsz=1048576,hard,timeo=600,retrans=2,noresvport,_netdev 0
0\n" >> /etc/fstab; fi
- retryCnt=15; waitTime=30; while true; do mount -a -t nfs4 defaults; if [$? = 0] ||
[$retryCnt -lt 1]; then echo File system mounted successfully; break; fi; echo File
system not available, retrying to mount.; ((retryCnt--)); sleep $waitTime; done;
```


## 在啟動時掛載 Amazon FSx 檔案系統

### 在啟動時掛載新的或現有的 Amazon FSx 檔案系統

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)，接著選擇 Launch instance (啟動執行個體) 來開啟啟動執行個體精靈。
3. 在 Application and OS Images (應用程式和 OS 映像) 區段中，選取要使用的 AMI。
4. 在 Instance type (執行個體類型) 區段中，選取執行個體類型。
5. 在 Key pair (金鑰對) 區段中，選取現有的金鑰對或建立新金鑰對。
6. 在 Network settings (網路設定) 區段中，執行下列動作：

- a. 選擇編輯。
- b. 如果您想要掛載現有的檔案系統，請針對 Subnet (子網) 選擇檔案系統的偏好子網。我們推薦您在與檔案系統之偏好子網路相同的可用區域中啟動執行個體，以便最佳化效能。


如果您想要建立新的檔案系統以掛載到執行個體，請針對 Subnet (子網) 選擇要在其中啟動執行個體子網。

 Important

您必須選取一個子網才能在新的啟動執行個體精靈中啟用 Amazon FSx 功能。如果沒有選取子網，則無法掛載現有檔案系統或建立新的檔案系統。

7. 在 Storage (儲存) 區段中，執行下列動作：

- a. 根據需要設定磁碟區。
- b. 展開 File systems (檔案系統) 區段，然後選取 FSx。
- c. 選擇 Add shared file system (新增共用檔案系統)。
- d. 針對 File system (檔案系統)，選取要掛載的檔案系統。

 Note

此清單會顯示所選區域中您帳戶中適用於 NetApp ONTAP 的所有 Amazon FSx 和適用於 OpenZFS 檔案系統的 Amazon FSx。

- e. 若要自動建立並連接啟用存取檔案系統所需的安全群組，請選取 Automatically create and attach security groups (自動建立並連接安全群組)。如果您希望手動建立安全群組，請清除核取方塊。如需詳細資訊，請參閱 [安全群組](#)。
  - f. 若要自動連接掛載檔案系統所需的使用者資料指令碼，請選取 Automatically mount shared file system by attaching required user data script (透過連接所需的使用者資料指令碼自動掛載共享檔案系統)。如果您希望手動提供使用者資料指令碼，請清除核取方塊。如需詳細資訊，請參閱 [使用者資料指令碼](#)。
8. 在 Advanced (進階) 區段中，根據需要編輯其他執行個體設定。
  9. 選擇啟動。

## 搭配 Amazon EC2 使用亞馬遜檔案快取

Amazon 檔案快取是一種全受管的高速快取 AWS，可用來處理檔案資料，無論資料存放在何處。Amazon 檔案快取可做為暫存、高效能的儲存位置，儲存在現場部署檔案系統、AWS 檔案系統和 Amazon Simple Storage Service (Amazon S3) 存貯體中的資料。您可以使用此功能，AWS 透過統一檢視和高速 (低於一毫秒的延遲和高輸送量)，將分散的資料集提供給以檔案為基礎的應用程式。如需詳細資訊，請參閱[什麼是 Amazon 檔案快取？](#)。

您可以使用開放原始碼 Lustre 用戶端，從 Amazon EC2 執行個體存取快取。Amazon EC2 執行個體可從相同 Amazon 虛擬私有雲端 (Amazon VPC) 內的其他可用區域存取您的快取，前提是您的聯網允許跨 VPC 中的子網路進行存取。掛載快取之後，您可以像使用本機檔案系統一樣使用它的檔案和目錄。

若要開始使用，請參閱[開始使用 Amazon 檔案快取](#)。

## 執行個體磁碟區限制

可連接到執行個體的 Amazon EBS 磁碟區數目上限，取決於執行個體類型和執行個體大小。考慮要連接至執行個體的磁碟區數目時，您應考量是否需要提高 I/O 頻寬或提高儲存體容量。

### 頻寬和容量

如需一致且可預測的頻寬使用案例，請使用 Amazon EBS 優化執行個體搭配一般用途 SSD 磁碟區或佈建 IOPS SSD 磁碟區。如需獲得最大效能，請比對您為磁碟區所佈建的 IOPS 與適用於執行個體類型的頻寬。

針對 RAID 組態，您可能會發現大於 8 個磁碟區的陣列會因提高 I/O 額外負荷而降低效能。測試個別應用程式效能，並視需要調整。

### 主題

- [建置於 Nitro System 上的執行個體磁碟區限制](#)
- [Xen 型執行個體的磁碟區限制](#)

## 建置於 Nitro System 上的執行個體磁碟區限制

### 主題

- [專用 Amazon EBS 磁碟區限制](#)
- [共用 Amazon EBS 磁碟區限制](#)

## 專用 Amazon EBS 磁碟區限制

下列 Nitro 執行個體類型具有專用的 Amazon EBS 磁碟區限制，視執行個體大小而有所不同。此限制不會與其他裝置連接共用。換句話說，您可以連接任意數量的 Amazon EBS 磁碟區，最多不超過磁碟區連接限制，而不論連接的裝置 (例如 NVMe 執行個體儲存體磁碟區和網路介面) 數量為何。

- 一般用途：M7a、M7i、M7-柔性
- 運算最佳化：C7a、C7i
- 記憶體優化：R7a、R7i、R7iz

對於這些支援專用磁碟區限制的執行個體類型，磁碟區限制取決於執行個體大小。下表顯示每個執行個體大小的限制。

執行個體大小	磁碟區限制
medium   large   xlarge   2xlarge   4xlarge   8xlarge   12xlarge	32
16xlarge	48
24xlarge	64
32xlarge	88
48xlarge	128
metal-16x1   metal-24x 1	39
metal-32x1   metal-48x 1	79

## 共用 Amazon EBS 磁碟區限制

所有其他 Nitro 執行個體類型 (未列於中 [專用 Amazon EBS 磁碟區限制](#)) 都有磁碟區附件限制，該限制可在 Amazon EBS 磁碟區、網路界面和 NVMe 執行個體存放區磁碟區之間共用。您可以連接任意數量的 Amazon EBS 磁碟區 (最多不超過該限制)，減去連接網路界面和 NVMe 執行個體儲存體磁碟區的數



量。請記住，每個執行個體必須至少有一個網路介面，而且會在啟動時自動連接 NVMe 執行個體儲存體磁碟區。

大多數這些執行個體最多支援 28 個附件。例如，如果在 m5.xlarge 執行個體上沒有另外的網路介面，您最多可以連接 27 個 EBS 磁碟區 (28 個磁碟區限制 - 1 個網路介面)。如果在 m5.xlarge 執行個體上有另外兩個網路介面，您最多可以連接 25 個 EBS 磁碟區 (28 個磁碟區限制 - 3 個網路介面)。同樣地，如果在 m5d.xlarge 執行個體 (具有 1 個 NVMe 執行個體儲存體磁碟區) 上有另外兩個網路介面，您最多可以連接 24 個 EBS 磁碟區 (28 個磁碟區限制 - 3 個網路介面 - 1 個 NVMe 執行個體儲存體磁碟區)。

具有共用磁碟區限制的執行個體類型有下列例外情況：

- DL2q 執行個體最多可支援 19 個 EBS 磁碟區。
- 大多數裸機執行個體最多支援 31 個 EBS 磁碟區。
- 記憶體密集型虛擬化執行個體最多可支援 27 個 EBS 磁碟區。
- 記憶體密集型裸機執行個體最多支援 19 個 EBS 磁碟區。
- inf1.xlarge 和 inf1.2xlarge 執行個體最多支援 26 個 EBS 磁碟區。
- inf1.6xlarge 執行個體最多可支援 23 個 EBS 磁碟區。
- mac1.metal 執行個體最多可支援 16 個 EBS 磁碟區。
- mac2.metalmac2-m2.metal、和 mac2-m2pro.metal 執行個體最多支援 10 個 EBS 磁碟區。
- inf1.24xlarge 執行個體最多可支援 11 個 EBS 磁碟區。
- g5.48xlarge 執行個體最多可支援 9 個 EBS 磁碟區。
- d3.8xlarge 和 d3en.12xlarge 執行個體最多支援 3 個 EBS 磁碟區。
- 對於加速運算執行個體，連接的加速器會計入共用磁碟區限制。例如，對於 p4d.24xlarge 執行個體 (其具有的共用磁碟區限制為 28 個、8 個 GPU 和 8 個 NVMe 執行個體儲存體磁碟區)，您最多可以連接 11 個 Amazon EBS 磁碟區 (28 個磁碟區限制 - 1 個網路介面 - 8 個 GPU - 8 個 NVMe 執行個體儲存體磁碟區)。

## Xen 型執行個體的磁碟區限制

### Linux 執行個體

將 40 個以上的磁碟區連接到 Xen 型 Linux 執行個體，可能會導致開機失敗。此數量包括根磁碟區，以及任何已連接的執行個體儲存體磁碟區和 Amazon EBS 磁碟區。

如果具有大量磁碟區的執行個體發生開機問題，請停止執行個體、分離啟動程序非必要的磁碟區、啟動執行個體，然後在執行個體執行之後重新連接磁碟區。

#### Important

Xen 型 Linux 執行個體最佳情況可支援連接超過 40 個磁碟區，但不予保證。

## Windows 執行個體

下表根據使用的驅動程式，顯示 Xen 型 Windows 執行個體的磁碟區限制。這些數目包括根磁碟區，以及任何已連接的執行個體儲存體磁碟區和 Amazon EBS 磁碟區。

#### Important

Xen 型 Windows 執行個體最佳情況可支援連接超過以下數量的磁碟區，但不予保證。

驅動程式	磁碟區限制
AWS 光伏	26
Citrix PV	26
Red Hat PV	17

建議您不要將超過 26 個磁碟區連接至具有 AWS PV 或 Citrix PV 驅動程式的 Xen 型 Windows 執行個體，因為這可能會造成效能問題。若要判定執行個體使用哪些 PV 驅動程式，或將 Windows 執行個體從 Red Hat 升級至 Citrix PV 驅動程式，請參閱[the section called “升級 PV 驅動程式”](#)。

如需裝置名稱如何與磁碟區相關的詳細資訊，請參閱[將磁碟映射至 Windows 執行個體上的磁碟區](#)。

## Amazon EC2 執行個體根磁碟區

當您啟動執行個體時，系統會為執行個體建立根磁碟區。根磁碟區內含用來啟動執行個體的映像。每個執行個體都有一個根磁碟區。您可以在啟動期間或之後將儲存磁碟區新增至執行個體。

系統會為根磁碟區保留特定的裝置名稱。如需詳細資訊，請參閱[Amazon EC2 實例上的設備名稱](#)。

## 目錄

- [根磁碟區類型](#)
- [根磁碟區類型選擇一個 Linux AMI](#)
- [判斷 Linux 執行個體的根裝置類型](#)
- [將根磁碟區變更為持續存在](#)
- [變更根磁碟區的初始大小](#)
- [替換 EC2 實例根卷](#)

## 根磁碟區類型

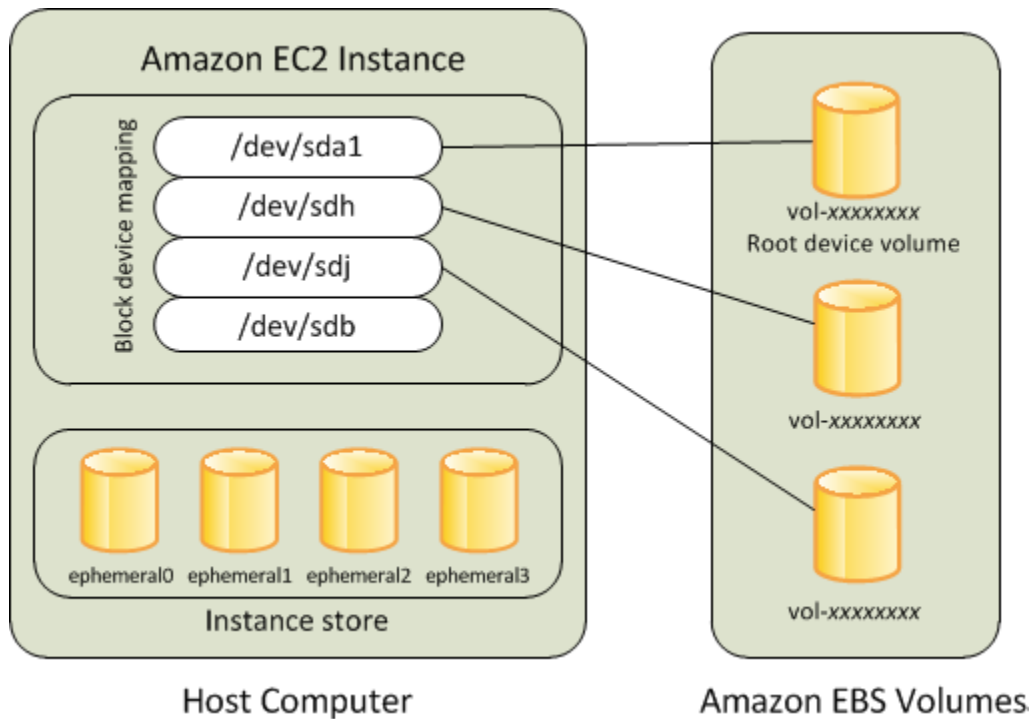
您用來啟動執行個體的 AMI 會決定根磁碟區的類型。您可以從 Amazon EBS-backed AMI (Linux 和 Windows 執行個體) 或執行個體 instance store-backed AMI (僅限 Linux 執行個體) 啟動執行個體。您可以對每種類型的 AMI 執行的操作之間存在顯著差異。如需這些差異的詳細資訊，請參閱 [根設備儲存](#)。

建議您使用 Amazon EBS 支援的 AMI，因這些執行個體啟動速度較快且採用持久性儲存。

### Amazon EBS 後端執行個體

使用 Amazon EBS 做為根磁碟區的執行個體會自動連接 Amazon EBS 磁碟區。啟動 Amazon EBS 後端執行個體時，我們會為您使用的 AMI 所參考的各個 Amazon EBS 快照，建立 Amazon EBS 磁碟區。您可根據執行個體類型選擇使用其他 Amazon EBS 磁碟區或執行個體存放區磁碟區。

停止後的 Amazon EBS 後端執行個體可在稍後重新啟動，不會影響連接磁碟區內存放的資料。Amazon EBS 支援的執行個體處於停止狀態時，您仍可執行多種執行個體和磁碟區的相關任務。例如，您可修改執行個體的屬性、變更其大小、更新其正使用的核心，或者您可將根磁碟區連接至另一個運作中執行個體以進行除錯或其他用途。如需詳細資訊，請參閱 [Amazon EBS 磁碟區](#)。



## 限制

您無法使用 `st1` 或 `sc1` EBS 磁碟區做為根磁碟區。

## 執行個體失敗

若 Amazon EBS 後端執行個體故障，您可採取下列方法之一來還原工作階段：

- 停止然後再次啟動 (請先嘗試此方法)。
- 自動拍攝所有相關磁碟區的快照，並建立新的 AMI。如需詳細資訊，請參閱 [創建一個 Amazon EBS-backed AMI](#)。
- 採取下列步驟將磁碟區連接至新的執行個體：
  1. 建立根磁碟區的快照。
  2. 使用快照註冊新的 AMI。
  3. 從新的 AMI 啟動新的執行個體。
  4. 將剩餘的 Amazon EBS 磁碟區自舊的執行個體分離。
  5. 將 Amazon EBS 磁碟區重新連接至新的執行個體。

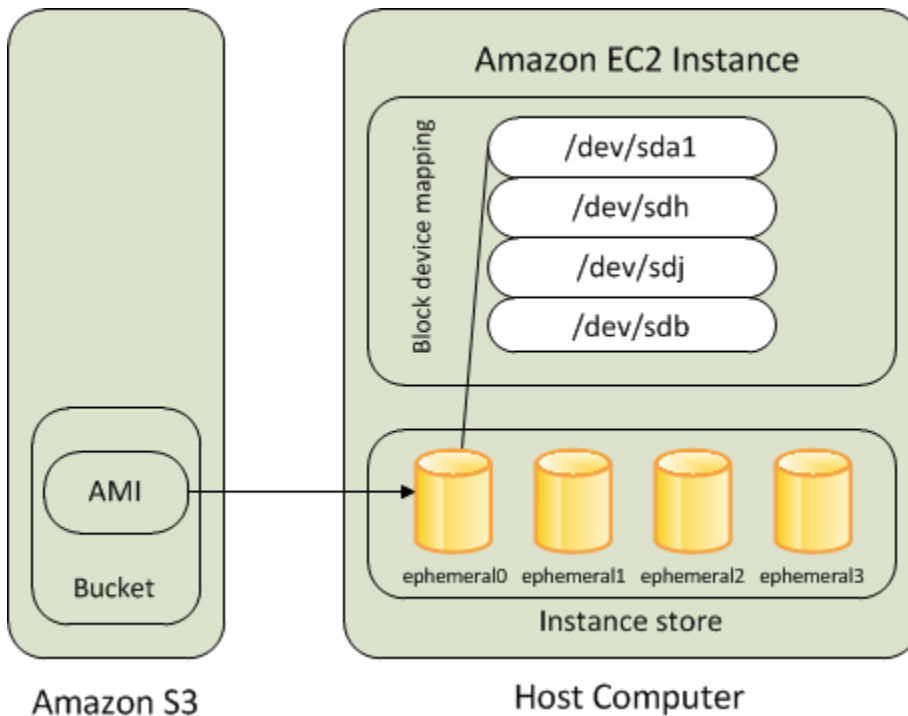
## 執行個體商店支援執行個體 (僅限 Linux 執行個體)

### Note

Windows 執行個體不支援執行個體儲存支援的根磁碟區。

將執行個體儲存體用於根磁碟區的執行個體，自動具備一個或多個可用的執行個體儲存體磁碟區，其中一個磁碟區會當成根磁碟區使用。執行個體啟動時，用於啟動執行個體的映像會複製到根磁碟區。請注意，您可根據執行個體類型選擇使用其他執行個體存放區磁碟區。

只要執行個體持續運作，執行個體存放區磁碟區上的資料就會永久保存，但若執行個體被終止 (執行個體後端執行個體不支援 Stop (停止) 動作) 或故障 (如底層磁碟區出現問題)，這些資料就會被刪除。如需詳細資訊，請參閱 [Amazon EC2 執行個體存放區](#)。



### 需求

只有下列執行個體類型支援執行個體儲存磁碟區做為根磁碟區：C3、D2、I2、M3 和 R3。

### 執行個體失敗

執行個體後端執行個體故障或終止後，將無法還原。若您計畫使用 Amazon EC2 執行個體後端執行個體，我們強烈建議您將執行個體存放區上的資料分散至多個可用區域。您也應將執行個體存放區磁碟區上的重要資料，定期備份為持久性儲存。

## 根磁碟區類型選擇一個 Linux AMI

### Note

所有視窗 AMI 都是 EBS 支援的。

您啟動執行個體時指定的 AMI，會判定您的執行個體所擁有的根設備磁碟區類型。您可以使用下列其中一種方法，依根裝置類型檢視 AMI。

### Console

使用主控台選擇 Amazon EBS 後端 AMI

1. 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 AMIs (AMI)。
3. 在篩選條件清單中，選取映像類型 (如 Public images (公有映像))。在搜尋列中，選擇「平台」以選取作業系統 (例如 Amazon Linux)，然後選擇根裝置類型以選取根磁碟區類型 (ebs 或執行個體存放區)。
4. (選用) 如要取得其他資訊以協助您做出選擇，請選擇偏好設定圖示，切換要顯示的欄位，然後選擇確認。
5. 選擇 AMI 並寫下其 AMI ID。

### AWS CLI

使用命令列驗證 AMI 根設備磁碟區的類型

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [describe-images](#) (AWS CLI)
- [Get-EC2Image](#) (AWS Tools for Windows PowerShell)

## 判斷 Linux 執行個體的根裝置類型

### Note

所有視窗執行個體都是 EBS 支援的。

您可以使用下列其中一種方法來檢視 Linux 執行個體的根裝置類型。

## Console

使用主控台判定執行個體的根設備類型

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取該執行個體。
3. 在 Storage (儲存體) 標籤的 Root device details (根裝置詳細資訊) 下，檢查 Root device type (根裝置類型) 的值，如下所示：
  - 若該值為 EBS，則此為 Amazon EBS 後端執行個體。
  - 若該值為 INSTANCE-STORE，則此為執行個體後端執行個體。

## AWS CLI

使用命令列判定執行個體的根設備類型

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [describe-instances](#) (AWS CLI)
- [Get-EC2Instance](#) (AWS Tools for Windows PowerShell)

## 將根磁碟區變更為持續存在

根據預設，執行個體終止時，以 Amazon EBS 為後端的 AMI 根磁碟區將被刪除。您可以變更預設行為，以確保在執行個體終止後磁碟區持續存在。若要變更此預設行為，請使用區塊型設備映射將 DeleteOnTermination 屬性設定為 false。

### 任務

- [將根磁碟區設定為執行個體啟動期間持續存在](#)
- [將根磁碟區設定為若為現有執行個體則持續存在](#)
- [確認根磁碟區已設定為持續存在](#)

## 將根磁碟區設定為執行個體啟動期間持續存在

您可以將根磁碟區設定為使用 Amazon EC2 主控台或命令列工具啟動執行個體時持續存在。

## Console

將根磁碟區設定為您使用主控台啟動執行個體時持續存在

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)，然後選擇 Launch instances (啟動執行個體)。
3. 依序選擇 Amazon Machine Image (AMI)、執行個體類型和金鑰對，然後設定網路設定。
4. 針對設定儲存，選擇進階。
5. 展開根磁碟區。
6. 針對在終止時刪除，選擇否。
7. 當您完成對執行個體的設定後，選擇啟動執行個體。

## AWS CLI

若要將根磁碟區設定為在使用 AWS CLI

使用 [run-instances](#) 命令，並包括將 DeleteOnTermination 屬性設定為 false 的區塊型設備映射。

```
aws ec2 run-instances --block-device-mappings file://mapping.json ...other
parameters...
```

在 mapping.json 中指定下列內容。

```
[
 {
 "DeviceName": "/dev/sda1",
 "Ebs": {
 "DeleteOnTermination": false
 }
 }
]
```

## Tools for Windows PowerShell

若要將根磁碟區設定為在使用 Windows 的工具啟動執行個體時持續存在 PowerShell

使用指 [New-EC2Instance](#) 令並包括將 DeleteOnTermination 屬性設定為的圖塊設備對映 false。



```
C:\> $ebs = New-Object Amazon.EC2.Model.EbsBlockDevice
C:\> $ebs.DeleteOnTermination = $false
C:\> $bdm = New-Object Amazon.EC2.Model.BlockDeviceMapping
C:\> $bdm.DeviceName = "dev/xvda"
C:\> $bdm.Ebs = $ebs
C:\> New-EC2Instance -ImageId ami-0abcdef1234567890 -BlockDeviceMapping
$bdm ...other parameters...
```

## 將根磁碟區設定為若為現有執行個體則持續存在

您只能使用命令列工具，將根磁碟區設定為若執行個體執行中則持續存在。

### AWS CLI

若要將根磁碟區設定為保留現有執行個體，請使用 AWS CLI

使用 [modify-instance-attribute](#) 命令，搭配將 DeleteOnTermination 屬性設定為 false 的區塊型設備映射。

```
aws ec2 modify-instance-attribute --instance-id i-1234567890abcdef0 --block-device-mappings file://mapping.json
```

在 mapping.json 中指定下列內容。

```
[
 {
 "DeviceName": "/dev/xvda",
 "Ebs": {
 "DeleteOnTermination": false
 }
 }
]
```

### Tools for Windows PowerShell

使用 AWS Tools for Windows PowerShell將根磁碟區設定為若為現有執行個體則持續存在

將指[Edit-EC2InstanceAttribute](#)令與將DeleteOnTermination屬性設定為的圖塊設備對映搭配使用false。

```
C:\> $ebs = New-Object Amazon.EC2.Model.EbsInstanceBlockDeviceSpecification
```

```
C:\> $ebs.DeleteOnTermination = $false
C:\> $bdm = New-Object Amazon.EC2.Model.InstanceBlockDeviceMappingSpecification
C:\> $bdm.DeviceName = "/dev/xvda"
C:\> $bdm.Ebs = $ebs
C:\> Edit-EC2InstanceAttribute -InstanceId i-1234567890abcdef0 -BlockDeviceMapping
$bdm
```

## 確認根磁碟區已設定為持續存在

您可以使用 Amazon EC2 主控台或命令列工具，來確認根磁碟區已設定為持續存在。

### Console

使用 Amazon EC2 主控台來確認根磁碟區已設定為持續存在

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取執行個體。
3. 在 Storage (儲存體) 標籤的 Block devices (區塊型裝置) 下，找出根磁碟區的項目。如果 Delete on termination (終止時刪除) 為 No，則磁碟區會設定為持續存在。

### AWS CLI

若要確認根磁碟區已設定為持續使用 AWS CLI

使用 [describe-instance](#) 命令，並驗證 DeleteOnTermination 回應元素中的 BlockDeviceMappings 屬性是否已設定為 false。

```
aws ec2 describe-instances --instance-id i-1234567890abcdef0
```

```
...
 "BlockDeviceMappings": [
 {
 "DeviceName": "/dev/sda1",
 "Ebs": {
 "Status": "attached",
 "DeleteOnTermination": false,
 "VolumeId": "vol-1234567890abcdef0",
 "AttachTime": "2013-07-19T02:42:39.000Z"
 }
 }
]
```

```
}
...
```

## Tools for Windows PowerShell

若要確認根磁碟區已設定為持續使用 AWS Tools for Windows PowerShell

使用 [Get-EC2Instance](#) 並確認 `BlockDeviceMappings` 回應元素中的 `DeleteOnTermination` 屬性是否設定為 `false`。

```
C:\> (Get-EC2Instance -InstanceId i-
i-1234567890abcdef0).Instances.BlockDeviceMappings.Ebs
```

## 變更根磁碟區的初始大小

根據預設，根磁碟區的大小取決於快照的大小。您可以使用執行個體的區塊型裝置映射來增加根磁碟區的初始大小，如下所示。

1. 確定 AMI 中指定根磁碟區的裝置名稱，如 [檢視 AMI 區塊型儲存裝置對應的 EBS 磁碟區](#) 中所述。
2. 確認 AMI 區塊裝置對應中指定的快照大小。
3. 使用執行個體區塊型裝置映射覆寫根磁碟區的大小，如 [啟動執行個體時更新區塊型儲存裝置對應](#) 中所述，指定大於快照大小的磁碟區大小。

例如，執行個體區塊型裝置映射的下列項目會將 `/dev/xvda` 根磁碟區的大小增加為 100 GiB。您可以在執行個體區塊型裝置映射中省略快照 ID，因為快照 ID 已經在 AMI 區塊型裝置映射中指定。

```
{
 "DeviceName": "/dev/xvda",
 "Ebs": {
 "VolumeSize": 100
 }
}
```

如需詳細資訊，請參閱 [區塊型儲存設備對應](#)。

## 替換 EC2 實例根卷

Amazon EC2 可讓您取代執行中執行個體的根 Amazon EBS 磁碟區，同時保留下列項目：

- 儲存在執行個體存放區磁碟區上的資料 - 在還原根磁碟區後，執行個體存放區磁碟區仍會保持連接至執行個體。
- 資料 (非根) Amazon EBS 磁碟區上儲存的資料 - 在還原根磁碟區後，非根 Amazon EBS 磁碟區會保持連接到執行個體。
- 網路組態 - 所有網路介面都會保持連接至執行個體，並保留其 IP 地址、識別符和附件 ID。當執行個體可用時，會清除所有擱置中的網路流量。此外，執行個體會保留在相同的實體主機上，因此會保留其公有和私有 IP 地址和 DNS 名稱。
- IAM 政策 - IAM 描述檔和與執行個體相關聯的政策 (例如索引標籤政策) 會保留並強制執行。

## 主題

- [其運作方式？](#)
- [取代根磁碟區](#)
- [檢視根磁碟區取代任務](#)

## 其運作方式？

當您取代執行個體的根磁碟區時，會以下列其中一種方式還原新的 (取代) 根磁碟區：

- 至初始啟動狀態 - 磁碟區會在執行個體啟動時還原至其初始狀態。如需詳細資訊，請參閱 [將根磁碟區還原至其啟動狀態](#)。
- 從快照 (來自與目前根磁碟區相同的譜系) - 這可讓您修正問題，例如根磁碟區損毀或訪客作業系統網路組態錯誤。如需詳細資訊，請參閱 [使用快照取代根磁碟區](#)。
- 從 AMI (具有與執行個體相同的索引鍵屬性) - 這可讓您執行作業系統和應用程式修補或升級。如需詳細資訊，請參閱 [使用 AMI 取代根磁碟區](#)。

原始根磁碟區會從執行個體中分離，新根磁碟區會連接至其位置中的執行個體。執行個體的區塊型裝置映射已更新，以反映取代根磁碟區的 ID。您可以選擇是否要在根磁碟區取代程序完成後保留原始根磁碟區。如果您選擇在取代程序完成之後刪除原始根磁碟區，則原始根磁碟區會自動刪除並變得無法復原。如果您選擇在程序完成後保留原始根磁碟區，則該磁碟區仍會在您的帳戶中佈建；當您不再需要它時必須手動刪除。

如果根磁碟區取代任務失敗，則執行個體會重新啟動，而原始根磁碟區會保持連接到執行個體。

## 根磁碟區取代的考量事項

- 執行個體必須處於 `running` 狀態。

- 執行個體會在過程中自動重新啟動。重新開機期間會清除記憶體 (RAM) 的內容。不需要手動重新啟動。
- 如果根磁碟區是執行個體存放區磁碟區，則無法取代根磁碟區。僅支援具有 Amazon EBS 根磁碟區的執行個體。
- 您可取代所有虛擬化執行個體類型和 EC2 Mac 裸機執行個體的根磁碟區。不支援所有其他裸機執行個體類型。
- 您只能使用屬於相同關係的快照作為執行個體以前的根磁碟區。
- 如果帳戶在目前區域中已啟用 Amazon EBS 預設加密，無論指定快照的加密狀態或指定 AMI 的根磁碟區為何，都會一律加密根磁碟區取代任務建立的取代根磁碟區。
- 下表摘要提供了可能的加密結果的摘要。

	原始根磁碟區	指定的快照或 AMI	預設加密	取代根磁碟區	用於取代根磁碟區的加密金鑰
將取代根磁碟區還原至初始啟動狀態	Encrypted	不適用	不考慮	Encrypted	與原始根磁碟區相同的 KMS 金鑰
	未加密	不適用	已停用	未加密	不適用
	未加密	不適用	已啟用	Encrypted	帳戶的 Amazon EBS 加密的預設 KMS 金鑰
從快照或 AMI 還原取代根磁碟區	Encrypted	未加密	不考慮	Encrypted	與原始根磁碟區相同的 KMS 金鑰
	Encrypted	Encrypted	不考慮	Encrypted	與原始根磁碟區相同的 KMS 金鑰
	未加密	未加密	已停用	未加密	不適用

	原始根磁碟區	指定的快照或 AMI	預設加密	取代根磁碟區	用於取代根磁碟區的加密金鑰
	未加密	未加密	已啟用	Encrypted	帳戶的 Amazon EBS 加密的預設 KMS 金鑰
	未加密	Encrypted	不考慮	Encrypted	如果 AMI 或快照由帳戶擁有，則會使用 AMI 或快照的 KMS 金鑰加密取代磁碟區。如果與帳戶共用 AMI 或快照，則會使用該帳戶的 Amazon EBS 加密的預設 KMS 金鑰來加密取代磁碟區。

## 主題

- [將根磁碟區還原至其啟動狀態](#)
- [使用快照取代根磁碟區](#)
- [使用 AMI 取代根磁碟區](#)

### 將根磁碟區還原至其啟動狀態

您可以執行根磁碟區取代，這會將執行個體的根磁碟區取代為已還原至原始根磁碟區啟動狀態的取代根磁碟區。取代磁碟區會在執行個體啟動期間從用來建立原始磁碟區的快照中自動還原。

取代根磁碟區會獲得與原始根磁碟區相同的類型、大小和「在終止時刪除」屬性。

### 使用快照取代根磁碟區

您可以執行根磁碟區取代，這會將執行個體的根磁碟區取代為已還原至特定快照的取代磁碟區。這可讓您將執行個體的根磁碟區還原至先前從該根磁碟區建立的特定快照。

取代根磁碟區會獲得與原始根磁碟區相同的類型、大小和「在終止時刪除」屬性。

### 使用快照時的考量事項

- 您只能使用屬於相同關係的快照作為執行個體的目前根磁碟區。
- 您無法使用從根磁碟區擷取的快照中建立的快照複本。
- 在成功取代根磁碟區之後，從原始根磁碟區中取得的快照仍然可用來取代新的 (取代) 根磁碟區。

### 使用 AMI 取代根磁碟區

您可以使用您擁有的 AMI 或與您共用的 AMI 來執行根磁碟區取代。AMI 必須具有與執行個體相同的產品代碼、帳單資訊、架構類型和虛擬化類型。

如果執行個體已針對 ENA 或 sriov-net 啟用，則您必須使用支援這些功能的 AMI。如果執行個體未針對 ENA 或 sriov-net 啟用，則您可以選取不支援這些功能的 AMI，或者如果您選取支援 ENA 或 sriov-net 的 AMI，則可以自動新增支援。

如果執行個體已啟用 NitRoTPM，則您必須使用已啟用 NitRoTPM 的 AMI。如果未設定執行個體，則不會啟用 NitRoTPM 支援，無論您選取的 AMI 為何。

只要執行個體支援 AMI 的開機模式，您就可以選取開機模式與執行個體不同的 AMI。如果執行個體不支援該開機模式，則請求會失敗。如果執行個體支援該開機模式，則新的開機模式會傳播至執行個體，並相應地更新其 UEFI 資料。如果您已手動修改開機順序，或已新增私有 UEFI 安全開機金鑰來載入私有核心模組，則變更會在根磁碟區取代期間遺失。

取代根磁碟區獲得與原始根磁碟區相同的磁碟區類型和終止時刪除屬性，且它會獲得根磁碟區的 AMI 區塊型裝置映射的大小。

#### Note

AMI 根磁碟區區塊型裝置映射的大小必須等於或大於原始根磁碟區的大小。如果 AMI 根磁碟區區塊型裝置映射的大小小於原始根磁碟區的大小，則請求失敗。

根磁碟區置換工作完成後，當您使用主控台 AWS CLI 或 AWS SDK 描述執行個體時，會反映下列新增和更新的資訊：

- 新的 AMI ID
- 根磁碟區的新磁碟區 ID
- 已更新開機模式組態 (如果由 AMI 變更)
- 已更新 NitroTPM 組態 (如果由 AMI 啟用)
- 已更新 ENA 組態 (如果由 AMI 啟用)
- 已更新 sriov-net 組態 (如果由 AMI 啟用)

新的 AMI ID 也會反映在執行個體中繼資料中。

### 使用 AMI 時的考量事項

- 如果您使用具有多個區塊型裝置映射的 AMI，則只會使用 AMI 的根磁碟區。其他 (非根) 磁碟區會被忽略。
- 只有在您擁有使用 AMI 及其關聯根磁碟區快照的權限時，才能使用此功能。您無法搭配 AWS Marketplace AMI 使用此功能。
- 只有在執行個體沒有產品代碼的情況下，才能使用不含產品代碼的 AMI。
- AMI 根磁碟區區塊型裝置映射的大小必須等於或大於原始根磁碟區的大小。如果 AMI 根磁碟區區塊型裝置映射的大小小於原始根磁碟區的大小，則請求失敗。
- 執行個體的執行個體身分文件會自動更新。
- 如果執行個體支援 NitroTPM，則會重設執行個體的 NitroTPM 資料，並產生新的金鑰。

## 取代根磁碟區

取代執行個體的根磁碟區時，會建立根磁碟區取代任務。您可以使用根磁碟區取代任務來監視取代過程的進度和結果。如需詳細資訊，請參閱 [檢視根磁碟區取代任務](#)。

您可以使用下列其中一種方法來取代執行個體的根磁碟區。

### Note

如果您使用 Amazon EC2 主控台，只有新主控台才能使用此功能。



## New console

### 取代根磁碟區

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取要取代根磁碟區的執行個體，然後依序選取 Actions (動作)、Monitor and troubleshoot (監控與疑難排解)、Replace root volume (取代根磁碟區)。

#### Note

如果選定的執行個體不在 `running` 狀態，則會停用取代根磁碟區操作。

4. 在 Replace root volume (取代根磁碟區) 畫面中，執行下列其中一項動作：
  - 若要將取代根磁碟區還原至其初始啟動狀態，請選取 Create replacement task (建立取代任務)，但不選取快照。
  - 若要將取代根磁碟區還原至特定快照，請在 Snapshot (快照) 中選取要使用的快照，然後選取 Create replacement task (建立取代任務)。
  - 若要使用 AMI 來還原取代根磁碟區，請在 AMI 中選取要使用的 AMI，然後選取 Create replacement task (建立取代任務)。
5. 若要在取代任務完成後刪除原始根磁碟區，請選取 Delete replaced root volume (刪除已取代的根磁碟區)。

## AWS CLI

### 若要將取代根磁碟區還原至啟動狀態

使用 [create-replace-root-volume-任務](#) 命令。若為 `--instance-id`，請指定要對其取代根磁碟區的執行個體 ID。省略 `--snapshot-id` 和 `--image-id` 參數。若要在取代原始根磁碟區之後將其刪除，請包含 `--delete-replaced-root-volume` 並指定 `true`。

```
$ aws ec2 create-replace-root-volume-task \
--instance-id i-1234567890abcdef0 \
--delete-replaced-root-volume true
```

### 若要將取代根磁碟區還原至特定快照

使用 [create-replace-root-volume-task](#) 指令。若為 `--instance-id`，請指定要對其取代根磁碟區的執行個體 ID。若為 `--snapshot-id`，請指定要使用的快照 ID。若要在取代原始根磁碟區之後將其刪除，請包含 `--delete-replaced-root-volume` 並指定 `true`。

```
$ aws ec2 create-replace-root-volume-task \
--instance-id i-1234567890abcdef0 \
--snapshot-id snap-9876543210abcdef0 \
--delete-replaced-root-volume true
```

若要使用 AMI 來還原取代根磁碟區

使用 [create-replace-root-volume-任務](#) 命令。若為 `--instance-id`，請指定要對其取代根磁碟區的執行個體 ID。若為 `--image-id`，請指定要使用的 AMI ID。若要在取代原始根磁碟區之後將其刪除，請包含 `--delete-replaced-root-volume` 並指定 `true`。

```
$ aws ec2 create-replace-root-volume-task \
--instance-id i-01234567890abcdef \
--image-id ami-09876543210abcdef \
--delete-replaced-root-volume true
```

## Tools for Windows PowerShell

若要將取代根磁碟區還原至啟動狀態

使用 [New-EC2ReplaceRootVolumeTask](#) 命令。若為 `-InstanceId`，請指定要對其取代根磁碟區的執行個體 ID。省略 `-SnapshotId` 和 `-ImageId` 參數。若要在取代原始根磁碟區之後將其刪除，請包含 `-DeleteReplacedRootVolume` 並指定 `$true`。

```
PS C:\> New-EC2ReplaceRootVolumeTask -InstanceId i-1234567890abcdef0 -
DeleteReplacedRootVolume $true
```

若要將取代根磁碟區還原至特定快照

使用 [New-EC2ReplaceRootVolumeTask](#) 命令。若為 `--InstanceId`，請指定要對其取代根磁碟區的執行個體 ID。若為 `-SnapshotId`，請指定要使用的快照 ID。若要在取代原始根磁碟區之後將其刪除，請包含 `-DeleteReplacedRootVolume` 並指定 `$true`。

```
PS C:\> New-EC2ReplaceRootVolumeTask -InstanceId i-1234567890abcdef0 -
SnapshotId snap-9876543210abcdef0 -DeleteReplacedRootVolume $true
```

若要使用 AMI 來還原取代根磁碟區

使用 [New-EC2ReplaceRootVolumeTask](#) 命令。若為 `-InstanceId`，請指定要對其取代根磁碟區的執行個體 ID。若為 `-ImageId`，請指定要使用的 AMI ID。若要在取代原始根磁碟區之後將其刪除，請包含 `-DeleteReplacedRootVolume` 並指定 `$true`。

```
PS C:\> New-EC2ReplaceRootVolumeTask -InstanceId i-1234567890abcdef0 -
ImageId ami-09876543210abcdef -DeleteReplacedRootVolume $true
```

## 檢視根磁碟區取代任務

取代執行個體的根磁碟區時，會建立根磁碟區取代任務。在此過程中，根磁碟區取代任務將通過以下狀態進行轉換：

- `pending` – 正在建立取代磁碟區。
- `in-progress` – 原始磁碟區正在分離，且取代磁碟區正在連接。
- `succeeded` – 取代磁碟區已成功連接至執行個體，且執行個體可用。
- `failing` — 取代任務處於失敗的程序中。
- `failed` – 取代任務失敗，但仍連接至原始根磁碟區。
- `failing-detached` - 取代任務正在失敗的過程中，執行個體可能不會連接根磁碟區。
- `failed-detached` – 取代任務失敗，且執行個體沒有連接根磁碟區。

您可以使用下列其中一種方法來檢視執行個體的根磁碟區取代任務。

### Note

如果您使用 Amazon EC2 主控台，只有新主控台才能使用此功能。

## Console

### 檢視根磁碟區取代任務

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取要檢視根磁碟區取代任務的執行個體，然後選取 Storage (儲存體) 索引標籤。
4. 在 Storage (儲存體) 索引標籤中，展開 Recent root volume replacement tasks (最近的根磁碟區取代任務)。

## AWS CLI

### 檢視根磁碟區取代任務的狀態

使用 [describe-replace-root-volume-tasks](#) 命令，並指定要檢視的根磁碟區取代任務的 ID。

```
$ aws ec2 describe-replace-root-volume-tasks \
--replace-root-volume-task-ids replacevol-1234567890abcdef0
```

```
{
 "ReplaceRootVolumeTasks": [
 {
 "ReplaceRootVolumeTaskId": "replacevol-1234567890abcdef0",
 "InstanceId": "i-1234567890abcdef0",
 "TaskState": "succeeded",
 "StartTime": "2020-11-06 13:09:54.0",
 "CompleteTime": "2020-11-06 13:10:14.0",
 "SnapshotId": "snap-01234567890abcdef",
 "DeleteReplacedRootVolume": "True"
 }
]
}
```

或者，指定要依執行個體篩選結果的 `instance-id` 篩選條件。

```
$ aws ec2 describe-replace-root-volume-tasks \
--filters Name=instance-id,Values=i-1234567890abcdef0
```

## Tools for Windows PowerShell

### 檢視根磁碟區取代任務的狀態

使用指 [Get-EC2ReplaceRootVolumeTask](#) 令並指定要檢視之根磁碟區取代工作的 ID。

```
PS C:\> Get-EC2ReplaceRootVolumeTask -
ReplaceRootVolumeTaskIds replacevol-1234567890abcdef0
```

或者，指定要依執行個體篩選結果的 `instance-id` 篩選條件。

```
PS C:\> Get-EC2ReplaceRootVolumeTask -Filters @{Name = 'instance-id'; Values =
'i-1234567890abcdef0'} | Format-Table
```

## Amazon EC2 實例上的設備名稱

當您將磁碟區連接到您的執行個體時，您會在其中包含磁碟區的裝置名稱。Amazon EC2 會使用此裝置名稱。執行個體的區塊型儲存設備驅動程式會在掛載磁碟區時指派實際的磁碟區名稱，其指派的名稱會與 Amazon EC2 使用的名稱不同。

您執行個體可以支援的磁碟區數目取決於作業系統。如需詳細資訊，請參閱 [執行個體磁碟區限制](#)。

### 目錄

- [可用裝置名稱](#)
- [裝置名稱考量](#)

## 可用裝置名稱

### Linux 執行個體

Linux 執行個體有兩種可用的虛擬化類型：全虛擬化 (PV) 或硬體虛擬機器 (HVM)。執行個體的虛擬化類型會由用來啟動執行個體的 AMI 決定。所有執行個體類型都支援 HVM AMI。某些前一代執行個體類型支援 PV AMI。請注意您 AMI 的虛擬化類型，因為您可以使用的建議及可用裝置名稱會取決於您執行個體的虛擬化類型。如需詳細資訊，請參閱 [AMI 虛擬化類型](#)。

下表列出您可以在區塊型設備映射中或在連結 EBS 磁碟區時指定的可用裝置名稱。

虛擬化類型	可用性	預留供根磁碟區使用	建議用於 EBS 磁碟區	執行個體存放區磁碟區
全虛擬化	/dev/sd[a-z]	/dev/sda1	/dev/sd[f-p]	/dev/sd[b-e]
	/dev/sd[a-z] [1-15]		/dev/sd[f-p][1-6]	
	/dev/hd[a-z]			
	/dev/hd[a-z] [1-15]			
HVM	/dev/sd[a-z]	根據 AMI 而不同	/dev/sd[f-p] *	/dev/sd[b-e]
	/dev/xvd[a-d][a-z]			

虛擬化類型	可用性	預留供根磁碟區使用	建議用於 EBS 磁碟區	執行個體存放區磁碟區
	/dev/xvd[e-z]	/dev/sda1 或 /dev/xvda		/dev/sd[b-h] (h1.16xlarge)  /dev/sd[b-y] (d2.8xlarge)  /dev/sd[b-i] (i2.8xlarge)  **

\* 您在區塊型設備映射中針對 NVMe EBS 磁碟區指定的裝置名稱會使用 NVMe 裝置名稱 (/dev/nvme[0-26]n1) 重新命名。區塊型儲存設備驅動程式指派 NVMe 設備名稱的順序，可能會與您在區塊型設備映射中為磁碟區指定的順序不同。

\*\* NVMe 執行個體存放區磁碟區會自動列舉並指派 NVMe 裝置名稱。

## Windows 執行個體

Windows AMI 使用下列其中一組驅動程式來允許存取虛擬化硬體：AWS PV、Citrix 光伏和 RedHat 光伏。如需詳細資訊，請參閱 [the section called “窗戶光伏驅動”](#)。

下表列出您可以在區塊型設備映射中或在連結 EBS 磁碟區時指定的可用裝置名稱。

驅動器類型	可用性	預留供根磁碟區使用	建議用於 EBS 磁碟區	執行個體存放區磁碟區
AWS 光伏、思杰 光伏	xvd[b-z]	/dev/sda1	xvd[f-z] *	xvdc[a-x]
	xvd[b-c][a-z]			xvd[a-e]
	/dev/sda1			**
	/dev/sd[b-e]			
Red Hat PV	xvd[a-z]	/dev/sda1	xvd[f-p]	xvdc[a-x]
	xvd[b-c][a-z]			xvd[a-e]

驅動器類型	可用性	預留供根磁碟區使用	建議用於 EBS 磁碟區	執行個體存放區磁碟區
	/dev/sda1			
	/dev/sd[b-e]			

\* 對於思杰光伏和紅帽光伏，如果您使用名稱對映 EBS 磁碟區 xvda，Windows 將無法識別該磁碟區 (AWS PV 或 AWS NVMe 可見磁碟區)。

\*\* NVMe 執行個體存放區磁碟區會自動列舉並指派 Windows 磁碟機代號。

如需執行個體存放區磁碟區的詳細資訊，請參閱 [Amazon EC2 執行個體存放區](#)。如需 NVMe EBS 磁碟區 (以硝基為基礎的執行個體) 的詳細資訊，包括如何識別 EBS 裝置，請參閱 [Amazon EBS 使用者指南](#) 中的 [Amazon EBS 和 NVMe](#)。

## 裝置名稱考量

當您選取裝置名稱時，請注意下列項目：

- 雖然您可以使用用來連接執行個體存放區磁碟區的裝置名稱來連接您的 EBS 磁碟區，但我們強烈建議您不要這麼做，因為該行為可能會無法預期。
- 執行個體的 NVMe 執行個體存放區磁碟區數目取決於執行個體的大小。NVMe 執行個體儲存磁碟區會自動列舉並指派 NVMe 裝置名稱 (Linux 執行個體) 或 Windows 磁碟機代號 (Windows 執行個體)。
- (Windows 執行個體) AWS Windows AMI 隨附額外的軟體，可在首次啟動時準備執行個體。可能是 EC2Config 服務 (Windows Server 2016 前的 Windows AMI) 或 EC2Launch (Windows Server 2016 和更新版本)。在裝置映射到磁碟機後，便會初始化並掛載。根磁碟機會初始化並掛載為 C:\。根據預設，EBS 磁碟區連接至 Windows 執行個體時，可能會在執行個體上顯示任何磁碟機代號。您可以根據您的規格，變更設定以設定磁碟區的磁碟機代號。對於實例存儲卷，默認值取決於驅動程序。AWS PV 驅動程式和 Citrix PV 驅動程式會指派執行個體儲存磁碟區磁碟區代號，從 Z: 到 A:。Red Hat 驅動程式會為執行個體存放區磁碟區指派 D: 到 Z: 的磁碟機代號。如需詳細資訊，請參閱 [設定 Amazon EC2 視窗執行個體的啟動設定](#) 和 [將磁碟映射至 Windows 執行個體上的磁碟區](#)。
- (Linux 執行個體) 視核心的區塊裝置驅動程式而定，裝置可能會使用與您指定的名稱不同。例如，假設您指定了裝置名稱 /dev/sdh，您的裝置可以重新命名為 /dev/xvdh 或 /dev/hdh。在大部分情況下，結尾的字母會維持相同。在 Red Hat Enterprise Linux (及其變體，例如 CentOS) 的某些版本中，結尾的字母可能會變更 (/dev/sda 可能會變成 /dev/xvde)。在這些情況下，各裝置名稱的結尾字母會以相同的次數遞增。例如，如果 /dev/sdb 重新命名為 /dev/xvdf，則 /dev/sdc 重

新命名為 `/dev/xvdg`。Amazon Linux 會為您指定給已重新命名裝置的名稱建立符號連結。其他作業系統可能會有不同的行為。

- (Linux 執行個體) HVM AMI 不支援在裝置名稱上使用尾隨數字，除了 `/dev/sda1` 保留給根裝置使用，以及 `/dev/sda2` 可以使用 `/dev/sda2` 時，不建議使用此裝置映射與 HVM 執行個體搭配。
- (Linux 執行個體) 使用 PV AMI 時，您無法附加共用相同裝置字母的磁碟區，且不含尾隨位數字。例如，若您將磁碟區連接為 `/dev/sdc`，並將另一個磁碟區連接為 `/dev/sdc1`，執行個體只會看見 `/dev/sdc`。若要在裝置名稱中使用尾碼，您必須在所有共享相同基礎代號的裝置名稱上使用尾碼 (例如 `/dev/sdc1`、`/dev/sdc2`、`/dev/sdc3`)。
- (Linux 執行個體) 某些自訂核心可能有限制使用 `/dev/sd[f-p]` 或 `/dev/sd[f-p][1-6]`。若您在 `/dev/sd[q-z]` 或 `/dev/sd[q-z][1-6]` 上發生問題，請嘗試切換為 `/dev/sd[f-p]` 或 `/dev/sd[f-p][1-6]`。

在您指定已選取的裝置名稱之前，請確認該名稱是否可用。否則，您將收到設備名稱已在使用中的錯誤。若要檢視磁碟裝置及其掛接點，請使用 `lsblk` 指令 (Linux 執行個體) 或「磁碟管理」公用程式或 `diskpart` 指令 (Windows 執行個體)。

## 區塊型儲存設備對應

您啟動的每個執行個體均有一個相關聯的根設備磁碟區，這可能是 Amazon EBS 磁碟區或執行個體存放磁碟區。您可以使用區塊型設備映射，來指定執行個體啟動時要連接的其他 EBS 磁碟區或執行個體存放磁碟區。您也可以將其他 EBS 磁碟區附加至執行中的執行個體。但是，唯一能將執行個體存放磁碟區連接至執行個體的方式，就是使用區塊型儲存設備映射來連接磁碟區，如同執行個體啟動時一樣。

### 目錄

- [區塊型儲存設備映射概念](#)
- [AMI 區塊型儲存設備映射](#)
- [執行個體區塊型儲存設備映射](#)

## 區塊型儲存設備映射概念

區塊型儲存設備是一種儲存設備，它可依位元組或位元 (區塊) 的順序移動資料。這些裝置支援隨機存取，且通常使用緩衝 I/O。例如硬碟、CD-ROM 光碟機和快閃磁碟機。區塊型設備可實際連接至電腦，或如同實際連接至電腦一般進行遠端存取。

Amazon EC2 支援兩種類型的區塊型設備：



- 執行個體存放磁碟區 (基礎硬體實際連接到執行個體主機電腦的虛擬設備)
- EBS 磁碟區 (遠端儲存設備)

區塊型設備映射定義要連接到執行個體的區塊型儲存設備 (執行個體存放磁碟區和 EBS 磁碟區)。您可在建立 AMI 時指定區塊型設備映射，如此從 AMI 啟動的所有執行個體都會使用該映射。或者，也可在啟動執行個體時指定區塊型設備映射，則此映射將覆寫啟動執行個體之 AMI 中指定的映射。請注意，某一執行個體類型支援的所有 NVMe 執行個體存放磁碟區會在執行個體啟動時自動列舉，並獲指派設備名稱；將這些磁碟區加入區塊型設備映射中並無作用。

## 內容

- [區塊型儲存設備映射項目](#)
- [區塊型儲存設備映射執行個體存放區 Caveat](#)
- [區塊型儲存設備映射範例](#)
- [如何在作業系統中將設備變為可用](#)

## 區塊型儲存設備映射項目

建立區塊型設備映射時，您需為要連接至執行個體的每個區塊型儲存設備指定下列資訊：

- Amazon EC2 內使用的裝置名稱。執行個體的區塊型儲存設備驅動程式，會在掛載磁碟區時指派實際的磁碟區名稱。指派的名稱可能和 Amazon EC2 建議的名稱不一樣。如需詳細資訊，請參閱 [Amazon EC2 實例上的設備名稱](#)。

對於執行個體存放區磁碟區，您也可以指定下列資訊：

- 虛擬裝置：ephemeral[0-23]。請注意，可供執行個體使用的執行個體存放磁碟區數量和大小，根據執行個體類型而異。

對於 NVMe 執行個體存放區磁碟區，也適用下列資訊：

- 這些磁碟區會自動列舉，並獲指派設備名稱；將這些磁碟區加入區塊型設備映射中並無作用。

對於 EBS 磁碟區，您也可以指定下列資訊：

- 用於建立區塊型儲存設備之快照的 ID (snap-xxxxxxx)。若已指定磁碟區大小，則此值是選用的。無法指定已封存快照的 ID。

- 磁碟區的大小 (GiB)。指定的大小必須大於或等於指定快照的大小。
- 是否在執行個體終止時刪除磁碟區 (true 或 false)。根設備磁碟區的預設值為 true，連接磁碟區的預設值為 false。建立 AMI 時，其區塊型設備映射會從執行個體繼承此設定。啟動執行個體時，其會從 AMI 繼承此設定。
- 磁碟區類型對於一般用途 SSD 為 gp2 和 gp3，對於佈建 IOPS SSD 為 io1 和 io2，對於輸送量最佳化 HDD 為 st1，對於冷 HDD 為 sc1，對於磁帶為 standard。
- 磁碟區支援的每秒輸入/輸出操作數 (IOPS)。(僅與 io1 和 io2 磁碟區搭配使用。)

## 區塊型儲存設備映射執行個體存放區 Caveat

使用區塊型設備映射中有執行個體存放磁碟區的 AMIs 啟動執行個體時，有幾個注意事項。

- 部分執行個體類型包含的執行個體存放磁碟區較多，有些執行個體類型則完全沒有執行個體存放磁碟區。如果您的執行個體類型支援一個執行個體存放磁碟區，而 AMI 有兩個執行個體存放磁碟區的映射，則執行個體將以一個執行個體存放磁碟區啟動。
- 執行個體存放磁碟區只能在啟動時映射。您無法停止沒有執行個體存放磁碟區的執行個體 (例如 t2.micro)，請將執行個體變更為支援執行個體存放磁碟區的類型，然後重新啟動含執行個體存放磁碟區的執行個體。但您可從執行個體建立 AMI，並在支援執行個體存放磁碟區的執行個體類型上啟動，然後將這些執行個體存放磁碟區映射至執行個體。
- 如果啟動有執行個體存放磁碟區映射的執行個體，接著停止執行個體，將其變更為執行個體存放磁碟區較少的執行個體類型，並重新啟動執行個體，則一開始啟動的執行個體存放磁碟區映射仍會出現在執行個體中繼資料內。但是，只有該執行個體類型支援的執行個體存放磁碟區數量上限可供執行個體使用。

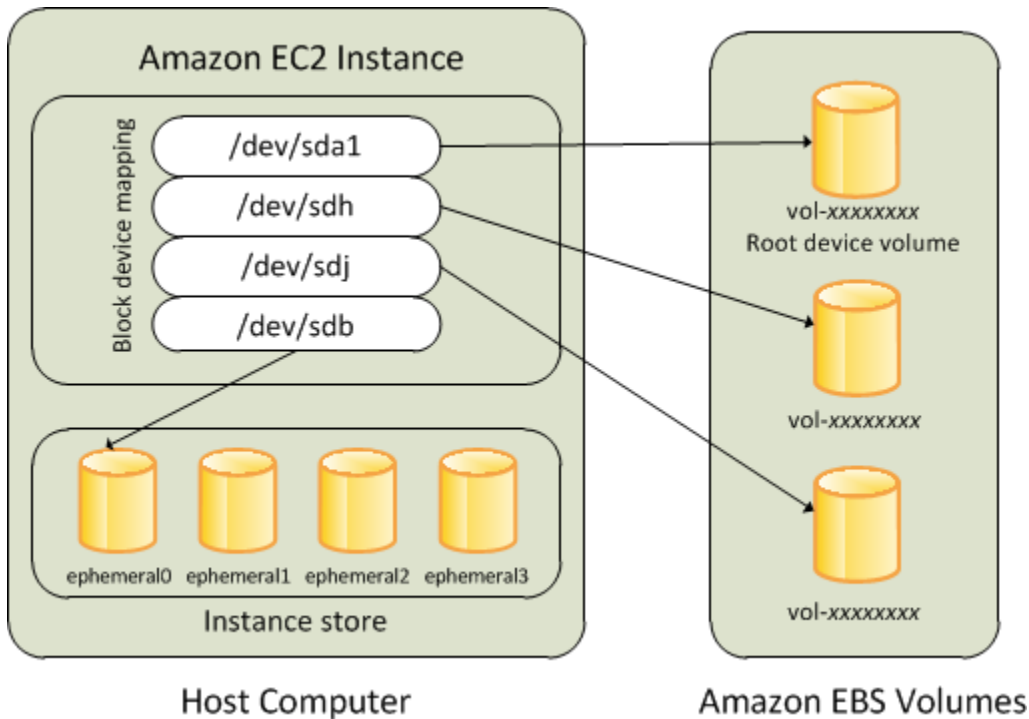
### Note

停止執行個體時，執行個體存放磁碟區上的所有資料都會遺失。

- 根據啟動時的執行個體存放區容量而定，除非於啟動時指定，否則 M3 執行個體啟動時可能會忽略 AMI 執行個體存放區的區塊型設備映射。您應在啟動時指定執行個體存放區的區塊型設備映射，即使要啟動的 AMI 在 AMI 內具有執行個體存放磁碟區映射，以確保執行個體存放磁碟區在執行個體啟動時可用。

## 區塊型儲存設備映射範例

下圖顯示 EBS 後端執行個體的區塊型設備映射範例。其將 `/dev/sdb` 映射至 `ephemeral0`，並映射兩個 EBS 磁碟區，一個映射至 `/dev/sdh`，另一個映射至 `/dev/sdj`。亦顯示為根設備磁碟區 `/dev/sda1` 的 EBS 磁碟區。



請注意，此區塊型設備映射範例使用於本主題的命令和 API 範例中。您可在[指定 AMI 的區塊型儲存裝置對應](#)和[啟動執行個體時更新區塊型儲存裝置對應](#)中找到建立區塊型設備映射的命令和 API 範例。

### 如何在作業系統中將設備變為可用

Amazon EC2 使用 `/dev/sdh` 和 `xvdh` 等設備名稱來描述區塊型儲存設備。Amazon EC2 使用區塊型設備映射來指定要連接到 EC2 執行個體的區塊型儲存設備。區塊型儲存設備連接至執行個體之後，作業系統必須掛載儲存設備，接著方可存取。區塊型儲存設備與執行個體分離之後，作業系統會卸載儲存設備，於是便無法再存取。

Linux 執行個體 — 區塊裝置對應中指定的裝置名稱會在執行個體第一次開機時對應至其對應的區塊裝置。執行個體類型將決定預設要格式化及掛載的執行個體存放磁碟區。您可在啟動時掛載額外的執行個體存放磁碟區，只要不超過執行個體類型可用的執行個體存放磁碟區數量即可。如需詳細資訊，請參閱[Amazon EC2 執行個體存放區](#)。執行個體的區塊型儲存設備驅動程式決定格式化及掛載磁碟區時要使用的設備。

Windows 執行個體 — 區塊裝置對應中指定的裝置名稱會在執行個體第一次開機時對應至其對應的區塊裝置，然後 EC2Config 服務會初始化並掛接磁碟機。根設備磁碟區將掛載為 C:\。執行個體存放磁碟區將掛載為 Z:\、Y:\，並以此類推。掛載 EBS 磁碟區時，其可用任何可用的磁碟機代號掛載。不過，您可以設定磁碟機代號指派給 EBS 磁碟區的方式；如需詳細資訊，請參閱[the section called “設定 Windows 啟動代理程式”](#)。

## AMI 區塊型儲存設備映射

每個 AMI 皆有區塊型設備映射，其指定從 AMI 啟動時要連接至執行個體的區塊型儲存設備。若要新增更多區塊型儲存設備至 AMI，您必須建立自己的 AMI。

### 內容

- [指定 AMI 的區塊型儲存裝置對應](#)
- [檢視 AMI 區塊型儲存裝置對應的 EBS 磁碟區](#)

## 指定 AMI 的區塊型儲存裝置對應

有兩種方法可在建立 AMI 時指定除了根磁碟區以外的其他磁碟區。如果您從執行個體建立 AMI 之前已將磁碟區連接至執行中的執行個體，AMI 的區塊型設備映射將包含這些相同的磁碟區。如為 EBS 磁碟區，現有資料將儲存至新快照，而區塊型設備映射中將指定這個新快照。如為執行個體存放磁碟區，資料將不會保留。

如為 EBS 後端 AMI，您可用區塊型設備映射新增 EBS 磁碟區和執行個體存放磁碟區。如為執行個體後端 AMI，您只能在登錄映像時修改映像工作資訊清單檔案中的區塊型設備映射項目，以此方式來新增執行個體存放磁碟區。

### Note

如為 M3 執行個體，您必須在啟動時在執行個體的區塊型設備映射中指定執行個體存放磁碟區。啟動 M3 執行個體時，若未指定為執行個體區塊型設備映射，AMI 區塊型設備映射中指定的執行個體存放磁碟區將遭忽略。

## Console

### 使用主控台新增磁碟區至 AMI

1. 開啟 Amazon EC2 主控台。

2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體並選取 Actions (動作)、Image and templates (影像和範本)、Create image (建立影像)。
4. 輸入影像的名稱和說明。
5. 執行個體磁碟區會出現在 Instance volumes (執行個體磁碟區) 下。若要新增其他磁碟區，請選擇 Add volume (新增磁碟區)。
6. 針對 Volume Type (磁碟區類型)，選擇一種磁碟區類型。對於 Device (裝置)，請選擇裝置名稱。對於 EBS 磁碟區，您可以指定其他詳細資訊，例如：快照、磁碟區大小、磁碟區類型、IOPS 和加密狀態。
7. 選擇建立映像。

## Command line

To add volumes to an AMI using the command line (使用命令列新增磁碟區至 AMI)

使用 [建立映像](#) 指 AWS CLI 令來指定 EBS 支援 AMI 的區塊裝置對應。使用 [寄存器映像檔](#) 指 AWS CLI 令，為執行個體存放區支援的 AMI 指定區塊裝置對應。

使用 `--block-device-mappings` 參數指定區塊型儲存設備映射。您可以在命令列上直接提供以 JSON 編碼的引數，或透過檔案的參考來提供：

```
--block-device-mappings [mapping, ...]
--block-device-mappings [file://mapping.json]
```

若要新增執行個體存放磁碟區，請使用下列映射。

```
{
 "DeviceName": "device_name",
 "VirtualName": "ephemeral0"
}
```

若要新增空的 100 GiB gp2 磁碟區，請使用下列映射。

```
{
 "DeviceName": "device_name",
 "Ebs": {
 "VolumeSize": 100
 }
}
```

```
}
}
```

若要以快照為基礎新增 EBS 磁碟區，請使用下列映射。

```
{
 "DeviceName": "device_name",
 "Ebs": {
 "SnapshotId": "snap-xxxxxxxx"
 }
}
```

若要省略設備的映射，請使用下列映射。

```
{
 "DeviceName": "device_name",
 "NoDevice": ""
}
```

或者，您可用 `-BlockDeviceMapping` 參數結合下列命令 (AWS Tools for Windows PowerShell)：

- [New-EC2Image](#)
- [Register-EC2Image](#)

## 檢視 AMI 區塊型儲存裝置對應的 EBS 磁碟區

您可輕鬆列舉 AMI 區塊型設備映射中的 EBS 磁碟區。

### Console

使用主控台檢視 AMI 的 EBS 磁碟區

1. 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 AMIs (AMI)。
3. 從 Filter (篩選條件) 清單中選擇 EBS images (EBS 映像)，以顯示 EBS 後端 AMI 的清單。
4. 選取所要的 AMI，然後檢視 Details (詳細資訊) 索引標籤。將至少顯示根儲存設備的以下資訊：

- Root Device Type (根設備類型) ebs ()
- Root Device Name (根設備名稱) (例如, /dev/sda1)
- Block Devices (區塊型儲存設備) (例如, /dev/sda1=snap-1234567890abcdef0:8:true)

如果使用區塊型設備映射建立含額外 EBS 磁碟區的 AMI, Block Devices (區塊型儲存設備) 欄位也會顯示額外磁碟區的映射。(此畫面不會顯示執行個體存放磁碟區。)

## Command line

To view the EBS volumes for an AMI using the command line (使用命令列檢視 AMI 的 EBS 磁碟區)

使用 [描述影像](#) (AWS CLI) 指令或 [Get-EC2Image](#) (AWS Tools for Windows PowerShell) 指令列舉 AMI 的區塊裝置對應中的 EBS 磁碟區。

## 執行個體區塊型儲存設備映射

根據預設, 您所啟動的執行個體包含啟動執行個體之 AMI 區塊型設備映射中指定的任何儲存設備。您可在啟動執行個體時指定變更執行個體的區塊型設備映射, 這些更新將覆寫或合併 AMI 的區塊型設備映射。

### 限制

- 如為根磁碟區, 只能修改: 磁碟區大小、磁碟區類型和 Delete on Termination (在終止時刪除) 旗標。
- 修改 EBS 磁碟區時, 您無法減少其大小。因此, 您所指定的快照大小必須等於或大於 AMI 區塊型設備映射中的快照。

### 內容

- [啟動執行個體時更新區塊型儲存裝置對應](#)
- [更新執行中執行個體的區塊型儲存裝置對應](#)
- [檢視執行個體區塊型儲存裝置對應的 EBS 磁碟區](#)
- [檢視執行個體區塊型儲存裝置對應的執行個體存放區的磁碟區](#)

## 啟動執行個體時更新區塊型儲存裝置對應

您可在啟動執行個體時新增 EBS 磁碟區和執行個體存放磁碟區至執行個體。但請注意，更新執行個體的區塊型設備映射，不會永久變更啟動執行個體之 AMI 的區塊型設備映射。

### Console

使用主控台新增磁碟區至執行個體

1. 開啟 Amazon EC2 主控台。
2. 在儀表中，選擇 Launch Instance (啟動執行個體)。
3. 在 Choose an Amazon Machine Image (AMI) (選取 Amazon Machine Image (AMI)) 頁面中選取欲使用的 AMI，然後選取 Select (選取)。
4. 遵循精靈的指示，來完成 Choose an Instance Type (選擇執行個體類型) 和 Configure Instance Details (設定執行個體的詳細資訊) 頁面上的操作。
5. 您可在 Add Storage (新增儲存) 頁面上修改根磁碟區、EBS 磁碟區和執行個體存放磁碟區，如下所示：
  - 若要變更根磁碟區的大小，請在 Type (類型) 資料行中尋找 Root (根) 磁碟區，並變更其 Size (大小) 欄位的值。
  - 若要抑制用於啟動執行個體之 AMI 的區塊型設備映射中指定的 EBS 磁碟區，請找出磁碟區並按一下其 Delete (刪除) 圖示。
  - 若要新增 EBS 磁碟區，請選擇 Add New Volume (新增新磁碟區)，選擇 Type (類型) 清單中的 EBS，然後填寫欄位 (Device (裝置)、Snapshot (快照) 等)。
  - 若要抑制用於啟動執行個體之 AMI 的區塊型設備映射中指定的執行個體存放磁碟區，請找出磁碟區並選擇其 Delete (刪除) 圖示。
  - 若要新增執行個體存放磁碟區，請選取 Add New Volume (新增新磁碟區)，選取 Type (類型) 清單中的 Instance Store (執行個體存放區)，然後從 Device (設備) 中選取設備名稱。
6. 完成剩餘的精靈頁面，然後選擇 Launch (啟動)。

### Command line

若要使用新增磁碟區至執行個體 AWS CLI

使用 [run-instance](#) 指 AWS CLI 令搭配 `--block-device-mappings` 選項來指定啟動時執行個體的區塊裝置對映。



例如，假設 EBS 支援的 AMI 會為 Linux 執行個體指定下列區塊裝置對應：

- `/dev/sdb = ephemeral0`
- `/dev/sdh = snap-1234567890abcdef0`
- `/dev/sdj = 100`

若要避免 `/dev/sdj` 連接至從此 AMI 啟動的執行個體，請使用下列映射。

```
{
 "DeviceName": "/dev/sdj",
 "NoDevice": ""
}
```

若要增加 `/dev/sdh` 到的大小 300 GiB，請指定下列對映。請注意，您不需要指定 `/dev/sdh` 的快照 ID，因為指定設備名稱足以找出磁碟區。

```
{
 "DeviceName": "/dev/sdh",
 "Ebs": {
 "VolumeSize": 300
 }
}
```

若要在執行個體啟動時增加根磁碟區的大小，請先呼叫具有 AMI ID 的 [describe-images](#) 來驗證根磁碟區的裝置名稱。例如：`"RootDeviceName": "/dev/xvda"`。若要覆寫根磁碟區的大小，請指定 AMI 使用之根裝置的裝置名稱和新磁碟區大小。

```
{
 "DeviceName": "/dev/xvda",
 "Ebs": {
 "VolumeSize": 100
 }
}
```

若要連接額外的執行個體存放磁碟區 `/dev/sdc`，請指定下列映射。如果執行個體類型不支援多重執行個體存放磁碟區，此映射將沒有效果。如果執行個體支援 NVMe 執行個體存放區磁碟區，它們會自動列舉並指派 NVMe 裝置名稱。

```
{
```

```
"DeviceName": "/dev/sdc",
"VirtualName": "ephemeral1"
}
```

若要使用新增磁碟區至執行個體 AWS Tools for Windows PowerShell

透過 [New-EC2Instance](#) 指令使用 `-BlockDeviceMapping` 參數 (AWS Tools for Windows PowerShell)。

## 更新執行中執行個體的區塊型儲存裝置對應

您可以使用 [修改執行個體屬性 AWS CLI 指令來更新執行](#) 中執行個體的圖塊裝置對應。您不需要為了變更屬性而停止執行個體。

```
aws ec2 modify-instance-attribute --instance-id i-1a2b3c4d --block-device-mappings
file://mapping.json
```

例如，若要在執行個體終止時保留根磁碟區，請在 `mapping.json` 中指定下列項目。

```
[
 {
 "DeviceName": "/dev/sda1",
 "Ebs": {
 "DeleteOnTermination": false
 }
 }
]
```

或者，您可以將 `-BlockDeviceMapping` 參數與 [Edit-EC2InstanceAttribute](#) 指令 (AWS Tools for Windows PowerShell) 搭配使用。

## 檢視執行個體區塊型儲存裝置對應的 EBS 磁碟區

您可輕鬆列舉映射至執行個體的 EBS 磁碟區。

### Note

對於在 2009-10-31 API 發行之之前啟動的執行個體，AWS 無法顯示區塊裝置對應。您必須分離並重新連接磁碟區，才 AWS 能顯示區塊裝置對應。

## Console

使用主控台檢視執行個體的 EBS 磁碟區

1. 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 在搜尋方塊中，輸入 Root device type (根裝置類型)，再選擇 EBS。將顯示 EBS 後端執行個體的清單。
4. 選取所要的執行個體，並查看 Storage (儲存) 索引標籤中顯示的詳細資訊。將至少顯示根儲存設備的以下資訊：
  - 根裝置類型 (例如：EBS)
  - Root device name (根裝置名稱) (例如：/dev/xvda)
  - Block devices (區塊型儲存裝置) (例如：/dev/xvda、/dev/sdf 和 /dev/sdj)

如果執行個體是使用區塊型儲存裝置對應的其他 EBS 磁碟區來啟動，則其會顯示在 Block devices (區塊型儲存裝置)。任何執行個體存放區的磁碟區都不會顯示在此索引標籤上。

5. 若要顯示 EBS 磁碟區的其他資訊，請選擇其磁碟區 ID 以移至磁碟區頁面。

## Command line

To view the EBS volumes for an instance using the command line (使用命令列檢視執行個體的 EBS 磁碟區)

使用 [describe-instance](#) (AWS CLI) 指令或 [Get-EC2Instance](#) (AWS Tools for Windows PowerShell) 指令列舉執行個體的區塊裝置對應中的 EBS 磁碟區。

## 檢視執行個體區塊型儲存裝置對應的執行個體存放區的磁碟區

執行個體類型決定執行個體可用的執行個體儲存磁碟區數量和類型。如果區塊型設備映射中的執行個體存放磁碟區的數量，超過執行個體可用的執行個體存放磁碟區，請忽略額外的磁碟區。若要檢視執行個體的執行個體存放區磁碟區，請執行 lsblk 指令 (Linux 執行個體) 或開啟 Windows 磁碟管理 (Windows 執行個體)。要了解每個執行個體類型支援多少個執行個體存放區磁碟區，請參閱 [Amazon EC2 執行個體類型規格](#)。

檢視執行個體的區塊型設備映射時，您只會看見 EBS 磁碟區，而非執行個體存放磁碟區。您用來檢視執行個體的執行個體存放區磁碟區的方法取決於磁碟區類型。

## NVMe 執行個體存放區磁碟區

### Linux 執行個體

您可以使用 NVMe 命令列套件 ([nvme-cli](#)) 來查詢區塊型設備映射中的 NVMe 執行個體存放區磁碟區。在您的執行個體上下載並安裝套件，然後執行下列命令。

```
[ec2-user ~]$ sudo nvme list
```

以下是執行個體的範例輸出。Model (模型) 欄中的文字指出磁碟區是 EBS 磁碟區還是執行個體存放區磁碟區。在本範例中，`/dev/nvme1n1` 和 `/dev/nvme2n1` 是執行個體存放區磁碟區。

Node	SN	Model	
Namespace			
-----			
-----			
<code>/dev/nvme0n1</code>	<code>vol06afc3f8715b7a597</code>	<code>Amazon Elastic Block Store</code>	<code>1</code>
<code>/dev/nvme1n1</code>	<code>AWS2C1436F5159EB6614</code>	<code>Amazon EC2 NVMe Instance Storage</code>	<code>1</code>
<code>/dev/nvme2n1</code>	<code>AWSB1F4FF0C0A6C281EA</code>	<code>Amazon EC2 NVMe Instance Storage</code>	<code>1</code>
<code>...</code>			

### Windows 執行個體

您可以使用磁碟管理，或 PowerShell 同時列出 EBS 和執行個體儲存 NVMe 磁碟區。如需詳細資訊，請參閱 [the section called “列出 NVMe 磁碟區”](#)。

### HDD 或 SSD 執行個體存放區磁碟區

您可以使用執行個體中繼資料來查詢區塊型設備映射中的 HDD 或 SSD 執行個體存放區磁碟區。不包含 NVMe 執行個體存放磁碟區。

執行個體中繼資料所有請求的基底 URI 為 `http://169.254.169.254/latest/`。如需詳細資訊，請參閱 [使用執行個體中繼資料](#)。

### Linux 執行個體

首先，連接至執行中的執行個體。從執行個體使用此查詢，取得其區塊型設備映射。

## IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/block-device-mapping/
```

## IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/block-device-mapping/
```

回應將包含執行個體的區塊型儲存設備名稱。例如，執行個體存放區支援的 `m1.small` 執行個體的輸出看起來如下。

```
ami
ephemeral0
root
swap
```

`ami` 設備並非執行個體所看見的根設備。執行個體存放磁碟區將命名為 `ephemeral[0-23]`。`swap` 設備用於分頁檔。如果您也映射了 EBS 磁碟區，其將顯示為 `ebs1`、`ebs2` 等等。

若要取得區塊型設備映射中個別區塊型儲存設備的詳細資訊，請將其名稱附加到先前的查詢，如下所示。

## IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/block-device-mapping/ephemeral0
```

## IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/block-device-mapping/ephemeral0
```

## Windows 執行個體

首先，連接至執行中的執行個體。從執行個體使用此查詢，取得其區塊型設備映射。

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/block-device-mapping/
```

回應將包含執行個體的區塊型儲存設備名稱。例如，執行個體存放區支援的 m1.small 執行個體的輸出看起來如下。

```
ami
ephemeral0
root
swap
```

ami 設備並非執行個體所看見的根設備。執行個體存放磁碟區將命名為 ephemeral[0-23]。swap 設備用於分頁檔。如果您也映射了 EBS 磁碟區，其將顯示為 ebs1、ebs2 等等。

若要取得區塊型設備映射中個別區塊型儲存設備的詳細資訊，請將其名稱附加到先前的查詢，如下所示。

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/block-device-mapping/ephemeral0
```

## 將磁碟映射至 Windows 執行個體上的磁碟區

### Note

本主題僅適用於 Windows 執行個體。

Windows 執行個體隨附 EBS 磁碟區，此磁碟區可做為根磁碟區使用。如果您的 Windows 執行個體使用 AWS PV 或 Citrix 光伏驅動程式，您可以選擇性地新增最多 25 個磁碟區，總共計 26 個磁碟區。如需詳細資訊，請參閱 [執行個體磁碟區限制](#)。

取決於執行個體的類型，執行個體將可使用 0 到 24 個可能的執行個體存放磁碟區。若要使用執行個體可用的任何執行個體存放磁碟區，您必須在建立 AMI 或啟動執行個體時指定這些磁碟區。您也可以在建​​立 AMI 或啟動執行個體時新增 EBS 磁碟區，或是在執行個體執行時連結這些磁碟區。

將磁碟區新增到執行個體時，您會指定 Amazon EC2 所使用的裝置名稱。如需詳細資訊，請參閱 [Amazon EC2 實例上的設備名稱](#)。AWS Windows Amazon Machine Image (AMI) 包含一組 Amazon EC2 會使用的驅動程式，可用來將執行個體存放區和 EBS 磁碟區對應到 Windows 磁碟和磁碟編號。如果您從使用 AWS PV 或 Citrix 光伏驅動程式的 Windows AMI 啟動執行個體，則可以使用本頁所述的關係將 Windows 磁碟對應至執行個體存放區和 EBS 磁碟區。如果 Windows AMI 使用 Red Hat PV 驅動程式，您可以更新執行個體來使用 Citrix 驅動程式。如需詳細資訊，請參閱 [the section called “升級 PV 驅動程式”](#)。

## 內容

- [列出 NVMe 磁碟區](#)
  - [使用磁碟管理列出 NVMe 磁碟](#)
  - [列出 NVMe 磁碟使用方式 PowerShell](#)
  - [映射 NVMe EBS 磁碟區](#)
- [列出磁碟區](#)
  - [使用磁碟管理列出磁碟](#)
  - [將磁碟裝置映射至裝置名稱](#)
    - [執行個體儲存體磁碟區](#)
    - [EBS 磁碟區](#)
  - [列出磁碟使用 PowerShell](#)

## 列出 NVMe 磁碟區

您可以使用磁碟管理公用程式或 Powershell，來搜尋 Windows 執行個體上的磁碟。

### 使用磁碟管理列出 NVMe 磁碟

您可以使用磁碟管理公用程式，來搜尋 Windows 執行個體上的磁碟。

### 搜尋 Windows 執行個體上的磁碟

1. 使用遠端桌面登入 Windows 執行個體。如需詳細資訊，請參閱 [連線至 Windows 執行個體](#)。
2. 啟動磁碟管理公用程式。
3. 檢閱磁碟。根磁碟區是掛載為 C:\ 的 EBS 磁碟區。如果沒有顯示其他的磁碟，表示當您建立 AMI 或啟動執行個體時，並未指定其他的磁碟區。

下列範例顯示當您啟動具有兩個額外 EBS 磁碟區的 r5d.4xlarge 執行個體時，可以使用的磁碟。

The screenshot shows the Windows Disk Management console. At the top, there is a table listing the volumes. Below this, each disk is shown with its details and the partitions it contains.

Volume	Layout	Type	File System	Status	Capacity	Free Spa...	% Free
(C:)	Simple	Basic	NTFS	Healthy (S...	30.00 GB	13.22 GB	44 %
New Volume (D:)	Simple	Basic	NTFS	Healthy (P...	8.00 GB	7.97 GB	100 %
New Volume (E:)	Simple	Basic	NTFS	Healthy (P...	8.00 GB	7.97 GB	100 %
New Volume (F:)	Simple	Basic	NTFS	Healthy (P...	279.39 GB	279.28 GB	100 %
New Volume (G:)	Simple	Basic	NTFS	Healthy (P...	279.39 GB	279.28 GB	100 %

Disk	Details	Partitions
Disk 0 Basic 30.00 GB Online	(C:) 30.00 GB NTFS Healthy (System, Boot, Page File, Active, Crash Dump, Primary Partition)	
Disk 1 Basic 8.00 GB Online	New Volume (D:) 8.00 GB NTFS Healthy (Primary Partition)	
Disk 2 Basic 8.00 GB Online	New Volume (E:) 8.00 GB NTFS Healthy (Primary Partition)	
Disk 3 Basic 279.40 GB Online	New Volume (F:) 279.39 GB NTFS Healthy (Primary Partition)	
Disk 4 Basic 279.40 GB Online	New Volume (G:) 279.39 GB NTFS Healthy (Primary Partition)	

Legend: ■ Unallocated ■ Primary partition



## 列出 NVMe 磁碟使用方式 PowerShell

下列 PowerShell 指令碼會列出每個磁碟及其對應的裝置名稱和磁碟區。它適用於 N [AWS itro 系統上建置的執行個體](#)，這些執行個體使用 NVMe EBS 和執行個體存放區磁碟區。

Connect 至您的 Windows 執行個體，並執行下列命令以啟用指 PowerShell 令碼執行。

```
Set-ExecutionPolicy RemoteSigned
```

複製下列指令碼，並在您的 Windows 執行個體上將其儲存為 mapping.ps1。

```
List the disks for NVMe volumes

function Get-EC2InstanceMetadata {
 param([string]$Path)
 (Invoke-WebRequest -Uri "http://169.254.169.254/latest/$Path").Content
}

function GetEBSVolumeId {
 param($Path)
 $SerialNumber = (Get-Disk -Path $Path).SerialNumber
 if($SerialNumber -clike 'vol*'){
 $EbsVolumeId = $SerialNumber.Substring(0,20).Replace("vol","vol-")
 }
 else {
 $EbsVolumeId = $SerialNumber.Substring(0,20).Replace("AWS","AWS-")
 }
 return $EbsVolumeId
}

function GetDeviceName{
 param($EbsVolumeId)
 if($EbsVolumeId -clike 'vol*'){

 $Device = ((Get-EC2Volume -VolumeId $EbsVolumeId).Attachment).Device
 $VolumeName = ""
 }
 else {
 $Device = "Ephemeral"
 $VolumeName = "Temporary Storage"
 }
 Return $Device,$VolumeName
}
```

```
function GetDriveLetter{
 param($Path)
 $DiskNumber = (Get-Disk -Path $Path).Number
 if($DiskNumber -eq 0){
 $VirtualDevice = "root"
 $DriveLetter = "C"
 $PartitionNumber = (Get-Partition -DriveLetter C).PartitionNumber
 }
 else
 {
 $VirtualDevice = "N/A"
 $DriveLetter = (Get-Partition -DiskNumber $DiskNumber).DriveLetter
 if(!$DriveLetter)
 {
 $DriveLetter = ((Get-Partition -DiskId $Path).AccessPaths).Split(",")[0]
 }
 $PartitionNumber = (Get-Partition -DiskId $Path).PartitionNumber
 }

 return $DriveLetter,$VirtualDevice,$PartitionNumber
}

$Report = @()
foreach($Path in (Get-Disk).Path)
{
 $Disk_ID = (Get-Partition -DiskId $Path).DiskId
 $Disk = (Get-Disk -Path $Path).Number
 $EbsVolumeId = GetEBSVolumeId($Path)
 $Size =(Get-Disk -Path $Path).Size
 $DriveLetter,$VirtualDevice, $Partition = (GetDriveLetter($Path))
 $Device,$VolumeName = GetDeviceName($EbsVolumeId)
 $Disk = New-Object PSObject -Property @{
 Disk = $Disk
 Partitions = $Partition
 DriveLetter = $DriveLetter
 EbsVolumeId = $EbsVolumeId
 Device = $Device
 VirtualDevice = $VirtualDevice
 VolumeName = $VolumeName
 }
 $Report += $Disk
}
}
```

```
$Report | Sort-Object Disk | Format-Table -AutoSize -Property Disk, Partitions,
DriveLetter, EbsVolumeId, Device, VirtualDevice, VolumeName
```

執行指令碼，如下所示：

```
PS C:\> .\mapping.ps1
```

以下是具有根磁碟區、兩個 EBS 磁碟區和兩個執行個體存放區磁碟區之執行個體的範例輸出。

Disk	Partitions	DriveLetter	EbsVolumeId	Device	VirtualDevice	VolumeName
0	1	C	vol-03683f1d861744bc7	/dev/sda1	root	
1	1	D	vol-082b07051043174b9	xvdb	N/A	
2	1	E	vol-0a4064b39e5f534a2	xvdc	N/A	
3	1	F	AWS-6AAD8C2AE1193F0	Ephemeral	N/A	Temporary
Storage						
4	1	G	AWS-13E7299C2BD031A28	Ephemeral	N/A	Temporary
Storage						

如果您未在 Windows 執行個體 PowerShell 上設定 Windows 適用工具的身分證明，則指令碼將無法取得 EBS 磁碟區識別碼，並在資料行 EbsVolumeId 中使用 N/A。

## 映射 NVMe EBS 磁碟區

在 N [AWS Nitro 系統上建置的執行個體](#) 時，EBS 磁碟區會公開為 NVMe 裝置。您可以使用 [Get-Disk](#) 命令，將 Windows 磁碟編號映射到 EBS 磁碟區 ID。

```
PS C:\> Get-Disk
Number Friendly Name Serial Number HealthStatus
OperationalStatus Total Size Partition
Style

3 NVMe Amazo... AWS6AAD8C2AE1193F0_00000001. Healthy Online
279.4 GB MBR
4 NVMe Amazo... AWS13E7299C2BD031A28_00000001. Healthy Online
279.4 GB MBR
2 NVMe Amazo... vol0a4064b39e5f534a2_00000001. Healthy Online
8 GB MBR
```

0	NVMe Amazo... vol03683f1d861744bc7_00000001. 30 GB MBR	Healthy	Online
1	NVMe Amazo... vol082b07051043174b9_00000001. 8 GB MBR	Healthy	Online

您也可以執行 `ebsnvme-id` 命令，將 NVMe 磁碟編號映射至 EBS 磁碟區 ID 和裝置名稱。

```
PS C:\> C:\PROGRAMDATA\Amazon\Tools\ebsnvme-id.exe
Disk Number: 0
Volume ID: vol-03683f1d861744bc7
Device Name: sda1

Disk Number: 1
Volume ID: vol-082b07051043174b9
Device Name: xvdb

Disk Number: 2
Volume ID: vol-0a4064b39e5f534a2
Device Name: xvdc
```

## 列出磁碟區

您可以使用磁碟管理公用程式或 Powershell，來搜尋 Windows 執行個體上的磁碟。

### 使用磁碟管理列出磁碟

您可以使用磁碟管理公用程式，來搜尋 Windows 執行個體上的磁碟。

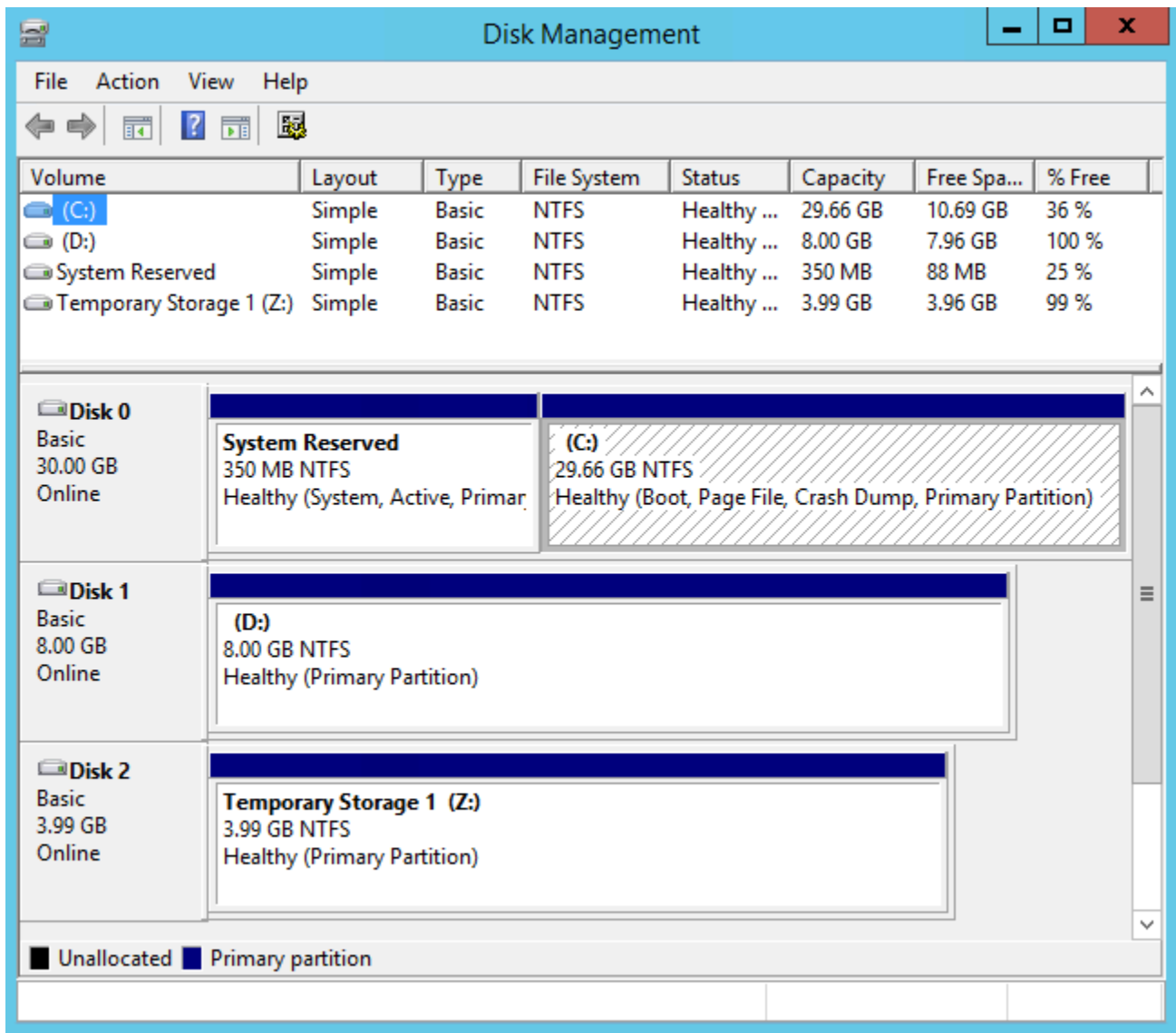
### 搜尋 Windows 執行個體上的磁碟

1. 使用遠端桌面登入 Windows 執行個體。如需詳細資訊，請參閱 [連線至 Windows 執行個體](#)。
2. 啟動磁碟管理公用程式。

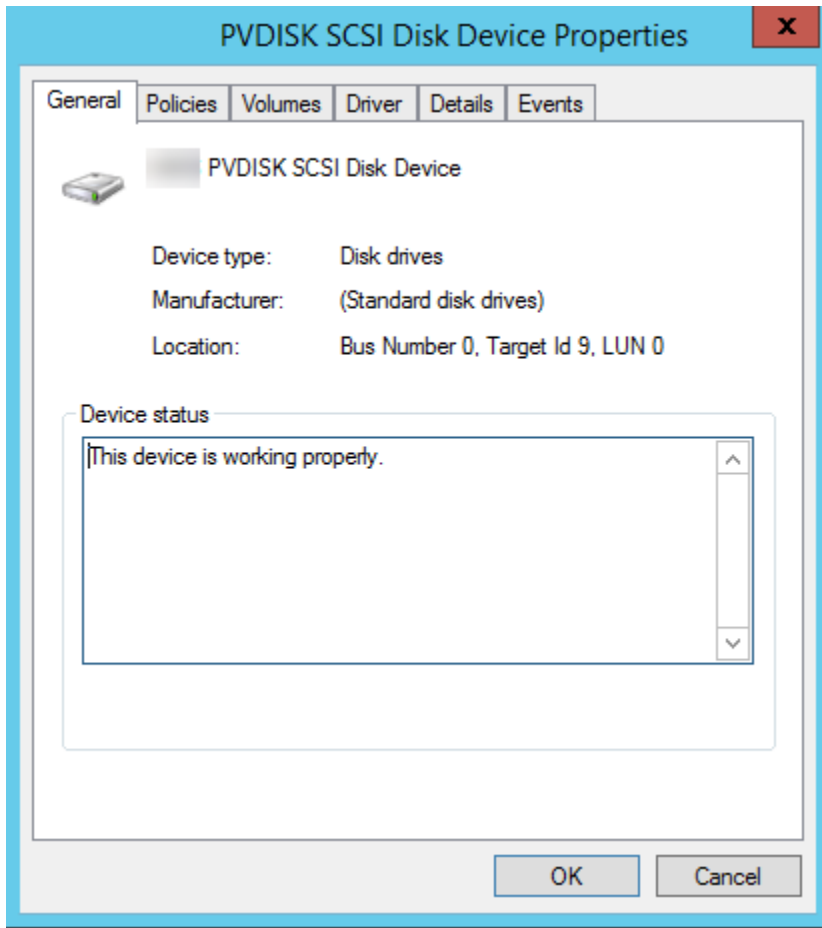
在工作列上，以滑鼠右鍵按一下 Windows 標誌，然後選擇 [磁碟管理]。

3. 檢閱磁碟。根磁碟區是掛載為 C:\ 的 EBS 磁碟區。如果沒有顯示其他的磁碟，表示當您建立 AMI 或啟動執行個體時，並未指定其他的磁碟區。

下列的範例，會顯示當您使用執行個體存放磁碟區 (磁碟 2) 和其他的 EBS 磁碟區 (磁碟 1) 來啟動 `m3.medium` 執行個體時，可以使用的磁碟。



- 在以灰色窗格標示的磁碟 1 上，按一下滑鼠右鍵，然後選取 Properties (屬性)。記下 Location (位置) 的值，然後在 [將磁碟裝置映射至裝置名稱](#) 的表格中查閱。例如，下列磁碟的位置為匯流排編號 0，目標 Id 9，LUN 0。根據 EBS 磁碟區的對應表，此位置的裝置名稱為 xvdi。



## 將磁碟裝置映射至裝置名稱

執行個體的區塊型儲存設備驅動程式，會在掛載磁碟區時指派實際的磁碟區名稱。

### 映射項目

- [執行個體儲存體磁碟區](#)
- [EBS 磁碟區](#)

### 執行個體儲存體磁碟區

下表說明 Citrix 光伏和光 AWS 伏驅動程式如何將非 NVMe 執行個體儲存磁碟區與 Windows 磁碟區對應。可用執行個體存放磁碟區的數量，會取決於執行個體的類型。如需詳細資訊，請參閱 [執行個體儲存體磁碟區](#)。

位置	裝置名稱
匯流排編號 0，目標 ID 78，LUN 0	xvdca
匯流排編號 0，目標 ID 79，LUN 0	xvdcb
匯流排編號 0，目標 ID 80，LUN 0	xvdcc
匯流排編號 0，目標 ID 81，LUN 0	xvdcd
匯流排編號 0，目標 ID 82，LUN 0	xvdce
匯流排編號 0，目標 ID 83，LUN 0	xvdcf
匯流排編號 0，目標 ID 84，LUN 0	xvdcg
匯流排編號 0，目標 ID 85，LUN 0	xvdch
匯流排編號 0，目標 ID 86，LUN 0	xvdci
匯流排編號 0，目標 ID 87，LUN 0	xvdcj
匯流排編號 0，目標 ID 88，LUN 0	xvdck
匯流排編號 0，目標 ID 89，LUN 0	xvdcl

## EBS 磁碟區

下表說明 Citrix 光伏和光 AWS 伏驅動程式如何將非 NVMe EBS 磁碟區對應至 Windows 磁碟區。

位置	裝置名稱
匯流排編號 0，目標 ID 0，LUN 0	/dev/sda1
匯流排編號 0，目標 ID 1，LUN 0	xvdb
匯流排編號 0，目標 ID 2，LUN 0	xvdc
匯流排編號 0，目標 ID 3，LUN 0	xvdd

位置	裝置名稱
匯流排編號 0，目標 ID 4，LUN 0	xvde
匯流排編號 0，目標 ID 5，LUN 0	xvdf
匯流排編號 0，目標 ID 6，LUN 0	xvdg
匯流排編號 0，目標 ID 7，LUN 0	xvdh
匯流排編號 0，目標 ID 8，LUN 0	xvdi
匯流排編號 0，目標 ID 9，LUN 0	xvdj
匯流排編號 0，目標 ID 10，LUN 0	xvdk
匯流排編號 0，目標 ID 11，LUN 0	xvdl
匯流排編號 0，目標 ID 12，LUN 0	xvdm
匯流排編號 0，目標 ID 13，LUN 0	xvdn
匯流排編號 0，目標 ID 14，LUN 0	xvdo
匯流排編號 0，目標 ID 15，LUN 0	xvdp
匯流排編號 0，目標 ID 16，LUN 0	xvdq
匯流排編號 0，目標 ID 17，LUN 0	xvdr
匯流排編號 0，目標 ID 18，LUN 0	xvds
匯流排編號 0，目標 ID 19，LUN 0	xvdt
匯流排編號 0，目標 ID 20，LUN 0	xvdu
匯流排編號 0，目標 ID 21，LUN 0	xvdv
匯流排編號 0，目標 ID 22，LUN 0	xvdw
匯流排編號 0，目標 ID 23，LUN 0	xvdx



位置	裝置名稱
匯流排編號 0，目標 ID 24，LUN 0	xvdy
匯流排編號 0，目標 ID 25，LUN 0	xvdz

## 列出磁碟使用 PowerShell

下列 PowerShell 指令碼會列出每個磁碟及其對應的裝置名稱和磁碟區。

### 要求與限制

- 需要 Windows Server 2012 或更新版本。
- 需要登入資料才能取得 EBS 磁碟區 ID。您可以使用的工具來設定設定檔 PowerShell，或將 IAM 角色附加至執行個體。
- 不支援 NVMe 磁碟區。
- 不支援動態磁碟。

Connect 至您的 Windows 執行個體，並執行下列命令以啟用指 PowerShell 令碼執行。

```
Set-ExecutionPolicy RemoteSigned
```

複製下列指令碼，並在您的 Windows 執行個體上將其儲存為 mapping.ps1。

```
List the disks
function Convert-SCSITargetIdToDeviceName {
 param([int]$SCSITargetId)
 If ($SCSITargetId -eq 0) {
 return "sda1"
 }
 $deviceName = "xvd"
 If ($SCSITargetId -gt 25) {
 $deviceName += [char](0x60 + [int]($SCSITargetId / 26))
 }
 $deviceName += [char](0x61 + $SCSITargetId % 26)
 return $deviceName
}

[string[]]$array1 = @()
```

```
[string[]]$array2 = @()
[string[]]$array3 = @()
[string[]]$array4 = @()

Get-WmiObject Win32_Volume | Select-Object Name, DeviceID | ForEach-Object {
 $array1 += $_.Name
 $array2 += $_.DeviceID
}

$i = 0
While ($i -ne ($array2.Count)) {
 $array3 += ((Get-Volume -Path $array2[$i] | Get-Partition | Get-Disk).SerialNumber) -
replace "_[^]*$" -replace "vol", "vol-"
 $array4 += ((Get-Volume -Path $array2[$i] | Get-Partition | Get-Disk).FriendlyName)
 $i ++
}

[array[]]$array = $array1, $array2, $array3, $array4

Try {
 $InstanceId = Get-EC2InstanceMetadata -Category "InstanceId"
 $Region = Get-EC2InstanceMetadata -Category "Region" | Select-Object -ExpandProperty
SystemName
}
Catch {
 Write-Host "Could not access the instance Metadata using AWS Get-EC2InstanceMetadata
CMDLet.
Verify you have AWSPowershell SDK version '3.1.73.0' or greater installed and Metadata
is enabled for this instance." -ForegroundColor Yellow
}
Try {
 $BlockDeviceMappings = (Get-EC2Instance -Region $Region -Instance
$InstanceId).Instances.BlockDeviceMappings
 $VirtualDeviceMap = (Get-EC2InstanceMetadata -Category
"BlockDeviceMapping").GetEnumerator() | Where-Object { $_.Key -ne "ami" }
}
Catch {
 Write-Host "Could not access the AWS API, therefore, VolumeId is not available.
Verify that you provided your access keys or assigned an IAM role with adequate
permissions." -ForegroundColor Yellow
}

Get-disk | ForEach-Object {
 $DriveLetter = $null
```

```

$VolumeName = $null
$VirtualDevice = $null
$DeviceName = $_.FriendlyName

$DiskDrive = $_
$Disk = $_.Number
$Partitions = $_.NumberOfPartitions
$EbsVolumeID = $_.SerialNumber -replace "_[^]*$" -replace "vol", "vol-"
if ($Partitions -ge 1) {
 $PartitionsData = Get-Partition -DiskId $_.Path
 $DriveLetter = $PartitionsData.DriveLetter | Where-object { $_ -notin @("",
 $null) }
 $VolumeName = (Get-PSDrive | Where-Object { $_.Name -in
 @($DriveLetter) }).Description | Where-object { $_ -notin @("", $null) }
}
If ($DiskDrive.path -like "*PROD_PVDISK*") {
 $BlockDeviceName = Convert-SCSITargetIdToDeviceName((Get-WmiObject -Class
 Win32_Diskdrive | Where-Object { $_.DeviceID -eq ("\\.\PHYSICALDRIVE" +
 $DiskDrive.Number) }).SCSITargetId)
 $BlockDeviceName = "/dev/" + $BlockDeviceName
 $BlockDevice = $BlockDeviceMappings | Where-Object { $BlockDeviceName -like "*" +
 $_.DeviceName + "*" }
 $EbsVolumeID = $BlockDevice.Ebs.VolumeId
 $VirtualDevice = ($VirtualDeviceMap.GetEnumerator() | Where-Object { $_.Value -eq
 $BlockDeviceName }).Key | Select-Object -First 1
}
ElseIf ($DiskDrive.path -like "*PROD_AMAZON_EC2_NVME*") {
 $BlockDeviceName = (Get-EC2InstanceMetadata -Category
 "BlockDeviceMapping").ephemeral((Get-WmiObject -Class Win32_Diskdrive | Where-Object
 { $_.DeviceID -eq ("\\.\PHYSICALDRIVE" + $DiskDrive.Number) }).SCSIPort - 2)
 $BlockDevice = $null
 $VirtualDevice = ($VirtualDeviceMap.GetEnumerator() | Where-Object { $_.Value -eq
 $BlockDeviceName }).Key | Select-Object -First 1
}
ElseIf ($DiskDrive.path -like "*PROD_AMAZON*") {
 if ($DriveLetter -match '^[^a-zA-Z0-9]') {
 $i = 0
 While ($i -ne ($array3.Count)) {
 if ($array[2][$i] -eq $EbsVolumeID) {
 $DriveLetter = $array[0][$i]
 $DeviceName = $array[3][$i]
 }
 $i ++
 }
 }
}

```

```

}
$BlockDevice = ""
$BlockDeviceName = ($BlockDeviceMappings | Where-Object { $_.ebs.VolumeId -eq
$EbsVolumeID }).DeviceName
}
ElseIf ($DiskDrive.path -like "*NETAPP*") {
 if ($DriveLetter -match '^[a-zA-Z0-9]') {
 $i = 0
 While ($i -ne ($array3.Count)) {
 if ($array[2][$i] -eq $EbsVolumeID) {
 $DriveLetter = $array[0][$i]
 $DeviceName = $array[3][$i]
 }
 $i ++
 }
 }
 $EbsVolumeID = "FSxN Volume"
 $BlockDevice = ""
 $BlockDeviceName = ($BlockDeviceMappings | Where-Object { $_.ebs.VolumeId -eq
$EbsVolumeID }).DeviceName
}
Else {
 $BlockDeviceName = $null
 $BlockDevice = $null
}
New-Object PSObject -Property @{
 Disk = $Disk;
 Partitions = $Partitions;
 DriveLetter = If ($DriveLetter -eq $null) { "N/A" } Else { $DriveLetter };
 EbsVolumeId = If ($EbsVolumeID -eq $null) { "N/A" } Else { $EbsVolumeID };
 Device = If ($BlockDeviceName -eq $null) { "N/A" } Else
{ $BlockDeviceName };
 VirtualDevice = If ($VirtualDevice -eq $null) { "N/A" } Else { $VirtualDevice };
 VolumeName = If ($VolumeName -eq $null) { "N/A" } Else { $VolumeName };
 DeviceName = If ($DeviceName -eq $null) { "N/A" } Else { $DeviceName };
}
} | Sort-Object Disk | Format-Table -AutoSize -Property Disk, Partitions, DriveLetter,
EbsVolumeId, Device, VirtualDevice, DeviceName, VolumeName

```

執行指令碼，如下所示：

```
PS C:\> .\mapping.ps1
```

下列為範例輸出。

Disk DeviceName	Partitions	DriveLetter	EbsVolumeId VolumeName	Device	VirtualDevice
0	1	C	vol-0561f1783298efedd	/dev/sda1	N/A
NVMe Amazon Elastic B		N/A			
1	1	D	vol-002a9488504c5e35a	xvdb	N/A
NVMe Amazon Elastic B		N/A			
2	1	E	vol-0de9d46fcc907925d	xvdc	N/A
NVMe Amazon Elastic B		N/A			

如果您未在 Windows 執行個體上提供登入資料，則指令碼無法取得 EBS 磁碟區 ID，並會在 EbsVolumeId 欄中使用 N/A。

## 以 Windows VSS 為基礎的應用程式一致的 Amazon EBS 快照

### Note

以 Windows VSS 為基礎的應用程式一致性快照僅支援 Windows 執行個體。

您可以使用執 [AWS Systems Manager 行](#) 命令，為連接到 Amazon EC2 Windows 執行個體的所有 Amazon EBS 磁碟區拍攝應用程式一致的快照。快照程序會使用 Windows [磁碟區陰影複製服務 \(VSS\)](#) 來取得 VSS 感知應用程式的 EBS 磁碟區層級備份。快照包括來自這些應用程式和磁碟之間擱置中交易的資料。若需要備份所有連接的磁碟區，您不需要關閉執行個體或中斷其連結。

使用以 VSS 為基礎的 EBS 快照無需額外費用。您只需為備份程序所建立的 EBS 快照支付費用。[如需詳細資訊，請參閱我的 Amazon EBS EBS 快照如何計費？](#)

### 目錄

- [什麼是 IVS？](#)
- [必要條件](#)
- [建立具備 VSS 功能的 EBS 快照](#)
- [疑難排解以視窗 VSS 為基礎的 EBS 快照](#)
- [從具備 VSS 功能的 EBS 快照中還原 EBS 磁碟區](#)
- [AWS VSS 解決方案版本紀錄](#)

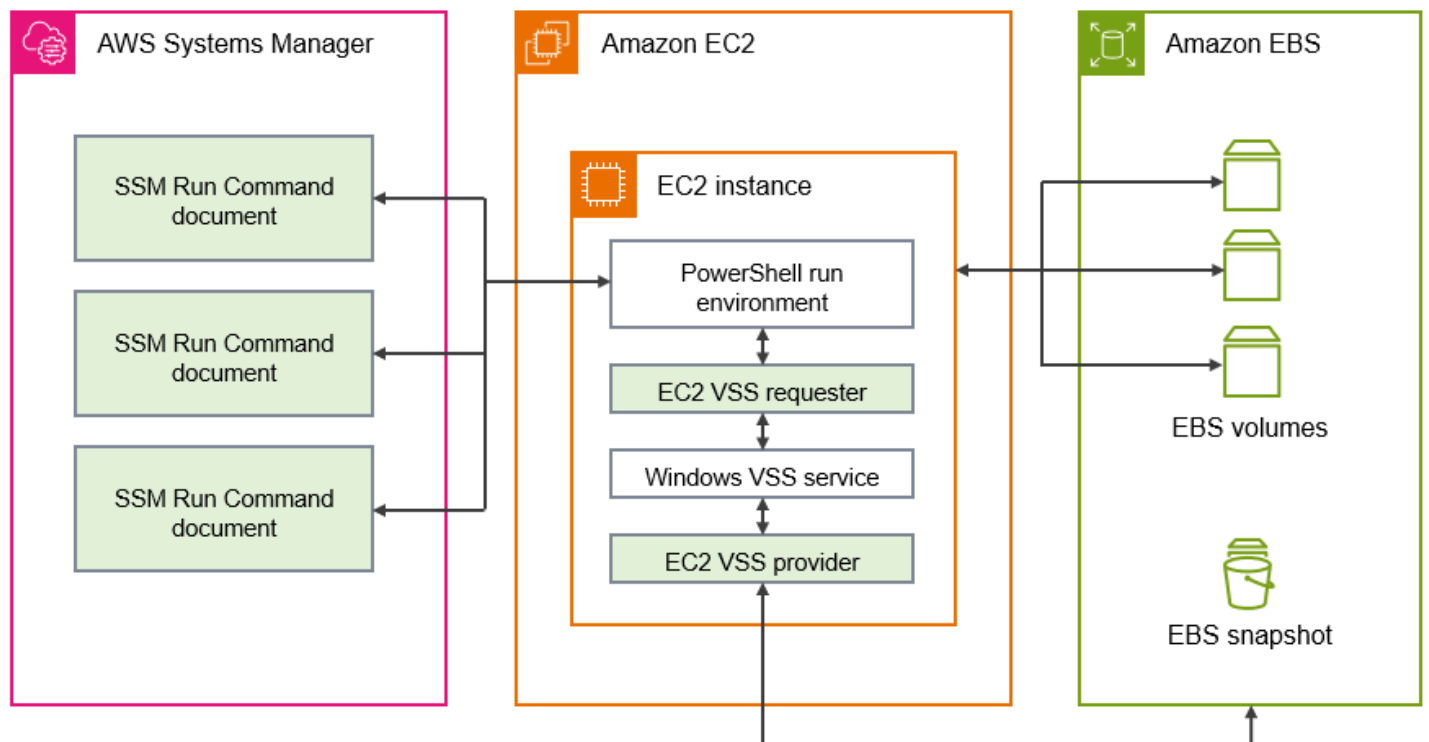
## 什麼是 IVS ？

磁碟區快照複製服務 (VSS) 是一種包含在 Microsoft Windows 中的備份和復原技術。當正在使用電腦檔案或磁碟區時，其可建立其備份複本或快照。如需詳細資訊，請參閱[磁碟區陰影複製服務](#)。

若要建立應用程式一致性快照，會涉及以下軟體元件。

- VSS 服務 – Windows 作業系統的一部分
- VSS 請求程式 – 請求建立陰影複製的軟體
- VSS 寫入程式 – 通常是作為應用程式 (例如 SQL Server) 的一部分提供，以確保要備份的資料集一致
- VSS 供應程式 – 建立基礎磁碟區陰影複本的元件

以 Windows VSS 為基礎的 Amazon EBS 快照解決方案包含多個可協助備份建立的 Systems Manager (SSM) 執行命令文件，以及一個稱為[AwsVssComponents](#)系統管理員代理商套件，其中包括 EC2 VSS 請求者和 EC2 VSS 提供者。此 AwsVssComponents 套件必須安裝在 EC2 Windows 執行個體上，才能擷取 EBS 磁碟區的應用程式一致性快照。下圖說明這些軟體元件之間的關係。



### 以 VSS 為基礎的 Amazon EBS 快照解決方案如何運作

採取應用程式一致性、以 VSS 為基礎的 EBS 快照指令碼的程序包含下列步驟。

1. 完成[必要條件](#)。
2. 輸入 AWSEC2-VssInstallAndSnapshot SSM 文件的參數，然後使用執行命令來執行此文件。如需詳細資訊，請參閱 [運行 AWSEC 2-VssInstallAndSnapshot 命令文檔 \(推薦\)](#)。
3. 執行個體上的 Windows VSS 服務會針對執行中的應用程式，協調所有正在進行的 I/O 操作。
4. 系統會排清所有 I/O 緩衝區，並暫時暫停所有 I/O 操作。暫停時間最多持續 10 秒。
5. 在暫停期間，系統會為連接至執行個體的所有磁碟區建立快照。
6. 系統會取消暫停，並恢復 I/O 操作。
7. 系統會將所有新建立的快照新增至 EBS 快照清單。系統會使用: true 標記此程序成功建立的所有已啟用 VSS 的 EBS 快照。AppConsistent
8. 如果您需要從快照還原，則可採用從快照建立磁碟區的標準 EBS 程序。或者，您能夠透過範例指令碼將所有磁碟區還原至執行個體 (如 [從具備 VSS 功能的 EBS 快照中還原 EBS 磁碟區](#) 中所述)。

## 必要條件

您可以使用系統管理員執行命令或 Amazon 資料生命週期管理員建立 VSS 型 EBS 快照。AWS Backup 以下先決條件適用所有解決方案。

### 必要條件

- [系統要求](#)
- [IAM 許可](#)
- [VSS 元件](#)

## 系統要求

### 安裝系統管理員代理程式

VSS 是由 AWS Systems Manager (Systems Manager) 使用 PowerShell 請確保已在 EC2 執行個體上安裝 SSM Agent 3.0.502.0 版或更高版本。如果已在舊版 SSM Agent，則可利用 Run Command 進行更新。如需詳細資訊，請參閱《AWS Systems Manager 使用者指南》中的[針對 Amazon EC2 執行個體設定 Systems Manager](#) 以及[在適用於 Windows Server 的 Amazon EC2 執行個體上使用 SSM Agent](#)。

### Amazon EC2 Windows 執行個體需求

執行 Windows 伺服器 2012 及更新版本的執行個體支援以 VSS 為基礎的 EBS 快照集。如需較舊版本的 Windows，請參閱 [AWS VSS 解決方案版本紀錄](#) 中的 Windows 版本支援表。

## .NET Framework 版本

AwsVssComponents 套件需要 .NET Framework 版本 4.6 或更高版本。視窗伺服器 2016 年之前的 Windows 作業系統版本預設為舊版的 .NET 架構。如果您的執行個體使用舊版的 .NET 架構，您必須使用 Windows Update 安裝 4.6 或更新版本。

## AWS Tools for Windows PowerShell 版本

確定您的執行個體執行 AWS Tools for Windows PowerShell 版本 3.3.48.0 或更新版本。若要檢查您的版本，請在執行個體的 PowerShell 終端機中執行下列命令。

```
C:\> Get-AWSPowerShellVersion
```

如果您需要更新 AWS Tools for Windows PowerShell 執行個體，請參閱《AWS Tools for Windows PowerShell 使用者指南》AWS Tools for Windows PowerShell 中的〈[安裝](#)〉。

## 視窗 PowerShell 版

確定您的執行個體執行的是 Windows PowerShell 主要版本 3、4 或 5。若要檢查您的版本，請在執行個體的 PowerShell 終端機中執行下列命令。

```
C:\> $PSVersionTable.PSVersion
```

## PowerShell 語言模式

確定您的執行個體已將 PowerShell 語言模式設定為 FullLanguage。如需詳細資訊，請參閱 Microsoft 文件中的 [about\\_Language\\_Modes](#)。

## IAM 許可

附加到 Amazon EC2 Windows 執行個體的 IAM 角色必須具有使用 VSS 建立應用程式一致快照的權限。若要授與必要的權限，您可以將 AWSEC2VssSnapshotPolicy 原則附加至您的執行個體設定檔。

此原則可讓 Systems Manager 執行下列動作：

- 建立和標記 EBS 快照
- 建立和標記 Amazon 機器映像 (AMI)
- 將中繼資料 (例如裝置 ID) 附加至 VSS 建立的預設快照標籤。

## 主題



- [將啟用 VSS 的快照政策附加到您的執行個體設定檔](#)
- [用於建立 VSS 快照的受管理原則](#)
- [舊版原則 \(不再支援\)](#)

將啟用 VSS 的快照政策附加到您的執行個體設定檔

若要授與執行個體啟用 VSS 之快照的權限，您可以將AWSEC2VssSnapshotPolicy受管政策附加到執行個體設定檔角色，如下所示。確保您的執行個體符合所有項目是非常重要的[系統要求](#)。

#### Note

若要使用受管理政策，您的執行個體必須安裝AwsVssComponents套件版本2.3.1或更新版本。如需版本歷程記錄，請參閱[AwsVssComponents 套件版本](#)。  
如果您的執行個體上安裝了先前版本的AwsVssComponents套件，請參閱[舊版政策](#)。

1. 前往 <https://console.aws.amazon.com/iam/> 開啟 IAM 主控台。
2. 在導覽窗格中，選擇 [角色] 以查看您有權存取的 IAM 角色清單。
3. 選取連結至執行個體之角色的角色名稱連結。這會開啟角色詳細資料頁面。
4. 若要附加受管理的策略，請選擇位於清單面板右上角的 [新增權限]。然後從下拉式清單中選取 [附加原則]。
5. 若要簡化結果，請在搜尋列 (AWSEC2VssSnapshotPolicy) 中輸入策略名稱。
6. 選取要附加之原則名稱旁的核取方塊，然後選擇 [新增權限]。

用於建立 VSS 快照的受管理原則

受 AWS 管政策是 Amazon 為 AWS 客戶提供的獨立政策。AWS 受管理的原則是為了授與常見使用案例的權限而設計。您無法變更受 AWS 管理原則中定義的權限。不過，您可以複製政策，並將其用作您使用案例特定之[客戶管理策略](#)的基準。

如需 AWS 受管政策的詳細資訊，請參閱 IAM 使用者指南中的[AWS 受管政策](#)。

若要使用受管AWSEC2VssSnapshotPolicy政策，您可以將該政策附加到附加到 EC2 Windows 執行個體的 IAM 角色。此政策可讓 EC2 VSS 解決方案建立和新增標籤至 Amazon 機器映像 (AMI) 和 EBS 快照。若要附加原則，請參閱[將啟用 VSS 的快照政策附加到您的執行個體設定檔](#)。

## AWSEC2VssSnapshotPolicy 授予的許可

該AWSEC2VssSnapshotPolicy政策包括以下 Amazon EC2 許可：

- ec2: CreateTags — 將標籤新增至 EBS 快照和 AMI，以協助識別和分類資源。
- ec2 : DescribeInstance屬性 — 擷取附加至目標執行個體的 EBS 磁碟區和對應的區塊裝置對映。
- ec2: CreateSnapshots — 建立 EBS 磁碟區的快照。
- ec2 : CreateImage— 從正在運行的 EC2 實例創建一個 AMI。
- ec2: DescribeImages — 擷取 EC2 AMI 和快照的資訊。
- ec2: DescribeSnapshots — 決定快照的建立時間和狀態，以驗證應用程式的一致性。

## 政策範例

以下是AWSEC2VssSnapshotPolicy策略的範例。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "DescribeInstanceInfo",
 "Effect": "Allow",
 "Action": [
 "ec2:DescribeInstanceAttribute"
],
 "Resource": [
 "arn:aws:ec2:*:*:instance/*"
],
 "Condition": {
 "StringLike": {
 "ec2:SourceInstanceARN": "*${ec2:InstanceId}"
 }
 }
 },
 {
 "Sid": "CreateSnapshotsWithTag",
 "Effect": "Allow",
 "Action": [
 "ec2:CreateSnapshots"
],
 "Resource": [
 "arn:aws:ec2:*:*:snapshot/*"
]
 }
]
}
```

```

],
 "Condition": {
 "StringLike": {
 "aws:RequestTag/AwsVssConfig": "*"
 }
 }
 },
 {
 "Sid": "CreateSnapshotsAccessInstance",
 "Effect": "Allow",
 "Action": [
 "ec2:CreateSnapshots"
],
 "Resource": [
 "arn:aws:ec2:*:*:instance/*"
],
 "Condition": {
 "StringLike": {
 "ec2:SourceInstanceARN": "*${ec2:InstanceId}"
 }
 }
 },
 {
 "Sid": "CreateSnapshotsAccessVolume",
 "Effect": "Allow",
 "Action": [
 "ec2:CreateSnapshots"
],
 "Resource": [
 "arn:aws:ec2:*:*:volume/*"
]
 },
 {
 "Sid": "CreateImageWithTag",
 "Effect": "Allow",
 "Action": [
 "ec2:CreateImage"
],
 "Resource": [
 "arn:aws:ec2:*:*:snapshot/*",
 "arn:aws:ec2:*:*:image/*"
],
 "Condition": {
 "StringLike": {

```

```

 "aws:RequestTag/AwsVssConfig": "*"
 }
}
},
{
 "Sid": "CreateImageAccessInstance",
 "Effect": "Allow",
 "Action": [
 "ec2:CreateImage"
],
 "Resource": [
 "arn:aws:ec2:*:*:instance/*"
],
 "Condition": {
 "StringLike": {
 "ec2:SourceInstanceARN": "*${ec2:InstanceId}"
 }
 }
},
{
 "Sid": "CreateTagsOnResourceCreation",
 "Effect": "Allow",
 "Action": "ec2:CreateTags",
 "Resource": [
 "arn:aws:ec2:*:*:snapshot/*",
 "arn:aws:ec2:*:*:image/*"
],
 "Condition": {
 "StringEquals": {
 "ec2:CreateAction": [
 "CreateImage",
 "CreateSnapshots"
]
 }
 }
},
{
 "Sid": "CreateTagsAfterResourceCreation",
 "Effect": "Allow",
 "Action": "ec2:CreateTags",
 "Resource": [
 "arn:aws:ec2:*:*:snapshot/*",
 "arn:aws:ec2:*:*:image/*"
],
}

```

```

 "Condition": {
 "StringLike": {
 "ec2:ResourceTag/AwsVssConfig": "*"
 },
 "ForAllValues:StringEquals": {
 "aws:TagKeys": [
 "AppConsistent",
 "Device"
]
 }
 },
 {
 "Sid": "DescribeImagesAndSnapshots",
 "Effect": "Allow",
 "Action": [
 "ec2:DescribeImages",
 "ec2:DescribeSnapshots"
],
 "Resource": "*"
 }
]
}

```

### 簡化特定使用案例的權限 (進階)

受AWSEC2VssSnapshotPolicy管理的原則包含您可以建立啟用 VSS 之快照的所有方式的權限。您可以建立僅包含所需權限的自訂原則。

使用案例：建立 AMI，使用案例：使用 AWS Backup 服務

如果您專門使用此CreateAmi選項，或者僅透過 AWS Backup 服務建立啟用 VSS 的快照，則可以依照下列方式簡化原則陳述式。

- 省略下列陳述式識別碼 (SID) 所識別的原則陳述式：
  - CreateSnapshotsWithTag
  - CreateSnapshotsAccessInstance
  - CreateSnapshotsAccessVolume
- 調整CreateTagsOnResourceCreation語句，如下所示：
  - arn:aws:ec2:\*:\*:snapshot/\*從資源中移除。
  - CreateSnapshots從ec2:CreateAction條件中移除。

- 調整CreateTagsAfterResourceCreation陳述式以arn:aws:ec2:\*:\*:snapshot/\*從資源中移除。
- 調整DescribeImagesAndSnapshots陳述式以ec2:DescribeSnapshots從陳述式動作中移除。

使用案例：僅限快照

如果您不使用該CreateAmi選項，則可以按照以下方式簡化政策聲明。

- 省略下列陳述式識別碼 (SID) 所識別的原則陳述式：
  - CreateImageAccessInstance
  - CreateImageWithTag
- 調整CreateTagsOnResourceCreation語句，如下所示：
  - arn:aws:ec2:\*:\*:image/\*從資源中移除。
  - CreateImage從ec2:CreateAction條件中移除。
- 調整CreateTagsAfterResourceCreation陳述式以arn:aws:ec2:\*:\*:image/\*從資源中移除。
- 調整DescribeImagesAndSnapshots陳述式以ec2:DescribeImages從陳述式動作中移除。

#### Note

為了確保您的自訂政策能如預期般執行，建議您定期檢閱受管理原則並加入更新。

舊版原則 (不再支援)

授予啟用 VSS 之快照權限的舊版政策包括在發行AWSEC2VssSnapshotPolicy受管政策之前建議的IAM 許可。

如果您已使用舊版政策設定執行個體角色，則可以繼續使用它。不過，為了確保您的政策依據最新的 IAM 最佳做法和範圍政策聲明保持最新狀態，我們建議您將舊版政策取代為AWSEC2VssSnapshotPolicy受管政策。

政策範例

下列政策範例使用 `ec2:DescribeInstanceAttributeAwsVssComponents` 套件 2.2.1 及更新版本所支援的。如果您已安裝較舊版本的 `AwsVssComponents` 套件，則應使用 `ec2:DescribeInstances` 動作取代該套件。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "ec2:CreateTags",
 "Resource": [
 "arn:aws:ec2:*::snapshot/*",
 "arn:aws:ec2:*::image/*"
]
 },
 {
 "Effect": "Allow",
 "Action": [
 "ec2:DescribeInstanceAttribute",
 "ec2:CreateSnapshot",
 "ec2:CreateSnapshots",
 "ec2:CreateImage",
 "ec2:DescribeImages",
 "ec2:DescribeSnapshots"
],
 "Resource": "*"
 }
]
}
```

如需 IAM 受管政策的詳細資訊，請參閱 IAM 使用者指南中的 [AWS 受管政策](#)。

## VSS 元件

若要在 Windows 作業系統上建立應用程式一致快照，則必須在執行個體上安裝 `AwsVssComponents` 套件。該套件包含執行個體上的 EC2 VSS 代理程式，它可作為 EBS 磁碟區的 VSS 請求程式和 EC2 VSS 供應程式。

有幾種方式可以將元件安裝到現有執行個體上：

- (建議) [運行 AWSEC 2-VssInstallAndSnapshot 命令文檔 \(推薦\)](#)。這會在每次執行時視需要自動安裝或更新。

- [在執行個體上手動安裝 VSS 元件](#).
- [依照排程更新執行個體上的 VSS 元件](#).

您也可以使用 EC2 Image Builder 建立 AMI，以使用 `aws-vss-components-windows` 受管元件安裝映像的 `AwsVssComponents` 套件。受管理元件會使用「AWS Systems Manager 散發者」來安裝套件。Image Builder 建立映像後，您從相關聯 AMI 啟動的每個執行個體都會安裝 VSS 套件。如需如何在安裝 VSS 套件的情況下建立 AMI 的詳細資訊，請參閱 EC2 Image Builder 使用者指南中的 [Windows 的 Distributor 套件受管元件](#)。

## 目錄

- [在執行個體上手動安裝 VSS 元件](#)
- [依照排程更新執行個體上的 VSS 元件](#)

## 在執行個體上手動安裝 VSS 元件

您的 EC2 Windows 執行個體須已安裝 VSS 元件，您才能使用 Systems Manager 建立應用程式一致快照。如果您不執行 `AWSEC2-VssInstallAndSnapshot` 命令文件，以便在每次建立應用程式一致快照時自動安裝或更新套件，則必須手動安裝套件。

如果您打算使用以下其中一種方法，從 EC2 執行個體建立應用程式一致快照，也必須手動安裝。

- 使用建立 VSS 快照 AWS Backup
- 使用 Amazon Data Lifecycle Manager 建立 VSS 快照

如果您需要執行手動安裝，建議您使用最新的 AWS VSS 元件套件，以提高 EC2 Windows 執行個體上應用程式一致性快照的可靠性和效能。

### Note

若要在建立應用程式一致快照時自動安裝或更新 `AwsVssComponents` 套件，建議您使用 Systems Manager 來執行 `AWSEC2-VssInstallAndSnapshot` 文件。如需詳細資訊，請參閱 [運行 AWSEC 2-VssInstallAndSnapshot 命令文檔 \(推薦\)](#)。

若要在 Amazon EC2 Windows 執行個體上安裝 VSS 元件，請依照適用您偏好之環境的步驟執行。



## Console

使用 SSM Distributor 安裝 VSS 元件

1. [請在以下位置開啟 AWS Systems Manager 主控台。](https://console.aws.amazon.com/systems-manager/) <https://console.aws.amazon.com/systems-manager/>
2. 在導覽窗格中，選擇 執行命令。
3. 選擇 執行命令。
4. 對於命令文件，請選擇 AWS AWSPackage 設定旁邊的按鈕。
5. 在 Command parameters (命令參數) 中，執行以下操作：
  - a. 驗證 Action (動作) 設為 Install (安裝)。
  - b. 對於名稱，輸入 AwsVssComponents。
  - c. 在版本中，輸入版本或將欄位留空，以便 Systems Manager 安裝最新版本。
6. 在 Targets (目標) 中，手動指定標籤或選取執行個體，以識別您要執行這項操作的執行個體。

### Note


如果您選取手動選取執行個體，但您預期看到的執行個體並未出現在清單上，請參閱 <https://docs.aws.amazon.com/systems-manager/latest/userguide/troubleshooting-remote-commands.html#where-are-instances> 使用者指南中的 AWS Systems Manager 我的執行個體在哪裡？以取得故障診斷秘訣。

7. 對於 Other parameters (其他參數)：
  - (選用) 在 Comment (註解) 中，輸入此命令的相關資訊。
  - 在 Timeout (seconds) (逾時 (秒)) 中，指定在命令執行全面失敗之前，系統要等候的秒數。
8. (選用) 在 Rate control (速率控制) 中：
  - 在 Concurrency (並行) 中，指定可同時執行命令的執行個體數目或百分比。

### Note

如果您已透過選取 Amazon EC2 標籤來選取目標，而且不確定有多少執行個體會使用所選的標籤，請指定百分比來限制可同時執行文件的執行個體數目。

- 在 Error threshold (錯誤閾值) 中，指定在特定數目或百分比之執行個體上的命令失敗之後，停止在其他執行個體上執行命令。例如，如果您指定三個錯誤，則 Systems Manager 會在收到第四個錯誤時停止傳送命令。仍在處理命令的執行個體也可能會傳送錯誤。
9. (選用) 在 Output options (輸出選項) 區段中，如果您想要將命令輸出儲存至檔案，請選取 Enable writing to an S3 bucket (啟用寫入至 S3 儲存貯體) 旁的按鈕。指定儲存貯體和 (選用) 字首 (資料夾) 名稱。

 Note

授予能力以將資料寫入至 S3 儲存貯體的 S3 許可，會是指派給執行個體之執行個體描述檔的許可，而不是執行此工作之使用者的許可。如需詳細資訊，請參閱《AWS Systems Manager 使用者指南》中的[建立 Systems Manager 的 IAM 執行個體設定檔](#)。

10. (選用) 指定 SNS notifications (SNS 通知) 的選項。

如需針對執行命令設定 Amazon SNS 通知的資訊，請參閱[針對 AWS Systems Manager 設定 Amazon SNS 通知](#)。

11. 選擇執行。

## AWS CLI

透過以下程序，即可使用 AwsVssComponents 中的執行命令來下載 AWS CLI 套件，並在執行個體上進行安裝。該套件安裝兩個元件：VSS 請求程式和 VSS 供應程式。系統會將這些元件複製到執行個體上的目錄，接著註冊供應程式 DLL，以將其做為 VSS 供應程式。

若要使用安裝 VSS 套件 AWS CLI

- 執行下列命令，藉此下載並安裝 Systems Manager 所需的 VSS 元件。

```
aws ssm send-command \
 --document-name "AWS-ConfigureAWSPackage" \
 --instance-ids "i-01234567890abcdef" \
 --parameters '{"action":["Install"],"name":["AwsVssComponents"]}'
```

## PowerShell

使用下列程序，使用 Windows 工具中的執行命令，在執行個體上下載並安裝 `AwsVssComponents` 套件 PowerShell。該套件安裝兩個元件：VSS 請求程式和 VSS 供應程式。系統會將這些元件複製到執行個體上的目錄，接著註冊供應程式 DLL，以將其做為 VSS 供應程式。

若要使用安裝 VSS 套件 AWS Tools for Windows PowerShell

- 執行下列命令，藉此下載並安裝 Systems Manager 所需的 VSS 元件。

```
Send-SSMCommand -DocumentName AWS-ConfigureAWSPackage -InstanceId
"i-01234567890abcdef" -Parameter
{'action'='Install';'name'='AwsVssComponents'}
```

### 驗證 AWS VSS 元件上的簽章

請使用下列程序驗證 `AwsVssComponents` 套件上的簽章。

- 連接至 Windows 執行個體。如需詳細資訊，請參閱 [連線至 Windows 執行個體](#)。
- 瀏覽至 `C:\Program 檔案\Amazon\AwsVss` 元件。
- 開啟 `ec2-vss-agent.exe` 的內容選單 (按一下右鍵)，然後選擇屬性。
- 導覽至數位簽章索引標籤，並確認簽署者的名稱是否為 Amazon Web Services Inc.
- 使用上述步驟驗證 `Ec2VssInstaller` 和 `Ec2VssProvider.dll` 上的簽章。

### 依照排程更新執行個體上的 VSS 元件

建議維持將 VSS 元件更新為最新的建議版本。在新版 `AwsVssComponents` 套件發行時，您可以透過多種不同方式更新元件。

### 更新方法

- 您可以重複發行新版本的 AWS VSS 元件 [在執行個體上手動安裝 VSS 元件](#) 時所述的步驟。
- 您可以將 Systems Manager State Manager 關聯設定為在 `AwsVssComponents` 套件可用時自動下載並安裝全新或更新後的 VSS 元件。
- 您可以在使用 Systems Manager 執行 `AWSEC2-VssInstallAndSnapshot` 文件時，隨時在建立應用程式一致快照時自動安裝或更新 `AwsVssComponents` 套件。

**Note**

建議您使用 Systems Manager 執行 `AwsVssComponents` 命令文件，其會在建立應用程式一致快照時，自動安裝或更新 `AWSEC2-VssInstallAndSnapshot` 套件。如需詳細資訊，請參閱 [運行 AWSEC 2-VssInstallAndSnapshot 命令文檔 \(推薦\)](#)。

若要建立 Systems Manager State Manager 關聯，請依照適用於您偏好之環境的步驟執行。

**Console**

使用主控台建立 State Manager 關聯

1. [請在以下位置開啟 AWS Systems Manager 主控台](https://console.aws.amazon.com/systems-manager/)。 <https://console.aws.amazon.com/systems-manager/>
2. 在導覽窗格中，選擇 State Manager (狀態管理員)。

或者，如果 Systems Manager 首頁先開啟，請開啟導覽窗格，然後選擇 State Manager。


3. 選擇 Create association (建立關聯)。
4. 在 Name (名稱) 欄位中，輸入描述性名稱。
5. 在 [文件] 清單中，選擇 [AWS-AWSPackage 設定]。
6. 在 Parameters (參數) 區段中，從 Action (動作) 清單中選擇 Install (安裝)。
7. 對於 Installation type (安裝類型)，選擇 Uninstall and reinstall (解除安裝並重新安裝)。
8. 在 Name (名稱) 欄位中，輸入 `AwsVssComponents`。可將 Version (版本) 和 Additional Arguments (其他引數) 保留為空白。
9. 在 Targets (目標) 區段中，選擇一個選項。

**Note**

如果您選擇使用標籤將執行個體設為目標，且指定映射到 Linux 執行個體的標籤，則關聯會在 Windows 執行個體上執行成功，但在 Linux 執行個體上執行失敗。關聯的整體狀態會顯示為 Failed (失敗)。

10. 在 Specify schedule (指定排程) 區段中，選擇一個選項。
11. 在 Advanced options (進階選項) 區段中，針對 Compliance severity (合規嚴重性)，選擇關聯的嚴重性等級。如需詳細資訊，請參閱[關於 State Manager 關聯合規性](#)。針對變更行事曆，請選取預先設定的變更行事曆。如需詳細資訊，請參閱[AWS Systems Manager 變更行事曆](#)。

12. 針對比率控制，請執行下列操作：
  - 在 Concurrency (並行) 中，指定可同時執行命令的受管節點數目或百分比。
  - 在 Error threshold (錯誤閾值) 中，指定在特定數目或百分比之節點上的命令失敗之後，停止在其他受管節點上執行命令。
13. (選用) 針對輸出選項，若要將命令輸出儲存至檔案，請選取啟用將輸出寫入 S3。在方塊中輸入儲存貯體和字首 (資料夾) 名稱。
14. 選擇 Create association (建立關聯)，接著選擇 Close (關閉)。系統會嘗試在執行個體上建立關聯，並立即套用狀態。

 Note

如果適用於 Windows Server 的 EC2 執行個體顯示 [失敗] 狀態，請確認 SSM 代理程式正在執行個體上執行，並確認執行個體已設定為 Systems Manager 的 AWS Identity and Access Management (IAM) 角色。如需詳細資訊，請參閱[設定 AWS Systems Manager](#)。

## AWS CLI

您可以執行[建立關聯 AWS CLI 命令](#)，[安排更新散發](#)者套件，而不必讓相關應用程式離線。只會取代套件中的新檔案或更新檔案。

若要使用建立「狀態管理員」關聯 AWS CLI

1. 安裝和配置 AWS CLI, 如果你還沒有. 如需相關資訊，請參閱[安裝或更新最新版本的 AWS CLI](#)。
2. 執行下列命令以建立關聯。--name 值 (文件名稱) 一律為 AWS-ConfigureAWSPackage。以下命令會使用索引鍵 InstanceIds 來指定目標執行個體。

```
aws ssm create-association \
 --name "AWS-ConfigureAWSPackage" \
 --parameters '{"action":["Install"],"installationType":["Uninstall and reinstall"],"name":["AwsVssComponents"]}' \
 --targets [{"Key\":\"InstanceIds\",\"Values\":[\"i-01234567890abcdef\", \"i-000011112222abcde\"}]}
```

若要取得有關可與指令配合使用的其他選項的資訊，請參閱《[create-association指令參考](#)》— AWS Systems Manager 節中的 [〈建立關聯〉](#)。AWS CLI

## 建立具備 VSS 功能的 EBS 快照

本節包含建立啟用 VSS 之 EBS 快照的步驟。

可以為附接至 EC2 執行個體的 EBS 磁碟區建立具備 VSS 功能的 EBS 快照。嘗試建立具備 VSS 功能的快照之前，請確保符合 [必要條件](#)。

### 主題

- [使用 AWS Systems Manager 命令文件建立 VSS 快照](#)
- [使用建立 VSS 快照 AWS Backup](#)
- [使用 Amazon Data Lifecycle Manager 建立 VSS 快照](#)

## 使用 AWS Systems Manager 命令文件建立 VSS 快照

您可以使用 AWS Systems Manager 命令文件建立啟用 VSS 的快照。下列內容會介紹可用的命令文件，以及文件用來建立快照的執行期參數。

在您使用任何 Systems Manager 命令文件之前，請確保您已符合所有 [必要條件](#)。

### 主題

- [Systems Manager VSS 快照文件的參數](#)
- [執行 Systems Manager VSS 快照命令文件](#)

## Systems Manager VSS 快照文件的參數

建立 VSS 快照的 Systems Manager 文件全部都會使用下列參數，除非另有說明：

ExcludeBoot體積 ( 字符串，可選 )

若您建立快照，此設定會從備份程序中排除開機磁碟區。若要從快照中排除開機磁碟區，請True將 [ExcludeBoot磁碟區] 設定CreateAmi為和False。

如果您為備份建立 AMI，則應將此參數設定為 False。此參數的預設值為 False。

### NoWriters ( 字符串, 可選 )

若要從快照程序中排除應用程式 VSS 寫入程式, 請將此參數設定為 True。排除應用程式 VSS 寫入程式有助於您解決與第三方 VSS 備份元件的衝突。此參數的預設值為 False。

### CopyOnly ( 字符串, 可選 )

如果除了 AWS VSS 之外還使用原生 SQL Server 備份, 則執行僅複製備份可防止 AWS VSS 中斷原生差異備份鏈結。若要執行僅限複製備份操作, 請將此參數設定為 True。

此參數的預設值為 False, 這會導致 AWS VSS 執行完整備份作業。

### CreateAmi ( 字符串, 可選 )

若要建立啟用 VSS 的 Amazon Machine Image (AMI) 來備份您的執行個體, 請將此參數設定為 True。此參數的預設值為 False, 因此會改為使用 EBS 快照備份您的執行個體。

如需從執行個體建立 AMI 的詳細資訊, 請參閱[創建一個 Amazon EBS-backed AMI](#)。

### AmiName ( 字符串, 可選 )

如果選 CreateAmi 項設定為 True, 請指定備份建立的 AMI 名稱。

### description ( 字串, 選用 )

指定此程序所建立之快照或映像的描述。

### tags ( 字串, 選用 )

我們建議您為快照和映像加上標籤, 以協助您尋找和管理資源, 例如從快照清單還原磁碟區。系統會新增 Name 金鑰, 其中包含空白值, 您可以在其中指定要套用至輸出快照或影像的名稱。

如果要指定其他標籤, 請在兩者之間以分號分隔標籤。例如  
Key=Environment,Value=Test;Key=User,Value=TestUser1。

依預設, 系統會為啟用 VSS 的快照和映像新增下列保留標記。

- 裝置 — 對於啟用 VSS 的快照, 這是快照擷取之 EBS 磁碟區的裝置名稱。
- AppConsistent — 此標籤表示已成功建立啟用 VSS 的快照或 AMI。
- AwsVssConfig — 這可識別在啟用 VSS 的情況下建立的快照和 AMI。該標籤包括元信息, 例如 AwsVssComponents 版本。

#### Warning

在參數清單中指定這些保留標籤中的任何一個都會導致錯誤。

## executionTimeout (字串, 選用)

指定在執行個體上執行快照建立程序或從執行個體建立 AMI 的時間上限 (以秒為單位)。增加此逾時時間可讓命令等待更長時間, 使 VSS 開始凍結並完成標記其建立的資源。此逾時僅適用於快照或 AMI 建立步驟。安裝或更新 AwsVssComponents 套件的初始步驟不包含在逾時中。

## CollectDiagnostic日誌 ( 字串, 可選 )

若要在快照和 AMI 建立步驟期間收集更多資訊, 請將此參數設定為 True ""。此參數的預設值為 "False"。合併的診斷記錄會以 .zip 格式封存格式儲存在執行個體的下列位置:

```
C:\ProgramData\Amazon\AwsVss\Logs\timestamp.zip
```

## VssVersion ( 字串, 可選 )

在使用 AWSEC2-VssInstallAndSnapshot 文件的情況下, 您可以指定 VssVersion 參數以在執行個體上安裝特定版本的 AwsVssComponents 套件。將此參數保留空白, 以安裝建議的預設版本。

如果已安裝指定版本的 AwsVssComponents 套件, 則指令碼會略過安裝步驟, 並繼續進行備份步驟。如需 AwsVssComponents 套件版本和操作支援的清單, 請參閱 [AWS VSS 解決方案版本紀錄](#)。

## 執行 Systems Manager VSS 快照命令文件

您可以使用命令文件建立啟用 VSS 的 EBS 快照, 如下所 AWS Systems Manager 示。

## 運行 AWSEC 2-VssInstallAndSnapshot 命令文檔 ( 推薦 )

當您使用執 AWS Systems Manager 行 AWSEC2-VssInstallAndSnapshot 文件時, 指令碼會執行下列步驟。

1. 該指令碼會先在執行個體上安裝或更新 AwsVssComponents 套件, 具體取決於是否已安裝執行個體。
2. 指令碼會在完成第一個步驟後, 建立應用程式一致快照。

若要執行 AWSEC2-VssInstallAndSnapshot 文件, 請依照適用於您偏好之環境的步驟執行。



## Console

從主控台建立啟用 VSS 的 EBS 快照

1. [請在以下位置開啟 AWS Systems Manager 主控台。](https://console.aws.amazon.com/systems-manager/) <https://console.aws.amazon.com/systems-manager/>
2. 從導覽窗格中選取執行命令。這會顯示您帳戶中目前正在執行的命令清單 (若適用)。
3. 選擇 執行命令。這會開啟您具有存取權限的命令文件清單。
4. 從命令文件清單中選取 AWSEC2-VssInstallAndSnapshot。若要簡化結果，可輸入全部或部分文件名稱。您也可以依擁有者、平台類型或標籤進行篩選。

在您選取命令文件時，系統會將詳細資訊填入清單下方。

5. 從文件版本清單中選取 Default version at runtime。
6. 設定命令參數，以定義 AWSEC2-VssInstallAndSnapshot 如何安裝 AwsVssComponents 套件並使用 VSS 快照或 AMI 來備份。如需參數詳細資訊，請參閱 [Systems Manager VSS 快照文件的參數](#)。
7. 針對目標選取，手動指定標籤或選取執行個體，以識別要執行這項操作的執行個體。

### Note

如果您手動選取執行個體，但預期看到的執行個體並未出現在清單中，請參閱[我的執行個體在哪裡？](#)，以取得疑難排解秘訣。

8. 如需定義 Systems Manager Run Command 行為的其他參數 (例如速率控制)，請依照[從主控台執行命令](#)中的說明輸入值。
9. 選擇 Run (執行)。

如果操作成功，該命令就會將新快照填入 EBS 快照清單中。您可以搜尋指定的標籤，或是搜尋 AppConsistent，藉此在 EBS 快照清單中找到這些快照。如果命令執行失敗，請檢視 Systems Manager 命令輸出的詳細資訊，以了解執行失敗的原因。若系統成功完成命令，但無法備份特定磁碟區，則您可以在 EBS 磁碟區清單中進行疑難排解。

## AWS CLI

您可以在中執行下列命令，AWS CLI 以建立啟用 VSS 的 EBS 快照並取得快照建立的狀態。

建立具備 VSS 功能的 EBS 快照

執行下列命令，藉此建立具備 VSS 功能的 EBS 快照。若要建立快照，您必須使用 `--instance-ids` 參數來識別執行個體。如需您可以查看之其他參數的詳細資訊，請參閱 [Systems Manager VSS 快照文件的參數](#)。

```
aws ssm send-command \
 --document-name "AWSEC2-VssInstallAndSnapshot" \
 --instance-ids "i-01234567890abcdef" \
 --parameters '{"ExcludeBootVolume":["False"],"description":["Description"],"tags":
["Key=key_name,Value=tag_value"],"VssVersion":[""]}'
```

如果操作成功，該命令文件就會將新快照填入 EBS 快照清單中。您可以搜尋指定的標記，或是搜尋 `AppConsistent`，藉此在 EBS 快照清單中找到這些快照。如果命令執行失敗，請檢視命令輸出的詳細資訊，以了解執行失敗的原因。

### 取得命令狀態

若要取得快照目前的狀態，請使用從 `send-command` 傳回的命令 ID 執行下列命令。

```
aws ssm get-command-invocation
 --instance-ids "i-01234567890abcdef" \
 --command-id "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111" \
 --plugin-name "CreateVssSnapshot"
```

## PowerShell

使用執行下列命令，AWS Tools for Windows PowerShell 以建立啟用 VSS 的 EBS 快照，並取得建立輸出的目前執行階段狀態。指定前面清單中描述的參數，以修改快照處理程序的行為。

使用視窗適用的工具建立啟用 VSS 的 EBS 快照 PowerShell

執行下列命令，以建立啟用 VSS 的 EBS 快照或 AMI。

```
Send-SSMCommand -DocumentName "AWSEC2-VssInstallAndSnapshot" -InstanceId
 "i-01234567890abcdef" -Parameter
 @{ 'ExcludeBootVolume'='False'; 'description'='a_description'
 ; 'tags'='Key=key_name,Value=tag_value'; 'VssVersion'='' }
```

### 取得命令狀態

若要取得快照目前的狀態，請使用從 `Send-SSMCommand` 傳回的命令 ID 執行下列命令。

```
Get-SSMCommandInvocationDetail -InstanceId "i-01234567890abcdef" -CommandId
"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111" -PluginName "CreateVssSnapshot"
```

如果操作成功，該命令就會將新快照填入 EBS 快照清單中。您可以搜尋指定的標記，或是搜尋 AppConsistent，藉此在 EBS 快照清單中找到這些快照。如果命令執行失敗，請檢視命令輸出的詳細資訊，以了解執行失敗的原因。

## 運行 AWSEC 2-CreateVssSnapshot 命令文檔

若要執行 AWSEC2-CreateVssSnapshot 文件，請依照適用於您偏好之環境的步驟執行。

### Console

從主控台建立啟用 VSS 的 EBS 快照

1. [請在以下位置開啟 AWS Systems Manager 主控台。](https://console.aws.amazon.com/systems-manager/) <https://console.aws.amazon.com/systems-manager/>
2. 從導覽窗格中選取執行命令。這會顯示您帳戶中目前正在執行的命令清單 (若適用)。
3. 選擇 執行命令。這會開啟您具有存取權限的命令文件清單。
4. 從命令文件清單中選取 AWSEC2-CreateVssSnapshot。若要簡化結果，可輸入全部或部分文件名稱。您也可以依擁有者、平台類型或標籤進行篩選。

在您選取命令文件時，系統會將詳細資訊填入清單下方。

5. 從文件版本清單中選取 Default version at runtime。
6. 設定命令參數，以定義如何使用 VSS 快照或 AMI 來備份 AWSEC2-CreateVssSnapshot。如需參數詳細資訊，請參閱 [Systems Manager VSS 快照文件的參數](#)。
7. 針對目標選取，手動指定標籤或選取執行個體，以識別要執行這項操作的執行個體。

#### Note

如果您手動選取執行個體，但預期看到的執行個體並未出現在清單中，請參閱[我的執行個體在哪裡？](#)，以取得疑難排解秘訣。

8. 如需定義 Systems Manager Run Command 行為的其他參數 (例如速率控制)，請依照[從主控台執行命令](#)中的說明輸入值。
9. 選擇 Run (執行)。

如果操作成功，該命令就會將新快照填入 EBS 快照清單中。您可以搜尋指定的標籤，或是搜尋 AppConsistent，藉此在 EBS 快照清單中找到這些快照。如果命令執行失敗，請檢視 Systems Manager 命令輸出的詳細資訊，以了解執行失敗的原因。若系統成功完成命令，但無法備份特定磁碟區，則您可以在 EBS 磁碟區清單中進行疑難排解。

## AWS CLI

您可以在中執行下列命令，以建立啟用 VSS 的 EBS 快照。AWS CLI

### 建立具備 VSS 功能的 EBS 快照

執行下列命令，藉此建立具備 VSS 功能的 EBS 快照。若要建立快照，您必須使用 `--instance-ids` 參數來識別執行個體。如需您可以查看之其他參數的詳細資訊，請參閱 [Systems Manager VSS 快照文件的參數](#)。

```
aws ssm send-command \
 --document-name "AWSEC2-CreateVssSnapshot" \
 --instance-ids "i-01234567890abcdef" \
 --parameters '{"ExcludeBootVolume":["False"],"description":["Description"],"tags":
 [{"Key=key_name,Value=tag_value}]}'
```

如果操作成功，該命令文件就會將新快照填入 EBS 快照清單中。您可以搜尋指定的標記，或是搜尋 AppConsistent，藉此在 EBS 快照清單中找到這些快照。如果命令執行失敗，請檢視命令輸出的詳細資訊，以了解執行失敗的原因。

## PowerShell

使用執行下列命令，AWS Tools for Windows PowerShell 以建立啟用 VSS 的 EBS 快照。

### 使用視窗適用的工具建立啟用 VSS 的 EBS 快照 PowerShell

執行下列命令，藉此建立具備 VSS 功能的 EBS 快照。若要建立快照，您必須使用 InstanceId 參數來識別執行個體。您可以指定多個執行個體，藉此為其建立快照。如需您可以查看之其他參數的詳細資訊，請參閱 [Systems Manager VSS 快照文件的參數](#)。

```
Send-SSMCommand -DocumentName AWSEC2-CreateVssSnapshot -InstanceId
 "i-01234567890abcdef" -Parameter
 @{ 'ExcludeBootVolume'='False'; 'description'='a_description'
 ; 'tags'='Key=key_name,Value=tag_value' }
```

如果操作成功，該命令就會將新快照填入 EBS 快照清單中。您可以搜尋指定的標記，或是搜尋 `AppConsistent`，藉此在 EBS 快照清單中找到這些快照。如果命令執行失敗，請檢視命令輸出的詳細資訊，以了解執行失敗的原因。若系統成功完成命令，但無法備份特定磁碟區，則您可以在 EBS 快照清單中進行疑難排解。

使用共用 EBS 儲存體，為 Windows 容錯移轉叢集執行命令文件

可以使用上一節所述的任何命令列程序來建立具備 VSS 功能的快照。命令文件 (`AWSEC2-VssInstallAndSnapshot` 或 `AWSEC2-CreateVssSnapshot`) 必須在叢集中的主節點上執行。次要節點上的文件將會失敗，因為其無法存取共用磁碟。如果您的主要和次要動態變更，您可以在多個節點上 AWS Systems Manager 執行 Run Command 文件，並期望指令在主要節點上成功並在次要節點上失敗。

運行 `AWSEC 2-ManageVss IO SSM` 命令文檔

您可使用下列指令碼和預先定義的 `AWSEC2-ManageVssIO SSM` 文件，以暫時暫停 I/O、建立具備 VSS 功能的 EBS 快照，然後重新啟動 I/O。系統會在執行命令的使用者內容中執行此程序。如果使用者有足夠的權限來建立和標記快照，則 AWS Systems Manager 可以建立並標記啟用 VSS 的 EBS 快照，而不需要執行個體上的其他 IAM 快照角色。

相對的，命令文件 (`AWSEC2-VssInstallAndSnapshot` 或 `AWSEC2-CreateVssSnapshot`) 會要求您將 IAM 快照角色指派給每個執行個體，以便為這些執行個體建立 EBS 快照。如果您出於政策或法規因素，不願提供其他 IAM 許可給執行個體使用，則可利用下列指令碼。

開始之前

請注意以下與此程序相關的重要詳細資訊：

- 此程序會使用指 PowerShell 指令碼 (`CreateVssSnapshotAdvancedScript.ps1`)，為您指定的執行個體上的所有磁碟區建立快照，但根磁碟區除外。您必須使用 `AWSEC2-CreateVssSnapshot SSM` 文件，才能擷取根磁碟區的快照。
- 指令碼會呼叫 `AWSEC2-ManageVssIO` 文件兩次。第一次呼叫時，該指令碼會將 `Action` 參數設定為 `Freeze`，這會暫停執行個體上的所有 I/O。第二次呼叫時，`Action` 參數的設定會變更為 `Thaw`，讓系統能強制恢復 I/O。
- 請勿在未使用 `CreateVssSnapshotAdvancedScript.ps1` 指令碼的情況下嘗試使用 `AWSEC2-ManageVssIO` 文件。Microsoft 的 VSS 框架要求 `Freeze` 和 `Thaw` 動作的呼叫時間相差不得超過 10 秒鐘。此外，在未使用指令碼的情況下手動呼叫這些動作，可能會導致錯誤。

## 使用 **AWSEC2-ManagedVssIO** SSM 文件建立具備 VSS 功能的 EBS 快照

1. 下載 [CreateVssSnapshotAdvancedScript.zip](#) 檔案並解壓縮檔案內容。
2. `CreateVssSnapshotAdvancedScript.ps1`在文字編輯器中開啟，使用有效的 EC2 執行個體 ID、快照說明和所需標籤值編輯指令碼底部的範例呼叫，然後從中執行指令碼 PowerShell。

如果操作成功，該命令就會將新快照填入 EBS 快照清單中。您可以搜尋指定的標記，或是搜尋 `AppConsistent`，藉此在 EBS 快照清單中找到這些快照。如果命令執行失敗，請檢視 命令輸出的詳細資訊，以了解執行失敗的原因。若命令執行成功，但無法備份特定磁碟區，則您可以在 EBS 磁碟區清單中進行疑難排解。

### Note

若要自動備份，您可以建立使用 `AWSEC2-VssInstallAndSnapshot` 文件的 AWS Systems Manager 維護時段工作。如需詳細資訊，請參閱《AWS Systems Manager 使用者指南》中的 [使用維護時段 Windows \(主控台\)](#)。

## 使用建立 VSS 快照 AWS Backup

您可以在使用時透過在主控制台或 CLI 中啟 AWS Backup 用 VSS 來建立 VSS 備份。在建立具備 VSS 功能的備份計畫之前，請確保已符合 [先決條件](#)。如需詳細資訊，請參閱《AWS Backup 開發人員指南》中的 [建立 Windows VSS 備份](#)。

### Note

AWS Backup 不會自動在您的執行個體上安裝 `AwsVssComponents` 套件。您必須在執行個體上執行手動安裝。如需詳細資訊，請參閱 [在執行個體上手動安裝 VSS 元件](#)。

## 使用 Amazon Data Lifecycle Manager 建立 VSS 快照

您可以在快照生命週期政策中啟用前置和後置指令碼，以使用 Amazon Data Lifecycle Manager 建立 VSS 快照。如需詳細資訊，請參閱 <https://docs.aws.amazon.com/efs/latest/userguide/automate-app-consistent-backups.html>。

**Note**

Amazon Data Lifecycle Manager 不會在您的執行個體上自動安裝 `AwsVssComponents` 套件。您必須在執行個體上執行手動安裝。如需詳細資訊，請參閱 [在執行個體上手動安裝 VSS 元件](#)。

## 疑難排解以視窗 VSS 為基礎的 EBS 快照

在嘗試任何其他疑難排解步驟之前，建議您先確認下列詳細資料。

- 確保您符合所有 [必要條件](#)。
- 確認您使用您作業系統適用的 `AwsVssComponents` 套件最新 [支援 Windows OS 版本](#)。您觀察到的問題可能已在較新版本中得到解決。

### 主題

- [檢查記錄檔](#)
- [收集其他診斷記錄](#)
- [在已設定 Proxy 的執行個體上使用 VSS](#)
- [錯誤：Thaw 管道連線逾時、Thaw 發生錯誤、VSS Freeze 等候逾時或其他逾時錯誤](#)
- [錯誤：無法叫用方法。僅在此語言模式下的核心類型上支援方法調用](#)

### 檢查記錄檔

如果您在建立啟用 VSS 的 EBS 快照時遇到問題或收到錯誤訊息，您可以在 Systems Manager 主控台中檢視命令輸出。

對於建立 VSS 快照的 Systems Manager 文件，您可以在執行時間將 `CollectDiagnosticLogs` 參數設定為 `True`。當 `CollectDiagnosticLogs` 參數設定為 `True` 時，VSS 會收集其他記錄檔以協助偵錯。如需詳細資訊，請參閱 [收集其他診斷記錄](#)。

如果您收集診斷記錄檔，Systems Manager 文件會將它們儲存在您的執行個體中，位於下列位置：`C:\ProgramData\Amazon\AwsVss\Logs\timestamp.zip` `CollectDiagnosticLogs` 參數的預設值為 `False`。

**Note**

如需其他偵錯說明，您可以將 .zip 檔案傳送至 AWS Support。

無論您是否收集診斷記錄，都可以使用下列其他記錄檔：

- %ProgramData%\Amazon\SSM\InstanceData\*InstanceID*\document\orchestration\*SSMCommandID*\awsrunPowerShellScript\runPowerShellScript\stdout
- %ProgramData%\Amazon\SSM\InstanceData\*InstanceID*\document\orchestration\*SSMCommandID*\awsrunPowerShellScript\runPowerShellScript\stderr

您也可以開啟事件檢視器 Windows 應用程式，然後選擇 Windows 日誌、應用程式來檢視其他日誌。若要特別查看 EC2 Windows VSS 供應程式和磁碟區陰影複製服務的事件，請依條款 **Ec2VssSoftwareProvider** 和 **VSS** 的來源進行篩選。

如果您將 Systems Manager 與 VPC 端點搭配使用，且 Systems Manager [SendCommand](#) API 動作 (在主控台中執行命令) 失敗，請確認您已正確設定下列端點：com.amazonaws. **##** .ec2。

如果沒有定義 Amazon EC2 端點，列舉連接的 EBS 磁碟區的呼叫會失敗，進而導致 Systems Manager 命令失敗。如需有關使用 Systems Manager 設定 VPC 端點的詳細資訊，請參閱 <https://docs.aws.amazon.com/systems-manager/latest/userguide/setup-create-vpc.html> 使用者指南中的 AWS Systems Manager 建立 Virtual Private Cloud 端點。

## 收集其他診斷記錄

若要在使用 Systems Manager send 命令執行 VSS 快照集文件時收集其他診斷記錄，請在執行階段將 CollectDiagnosticLogs 輸入參數設定為 True ""。建議您在疑難排解時將此參數設定為 True ""。

若要查看指令行範例，請選取下列其中一個標籤。

### AWS CLI

下列範例會在中執行 AWSEC2-CreateVssSnapshot Systems Manager 文件 AWS CLI：

```
aws ssm send-command \
--document-name "AWSEC2-CreateVssSnapshot" \
--instance-ids "i-1234567890abcdef0" \

```



```
--parameters '{"description":["Example - create diagnostic logs at runtime."],"tags":["Key=tag_name,Value=tag_value"],"CollectDiagnosticLogs":["True"]}'
```

## PowerShell

下列範例會在中執行 AWSEC2-CreateVssSnapshot Systems Manager 文件 PowerShell：

```
Send-SSMCommand -DocumentName AWSEC2-CreateVssSnapshot -InstanceId
"i-1234567890abcdef0" -Parameter @{'description'='Example - create diagnostic logs at runtime.';
'tags'='Key=tag_name,Value=tag_value';
'CollectDiagnosticLogs'='True'}
```

## 在已設定 Proxy 的執行個體上使用 VSS

如果在使用代理連線 EC2 端點的執行個體上建立啟用 VSS 的 EBS 快照時發生問題，請確認以下事項：

- 代理伺服器經過設定，以便透過以 SYSTEM 身分執行來存取執 AWS Tools for Windows PowerShell 行個體區域和 IMDS 中的 EC2 服務端點。
- 已安裝 AwsVssComponents 2.0.1 版或更新版本。從 AwsVssComponents 2.0.1 版開始，EC2 VSS 提供者支援使用系統所設定的 WinHTTP 代理。如需設定 WinHTTP 代理的詳細資訊，請前往 Microsoft 網站參閱 [Windows 超文字傳輸通訊協定 \(WINHTTP\) 的 Netsh 命令](#)。

## 錯誤：Thaw 管道連線逾時、Thaw 發生錯誤、VSS Freeze 等候逾時或其他逾時錯誤

EC2 Windows VSS 供應程式可能會因為執行個體上的活動或服務導致具備 VSS 功能的 EBS 快照無法及時進行而逾時。Windows VSS 架構會提供無法設定的 10 秒時段，暫停與檔案系統的通訊。在這段期間，AWSEC2-CreateVssSnapshot 會對磁碟區執行快照。

下列問題可能造成 EC2 Windows VSS 供應程式在快照期間遇到時間限制：

- 磁碟區的過多 I/O
- 執行個體上的 EC2 API 回應緩慢
- 分段的磁碟區
- 與某些防毒軟體不相容
- VSS 應用程式寫入器的問題
- 為大量模組啟用「模 PowerShell 組記錄」時，可能會導致 PowerShell 指令碼執行緩慢

執行 `AWSEC2-CreateVssSnapshot` 命令文件時遇到的大部分逾時問題是與備份時執行個體的工作負載過高相關。下列動作可協助您成功建立快照：

- 重試 `AWSEC2-CreateVssSnapshot` 命令來查看快照是否成功。若在某些情況下重試成功，減少執行個體負載可能使快照更加成功。
- 等候一段時間，待執行個體上的工作負載減少，再重試 `AWSEC2-CreateVssSnapshot` 命令。或者，您可以在執行個體處於低壓力時嘗試快照。
- 當系統上的防毒軟體關閉時，請嘗試 VSS 快照。若這樣可解決問題，請見防毒軟體說明，並將其設定為允許 VSS 快照。
- 如果您的帳戶在執行快照的相同區域內出現大量 Amazon EC2 API 呼叫，則 API 限流可能會延遲快照作業。若要減少限流帶來的影響，請使用最新的 `AwsVssComponents` 套件 (2.1.0 版和更新版本，具有先決條件許可)。此套件利用 EC2 `CreateSnapshots` API 動作來減少每磁碟區快照建立和標記等變更動作的數量。
- 如果您同時執行多個 `AWSEC2-CreateVssSnapshot` 命令指令碼，您可以採取下列步驟來減少並行問題。
  - 請考慮在 API 活動較低期間安排快照時間。
  - 如果您在 Systems Manager 主控台 (或 API 中的 `SendCommand`) 使用 `Run Command` 來執行命令指令碼，您可以使用 Systems Manager 速率控制來減少並行。

您也可以使用 Systems Manager 速率控制來減少使用 Systems Manager 執行命令指令碼之類 AWS Backup 服務的並行性。

- 在 Shell 中執行命令 `vssadmin list writers`，並查看是否有回報系統上任何寫入器的上次錯誤欄位中任何錯誤。如有任何寫入器回報逾時錯誤，請考慮在執行個體負載較少時重試快照。
- 當您使用較小的執行個體類型 (例如 `t2` / `t3` / `t3a.nano` 或 `t2` / `t3` / `t3a.micro`)，可能會因記憶體和 CPU 限制而發生逾時。下列動作可能有助於減少逾時問題。
  - 在建立快照之前，請嘗試關閉記憶體或 CPU 密集型應用程式。
  - 嘗試在執行個體活動較低期間建立快照。

**錯誤：**無法叫用方法。僅在此語言模式下的核心類型上支援方法調用

PowerShell 語言模式未設定為時，您將會遇到此錯誤 `FullLanguage`。`AWSEC2-CreateVssSnapshot` 和 `AWSEC2-ManageVssIo` SSM 文件需 PowerShell 要設定為 `FullLanguage` 模式。

若要驗證語言模式，請在 PowerShell 主控台的執行個體上執行下列命令：

```
$ExecutionContext.SessionState.LanguageMode
```

如需語言模式的詳細資訊，請參閱 Microsoft 文件中的 [about\\_Language\\_Modes](#)。

## 從具備 VSS 功能的 EBS 快照中還原 EBS 磁碟區

您能夠善用 `RestoreVssSnapshotSampleScript.ps1` 指令碼，以從具備 VSS 功能的 EBS 快照還原執行個體上的磁碟區。此指令碼會執行下列任務：

- 停止執行個體
- 移除執行個體上的所有現有磁碟機 (若已排除開機磁碟區，則其將不會涵蓋在內)
- 從快照建立新的磁碟區
- 使用快照上的裝置 ID 標記來將磁碟區連接至執行個體
- 重新啟動執行個體

### Important

下列指令碼會分離所有連接至執行個體的磁碟區，然後從快照建立新的磁碟區。請確認您已正確地備份執行個體。系統不會刪除舊的磁碟區。如果您想刪除舊磁碟區，則可編輯指令碼。

### 從具備 VSS 功能的 EBS 快照還原磁碟區

1. 下載 [RestoreVssSnapshotSampleScript.zip](#) 檔案並解壓縮檔案內容。
2. `RestoreVssSnapshotSampleScript.ps1` 在文字編輯器中開啟，並使用有效的 EC2 執行個體 ID 和 EBS 快照 ID 編輯指令碼底部的範例呼叫，然後從 PowerShell 中執行指令碼。

## AWS VSS 解決方案版本紀錄

### 主題

- [AwsVssComponents 套件版本](#)
- [支援 Windows OS 版本](#)

## AwsVssComponents 套件版本

下表說明 AWS VSS 元件套件的已發行版本。

版本	詳細資訊	發行日期
2.3.2	修正解除安裝時未移除 VSS 提供者註冊的情況。	2024年5月9日
2.3.1	新增新的預設標籤， <code>AwsVssConfig</code> 以識別 AWS VSS 建立的快照和 AMI。	2024年3月7日
2.2.1	<ul style="list-style-type: none"> <li>增加了對使用 <code>DescribeInstanceAttribute</code> API 的支持。</li> <li>錯誤修正與可靠性改進。</li> <li>對視窗服務器 2012 年和 2012 R2 不推薦使用的支持。AWS 視窗伺服器 2012 和 2012 R2 上的 VSS 元件 2.2.1 版安裝將會失敗。AWS VSS 組件版本 2.1.0 是最後一個版本，以支持視窗服務器 2012 年和 2012 R2。</li> </ul>	2024年1月18日
2.1.0	增加了對使用 <code>CreateSnapshots</code> API 的支持。	2023年11月6日
2.0.1	新增 WinHTTP 代理設定的使用支援。	2023年10月26日
2.0.0	已新增 AWS VSS 元件建立快照和 AMI 的功能，以便與 PowerShell 模組記錄、指令碼區塊記錄和轉錄功能相容。	2023年4月28日
1.3.2.0	修復了未正確報告安裝失敗的情況。	2022年5月10日
1.3.1.0	<ul style="list-style-type: none"> <li></li> </ul>	2020年2月6日

版本	詳細資訊	發行日期
	<p>修正快照在網域控制器上失敗且與 NTDS VSS 寫入器記錄有關的錯誤。</p> <ul style="list-style-type: none"> <li>修正解除安裝 1.0 版 VSS 提供者時的 VSS 代理程式錯誤。</li> </ul>	
1.3.00	<ul style="list-style-type: none"> <li>降低不需要的詳細資訊以改善記錄。</li> <li>修正安裝期間的區域化問題。</li> <li>修正某些提供者註冊錯誤狀況的傳回代碼。</li> <li>修正各種安裝問題。</li> </ul>	2019 年 3 月 19 日
1.2.00	<ul style="list-style-type: none"> <li>為代理程式新增命令列參數 <code>-nw</code> (無寫入器) 和 <code>-copy</code> (僅限複製)。</li> <li>修 EventLog 復了內存分配調用不當導致的錯誤。</li> </ul>	2018 年 11 月 15 日
1.1	修正錯誤作為預設 Windows Backup 與還原提供者使用的 AWS VSS 元件。	2017 年 12 月 12 日
1.0	初始版本。	2017 年 11 月 20 日

## 支援 Windows OS 版本

下表顯示您應該在 Amazon EC2 上每個版本的 Windows 伺服器上執行哪些 AWS VSS 解決方案版本。

Windows Server 版本	AwsVssComponents 版本	AWSEC2-VsInstallAndSnapshot 版本名稱	AWSEC2-CreateVssSnapshot 版本名稱	AWSEC2-ManagedVssIO 版本名稱
Windows Server 2022	預設	預設	預設	預設
Windows Server 2019	預設	預設	預設	預設
Windows Server 2016	預設	預設	預設	預設
Windows Server 2012 R2	2.1.0	不支援	2012R2	2012R2
Windows Server 2012	2.1.0	不支援	2012R2	2012R2
Windows Server 2008 R2	1.3.1.0	不支援	2008R2	2008R2

## Linux 執行個體的寫入防止損毀

### Note

僅 Linux 執行個體支援防止寫入損毀。

防寫是一項區塊儲存功能，其設計目的是改善 I/O 密集型關聯式資料庫工作負載的效能，並在不影響資料復原能力的情況下減少延遲。AWS 使用 InnoDB 或 XtraDB 作為資料庫引擎的關聯式資料庫，例如 MySQL 和 MariaDB，將受益於預防未完成的寫入。

一般而言，使用大於儲存裝置電源故障不可分割性分頁的關聯式資料庫會使用資料記錄機制來預防未完成的寫入。MariaDB 和 MySQL 使用雙寫緩衝區檔案將資料寫入資料表之前記錄資料。在寫入不完整或未完成的狀況下，由於作業系統當機或寫入交易期間斷電，資料庫可以從雙寫緩衝區復原資料。與寫入雙寫緩衝區相關聯的額外 I/O 負荷會影響資料庫效能和應用程式延遲，並減少每秒可處理的交易數量。有關雙寫緩衝區的詳細資訊，請參閱 [MariaDB](#) 和 [MySQL](#) 文件。

透過預防未完成的寫入，資料會以全有或全無寫入交易的方式寫入儲存區，因此無需使用雙寫緩衝區。這可避免在寫入交易期間發生作業系統當機或斷電時，將部分或未完成的資料寫入儲存區。每秒處理的交易數量最多可增加 30%，且寫入延遲最多可減少 50%，並且不會影響工作負載的彈性。

## 定價

使用預防未完成的寫入不會產生額外成本。

## 支援的區塊大小和區塊邊界相符

預防未完成的寫入支援 4 KiB、8 KiB 和 16 KiB 資料區塊的寫入作業。資料區塊起始邏輯區塊定址 (LBA) 必須與 4 KiB、8 KiB 或 16 KiB 的各自區塊邊界大小相符。例如，對於 16 KiB 寫入作業，資料區塊起始 LBA 必須與 16 KiB 的區塊邊界大小相符。

下表顯示各種儲存體和執行個體類型的支援。

	4 KiB 區塊	8 KiB 區塊	16 KiB 區塊
執行個體儲存體磁碟區	所有 NVMe 執行個體儲存體磁碟區都會連接至新一代的 I 系列執行個體。	硝基固態硬碟支援的 i4i、IM4GN 和 iS4Gen 執行個體。AWS	
Amazon EBS 磁碟區	所有 Amazon EBS 磁碟區連接到在 <a href="#">AWS 硝基系統上建置的執行個體</a> 。		

若要確認您的執行個體和磁碟區是否支援預防未完成的寫入，請查詢執行個體是否支援預防未完成的寫入和其他詳細資訊，例如支援的區塊和邊界大小。如需詳細資訊，請參閱 [檢查預防未完成的寫入支援和設定](#)。

## 需求

若要預防未完成的寫入功能正常運作，I/O 作業必須符合 NTWPU、NTWGU、NTWBU 欄位中指定的大小、對齊和邊界要求。您必須設定作業系統，以確保特定的儲存子系統 (檔案系統、LVM、RAID 等) 在提交至裝置之前，不會修改儲存體堆疊下的 I/O 屬性，包括區塊合併、分割或區塊定址重新定位。

已透過下列組態測試預防未完成的寫入：

- 支援所需區塊大小的執行個體類型和儲存體類型。
- 核心版本 5.10 或更高版本的 Amazon Linux 2。
- 已啟用 `bigalloc` 且叢集大小為 16 KiB 的 `ext4`，以及最新的 `ext4` 公用程式 (`e2fsprogs 1.46.5` 或更高版本)。
- `O_DIRECT` 檔案存取模式繞過 Linux 核心緩衝區快取。

### Note

您不需要停用 MySQL 和 MariaDB 工作負載的 I/O 合併。

## 檢查預防未完成的寫入支援和設定

若要確認您的執行個體和磁碟區是否支援預防未完成的寫入，以及檢視 NVMe 命名空間廠商特定資料是否包含預防未完成的寫入資訊，請使用下列命令。

```
$ sudo nvme id-ns -v device_name
```

### Note


該命令以十六進位傳回供應商特定的資訊，並帶有 ASCII 解釋。您可能需要在可讀取和剖析輸出的應用程式中建置類似於 `ebsnvme-id` 的工具。

例如，下列命令會傳回 NVMe 命名空間廠商特定資料，其中包含 `/dev/nvme1n1` 的預防未完成的寫入資訊。

```
$ sudo nvme id-ns -v /dev/nvme1n1
```



如果您的執行個體和磁碟區支援損毀的寫入防護，它會在 NVMe 命名空間廠商特定資料中傳回下列 AWS 撕裂的寫入防護資訊。

 Note

下表中的位元組表示從 NVMe 命名空間廠商特定資料開頭算起的位移 (以位元組為單位)。

位元組	描述
0:31	例如，裝置配件掛載點的名稱，例如 <code>/dev/xvda</code> 。您可以在磁碟區附件請求期間提供此功能，且 Amazon EC2 執行個體可使用其來建立 NVMe 區塊型儲存裝置 (nvmeXn1) 的符號連結。
32:63	磁碟區 ID。例如 <code>vol01234567890abcdef</code> 。此欄位可用於將 NVMe 裝置對應至連接的磁碟區。
64:255	保留以供日後使用。
256:257	命名空間預防未完成的寫入單位大小 (NTWPU)。此欄位指出在停電或錯誤情況下保證自動寫入 NVM 之寫入作業的命名空間特定大小。此欄位邏輯區塊 (用以零為基礎的值表示) 中指定。
258:259	命名空間預防未完成的寫入精密度 (NTWPG)。此欄位指出在停電或錯誤情況下保證自動寫入 NVM 的寫入作業以下 NTWPU 的命名空間特定大小增量。也就是說，大小應為 $NTWPG * n \leq NTWPU$ ，其中 $n$ 是正整數。寫入作業 LBA 位移也必須與此欄位相符。此欄位邏輯區塊 (用以零為基礎的值表示) 中指定。
260:263	命名空間預防未完成的寫入邊界大小 (NTWPB)。此欄位指出此 NTWPU 值的不可分割邊界大小。在停電或錯誤情況下，不能保證向此命名空間寫入跨越不可分割邊界的內容會自動寫入 NVM。 <code>0h</code> 的值表示停電或錯誤情況沒有不可分割邊界。所有其他值會使用與 NTWPU 欄位相同的編碼方式，以邏輯區塊來指定大小。

## 設定軟體堆疊以預防未完成的寫入

[支援的執行個體類型 \(含支援的磁碟區\)](#) 預設啟用預防未完成的寫入。您不需要啟用任何其他設定，即可啟用磁碟區或執行個體，進而預防未完成的寫入。

### Note

不會對不支援預防未完成的寫入的工作負載造成效能影響。您不需要為這些工作負載進行任何變更。

確實支援預防未完成的寫入，但未設定為使用它的工作負載，會繼續使用雙寫緩衝區，而且不會獲得任何效能優勢。

若要設定 MySQL 或 MariaDB 軟體堆疊以停用雙寫緩衝區，並使用預防未完成的寫入，請完成以下步驟：

1. 配置您的磁碟區以使用 ext4 檔案系統與 BigAlloc 選項，並將叢集大小設定為 4 KiB、8 KiB 或 16 KiB。使 BigAlloc 用叢集大小為 4 KiB、8 KiB 或 16 KiB，可確保檔案系統配置與相應邊界對齊的檔案。

```
$ mkfs.ext4 -O bigalloc -C 4096/8192/16384 device_name
```

### Note

對於 MySQL 和 MariaDB，您必須使用 `-C 16384` 來比對資料庫頁面大小。將分配精密度設定為頁面大小倍數以外的值，可能會導致分配與儲存裝置的預防未完成的寫入邊界不符。

例如：

```
$ mkfs.ext4 -O bigalloc -C 16384 /dev/nvme1n1
```

2. 將 InnoDB 設定為使用 `0_DIRECT` 排清方法並關閉 InnoDB 雙寫。使用您偏好的文字編輯器開啟 `/etc/my.cnf`，並更新 `innodb_flush_method` 和 `innodb_doublewrite` 參數，如下所示：

```
innodb_flush_method=0_DIRECT
innodb_doublewrite=0
```

**⚠ Important**

如果您使用的是邏輯磁碟區管理工具 (LVM) 或其他儲存虛擬化層，請確定磁碟區的起始位移與 16 KiB 倍數相符。這與基礎 NVMe 儲存體相關，可用來說明儲存虛擬化層所使用的中繼資料標題和超級區塊。如果您新增 LVM 實體磁碟區的位移，可能會導致檔案系統分配與 NVMe 裝置的位移不相符，而這會使預防未完成的寫入失效。如需詳細資訊，請參閱 [Linux 手冊頁面](#) 中的 `--dataalignmentoffset`。

# 資源與標籤

Amazon EC2 提供您可以建立及使用的不同「資源」。這些資源中的一部分包含映像、執行個體、磁碟區和快照。當您建立資源時，我們會指派一個唯一的資源 ID 給資源。

有些資源可使用您定義的值建立標籤，協助您整理和識別它們。

下列主題說明資源和標籤，以及如何使用它們。

## 目錄

- [資源回收筒](#)
- [資源位置](#)
- [資源 ID](#)
- [列出與篩選您的資源](#)
- [Amazon EC2 全域檢視](#)
- [標記您的 Amazon EC2 資源](#)
- [Amazon EC2 服務配額](#)

## 資源回收筒

資源回收筒具有資料復原功能，可讓您還原意外刪除的 Amazon EBS 快照和 EBS 後端 AMI。使用資源回收筒時，若您的資源遭到刪除，在永久刪除以前，其會在您指定的期間內保留於資源回收筒中。

您可於保留期間到期之前，隨時從資源回收筒還原資源。在您從資源回收筒還原資源之後，資源會從資源回收筒中移除，且您可像在帳戶中使用該類型的任何其他資源一樣加以使用。若保留期間到期且未還原資源，則會從資源回收筒中永久刪除資源，且再也無法進行復原。

使用資源回收筒可藉由保護您的業務關鍵資料，避免意外刪除，協助確保業務持續性。

## 主題

- [其運作方式？](#)
- [支援的資源](#)
- [考量事項](#)
- [配額](#)
- [相關服務](#)

- [定價](#)
- [所需的 IAM 許可](#)
- [使用保留規則](#)
- [使用資源回收筒中的資源](#)
- [監控資源回收筒](#)

## 其運作方式？

若要啟用並使用資源回收筒，您必須在要保護資源的 AWS 區域中建立保留規則。保留規則會指定下列項目：

- 您要保護的資源類型。
- 刪除資源回收筒時您想要保留於資源回收筒中的資源。
- 在永久刪除資源之前，要在資源回收筒中保留資源的保留期限。

使用資源回收筒，您可以建立兩種類型的保留規則：

- 標籤層級保留規則 - 標籤層級保留規則會使用資源標籤，來識別要保留在資源回收筒中的資源。針對每個保留規則，您可以指定一或多個標籤鍵值組。以保留規則中指定的至少一個標籤鍵值組標記的指定類型資源會在刪除時自動保留在資源回收筒中。如果您想要根據其標籤來保護帳戶中的特定資源，請使用此類型的保留規則。
- 區域層級保留規則 - 區域層級保留規則未指定任何資源標籤。其會套用至其建立規則所在區域中指定類型的所有資源，即使是未標記資源也一樣。若想要保護特定區域中特定類型的所有資源，請使用此類型的保留規則。

當資源位於資源回收筒中時，您可以隨時將其還原以供使用。

資源會保留在資源回收筒中，直到發生下列其中一種情況：

- 您可以手動將其還原以供使用。當您從資源回收筒還原資源時，該資源會從資源回收筒中移除，且立即變成可用。您可像使用您帳戶中該類型的任何其他資源一樣使用還原的資源。
- 保留期間到期。若保留期間到期且未從資源回收筒還原資源，則會從資源回收筒中永久刪除該資源，且再也無法進行檢視或還原。

## 支援的資源

資源回收筒支援下列資源類型：

- Amazon EBS 快照

### Important

資源回收筒保留規則也適用於封存儲存層中的封存快照。如果您刪除符合保留規則的封存快照，該快照會保留在資源回收筒中，其保留時間為保留規則中定義的期間。封存的快照在資源回收筒中時，會以封存快照的費率計費。

- Amazon EBS 後端 Amazon Machine Images (AMI)

### Note

保留規則也會套用至已停用的 AMI。

## 考量事項

下列考量適用於使用資源回收筒和保留規則時。

### 一般考量

#### Important


在您建立第一個保留規則時，該規則最多可能需要 30 分鐘才能變為活動狀態，並開始保留資源。建立第一個保留規則後，後續保留規則將變為活動狀態，且幾乎立即開始保留資源。

- 若資源在刪除時符合多個保留規則，則保留期間最長的保留規則會優先採用。
- 您無法手動從資源回收筒刪除資源。資源將會在其保留期間到期之後自動刪除。
- 當資源位於資源回收筒中時，您只能檢視該資源、還原該資源或修改其標籤。如要以任何其他方式使用資源，首先必須將其還原。
- 如果有任何 AWS 服務資源 (例如 B AWS ackup 或 Amazon Data Lifecycle Manager) 會刪除符合保留規則的資源，資源回收筒會自動保留該資源。
- 將資源傳送到資源回收筒時，會將下列系統產生的標籤指派給該資源：
  - 標籤鍵 - `aws:recycle-bin:resource-in-bin`


- 標籤值 - true

您無法手動編輯或刪除此標籤。從資源回收筒還原資源時，該標籤將會自動移除。

## 快照的考量

-  **Important**  
若您有 AMI 及其關聯快照的保留規則，請讓快照的保留期限與 AMI 的保留期限相同或更長。此可確保資源回收筒在刪除 AMI 本身之前不會刪除與 AMI 相關聯的快照，因為這會使 AMI 無法復原。
- 如果在刪除快照時對快照啟用快速快照還原，快速快照還原會在快照傳送至資源回收筒後不久自動停用。
  - 如果您在對快照停用快速快照還原之前還原快照，該快照仍會保持啟用狀態。
  - 如果您還原快照，在快速快照還原停用之後，它會保持停用狀態。如有需要，您必須手動重新啟用快速快照還原。
- 若快照在刪除時為共享，則在該快照傳送至資源回收筒時，它會自動取消共享。如果您還原快照，所有先前的共用許可都會自動還原。
- 如果由其他 AWS 服務建立的快照集 (例如 AWS Backup 傳送至資源回收筒)，而您稍後從資源回收筒還原該快照集，則該快照將不再由建立該快照的 AWS 服務管理。若不再需要該快照，您必須手動將其刪除。

## AMI 的考量

- 僅支援 Amazon EBS 後端 AMI。
-  **Important**  
若您有 AMI 及其關聯快照的保留規則，請讓快照的保留期限與 AMI 的保留期限相同或更長。此可確保資源回收筒在刪除 AMI 本身之前不會刪除與 AMI 相關聯的快照，因為這會使 AMI 無法復原。
- 若 AMI 在刪除時為共享，則在其傳送至資源回收筒時，會自動取消共用。若您還原 AMI，所有先前的共享許可皆會自動還原。
- 您必須先從資源回收筒還原 AMI 的所有關聯快照，並確保其為 available 狀態，然後才能從資源回收筒還原 AMI。

- 若從資源回收筒中刪除與 AMI 相關聯的快照，則該 AMI 將再也無法復原。在保留期間過期後將會刪除 AMI。
- 如果由其他 AWS 服務（例如 B AWS ackup）創建的 AMI 被發送到回收站，並且您稍後從回收站還原該 AMI，則不再由創建它的 AWS 服務管理該 AMI。若不再需要該 AMI，您必須手動將其刪除。

### Amazon Data Lifecycle Manager 快照政策的考量

- 如果 Amazon Data Lifecycle Manager 刪除符合保留規則的快照，則資源回收筒會自動保留該快照。
- 如果 Amazon Data Lifecycle Manager 刪除快照並在達到政策的保留閾值時將其傳送到資源回收筒，並且您從資源回收筒手動還原快照，則必須在不再需要該快照時手動將其刪除。Amazon Data Lifecycle Manager 將不再管理快照。
- 如果您手動刪除由政策建立的快照，並且在達到政策保留閾值時該快照位於資源回收筒中，則 Amazon Data Lifecycle Manager 將不會刪除該快照。當快照儲存在資源回收筒中時，Amazon Data Lifecycle Manager 不會管理這些快照。

如果在達到政策的保留閾值之前從資源回收筒還原快照，則 Amazon Data Lifecycle Manager 將在達到政策的保留閾值時刪除快照。

如果在達到政策的保留閾值後從資源回收筒還原快照，Amazon Data Lifecycle Manager 將不再刪除該快照。您必須手動刪除不再需要的快照。

### AWS Backup 的注意事項

- 如果 B AWS ackup 刪除符合保留規則的快照，資源回收筒會自動保留該快照。

### 封存快照的考量事項

- 資源回收筒保留規則也適用於封存儲存層中的封存快照。如果您刪除符合保留規則的封存快照，該快照會保留在資源回收筒中，其保留時間為保留規則中定義的期間。

封存的快照在資源回收筒中時，會以封存快照的費率計費。

如果保留規則在最短封存期間 90 天之前，從資源回收筒刪除封存的快照，則會向您收取剩餘天數的費用。如需詳細資訊，請參閱 Amazon EBS 使用者指南中的[封存快照定價和帳單](#)。

若要使用資源回收筒中的封存快照，您必須先從資源回收筒復原快照，然後再從封存層將快照還原至標準層。



## 配額

下列配額適用於資源回收筒。

配額	預設配額			
每個區域的保留規則	250			
每個保留規則的標籤鍵值組	50			

## 相關服務

資源回收筒可以使用以下服務：

- AWS CloudTrail - 可讓您記錄資源回收筒中發生的事件。如需詳細資訊，請參閱 [使用監視回收站 AWS CloudTrail](#)。

## 定價

資源回收筒中的資源按其標準費率計費。使用資源回收筒和保留規則無須額外付費。如需詳細資訊，請參閱 [Amazon EBS 定價](#)。

### Note

有些資源可能會在保留期間過期 AWS CLI 且已永久刪除之後，短時間內仍會顯示在資源回收筒主控台或和 API 輸出中。不會向您收取這些資源的費用。一旦保留期間到期，計費就會停止。

使用時，您可以使用下列 AWS 產生的成本配置標籤來追蹤成本和配置目的 AWS Billing and Cost Management。

- 索引鍵：`aws:recycle-bin:resource-in-bin`
- 值：`true`

如需詳細資訊，請參閱 AWS Billing and Cost Management 使用者指南中的 [AWS產生的成本分配標籤](#)。

## 所需的 IAM 許可

依預設，使用者無權使用資源回收筒、保留規則或位於資源回收筒中的資源。若要允許使用者使用這些資源，您必須建立 IAM 政策，其會授予使用特定資源和 API 動作的許可。建立策略之後，您必須將權限新增至您的使用者、群組或角色。

### 主題

- [使用資源回收筒和保留規則的許可](#)
- [使用資源回收筒中資源的許可](#)
- [資源回收筒的條件索引鍵](#)

## 使用資源回收筒和保留規則的許可

如要使用資源回收筒和保留規則，使用者需要下列許可。

- `rbin:CreateRule`
- `rbin:UpdateRule`
- `rbin:GetRule`
- `rbin:ListRules`
- `rbin>DeleteRule`
- `rbin:TagResource`
- `rbin:UntagResource`
- `rbin:ListTagsForResource`
- `rbin:LockRule`
- `rbin:UnlockRule`

如要使用資源回收筒主控台，使用者需要 `tag:GetResources` 許可。

以下為一個範例 IAM 政策，該政策包含主控台使用者的 `tag:GetResources` 許可。若無需某些許可，則您可從政策中將其移除。

```
{
```

```
"Version": "2012-10-17",
"Statement": [{
 "Effect": "Allow",
 "Action": [
 "rbin:CreateRule",
 "rbin:UpdateRule",
 "rbin:GetRule",
 "rbin:ListRules",
 "rbin>DeleteRule",
 "rbin:TagResource",
 "rbin:UntagResource",
 "rbin:ListTagsForResource",
 "rbin:LockRule",
 "rbin:UnlockRule",
 "tag:GetResources"
],
 "Resource": "*"
}]
}
```

若要提供存取權，請新增權限至您的使用者、群組或角色：

- 使用者和群組位於 AWS IAM Identity Center：

建立權限合集。請按照 AWS IAM Identity Center 使用者指南 中的 [建立權限合集](#) 說明進行操作。

- 透過身分提供者在 IAM 中管理的使用者：

建立聯合身分的角色。請按照 IAM 使用者指南 的 [為第三方身分提供者 \(聯合\) 建立角色](#) 中的指示進行操作。

- IAM 使用者：

- 建立您的使用者可擔任的角色。請按照 IAM 使用者指南 的 [為 IAM 使用者建立角色](#) 中的指示進行操作。
- (不建議) 將政策直接附加至使用者，或將使用者新增至使用者群組。請遵循 IAM 使用者指南的 [新增許可到使用者 \(主控台\)](#) 中的指示。

## 使用資源回收筒中資源的許可

如需有關使用資源回收筒中資源所需的 IAM 許可的詳細資訊，請參閱下列內容：

- [使用資源回收筒中快照的許可](#)

- [使用資源回收筒中 AMI 的許可](#)

## 資源回收筒的條件索引鍵

資源回收筒定義下列條件索引鍵，讓您可以在 IAM 政策的 Condition 元素中將其用於控制政策陳述式適用的條件。如需詳細資訊，請參閱《IAM 使用者指南》中的 [IAM JSON 政策元素：條件](#)。

### 主題

- [rbin:Request/ResourceType 條件金鑰](#)
- [rbin:Attribute/ResourceType 條件金鑰](#)

### **rbin:Request/ResourceType** 條件金鑰

rbin:Request/ResourceType 條件索引鍵可用來根據 ResourceType request 參數指定的值來篩選存取 [CreateRule](#) 和要 [ListRules](#) 求。

#### 例子一- CreateRule

下列 IAM 政策範例只允許 IAM 主體在指定給請 CreateRule 求參數的值為 EBS\_SNAPSHOT 或 EC2\_IMAGE 時 ResourceType 提出要求。這允許主體僅為快照和 AMI 建立新的保留規則。

```
{
 "Version" : "2012-10-17",
 "Statement" : [
 {
 "Effect" : "Allow",
 "Action" : [
 "rbin:CreateRule"
],
 "Resource" : "*",
 "Condition" : {
 "StringEquals" : {
 "rbin:Request/ResourceType" : ["EBS_SNAPSHOT", "EC2_IMAGE"]
 }
 }
 }
]
}
```

## 例子二- ListRules

下列 IAM 政策範例只允許 IAM 主體在請ListRules求參數指定的值為時ResourceType提出要求。EBS\_SNAPSHOT這允許主體僅列出快照的保留規則，並防止其列出任何其他資源類型的保留規則。

```
{
 "Version" : "2012-10-17",
 "Statement" : [
 {
 "Effect" : "Allow",
 "Action" : [
 "rbin:ListRules"
],
 "Resource" : "*",
 "Condition" : {
 "StringEquals" : {
 "rbin:Request/ResourceType" : "EBS_SNAPSHOT"
 }
 }
 }
]
}
```

### **rbin:Attribute/ResourceType** 條件金鑰

rbin:Attribute/ResourceType條件索引鍵可用來根據保留規則ResourceType屬性值篩選[LockRuleUnlockRuleTagResourceUntagResource](#)、[DeleteRuleGetRuleUpdateRule](#)、[ListTagsForResource](#)要求的存取。

## 例子一- UpdateRule

下列 IAM 政策範例只允許 IAM 主體在要UpdateRule求的保留規則ResourceType屬性為EBS\_SNAPSHOT或EC2\_IMAGE時提出要求。這允許主體僅為快照和 AMI 更新保留規則。

```
{
 "Version" : "2012-10-17",
 "Statement" : [
 {
 "Effect" : "Allow",
 "Action" : [
 "rbin:UpdateRule"
],

```

```

 "Resource" : "*",
 "Condition" : {
 "StringEquals" : {
 "rbin:Attribute/ResourceType" : ["EBS_SNAPSHOT", "EC2_IMAGE"]
 }
 }
 }
]
}

```

## 例子二- DeleteRule

下列 IAM 政策範例只有在要DeleteRule求的保留規則ResourceType屬性為時，才允許 IAM 主體提出要求。EBS\_SNAPSHOT這允許主體僅為快照刪除保留規則。

```

{
 "Version" : "2012-10-17",
 "Statement" : [
 {
 "Effect" : "Allow",
 "Action" : [
 "rbin>DeleteRule"
],
 "Resource" : "*",
 "Condition" : {
 "StringEquals" : {
 "rbin:Attribute/ResourceType" : "EBS_SNAPSHOT"
 }
 }
 }
]
}

```

## 使用保留規則

若要啟用並使用資源回收筒，您必須在要保護資源的 AWS 區域中建立保留規則。保留規則會指定下列項目：

- 您要保護的資源類型。
- 刪除資源回收筒時您想要保留於資源回收筒中的資源。
- 在永久刪除資源之前，要在資源回收筒中保留資源的保留期限。

使用資源回收筒，您可以建立兩種類型的保留規則：

- **標籤層級保留規則** - 標籤層級保留規則會使用資源標籤，來識別要保留在資源回收筒中的資源。針對每個保留規則，您可以指定一或多個標籤鍵值組。以保留規則中指定的至少一個標籤鍵值組標記的指定類型資源會在刪除時自動保留在資源回收筒中。如果您想要根據其標籤來保護帳戶中的特定資源，請使用此類型的保留規則。
- **區域層級保留規則** - 區域層級保留規則未指定任何資源標籤。其會套用至其建立規則所在區域中指定類型的所有資源，即使是未標記資源也一樣。若想要保護特定區域中特定類型的所有資源，請使用此類型的保留規則。

在您建立保留規則之後，符合其條件的資源會在刪除後自動在資源回收筒中於指定期間內保留下來。

## 主題

- [建立保留規則](#)
- [檢視資源回收筒保留規則](#)
- [更新保留規則](#)
- [鎖定保留規則](#)
- [解除鎖定保留規則](#)
- [標記保留規則](#)
- [檢視保留規則標籤](#)
- [從保留規則移除標籤](#)
- [刪除資源回收筒保留規則](#)

## 建立保留規則

建立保留規則時，您必須指定下列必要的參數：

- 要受保留規則保護的資源類型。
- 要受保留規則保護的資源。您可以在標籤層級和區域層級建立保留規則。
  - 若要建立標籤層級保留規則，請指定資源標籤，而這些標籤會識別要保護的資源。您只能為每個規則指定最多 50 個標籤，並將相同的標籤鍵值組新增至最多五個保留規則。
  - 若要建立區域層級保留規則，請不要指定任何標籤鍵值組。在這種情況下，指定類型的所有資源都受到保護。
- 資源刪除後要在資源回收筒中保留的期限。期間可以長達 1 年 (365 天)。

您也可以指定以下選用參數：

- 保留規則的選用名稱。名稱長度上限為 255 個字元。
- 保留規則的選用描述。描述最長可為 255 個字元。

#### Note

建議您不要在保留規則說明中加入個人識別資訊、機密或敏感資訊。

- 選用保留規則標籤，可協助識別和組織您的保留規則。您可以將最多 50 個標籤指派給每個規則。

您也可以在建立時選擇鎖定保留規則。如果您在建立時鎖定保留規則，則還必須指定解除鎖定延遲期間 (可以是 7 到 30 天)。依預設，保留規則會維持解除鎖定狀態，除非您明確予以鎖定。

保留規則只會在其建立所在區域中運作。如果打算在其他區域中使用資源回收筒，您必須在這些區域中建立額外的保留規則。

您可以使用下列其中一種方法來建立資源回收筒保留規則。

## Recycle Bin console

### 建立保留規則

1. 開啟資源回收筒主控台，網址為 <https://console.aws.amazon.com/rbin/home/>
2. 在導覽窗格中，選擇 Retention Rules (保留規則)，然後選擇 Create retention rule (建立保留規則)。
3. 在 Rule details (規則詳細資訊) 區段中，執行下列動作：
  - a. (選用) 對於 Retention rule name (保留規則名稱)，輸入保留規則的描述性名稱。
  - b. (選用) 對於 Retention rule description (保留規則描述)，輸入保留規則的簡短描述。
4. 在 Rule settings (規則設定) 區段中，執行下列動作：
  - a. 對於 Resource Type (資源類型)，選取要保護之保留規則的資源類型。保留規則將僅於資源回收筒中保留此類型的資源。
  - b. 執行以下任意一項：
    - 如要建立符合區域中所有已刪除指定類型資源的區域層級保留規則，請選取 Apply to all resources (套用到所有資源)。保留規則會在刪除時，保留資源回收筒中指定的所有已刪除資源，即使資源並無任何標籤也一樣。



- 若要建立標籤層級保留規則，如為 Resource tags to match (要符合的資源標籤)，輸入要用來識別要保留在資源回收筒中之指定類型資源的標籤鍵值組。保留規則只會保留至少具有其中一個指定之標籤鍵值組的指定類型資源。
  - c. 若為 Retention period (保留期間)，請輸入保留規則要在資源回收筒中保留資源的天數。
5. (選用) 若要鎖定保留規則，針對 Rule lock settings (規則鎖定設定)，選取 Lock (鎖定)，然後針對 Unlock delay period (解除鎖定延遲期間)，指定解除鎖定延遲期間 (以天為單位)。無法修改或刪除鎖定的保留規則。若要修改或刪除規則，您必須先將其解除鎖定，然後等待解除鎖定延遲期間到期。如需更多資訊，請參閱[鎖定保留規則](#)
- 若要讓保留規則保持解除鎖定狀態，則針對 Rule lock settings (規則鎖定設定)，請選取 Unlock (解除鎖定)。您可以隨時修改或刪除解除鎖定的保留規則。如需詳細資訊，請參閱[解除鎖定保留規則](#)。
6. (選用) 在 Tags (標籤) 區段中，執行下列動作：
    - 若要使用自訂標籤來標記規則，請選擇 Add tag (新增標籤)，然後輸入標籤鍵值組。
  7. 選擇 Create retention rule (建立保留規則)。

## AWS CLI

### 建立保留規則

使用 [create-rule](#) AWS CLI 命令。對於 `--retention-period`，指定要在資源回收筒中保留已刪除快照的天數。若為 `--resource-type`，請為快照指定 `EBS_SNAPSHOT`，或為 AMI 指定 `EC2_IMAGE`。若要建立標籤層級保留規則，請針對 `--resource-tags`，指定要用來識別要保留之快照的標籤。若要建立區域層級保留規則，請省略 `--resource-tags`。若要鎖定保留規則，請包含 `--lock-configuration`，並指定解除鎖定延遲期間 (以天為單位)。

```
aws rbin create-rule \
--retention-period RetentionPeriodValue=number_of_days,RetentionPeriodUnit=DAYS \
--resource-type EBS_SNAPSHOT|EC2_IMAGE \
--description "rule_description" \
--lock-configuration
'UnlockDelay={UnlockDelayUnit=DAYS,UnlockDelayValue=unlock_delay_in_days}' \
--resource-tags ResourceTagKey=tag_key,ResourceTagValue=tag_value
```

### 範例 1

下列範例命令會建立解除鎖定的區域層級保留規則，其會將所有已刪除的快照保留 7 天。

```
aws rbin create-rule \
--retention-period RetentionPeriodValue=7,RetentionPeriodUnit=DAYS \
--resource-type EBS_SNAPSHOT \
--description "Match all snapshots"
```

## 範例 2

下列範例命令會建立標籤層級保留規則，其會將以 `purpose=production` 標記的已刪除快照保留 7 天。

```
aws rbin create-rule \
--retention-period RetentionPeriodValue=7,RetentionPeriodUnit=DAYS \
--resource-type EBS_SNAPSHOT \
--description "Match snapshots with a specific tag" \
--resource-tags ResourceTagKey=purpose,ResourceTagValue=production
```

## 範例 3

下列範例命令會建立鎖定的區域層級保留規則，其會將所有已刪除的快照保留 7 天。保留規則會鎖定，解除鎖定延遲期間為 7 天。

```
aws rbin create-rule \
--retention-period RetentionPeriodValue=7,RetentionPeriodUnit=DAYS \
--resource-type EBS_SNAPSHOT \
--description "Match all snapshots" \
--lock-configuration 'UnlockDelay={UnlockDelayUnit=DAYS,UnlockDelayValue=7}'
```

## 檢視資源回收筒保留規則

您可以使用下列其中一種方法來檢視資源回收筒保留規則。

### Recycle Bin console

#### 檢視保留規則

1. 開啟資源回收筒主控台，網址為 <https://console.aws.amazon.com/rbin/home/>
2. 在導覽窗格中，選擇 Retention rules (保留規則)。
3. 網格會列出所選區域的所有保留規則。若要檢視特定保留規則的詳細資訊，請在網格中選取該規則。

## AWS CLI

### 檢視所有保留規則

使用[清單規則](#) AWS CLI 命令，若為 `--resource-type`，請為快照指定 `EBS_SNAPSHOT`，或為 AMI 指定 `EC2_IMAGE`。

```
aws rbin list-rules --resource-type EBS_SNAPSHOT|EC2_IMAGE
```

### 範例

下列範例命令提供了保留快照的所有保留規則清單。

```
aws rbin list-rules --resource-type EBS_SNAPSHOT
```

### 檢視特定保留規則的資訊

使用取[得規則](#) AWS CLI 指令。

```
aws rbin get-rule --identifier rule_ID
```

### 範例

下列範例命令提供保留規則 `pwxIkFcvge4` 的相關資訊。

```
aws rbin get-rule --identifier pwxIkFcvge4
```

## 更新保留規則

建立後，您可隨時更新解除鎖定的保留規則的描述、資源標籤，以及保留期間。即使保留規則已解除鎖定，您也無法更新保留規則的資源類型或解除鎖定延遲期間。

您無法以任何方式更新鎖定的保留規則。如果您需要修改鎖定的保留規則，則必須先將其解除鎖定，然後等待解除鎖定延遲期間到期。

如果您需要修改鎖定的保留規則的解除鎖定延遲期間，則必須[解除鎖定保留規則](#)，並等待目前的解除鎖定延遲期間到期。解除鎖定延遲期間到期後，您必須[重新鎖定保留規則](#)，並指定新的解除鎖定延遲期間。

**Note**

建議您不要在保留規則說明中加入個人識別資訊、機密或敏感資訊。

在您更新保留規則之後，變更僅會套用至其保留的新資源。這些變更不會影響先前傳送至資源回收筒的資源。例如，若更新保留規則的保留期間，則只有更新後刪除的快照才會保留於新的保留期間內。在更新之前傳送至資源回收筒的快照仍會保留先前 (舊) 的保留期間。

您可以使用下列其中一種方法來更新保留規則。

## Recycle Bin console

### 更新保留規則

1. 開啟資源回收筒主控台，網址為 <https://console.aws.amazon.com/rbin/home/>
2. 在導覽窗格中，選擇 Retention rules (保留規則)。
3. 在網格中，選取要更新的保留規則，然後選取 Actions (動作)、Edit retention rule (編輯保留規則)。
4. 在 Rule details (規則詳細資訊) 區段中，視需要更新 Retention rule name (保留規則名稱) 和 Retention rule description (保留規則描述)。
5. 在 Rule settings (規則設定) 區段中，視需要更新 Resource type (資源類型)、Resource tags to match (要符合的資源標籤)，和 Retention period (保留期間)。
6. 在 Tags (標籤) 區段中，視需要新增或移除保留規則標籤。
7. 選擇 Save retention rule (儲存保留規則)。

## AWS CLI

### 更新保留規則

使用 [update-rule](#) AWS CLI 命令。對於 `--identifier`，指定要更新的保留規則 ID。對於 `--resource-types`，為快照指定 `EBS_SNAPSHOT`，或為 AMI 指定 `EC2_IMAGE`。

```
aws rbin update-rule \
--identifier rule_ID \
--retention-period RetentionPeriodValue=number_of_days,RetentionPeriodUnit=DAYS \
--resource-type EBS_SNAPSHOT|EC2_IMAGE \

```

```
--description "rule_description"
```

## 範例

下列範例命令會更新保留規則 61sJ2Fa9nh9，以將所有快照保留 7 天並更新其描述。

```
aws rbin update-rule \
--identifier 61sJ2Fa9nh9 \
--retention-period RetentionPeriodValue=7,RetentionPeriodUnit=DAYS \
--resource-type EBS_SNAPSHOT \
--description "Retain for three weeks"
```

## 鎖定保留規則

資源回收筒可讓您隨時鎖定區域層級的保留規則。

### Note

您無法鎖定標籤層級的保留規則。

無法修改或刪除鎖定的保留規則，即使是擁有必要 IAM 許可的使用者也無法修改或刪除。鎖定保留規則，以協助保護規則免受意外或惡意的修改和刪除。

鎖定保留規則時，您必須指定解除鎖定延遲期間。這是解除鎖定保留規則後必須等待的時間，這樣您才能進行修改或刪除。您無法在解除鎖定延遲期間修改或刪除保留規則。您僅可在解除鎖定延遲期間到期後才能修改或刪除保留規則。

鎖定保留規則之後，您就無法變更解除鎖定延遲期間。如果您的帳戶許可遭受入侵，解除鎖定延遲期間可讓您有更多時間偵測及回應安全威脅。此期間的長度應該比識別和回應安全漏洞所需的時間更長。若要設定正確的持續時間，您可以檢視先前的安全事件，以及識別和修復帳戶漏洞所需的時間。

建議您使用 Amazon EventBridge 規則通知您保留規則鎖定狀態變更。如需詳細資訊，請參閱 [使用 Amazon 監控回收站 EventBridge](#)。

## 考量

- 您只能鎖定區域層級的保留規則。
- 您可隨時鎖定解除鎖定的保留規則。

- 解除鎖定延遲期間必須為 7 到 30 天。
- 您可以在解除鎖定延遲期間重新鎖定保留規則。重新鎖定保留規則會重設解除鎖定延遲期間。

您可以使用下列其中一種方法來鎖定區域層級保留規則。

## Recycle Bin console

### 若要鎖定保留規則

1. 開啟資源回收筒主控台，網址為 <https://console.aws.amazon.com/rbin/home/>
2. 在導覽面板中，選擇 Retention rules (保留規則)。
3. 在網格中，選取要鎖定的解除鎖定的保留規則，然後選取 Actions (動作)、Edit retention rule lock (編輯保留規則鎖定)。
4. 在 Edit retention rule lock (編輯保留規則鎖定) 畫面中，選擇 Lock (鎖定)，然後針對 Unlock delay period (解除鎖定延遲期間)，指定解除鎖定延遲期間 (以天為單位)。
5. 選取 I acknowledge that locking the retention rule will prevent it from being modified or deleted (我確認鎖定保留規則將防止其遭受修改或刪除) 核取方塊，然後選擇 Save (儲存)。

## AWS CLI

### 若要鎖定解除鎖定的保留規則

使用 [lock-rule](#) AWS CLI 命令。對於 `--identifier`，指定要鎖定的保留規則 ID。對於 `--lock-configuration`，指定解除鎖定延遲期間 (以天為單位)。

```
aws rbin lock-rule \
--identifier rule_ID \
--lock-configuration
'UnlockDelay={UnlockDelayUnit=DAYS,UnlockDelayValue=number_of_days}'
```

### 範例

下列範例命令會鎖定保留規則 6lsJ2Fa9nh9，並將解除鎖定延遲期間設定為 15 天。

```
aws rbin lock-rule \
--identifier 6lsJ2Fa9nh9 \
--lock-configuration 'UnlockDelay={UnlockDelayUnit=DAYS,UnlockDelayValue=15}'
```

## 解除鎖定保留規則

您無法修改或刪除鎖定的保留規則。如果您需要修改鎖定的保留規則，必須先將其解除鎖定。解除鎖定保留規則之後，您必須等待解除鎖定延遲期間到期，才能對其進行修改或刪除。您無法在解除鎖定延遲期間修改或刪除保留規則。

擁有必要 IAM 許可的使用者可以隨時修改和刪除解除鎖定的保留規則。解除鎖定保留規則，可能會導致其遭受意外或惡意的修改和刪除。

### 考量

- 您可以在解除鎖定延遲期間重新鎖定保留規則。
- 您可以在解除鎖定延遲期間到期後重新鎖定保留規則。
- 您無法繞過解除鎖定延遲時間。
- 初始鎖定後，您就無法變更解除鎖定延遲時間。

建議您使用 Amazon EventBridge 規則通知您保留規則鎖定狀態變更。如需詳細資訊，請參閱 [使用 Amazon 監控回收站 EventBridge](#)。

您可以使用下列其中一種方法來解除鎖定鎖定的區域層級保留規則。

### Recycle Bin console

#### 若要解除鎖定保留規則

1. 開啟資源回收筒主控台，網址為 <https://console.aws.amazon.com/rbin/home/>
2. 在導覽面板中，選擇 Retention rules (保留規則)。
3. 在網格中，選取要解除鎖定的鎖定保留規則，然後選取 Actions (動作)、Edit retention rule lock (編輯保留規則鎖定)。
4. 在 Edit retention rule lock (編輯保留規則鎖定) 畫面上，選擇 Unlock (解除鎖定)，然後選擇 Save (儲存)。

### AWS CLI

#### 若要解除鎖定鎖定的保留規則

使用 [unlock-rule](#) AWS CLI 命令。對於 `--identifier`，指定要解除鎖定的保留規則 ID。

```
aws rbin unlock-rule \
```

```
--identifier rule_ID
```

## 範例

下列範例命令會解除鎖定保留規則 61sJ2Fa9nh9。

```
aws rbin unlock-rule \
--identifier 61sJ2Fa9nh9
```

## 標記保留規則

您可以將自訂標籤指派給保留規則，以不同的方式將其分類，例如：依用途、擁有者或環境。這可協助您根據所指派的自訂標籤，有效地找到特定的保留規則。

您可以使用下列其中一種方法，將標籤指派給保留規則。

### Recycle Bin console

#### 標記保留規則

1. 開啟資源回收筒主控台，網址為 <https://console.aws.amazon.com/rbin/home/>
2. 在導覽窗格中，選擇 Retention rules (保留規則)。
3. 選取要標記的保留規則、選取 Tags (標籤) 索引標籤，然後選取 Manage tags (管理標籤)。
4. 選擇 Add tag (新增標籤)。對於 Key (鍵)，輸入標籤鍵名稱。對於 Value (值)，輸入標籤值。
5. 選擇 Save (儲存)。

### AWS CLI

#### 標記保留規則

使用 [標籤資源命令](#) AWS CLI。對於 `--resource-arn`，指定要標記之保留規則的 Amazon Resource Name (ARN)，並針對 `--tags`，指定標籤鍵值組。

```
aws rbin tag-resource \
--resource-arn retention_rule_arn \
--tags key=tag_key,value=tag_value
```

## 範例



下列範例命令會以標籤 `arn:aws:rbin:us-east-1:123456789012:rule/n0oSBBtItF3` 標記保留規則 `purpose=production`。

```
aws rbin tag-resource \
--resource-arn arn:aws:rbin:us-east-1:123456789012:rule/n0oSBBtItF3 \
--tags key=purpose,value=production
```

## 檢視保留規則標籤

您可以使用下列其中一種方法，檢視已指派給保留規則的標籤。

### Recycle Bin console

#### 檢視保留規則的標籤

1. 開啟資源回收筒主控台，網址為 <https://console.aws.amazon.com/rbin/home/>
2. 在導覽窗格中，選擇 Retention rules (保留規則)。
3. 選取要檢視其標籤的保留規則，並選取 Tags (標籤) 索引標籤。

### AWS CLI

#### 檢視已指派給保留規則的標籤

使用 [list-tags-for-resource](#) AWS CLI 命令。對於 `--resource-arn`，指定保留規則的 ARN。

```
aws rbin list-tags-for-resource \
--resource-arn retention_rule_arn
```

#### 範例

下列範例命令會列出保留規則 `arn:aws:rbin:us-east-1:123456789012:rule/n0oSBBtItF3` 的標籤。

```
aws rbin list-tags-for-resource \
--resource-arn arn:aws:rbin:us-east-1:123456789012:rule/n0oSBBtItF3
```

## 從保留規則移除標籤

您可以使用下列其中一種方法，從保留規則移除標籤。

## Recycle Bin console

### 從保留規則移除標籤

1. 開啟資源回收筒主控台，網址為 <https://console.aws.amazon.com/rbin/home/>
2. 在導覽窗格中，選擇 Retention rules (保留規則)。
3. 選取要從中移除標籤的保留規則、選取 Tags (標籤) 索引標籤，然後選取 Manage tags (管理標籤)。
4. 選擇要移除之標籤旁邊的 Remove (移除)。
5. 選擇 Save (儲存)。

## AWS CLI

### 從保留規則移除標籤

使用 [untag-resource](#) AWS CLI 命令：對於 `--resource-arn`，指定保留規則的 ARN。對於 `--tagkeys`，指定要移除之標籤的標籤鍵。

```
aws rbin untag-resource \
--resource-arn retention_rule_arn \
--tagkeys tag_key
```

### 範例

下列範例命令會從保留規則 `purpose` 中移除標籤鍵為 `arn:aws:rbin:us-east-1:123456789012:rule/n0oSBBtItF3` 的標籤。

```
aws rbin untag-resource \
--resource-arn arn:aws:rbin:us-east-1:123456789012:rule/n0oSBBtItF3 \
--tagkeys purpose
```

## 刪除資源回收筒保留規則

您可隨時刪除保留規則。當您刪除保留規則時，在刪除資源回收筒中新的資源之後，該規則不再保留它們。在刪除保留規則之前傳送至資源回收筒的資源會根據保留規則中定義的保留期間，繼續保留在資源回收筒中。當此期間到期時，資源會從資源回收筒中永久刪除。

您可以使用下列其中一種方法來刪除保留規則。

## Recycle Bin console

### 刪除保留規則

1. 開啟資源回收筒主控台，網址為 <https://console.aws.amazon.com/rbin/home/>
2. 在導覽窗格中，選擇 Retention rules (保留規則)。
3. 在網格中，選取要刪除的保留規則，然後選取 Actions (動作)、Delete retention rule (刪除保留規則)。
4. 當出現提示時，輸入確認訊息，然後選擇 Delete retention rule (刪除保留規則)。

## AWS CLI

### 刪除保留規則

使用 [delete-rule](#) AWS CLI 命令。對於 `--identifier`，指定要刪除的保留規則 ID。

```
aws rbin delete-rule --identifier rule_ID
```

### 範例

下列範例命令會刪除保留規則 61sJ2Fa9nh9。

```
aws rbin delete-rule --identifier 61sJ2Fa9nh9
```

## 使用資源回收筒中的資源

資源回收筒支援下列資源類型：

- Amazon EBS 快照
- Amazon EBS 後端 Amazon Machine Images (AMI)

### 任務

- [從資源回收筒復原快照](#)
- [從資源回收筒復原 AMI](#)

## 從資源回收筒復原快照

資源回收筒具有資料復原功能，可讓您還原意外刪除的 Amazon EBS 快照和 EBS 後端 AMI。使用資源回收筒時，若您的資源遭到刪除，在永久刪除以前，其會在您指定的期間內保留於資源回收筒中。

您可於保留期間到期之前，隨時從資源回收筒還原資源。在您從資源回收筒還原資源之後，資源會從資源回收筒中移除，且您可像在帳戶中使用該類型的任何其他資源一樣加以使用。若保留期間到期且未還原資源，則會從資源回收筒中永久刪除資源，且再也無法進行復原。

資源回收筒中快照的計費費率與您帳戶中的一般快照相同。使用資源回收筒和保留規則無須額外付費。如需詳細資訊，請參閱 [Amazon EBS 定價](#)。

如需詳細資訊，請參閱 [資源回收筒](#)。

### 主題

- [使用資源回收筒中快照的許可](#)
- [檢視資源回收筒中的快照](#)
- [從資源回收筒還原快照](#)

### 使用資源回收筒中快照的許可

依預設，使用者無權使用資源回收筒中的快照。若要允許使用者使用這些資源，您必須建立 IAM 政策，其會授予使用特定資源和 API 動作的許可。建立策略之後，您必須將權限新增至您的使用者、群組或角色。

如要檢視及復原資源回收筒中的快照，使用者必須具有下列許可：

- `ec2:ListSnapshotsInRecycleBin`
- `ec2:RestoreSnapshotFromRecycleBin`

如要管理資源回收筒中快照的標籤，使用者需要下列其他許可。

- `ec2:CreateTags`
- `ec2>DeleteTags`

如要使用資源回收筒主控台，使用者需要 `ec2:DescribeTags` 許可。

IAM 政策範例如下。其包括適用於主控台使用者的 `ec2:DescribeTags` 許可，且其包括管理標籤的 `ec2:CreateTags` 和 `ec2>DeleteTags` 權限。若無需許可，則您可從政策中將其移除。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:ListSnapshotsInRecycleBin",
 "ec2:RestoreSnapshotFromRecycleBin"
],
 "Resource": "*"
 },
 {
 "Effect": "Allow",
 "Action": [
 "ec2:CreateTags",
 "ec2>DeleteTags",
 "ec2:DescribeTags"
],
 "Resource": "arn:aws:ec2:Region:account-id:snapshot/*"
 }
]
}
```

若要提供存取權，請新增權限至您的使用者、群組或角色：

- 使用者和群組位於 AWS IAM Identity Center：

建立權限合集。請按照 AWS IAM Identity Center 使用者指南 中的 [建立權限合集](#) 說明進行操作。

- 透過身分提供者在 IAM 中管理的使用者：

建立聯合身分的角色。請按照 IAM 使用者指南 的 [為第三方身分提供者 \(聯合\) 建立角色](#) 中的指示進行操作。

- IAM 使用者：

- 建立您的使用者可擔任的角色。請按照 IAM 使用者指南 的 [為 IAM 使用者建立角色](#) 中的指示進行操作。
- (不建議) 將政策直接附加至使用者，或將使用者新增至使用者群組。請遵循 IAM 使用者指南的 [新增許可到使用者 \(主控台\)](#) 中的指示。

如需更多使用資源回收筒所需許可的詳細資訊，請參閱 [使用資源回收筒和保留規則的許可](#)。

## 檢視資源回收筒中的快照

當快照位於資源回收筒中時，您可以檢視其有限的相關資訊，包括：

- 快照的 ID。
- 快照描述。
- 已從中建立快照的磁碟區 ID。
- 刪除快照及其進入資源回收筒的日期和時間。
- 保留期間到期的日期和時間。此時，會從資源回收筒中永久刪除快照。

您可以使用下列其中一種方法來檢視資源回收筒中的快照。

### Recycle Bin console

使用主控台檢視資源回收筒中的快照

1. 開啟資源回收筒主控台，網址為 <https://console.aws.amazon.com/rbin/home/>
2. 在導覽窗格中，選擇 Recycle Bin (資源回收筒)。
3. 網格會列出目前位於資源回收筒中的所有快照。若要檢視特定快照的詳細資訊，請在網格中選取該快照，然後選取 Actions (動作)、View details (檢視詳細資訊)。

### AWS CLI

使用檢視資源回收筒中的快照 AWS CLI

使用 [列表快照](#) AWS CLI 回收站命令。包括 `--snapshot-id` 選項來檢視特定快照。或者省略 `--snapshot-id` 選項來檢視資源回收筒中的所有快照。

```
aws ec2 list-snapshots-in-recycle-bin --snapshot-id snapshot_id
```

例如，下列命令會提供資源回收筒中快照 `snap-01234567890abcdef` 的相關資訊。

```
aws ec2 list-snapshots-in-recycle-bin --snapshot-id snap-01234567890abcdef
```

輸出範例：

```
{
 "SnapshotRecycleBinInfo": [
```

```
{
 "Description": "Monthly data backup snapshot",
 "RecycleBinEnterTime": "2021-12-01T13:00:00.000Z",
 "RecycleBinExitTime": "2021-12-15T13:00:00.000Z",
 "VolumeId": "vol-abcdef09876543210",
 "SnapshotId": "snap-01234567890abcdef"
}
]
```

## 從資源回收筒還原快照

當快照位於資源回收筒中時，您無法以任何方式使用該快照。若要使用該快照，首先必須將其還原。當您從資源回收筒還原快照時，該快照會立即可供使用，而且會從資源回收筒中移除。您可以採取您在帳戶中使用任何其他快照的同一方式來使用還原的快照。

您可以使用下列其中一種方法，從資源回收筒還原快照。

### Recycle Bin console

#### 使用主控台從資源回收筒還原快照

1. 開啟資源回收筒主控台，網址為 <https://console.aws.amazon.com/rbin/home/>
2. 在導覽窗格中，選擇 Recycle Bin (資源回收筒)。
3. 網格會列出目前位於資源回收筒中的所有快照。選取要還原的快照，然後選取 Recover (復原)。
4. 出現提示時，請選擇 Recover (復原)。

### AWS CLI

若要從資源回收筒還原已刪除的快照，請使用 AWS CLI

使用從[回收筒恢復快照命令](#) AWS CLI。對於 `--snapshot-id`，指定要還原的快照 ID。

```
aws ec2 restore-snapshot-from-recycle-bin --snapshot-id snapshot_id
```

例如，下列命令會從資源回收筒還原快照 `snap-01234567890abcdef`。

```
aws ec2 restore-snapshot-from-recycle-bin --snapshot-id snap-01234567890abcdef
```

輸出範例：

```
{
 "SnapshotId": "snap-01234567890abcdef",
 "Description": "Monthly data backup snapshot",
 "Encrypted": false,
 "OwnerId": "111122223333",
 "Progress": "100%",
 "StartTime": "2021-12-01T13:00:00.000000+00:00",
 "State": "recovering",
 "VolumeId": "vol-ffffffff",
 "VolumeSize": 30
}
```

## 從資源回收筒復原 AMI

資源回收筒具有資料復原功能，可讓您還原意外刪除的 Amazon EBS 快照和 EBS 後端 AMI。使用資源回收筒時，若您的資源遭到刪除，在永久刪除以前，其會在您指定的期間內保留於資源回收筒中。

您可於保留期間到期之前，隨時從資源回收筒還原資源。在您從資源回收筒還原資源之後，資源會從資源回收筒中移除，且您可像在帳戶中使用該類型的任何其他資源一樣加以使用。若保留期間到期且未還原資源，則會從資源回收筒中永久刪除資源，且再也無法進行復原。

資源回收筒中的 AMI 不會產生任何額外的費用。

如需詳細資訊，請參閱 [資源回收筒](#)。

### 主題

- [使用資源回收筒中 AMI 的許可](#)
- [檢視資源回收筒中的 AMI](#)
- [從資源回收筒還原 AMI](#)

### 使用資源回收筒中 AMI 的許可

依預設，使用者無權使用資源回收筒中的 AMI。若要允許使用者使用這些資源，您必須建立 IAM 政策，其會授予使用特定資源和 API 動作的許可。建立策略之後，您必須將權限新增至您的使用者、群組或角色。

若要檢視及復原資源回收筒中的 AMI，使用者必須具有下列許可：



- `ec2:ListImagesInRecycleBin`
- `ec2:RestoreImageFromRecycleBin`

如要管理資源回收筒中 AMI 的標籤，使用者需要下列其他許可。

- `ec2:CreateTags`
- `ec2>DeleteTags`

如要使用資源回收筒主控台，使用者需要 `ec2:DescribeTags` 許可。

IAM 政策範例如下。其包括適用於主控台使用者的 `ec2:DescribeTags` 許可，且其包括管理標籤的 `ec2:CreateTags` 和 `ec2>DeleteTags` 權限。若無需許可，則您可從政策中將其移除。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:ListImagesInRecycleBin",
 "ec2:RestoreImageFromRecycleBin"
],
 "Resource": "*"
 },
 {
 "Effect": "Allow",
 "Action": [
 "ec2:CreateTags",
 "ec2>DeleteTags",
 "ec2:DescribeTags"
],
 "Resource": "arn:aws:ec2:Region::image/*"
 }
]
}
```

若要提供存取權，請新增權限至您的使用者、群組或角色：

- 使用者和群組位於 AWS IAM Identity Center：

建立權限合集。請按照 AWS IAM Identity Center 使用者指南 中的 [建立權限合集](#) 說明進行操作。

- 透過身分提供者在 IAM 中管理的使用者：

建立聯合身分的角色。請按照 IAM 使用者指南的 [為第三方身分提供者 \(聯合\) 建立角色](#) 中的指示進行操作。

- IAM 使用者：
  - 建立您的使用者可擔任的角色。請按照 IAM 使用者指南的 [為 IAM 使用者建立角色](#) 中的指示進行操作。
  - (不建議) 將政策直接附加至使用者，或將使用者新增至使用者群組。請遵循 IAM 使用者指南的 [新增許可到使用者 \(主控台\)](#) 中的指示。

如需更多使用資源回收筒所需許可的詳細資訊，請參閱 [使用資源回收筒和保留規則的許可](#)。

## 檢視資源回收筒中的 AMI

當 AMI 位於資源回收筒中時，您可檢視其有限的相關資訊，包括：

- AMI 的名稱、說明和唯一 ID。
- 刪除 AMI 及其進入資源回收筒的日期和時間。
- 保留期間到期的日期和時間。AMI 將在此時遭到永久刪除。

您可使用下列其中一種方法來檢視資源回收筒中的 AMI。

## Recycle Bin console

如要使用主控台檢視資源回收筒中的 AMI

1. 開啟資源回收筒主控台，網址為 [console.aws.amazon.com/rbin/home/](https://console.aws.amazon.com/rbin/home/)。
2. 在導覽窗格中，選擇 Recycle Bin (資源回收筒)。
3. 網格會列出目前位於資源回收筒中的所有資源。如要檢視特定 AMI 的詳細資料，請在網格中加以選取，然後依序選取 Actions (動作)、View details (檢視詳細資料)。

## AWS CLI

若要檢視資源回收筒中已刪除的 AMI，請使用 AWS CLI

使用 [list-images-in-recycle-bin](#) AWS CLI 指令。如要檢視特定 AMI，請包含 `--image-id` 選項並指定要檢視的 AMI ID。您最多可以在單一請求中指定 20 個 ID。

如要檢視資源回收筒中的所有 AMI，請省略 `--image-id` 選項。若您不指定 `--max-items` 的值，依預設，該命令每頁會傳回 1,000 個項目。如需更多資訊，請參閱《Amazon EC2 API 參考》中的[分頁](#)。

```
aws ec2 list-images-in-recycle-bin --image-id ami_id
```

例如，下列命令會提供資源回收筒中 AMI `ami-01234567890abcdef` 的相關資訊。

```
aws ec2 list-images-in-recycle-bin --image-id ami-01234567890abcdef
```

輸出範例：

```
{
 "Images": [
 {
 "ImageId": "ami-0f740206c743d75df",
 "Name": "My AL2 AMI",
 "Description": "My Amazon Linux 2 AMI",
 "RecycleBinEnterTime": "2021-11-26T21:04:50+00:00",
 "RecycleBinExitTime": "2022-03-06T21:04:50+00:00"
 }
]
}
```

### Important

如果您收到下列錯誤訊息，您可能需要更新您的 AWS CLI 版本。如需詳細資訊，請參閱[命令找不到錯誤](#)。

```
aws.exe: error: argument operation: Invalid choice, valid choices are: ...
```

## 從資源回收筒還原 AMI

當 AMI 位於資源回收筒中時，您無法以任何方式加以使用。如要使用 AMI，首先必須將其還原。當您從資源回收筒還原 AMI 時，該 AMI 可立即使用，且會從資源回收筒中移除。您可以像在您帳戶中使用任何其他 AMI 一樣使用還原的 AMI。

您可使用下列其中一種方法，從資源回收筒還原 AMI。

## Recycle Bin console

如要使用主控台從資源回收筒還原 AMI

1. 開啟資源回收筒主控台，網址為 [console.aws.amazon.com/rbin/home/](https://console.aws.amazon.com/rbin/home/)。
2. 在導覽窗格中，選擇 Recycle Bin (資源回收筒)。
3. 網格會列出目前位於資源回收筒中的所有資源。選取要還原的 AMI，然後選取 Recover (復原)。
4. 出現提示時，請選擇 Recover (復原)。

## AWS CLI

要使用從回收站恢復已刪除的 AMI AWS CLI

使用 [restore-image-from-recycle-bin](#) AWS CLI 指令。若為 `--image-id`，請指定要還原的 AMI ID。

```
aws ec2 restore-image-from-recycle-bin --image-id ami_id
```

例如，下列命令會從資源回收筒還原 AMI `ami-01234567890abcdef`。

```
aws ec2 restore-image-from-recycle-bin --image-id ami-01234567890abcdef
```

成功時不會傳回任何輸出。

### Important

如果您收到下列錯誤訊息，您可能需要更新您的 AWS CLI 版本。如需詳細資訊，請參閱[命令找不到錯誤](#)。

```
aws.exe: error: argument operation: Invalid choice, valid choices are: ...
```

## 監控資源回收筒

您可以使用下列功能來監控資源回收筒。

### 主題

- [使用 Amazon 監控回收站 EventBridge](#)
- [使用監視回收站 AWS CloudTrail](#)

## 使用 Amazon 監控回收站 EventBridge

資源回收筒會將事件傳送至 Amazon，以瞭解 EventBridge 對保留規則執行的動作。使用 EventBridge，您可以建立規則來啟動程式設計動作以回應這些事件。例如，您可以建立 EventBridge 規則，在保留規則解除鎖定且進入解除鎖定延遲期間時，傳送通知至您的電子郵件。如需詳細資訊，請參閱[建立可回應事件的 Amazon EventBridge 規則](#)。

中的事件 EventBridge 會以 JSON 物件表示。事件的獨特欄位會包含在 JSON 物件的 detail 區段中。event 欄位則包含事件名稱。result 欄位包含初始化事件之動作的完成狀態。如需詳細資訊，請參閱 [Amazon EventBridge 使用者指南中的 Amazon EventBridge 事件模式](#)。

有關 Amazon 的更多信息 EventBridge，請參閱 [Amazon 是什麼 EventBridge?](#) 在 Amazon 用 EventBridge 戶指南。

### 事件

- [RuleLocked](#)
- [RuleChange](#) 嘗試
- [RuleUnlock](#)已排程
- [RuleUnlocking](#)通知
- [RuleUnlocked](#)

### RuleLocked

以下是成功鎖定保留規則時資源回收筒所產生之事件的範例。此事件可以由CreateRule和LockRule請求產生。產生事件的 API 會在 api-name 欄位中註明。

```
{
 "version": "0",
 "id": "exampleb-b491-4cf7-a9f1-bf370example",
 "detail-type": "Recycle Bin Rule Locked",
 "source": "aws.rbin",
 "account": "123456789012",
 "time": "2022-08-10T16:37:50Z",
 "region": "us-west-2",
 "resources": [
```

```
 "arn:aws:rbin:us-west-2:123456789012:rule/a12345abcde"
],
 "detail":
 {
 "detail-version": " 1.0.0",
 "rule-id": "a12345abcde",
 "rule-description": "locked account level rule",
 "unlock-delay-period": "30 days",
 "api-name": "CreateRule"
 }
}
```

## RuleChange嘗試

以下是資源回收筒因嘗試修改或刪除鎖定的規則失敗而產生的事件範例。此事件可以由DeleteRule和UpdateRule請求產生。產生事件的 API 會在 api-name 欄位中註明。

```
{
 "version": "0",
 "id": "exampleb-b491-4cf7-a9f1-bf370example",
 "detail-type": "Recycle Bin Rule Change Attempted",
 "source": "aws.rbin",
 "account": "123456789012",
 "time": "2022-08-10T16:37:50Z",
 "region": "us-west-2",
 "resources": [
 "arn:aws:rbin:us-west-2:123456789012:rule/a12345abcde"
],
 "detail":
 {
 "detail-version": " 1.0.0",
 "rule-id": "a12345abcde",
 "rule-description": "locked account level rule",
 "unlock-delay-period": "30 days",
 "api-name": "DeleteRule"
 }
}
```

## RuleUnlock已排程

以下是解除鎖定保留規則並啟動其解除鎖定延遲期間時資源回收筒所產生之事件的範例。

```
{
```

```
"version": "0",
"id": "exampleb-b491-4cf7-a9f1-bf370example",
"detail-type": "Recycle Bin Rule Unlock Scheduled",
"source": "aws.rbin",
"account": "123456789012",
"time": "2022-08-10T16:37:50Z",
"region": "us-west-2",
"resources": [
 "arn:aws:rbin:us-west-2:123456789012:rule/a12345abcde"
],
"detail":
{
 "detail-version": " 1.0.0",
 "rule-id": "a12345abcde",
 "rule-description": "locked account level rule",
 "unlock-delay-period": "30 days",
 "scheduled-unlock-time": "2022-09-10T16:37:50Z",
}
}
```

## RuleUnlocking通知

以下是資源回收筒每天在保留規則處於解除鎖定延遲期間 (解除鎖定延遲期間到期前一天) 產生的事件範例。

```
{
 "version": "0",
 "id": "exampleb-b491-4cf7-a9f1-bf370example",
 "detail-type": "Recycle Bin Rule Unlocking Notice",
 "source": "aws.rbin",
 "account": "123456789012",
 "time": "2022-08-10T16:37:50Z",
 "region": "us-west-2",
 "resources": [
 "arn:aws:rbin:us-west-2:123456789012:rule/a12345abcde"
],
 "detail":
 {
 "detail-version": " 1.0.0",
 "rule-id": "a12345abcde",
 "rule-description": "locked account level rule",
 "unlock-delay-period": "30 days",
 "scheduled-unlock-time": "2022-09-10T16:37:50Z"
 }
}
```

```
}
}
```

## RuleUnlocked

以下是資源回收筒在保留規則的解除鎖定延遲期間到期且可以修改或刪除保留規則時所產生的事件範例。

```
{
 "version": "0",
 "id": "exampleb-b491-4cf7-a9f1-bf370example",
 "detail-type": "Recycle Bin Rule Unlocked",
 "source": "aws.rbin",
 "account": "123456789012",
 "time": "2022-08-10T16:37:50Z",
 "region": "us-west-2",
 "resources": [
 "arn:aws:rbin:us-west-2:123456789012:rule/a12345abcde"
],
 "detail":
 {
 "detail-version": " 1.0.0",
 "rule-id": "a12345abcde",
 "rule-description": "locked account level rule",
 "unlock-delay-period": "30 days",
 "scheduled-unlock-time": "2022-09-10T16:37:50Z"
 }
}
```

## 使用監視回收站 AWS CloudTrail

資源回收筒服務已與整合 AWS CloudTrail。CloudTrail 是提供使用者、角色或服務所採取之動作記錄的 AWS 服務。CloudTrail 捕獲在回收站中執行的所有 API 調用作為事件。如果您建立追蹤，您可以啟用連續交付 CloudTrail 事件到 Amazon Simple Storage Service (Amazon S3) 儲存貯體。如果您未設定追蹤，您仍然可以在 [事件歷程記錄] 的 CloudTrail 主控台中檢視最近的管理事件。您可以使用收集的資訊 CloudTrail 來判斷向資源回收筒提出的要求、提出要求的 IP 位址、提出要求的人員、提出要求的時間，以及其他詳細資訊。

若要取得有關的更多資訊 CloudTrail，請參閱[AWS CloudTrail 使用者指南](#)。



## 資源回收筒資訊 CloudTrail

CloudTrail 在您創建 AWS 帳戶時，您的帳戶已啟用。當資源回收筒中發生受支援的事件活動時，該活動會與 CloudTrail 事件歷史記錄中的其他 AWS 服務事件一起記錄在事件中。您可以在帳戶中查看，搜索和下載最近的事 AWS 件。如需詳細資訊，請參閱[檢視具有事 CloudTrail 件記錄的事件](#)。

如需 AWS 帳戶中持續記錄事件 (包括資源回收筒的事件)，請建立追蹤。追蹤可 CloudTrail 將日誌檔傳遞至 S3 儲存貯體。根據預設，當您在主控台中建立追蹤時，追蹤會套用至所有 AWS 區域。追蹤記錄來自 AWS 分割區中所有區域的事件，並將日誌檔傳送到您指定的 S3 儲存貯體。此外，您還可以設定其他 AWS 服務，以進一步分析 CloudTrail 記錄中收集的事件資料並採取行動。如需詳細資訊，請參閱 AWS CloudTrail 使用者指南中的[建立追蹤的概觀](#)。

### 支援的 API 動作

對於資源回收筒，您可以使 CloudTrail 用將下列 API 動作記錄為管理事件。

- CreateRule
- UpdateRule
- GetRules
- ListRule
- DeleteRule
- TagResource
- UntagResource
- ListTagsForResource
- LockRule
- UnlockRule

如需有關記錄管理事件的詳細資訊，請參閱《CloudTrail 使用指南》中的[記錄追蹤的管理事件](#)。

### 身分資訊

每一筆事件或日誌項目都會包含產生請求者的資訊。身分資訊可協助您判斷下列事項：

- 該請求是否使用根使用者還是使用者憑證提出。
- 提出該請求時，是否使用了特定角色或聯合身分使用者的暫時安全憑證。
- 請求是否由其他 AWS 服務提出。

如需詳細資訊，請參閱使[CloudTrail 用者IdentityElement](#)。

## 了解資源回收筒日誌檔項目

追蹤是一種組態，可讓事件以日誌檔的形式傳遞至您指定的 S3 儲存貯體。CloudTrail 記錄檔包含一或多個記錄項目。事件代表來自任何來源的單一請求，包括有關請求的操作，動作的日期和時間，請求參數等信息。CloudTrail 日誌文件不是公共 API 調用的有序堆棧跟踪，因此它們不會以任何特定順序顯示。

以下是範例 CloudTrail 記錄項目。

### CreateRule

```
{
 "eventVersion": "1.08",
 "userIdentity": {
 "type": "AssumedRole",
 "principalId": "123456789012",
 "arn": "arn:aws:iam::123456789012:root",
 "accountId": "123456789012",
 "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
 "sessionContext": {
 "sessionIssuer": {
 "type": "Role",
 "principalId": "123456789012",
 "arn": "arn:aws:iam::123456789012:role/Admin",
 "accountId": "123456789012",
 "userName": "Admin"
 },
 "webIdFederationData": {},
 "attributes": {
 "mfaAuthenticated": "false",
 "creationDate": "2021-08-02T21:43:38Z"
 }
 }
 },
 "eventTime": "2021-08-02T21:45:22Z",
 "eventSource": "rbin.amazonaws.com",
 "eventName": "CreateRule",
 "awsRegion": "us-west-2",
 "sourceIPAddress": "123.123.123.123",
 "userAgent": "aws-cli/1.20.9 Python/3.6.14
Linux/4.9.230-0.1.ac.224.84.332.metal1.x86_64 boto-core/1.21.9",
```

```

"requestParameters": {
 "retentionPeriod": {
 "retentionPeriodValue": 7,
 "retentionPeriodUnit": "DAYS"
 },
 "description": "Match all snapshots",
 "resourceType": "EBS_SNAPSHOT"
},
"responseElements": {
 "identifier": "jkrnexample"
},
"requestID": "ex0577a5-amc4-pl4f-ef51-50fdexample",
"eventID": "714fafex-2eam-42pl-913e-926d4example",
"readOnly": false,
"eventType": "AwsApiCall",
"managementEvent": true,
"eventCategory": "Management",
"recipientAccountId": "123456789012",
"tlsDetails": {
 "tlsVersion": "TLSv1.2",
 "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
 "clientProvidedHostHeader": "rbin.us-west-2.amazonaws.com"
}
}

```

## GetRule

```

{
 "eventVersion": "1.08",
 "userIdentity": {
 "type": "AssumedRole",
 "principalId": "123456789012",
 "arn": "arn:aws:iam::123456789012:root",
 "accountId": "123456789012",
 "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
 "sessionContext": {
 "sessionIssuer": {
 "type": "Role",
 "principalId": "123456789012",
 "arn": "arn:aws:iam::123456789012:role/Admin",
 "accountId": "123456789012",
 "userName": "Admin"
 }
 }
 },

```

```

 "webIdFederationData": {},
 "attributes": {
 "mfaAuthenticated": "false",
 "creationDate": "2021-08-02T21:43:38Z"
 }
 },
 "eventTime": "2021-08-02T21:45:33Z",
 "eventSource": "rbin.amazonaws.com",
 "eventName": "GetRule",
 "awsRegion": "us-west-2",
 "sourceIPAddress": "123.123.123.123",
 "userAgent": "aws-cli/1.20.9 Python/3.6.14
Linux/4.9.230-0.1.ac.224.84.332.metal1.x86_64 boto-core/1.21.9",
 "requestParameters": {
 "identifier": "jkrnexample"
 },
 "responseElements": null,
 "requestID": "ex0577a5-amc4-pl4f-ef51-50fdexample",
 "eventID": "714fafex-2eam-42pl-913e-926d4example",
 "readOnly": true,
 "eventType": "AwsApiCall",
 "managementEvent": true,
 "eventCategory": "Management",
 "recipientAccountId": "123456789012",
 "tlsDetails": {
 "tlsVersion": "TLSv1.2",
 "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
 "clientProvidedHostHeader": "rbin.us-west-2.amazonaws.com"
 }
}

```

## ListRules

```

{
 "eventVersion": "1.08",
 "userIdentity": {
 "type": "AssumedRole",
 "principalId": "123456789012",
 "arn": "arn:aws:iam::123456789012:root",
 "accountId": "123456789012",
 "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
 "sessionContext": {

```

```
"sessionIssuer": {
 "type": "Role",
 "principalId": "123456789012",
 "arn": "arn:aws:iam::123456789012:role/Admin",
 "accountId": "123456789012",
 "userName": "Admin"
},
"webIdFederationData": {},
"attributes": {
 "mfaAuthenticated": "false",
 "creationDate": "2021-08-02T21:43:38Z"
}
},
"eventTime": "2021-08-02T21:44:37Z",
"eventSource": "rbin.amazonaws.com",
"eventName": "ListRules",
"awsRegion": "us-west-2",
"sourceIPAddress": "123.123.123.123",
"userAgent": "aws-cli/1.20.9 Python/3.6.14
Linux/4.9.230-0.1.ac.224.84.332.metal1.x86_64 botocore/1.21.9",
"requestParameters": {
 "resourceTags": [
 {
 "resourceTagKey": "test",
 "resourceTagValue": "test"
 }
]
},
"responseElements": null,
"requestID": "ex0577a5-amc4-pl4f-ef51-50fdexample",
"eventID": "714fafex-2eam-42pl-913e-926d4example",
"readOnly": true,
"eventType": "AwsApiCall",
"managementEvent": true,
"eventCategory": "Management",
"recipientAccountId": "123456789012",
"tlsDetails": {
 "tlsVersion": "TLSv1.2",
 "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
 "clientProvidedHostHeader": "rbin.us-west-2.amazonaws.com"
}
}
```

## UpdateRule

```
{
 "eventVersion": "1.08",
 "userIdentity": {
 "type": "AssumedRole",
 "principalId": "123456789012",
 "arn": "arn:aws:iam::123456789012:root",
 "accountId": "123456789012",
 "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
 "sessionContext": {
 "sessionIssuer": {
 "type": "Role",
 "principalId": "123456789012",
 "arn": "arn:aws:iam::123456789012:role/Admin",
 "accountId": "123456789012",
 "userName": "Admin"
 },
 "webIdFederationData": {},
 "attributes": {
 "mfaAuthenticated": "false",
 "creationDate": "2021-08-02T21:43:38Z"
 }
 }
 },
 "eventTime": "2021-08-02T21:46:03Z",
 "eventSource": "rbin.amazonaws.com",
 "eventName": "UpdateRule",
 "awsRegion": "us-west-2",
 "sourceIPAddress": "123.123.123.123",
 "userAgent": "aws-cli/1.20.9 Python/3.6.14
Linux/4.9.230-0.1.ac.224.84.332.metal1.x86_64 boto-core/1.21.9",
 "requestParameters": {
 "identifier": "jkrnexample",
 "retentionPeriod": {
 "retentionPeriodValue": 365,
 "retentionPeriodUnit": "DAYS"
 }
 },
 "description": "Match all snapshots",
 "resourceType": "EBS_SNAPSHOT"
},
"responseElements": null,
"requestID": "ex0577a5-amc4-pl4f-ef51-50fdexample",
"eventID": "714fafex-2eam-42pl-913e-926d4example",
```

```
"readOnly": false,
"eventType": "AwsApiCall",
"managementEvent": true,
"eventCategory": "Management",
"recipientAccountId": "123456789012",
"tlsDetails": {
 "tlsVersion": "TLSv1.2",
 "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
 "clientProvidedHostHeader": "rbin.us-west-2.amazonaws.com"
}
}
```

## DeleteRule

```
{
 "eventVersion": "1.08",
 "userIdentity": {
 "type": "AssumedRole",
 "principalId": "123456789012",
 "arn": "arn:aws:iam::123456789012:root",
 "accountId": "123456789012",
 "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
 "sessionContext": {
 "sessionIssuer": {
 "type": "Role",
 "principalId": "123456789012",
 "arn": "arn:aws:iam::123456789012:role/Admin",
 "accountId": "123456789012",
 "userName": "Admin"
 },
 "webIdFederationData": {},
 "attributes": {
 "mfaAuthenticated": "false",
 "creationDate": "2021-08-02T21:43:38Z"
 }
 }
 },
 "eventTime": "2021-08-02T21:46:25Z",
 "eventSource": "rbin.amazonaws.com",
 "eventName": "DeleteRule",
 "awsRegion": "us-west-2",
 "sourceIPAddress": "123.123.123.123",
```

```

"userAgent": "aws-cli/1.20.9 Python/3.6.14
Linux/4.9.230-0.1.ac.224.84.332.metal1.x86_64 boto-core/1.21.9",
"requestParameters": {
 "identifier": "jkrnexample"
},
"responseElements": null,
"requestID": "ex0577a5-amc4-pl4f-ef51-50fdexample",
"eventID": "714fafex-2eam-42pl-913e-926d4example",
"readOnly": false,
"eventType": "AwsApiCall",
"managementEvent": true,
"eventCategory": "Management",
"recipientAccountId": "123456789012",
"tlsDetails": {
 "tlsVersion": "TLSv1.2",
 "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
 "clientProvidedHostHeader": "rbin.us-west-2.amazonaws.com"
}
}

```

## TagResource

```

{
 "eventVersion": "1.08",
 "userIdentity": {
 "type": "AssumedRole",
 "principalId": "123456789012",
 "arn": "arn:aws:iam::123456789012:root",
 "accountId": "123456789012",
 "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
 "sessionContext": {
 "sessionIssuer": {
 "type": "Role",
 "principalId": "123456789012",
 "arn": "arn:aws:iam::123456789012:role/Admin",
 "accountId": "123456789012",
 "userName": "Admin"
 },
 "webIdFederationData": {},
 "attributes": {
 "mfaAuthenticated": "false",
 "creationDate": "2021-10-22T21:38:34Z"
 }
 }
 }
}

```



```

 }
 },
 "eventTime": "2021-10-22T21:43:15Z",
 "eventSource": "rbin.amazonaws.com",
 "eventName": "TagResource",
 "awsRegion": "us-west-2",
 "sourceIPAddress": "123.123.123.123",
 "userAgent": "aws-cli/1.20.26 Python/3.6.14
Linux/4.9.273-0.1.ac.226.84.332.metal1.x86_64 boto3/1.21.26",
 "requestParameters": {
 "resourceArn": "arn:aws:rbin:us-west-2:123456789012:rule/ABCDEF01234",
 "tags": [
 {
 "key": "purpose",
 "value": "production"
 }
]
 },
 "responseElements": null,
 "requestID": "examplee-7962-49ec-8633-795efexample",
 "eventID": "example4-6826-4c0a-bdec-0bab1example",
 "readOnly": false,
 "eventType": "AwsApiCall",
 "managementEvent": true,
 "eventCategory": "Management",
 "recipientAccountId": "123456789012",
 "tlsDetails": {
 "tlsVersion": "TLSv1.2",
 "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
 "clientProvidedHostHeader": "rbin.us-west-2.amazonaws.com"
 }
}

```

## UntagResource

```

{
 "eventVersion": "1.08",
 "userIdentity": {
 "type": "AssumedRole",
 "principalId": "123456789012",
 "arn": "arn:aws:iam::123456789012:root",
 "accountId": "123456789012",
 "accessKeyId": "AKIAIOSFODNN7EXAMPLE",

```

```
"sessionContext": {
 "sessionIssuer": {
 "type": "Role",
 "principalId": "123456789012",
 "arn": "arn:aws:iam::123456789012:role/Admin",
 "accountId": "123456789012",
 "userName": "Admin"
 },
 "webIdFederationData": {},
 "attributes": {
 "mfaAuthenticated": "false",
 "creationDate": "2021-10-22T21:38:34Z"
 }
},
"eventTime": "2021-10-22T21:44:16Z",
"eventSource": "rbin.amazonaws.com",
"eventName": "UntagResource",
"awsRegion": "us-west-2",
"sourceIPAddress": "123.123.123.123",
"userAgent": "aws-cli/1.20.26 Python/3.6.14
Linux/4.9.273-0.1.ac.226.84.332.metal1.x86_64 boto-core/1.21.26",
"requestParameters": {
 "resourceArn": "arn:aws:rbin:us-west-2:123456789012:rule/ABCDEF01234",
 "tagKeys": [
 "purpose"
]
},
"responseElements": null,
"requestID": "example7-6c1e-4f09-9e46-bb957example",
"eventID": "example6-75ff-4c94-a1cd-4d5f5example",
"readOnly": false,
"eventType": "AwsApiCall",
"managementEvent": true,
"eventCategory": "Management",
"recipientAccountId": "123456789012",
"tlsDetails": {
 "tlsVersion": "TLSv1.2",
 "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
 "clientProvidedHostHeader": "rbin.us-west-2.amazonaws.com"
}
}
```

## ListTagsForResource

```
{
 "eventVersion": "1.08",
 "userIdentity": {
 "type": "AssumedRole",
 "principalId": "123456789012",
 "arn": "arn:aws:iam::123456789012:root",
 "accountId": "123456789012",
 "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
 "sessionContext": {
 "sessionIssuer": {
 "type": "Role",
 "principalId": "123456789012",
 "arn": "arn:aws:iam::123456789012:role/Admin",
 "accountId": "123456789012",
 "userName": "Admin"
 },
 "webIdFederationData": {},
 "attributes": {
 "mfaAuthenticated": "false",
 "creationDate": "2021-10-22T21:38:34Z"
 }
 }
 },
 "eventTime": "2021-10-22T21:42:31Z",
 "eventSource": "rbin.amazonaws.com",
 "eventName": "ListTagsForResource",
 "awsRegion": "us-west-2",
 "sourceIPAddress": "123.123.123.123",
 "userAgent": "aws-cli/1.20.26 Python/3.6.14
Linux/4.9.273-0.1.ac.226.84.332.metal1.x86_64 boto-core/1.21.26",
 "requestParameters": {
 "resourceArn": "arn:aws:rbin:us-west-2:123456789012:rule/ABCDEF01234"
 },
 "responseElements": null,
 "requestID": "example8-10c7-43d4-b147-3d9d9example",
 "eventID": "example2-24fc-4da7-a479-c9748example",
 "readOnly": true,
 "eventType": "AwsApiCall",
 "managementEvent": true,
 "eventCategory": "Management",
 "recipientAccountId": "123456789012",
 "tlsDetails": {
```

```
"tlsVersion": "TLSv1.2",
"cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
"clientProvidedHostHeader": "rbin.us-west-2.amazonaws.com"
}
}
```

## LockRule

```
{
 "eventVersion": "1.08",
 "userIdentity": {
 "type": "AssumedRole",
 "principalId": "123456789012",
 "arn": "arn:aws:iam::123456789012:root",
 "accountId": "123456789012",
 "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
 "sessionContext": {
 "sessionIssuer": {
 "type": "Role",
 "principalId": "123456789012",
 "arn": "arn:aws:iam::123456789012:role/Admin",
 "accountId": "123456789012",
 "userName": "Admin"
 },
 "webIdFederationData": {},
 "attributes": {
 "creationDate": "2022-10-25T00:45:11Z",
 "mfaAuthenticated": "false"
 }
 }
 },
 "eventTime": "2022-10-25T00:45:19Z",
 "eventSource": "rbin.amazonaws.com",
 "eventName": "LockRule",
 "awsRegion": "us-west-2",
 "sourceIPAddress": "123.123.123.123",
 "userAgent": "python-requests/2.25.1",
 "requestParameters": {
 "identifier": "jkrnexample",
 "lockConfiguration": {
 "unlockDelay": {
 "unlockDelayValue": 7,
 "unlockDelayUnit": "DAYS"
 }
 }
 }
}
```

```

 }
 }
},
"responseElements": {
 "identifier": "jkrnexample",
 "description": "",
 "resourceType": "EBS_SNAPSHOT",
 "retentionPeriod": {
 "retentionPeriodValue": 7,
 "retentionPeriodUnit": "DAYS"
 },
 "resourceTags": [],
 "status": "available",
 "lockConfiguration": {
 "unlockDelay": {
 "unlockDelayValue": 7,
 "unlockDelayUnit": "DAYS"
 }
 },
 "lockState": "locked"
},
"requestID": "ex0577a5-amc4-pl4f-ef51-50fdexample",
"eventID": "714fafex-2eam-42pl-913e-926d4example",
"readOnly": false,
"eventType": "AwsApiCall",
"managementEvent": true,
"recipientAccountId": "123456789012",
"eventCategory": "Management",
"tlsDetails": {
 "tlsVersion": "TLSv1.2",
 "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
 "clientProvidedHostHeader": "rbin.us-west-2.amazonaws.com"
}
}

```

## UnlockRule

```

{
 "eventVersion": "1.08",
 "userIdentity": {
 "type": "AssumedRole",
 "principalId": "123456789012",
 "arn": "arn:aws:iam::123456789012:root",
 }
}

```

```
"accountId": "123456789012",
"accessKeyId": "AKIAIOSFODNN7EXAMPLE",
"sessionContext": {
 "sessionIssuer": {
 "type": "Role",
 "principalId": "123456789012",
 "arn": "arn:aws:iam::123456789012:role/Admin",
 "accountId": "123456789012",
 "userName": "Admin"
 },
 "webIdFederationData": {},
 "attributes": {
 "creationDate": "2022-10-25T00:45:11Z",
 "mfaAuthenticated": "false"
 }
},
"eventTime": "2022-10-25T00:46:17Z",
"eventSource": "rbin.amazonaws.com",
"eventName": "UnlockRule",
"awsRegion": "us-west-2",
"sourceIPAddress": "123.123.123.123",
"userAgent": "python-requests/2.25.1",
"requestParameters": {
 "identifier": "jkrnexample"
},
"responseElements": {
 "identifier": "jkrnexample",
 "description": "",
 "resourceType": "EC2_IMAGE",
 "retentionPeriod": {
 "retentionPeriodValue": 7,
 "retentionPeriodUnit": "DAYS"
 },
 "resourceTags": [],
 "status": "available",
 "lockConfiguration": {
 "unlockDelay": {
 "unlockDelayValue": 7,
 "unlockDelayUnit": "DAYS"
 }
 },
 "lockState": "pending_unlock",
 "lockEndTime": "Nov 1, 2022, 12:46:17 AM"
```

```

},
"requestID": "ex0577a5-amc4-pl14f-ef51-50fdexample",
"eventID": "714fafex-2eam-42pl-913e-926d4example",
"readOnly": false,
"eventType": "AwsApiCall",
"managementEvent": true,
"recipientAccountId": "123456789012",
"eventCategory": "Management",
"tlsDetails": {
 "tlsVersion": "TLSv1.2",
 "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
 "clientProvidedHostHeader": "rbin.us-west-2.amazonaws.com"
}
}

```

## 資源位置

Amazon EC2 資源專屬於它們所在的區 AWS 域或可用區域。

資源	類型	描述
Amazon EC2 資源識別符	區域性	每個資源識別符 (例如 AMI ID、執行個體 ID、EBS 磁碟區 ID 或 EBS 快照 ID) 都會與其區域繫結，因此只能在您建立該資源的區域中使用。
使用者提供的資源名稱	區域性	每個資源名稱 (例如：安全群組名稱或金鑰對名稱) 都會與其區域繫結，因此只能在您建立該資源的區域中使用。雖然您可以在多個區域中使用相同名稱建立資源，但它們彼此間不會產生關聯。
AMI	區域性	AMI 會與其檔案在 Amazon S3 中的區域繫結。您可以將 AMI 從其中一個區域複製到另外一個。如需詳細資訊，請參閱 <a href="#">複製 AMI</a> 。
EBS 快照	區域性	EBS 快照與區域繫結，因此僅能用於在相同區域中建立磁碟區。您可以將快照從其中一個區域複製到另外一個。

資源	類型	描述
EBS 磁碟區	可用區域	Amazon EBS 磁碟區與其可用區域繫結，因此僅能連接到位於相同可用區域中的執行個體。
彈性 IP 地址	區域性	彈性 IP 地址與區域繫結，因此僅能與相同區域中的執行個體相關聯。
執行個體	可用區域	執行個體與您啟動它的可用區域繫結。但是，其執行個體 ID 則與區域繫結。
金鑰對	全域性或區域性	<p>您使用 Amazon EC2 建立的金鑰對會和您建立它們的區域繫結。您可以建立自己的 RSA 金鑰對，並將其上傳到您希望使用的區域；如此一來，您便能透過上傳到每個區域，使您的金鑰對在全球服務中皆可使用。</p> <p>如需詳細資訊，請參閱 <a href="#">Amazon EC2 金鑰配對和 Amazon EC2 執行個體</a>。</p>
Security groups (安全群組)	區域性	安全群組與區域繫結，因此僅能指派給位於相同區域中的執行個體。您無法使用安全群組規則讓執行個體與區域外的執行個體通訊。來自其他區域中執行個體的流量會視為 WAN 頻寬。

## 資源 ID

當建立資源時，我們會指派一個唯一的資源 ID 給每個資源。資源 ID 的形式為資源識別符 (例如快照為 snap)，其後伴隨連字號和唯一的字母及數字組合。

每個資源識別符 (例如 AMI ID、執行個體 ID、EBS 磁碟區 ID 或 EBS 快照 ID) 都會與其區域繫結，因此只能在您建立該資源的區域中使用。

您可以使用資源 ID 來在 Amazon EC2 主控台中尋找您的資源。若您使用命令列工具或 Amazon EC2 API 使用 Amazon EC2，特定命令將需要資源 ID。例如，如果您使用 [stop-instance 指 AWS CLI 令停止執行個體](#)，則必須在指令中指定執行個體 ID。

### 資源 ID 長度



在 2016 年 1 月之前，指派給新建立特定資源類型之資源的 ID 是採用連字號後 8 個字元 (例如 i-1a2b3c4d) 的格式。從 2016 年 1 月到 2018 年 6 月，我們已將這些資源類型的 ID 變更為連字號後 17 個字元 (例如 i-1234567890abcdef0)。根據您建立帳戶的時間點，您的一些現有資源可能具有短 ID，但是新資源將收到較長 ID。

## 列出與篩選您的資源

您可以使用 Amazon EC2 主控台取得某些資源類型的清單。您可以使用資源對應的命令或 API 動作取得每種資源類型的清單。如果您有許多資源，您可篩選結果，只包含或排除符合特定條件的資源。

### 內容

- [使用主控台列出和篩選資源](#)
- [使用 CLI 和 API 列出和篩選](#)
- [使用 Amazon EC2 全域檢視來檢視跨區域資源](#)

## 使用主控台列出和篩選資源

### 內容

- [使用主控台列出資源](#)
- [使用主控台篩選資源](#)
  - [支援的篩選條件](#)

## 使用主控台列出資源

您可以使用主控台檢視最常見的 Amazon EC2 資源類型。若要檢視其他資源，請使用命令列介面或 API 動作。

### 使用主控台列出 EC2 資源

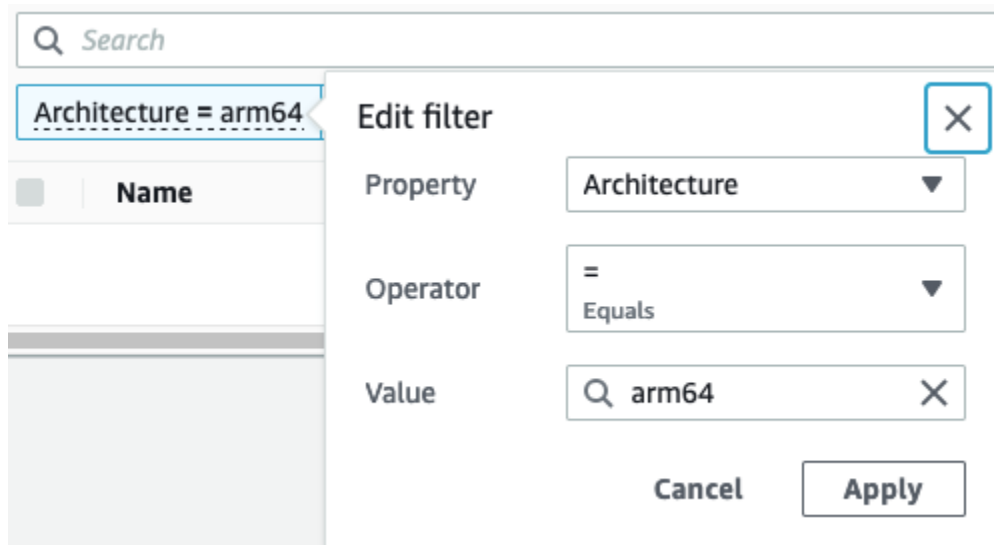
1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中選擇對應資源類型的選項。例如，若要列出執行個體，請選擇 Instances (執行個體)。

此頁面會顯示所選資源類型的所有資源。

## 使用主控台篩選資源

### 篩選資源清單

1. 在導覽窗格中，選取資源類型 (例如，Instances (執行個體))。
2. 選擇搜尋欄位。
3. 從清單中選取篩選條件。
4. 選取運算子，例如 = (等於)。某些屬性有更多可用的運算子可供選取。請注意，並非所有畫面均支援選取運算子。
5. 選取篩選條件值。
6. 若要編輯選取的篩選條件，請選取篩選條件字符 (藍色框)，進行所需的編輯，然後選取 Apply (套用)。請注意，並非所有畫面均支援編輯選取的篩選條件。



7. 完成後，請移除篩選器。

### 支援的篩選條件

Amazon EC2 主控台支援兩種篩選類型。

- API 篩選發生在伺服器端。篩選會套用在 API 呼叫上，這會減少伺服器傳回的資源數目。它允許快速篩選大量資源，並且可以減少伺服器和瀏覽器之間傳輸資料的時間和成本。API 篩選支援 = (等於) 和 : (包含) 運算子，並且一律區分大小寫。
- 用戶端篩選發生在用戶端。它可讓您篩選瀏覽器中已經可用的資料 (也就是 API 已傳回的資料)。用戶端篩選與 API 篩選器結合使用效果很好，可在瀏覽器中篩選至較小的資料集。除了 = (等於) 和 :

(包含) 運算子，用戶端篩選也可以支援範圍運算子，例如  $\geq$  (大於或等於) 和否定 (反向) 運算子，例如  $\neq$  (不等於)。

Amazon EC2 主控台支援以下類型的搜尋：

### 以關鍵字搜尋

以關鍵字搜尋是一種任意文字搜尋，可讓您在所有資源屬性或標籤中搜尋值，而不需指定要搜尋的屬性或標籤金鑰。

#### Note

所有關鍵字搜尋都使用用戶端篩選。

要以關鍵字搜尋，請在搜尋方塊中輸入或貼上您要找的內容，然後選擇 Enter。例如，搜尋 123 會符合任何屬性中具有 123 的所有執行個體，例如 IP 地址、執行個體 ID、VPC ID 或 AMI ID，或在其任何標籤中，例如 Name (名稱)。如果任意文字搜尋傳回非預期的相符項目，請套用其他篩選器。

### 依屬性搜尋

依屬性搜尋可讓您在所有資源中搜尋特定屬性。

#### Note

屬性搜尋會使用 API 篩選或用戶端篩選，視選取的屬性而定。執行屬性搜尋時，屬性會相應地分組。

例如，您可以搜尋所有執行個體的 Instance state (執行個體狀態) 屬性，以僅傳回處於 stopped 狀態的執行個體。若要執行此作業：

1. 在執行個體畫面的搜尋欄位中，開始輸入 Instance state。在您輸入字元時，針對以下執行個體狀態會出現兩種類型的篩選條件：API 篩選條件和用戶端篩選條件。
2. 若要在伺服器側進行搜尋，請在 API filters (API 篩選條件) 下選擇 Instance state (執行個體狀態)。若要在用戶端側搜尋，請在 Client filters (用戶端篩選條件) 下選擇 Instance state (client) (執行個體狀態 (用戶端))。

此時會顯示所選屬性的可能運算子清單。

### 3. 選擇 = (等於) 運算子。

此時會顯示所選屬性和運算子的可能值清單。

### 4. 從清單中選擇 Stopped (已停止)。

## 依標籤搜尋

依標籤搜尋，您可以透過標籤金鑰或標籤值篩選目前顯示的表中的資源。

標籤搜尋使用 API 篩選或者用戶端篩選，具體取決於 Preferences (偏好設定) 視窗中的設定。

### 確保標籤的 API 篩選

1. 開啟 Preferences (偏好設定) 視窗。
2. 清除 Use regular expression matching (使用常規運算式比對) 核取方塊。如果選取此核取方塊，則會執行用戶端篩選。
3. 選取 Use case sensitive matching (使用區分大小寫比對) 核取方塊。如果清除此核取方塊，則會執行用戶端篩選。
4. 選擇確認。

依標籤搜尋時，可以使用下列值：

- (empty) ((空白)) – 尋找具有指定標籤金鑰的所有資源，但必須沒有標籤值。
- All values (所有值) – 尋找具有指定標籤金鑰和任何標籤值的所有資源。
- Not tagged (未標記) – 尋找不含指定標籤金鑰的索引資源。
- 顯示的值 – 尋找具有指定標籤金鑰和指定標籤值的所有資源。

您可以使用下列技巧來強化或精簡搜尋：

## 反向搜尋

反向搜尋可讓您搜尋不符合指定值的資源。在 Instances (執行個體) 和 AMI 畫面中，透過選取 != (不等於) 或 !: (不包含) 運算子，然後選取一個值來執行反向搜尋。在其他畫面中，反向搜尋是透過在搜尋關鍵字前面加上驚嘆號 (!) 字元來執行。

### Note

只有用戶端篩選條件上的關鍵字搜尋和屬性搜尋才支援反向搜尋。API 篩選器上的屬性搜尋不支援此功能。

例如，您可以搜尋所有執行個體的 Instance state (執行個體狀態) 屬性，以排除所有處於 terminated 狀態的執行個體。若要執行此作業：

1. 在執行個體畫面的搜尋欄位中，開始輸入 Instance state。在您輸入字元時，針對以下執行個體狀態會出現兩種類型的篩選條件：API 篩選條件和用戶端篩選條件。
2. 在 Client filters (用戶端篩選條件) 下選擇 Instance state (client) (執行個體狀態 (用戶端))。只有用戶端篩選條件支援反向搜尋。

此時會顯示所選屬性的可能運算子清單。

3. 選擇 != (不等於)，然後選擇 terminated (終止)。

若要根據執行個體狀態屬性來篩選執行個體，您還可以使用 Instance state (執行個體狀態) 欄中的搜尋圖示



帶有加號 (+) 的搜尋圖示會顯示符合該屬性的所有執行個體。帶有減號 (-) 的搜尋圖示會排除符合該屬性的所有執行個體。

以下是使用反向搜尋的另一個範例：若要列出未獲指派名稱為 launch-wizard-1 之安全群組的所有執行個體，請在 Client filters (用戶端篩選條件) 下，依 Security group name (安全群組名稱) 屬性搜尋，選擇 !=，並在搜尋列中輸入 launch-wizard-1。

## 部分搜尋

透過部分搜尋，您可以搜尋部分字串值。若要執行部分搜尋，請僅輸入您要搜尋的關鍵字部分。在 Instances (執行個體) 和 AMIs 畫面中，部分搜尋只能使用 : (包含) 運算子執行。在其他畫面上，可以選取用戶端篩選條件屬性，並立即僅輸入您要搜尋的關鍵字部分。例如，在 Instance type (執行個體類型) 畫面上，若要搜尋所有 t2.micro、t2.small 和 t2.medium 執行個體，請依 Instance Type (執行個體類型) 屬性搜尋，並輸入關鍵字 t2。

## 常規運算式搜尋

若要使用常規運算式搜尋，必須選取 Preferences (偏好設定) 視窗中的 Use regular expression matching (使用常規運算式比對) 核取方塊。

當您需要將欄位中的值與特定模式匹配，就可以使用常規表達式。例如，若要搜尋開頭為 s 的值，請搜尋 ^s。若要搜尋結尾為 xyz 的值，請搜尋 xyz\$。或者，若要搜尋以數字開頭後跟一個或多個字元的值，請搜尋 [0-9]+.\*。

**Note**

只有用戶端篩選器上的關鍵字搜尋和屬性搜尋才支援常規運算式搜尋。API 篩選器上的屬性搜尋不支援此功能。

## 區分大小寫的搜尋

若要使用區分大小寫的搜尋，必須選取 Preferences (偏好設定) 視窗中的 Use case sensitive matching (使用區分大小寫比對) 核取方塊。區分大小寫的偏好設定僅適用於用戶端和標籤篩選條件。

**Note**

API 篩選條件一律區分大小寫。

## 萬用字元搜尋

使用 \* 萬用字元來比對零個或多個字元。使用 ? 萬用字元來比對零個或一個字元。例如，如果您有一個具有值 prod、prods 和 production 的資料集，則搜尋 prod\* 會比對出所有值，而 prod? 僅會比對出 prod 和 prods。若要使用常值，請使用反斜線 (\) 將其逸出。例如，"prod\"\*" 會符合 prod\*。

**Note**

只有 API 篩選器上的屬性和標籤搜尋才支援萬用字元搜尋。用戶端篩選器上的關鍵字搜尋和屬性及標籤搜尋不支援此功能。

## 組合搜尋

一般而言，具有相同屬性的多個篩選器會自動使用 OR 合併。例如，搜尋 Instance State : Running 和 Instance State : Stopped 會傳回執行中「或」已停止的所有執行個體。若要使用 AND 合併搜尋，請跨不同的屬性進行搜尋。例如，搜尋 Instance State : Running 和 Instance Type : c4.large 會傳回只屬於類型 c4.large 「且」處於執行中狀態的執行個體。

## 使用 CLI 和 API 列出和篩選

每種資源類型都有對應的 CLI 命令和 API 動作，可讓您用來列出該類型的資源。產生的資源清單可能很長，因此若要加快速度並提高實用性，請篩選結果，只包含符合特定條件的資源。

### 篩選考量

- 您可以在單一要求中指定多達 50 個篩選條件和每個篩選器最多 200 個值。
- 篩選字串的長度最多可達 255 個字元。
- 您可以使用萬用字元搭配篩選條件值。星號 (\*) 代表零或更多字元，問號 (?) 代表零或一個字元。
- 篩選條件值區分大小寫。
- 您的搜尋可包含萬用字元的文字值；您只需要在字元前加一個反斜線區隔它們即可。例如，\`*amazon\?\` 值會搜尋 `*amazon?\` 文字字串。

### 支援的篩選條件

若要查看每個 Amazon EC2 資源的支援篩選條件，請參閱下列文件：

- AWS CLI：[AWS CLI Command Reference-Amazon EC2](#) 中的 `describe` 命令。
- 適用於視窗的工具 PowerShell：指 `Get` 令程式中的命令 [AWS Tools for PowerShell 令參考資料-Amazon EC2](#)。
- 查詢 API：[Amazon EC2 API 參考](#) 中的 `Describe` API 動作。

### Example 範例：指定單一篩選條件

您可使用 [describe-instances](#) 列出 Amazon EC2 執行個體。在沒有篩選條件的情況下，回應會包含您所有資源的資訊。您可以使用下列命令，只在輸出中包含執行中的執行個體。

```
aws ec2 describe-instances --filters Name=instance-state-name,Values=running
```

若只要列出執行中執行個體的執行個體 ID，請新增 `--query` 參數，如下所示。

```
aws ec2 describe-instances --filters Name=instance-state-name,Values=running --query "Reservations[*].Instances[*].InstanceId" --output text
```

下列為範例輸出。

```
i-0ef1f57f78d4775a4
i-0626d4edd54f1286d
i-04a636d18e83cfacb
```

**Example 範例：** 指定多個篩選條件或篩選條件值

如果您指定多個篩選條件或多個篩選條件值，資源必須符合所有要包含在結果中的篩選條件。

您可以使用下列命令，列出類型為 `m5.large` 或 `m5d.large` 的所有執行個體。

```
aws ec2 describe-instances --filters Name=instance-type,Values=m5.large,m5d.large
```

您可以使用下列命令，列出類型為 `t2.micro` 的所有已停止的執行個體。

```
aws ec2 describe-instances --filters Name=instance-state-name,Values=stopped
Name=instance-type,Values=t2.micro
```

**Example 範例：** 在篩選條件值中使用萬用字元

如果您在使用 [describe-snapshots](#) 描述 EBS 快照時，將 `database` 指定為 `description` 篩選條件的篩選條件值，命令只會傳回描述為「資料庫」的快照。

```
aws ec2 describe-snapshots --filters Name=description,Values=database
```

\* 萬用字元符合零或更多字元。如果您將 `*database*` 指定為篩選條件值，命令只會傳回描述中包含 `database` (資料庫) 此單字的快照。

```
aws ec2 describe-snapshots --filters Name=description,Values=*database*
```

? 萬用字元確切符合 1 個字元。如果您指定 `database?` 做為篩選條件值，命令只會傳回描述為「database」或「database」後面接著一個字元的快照。

```
aws ec2 describe-snapshots --filters Name=description,Values=database?
```

如果您指定 `database????`，命令只會傳回描述為「資料庫」後面接著最多四個字元的快照。它會排除具有「資料庫」後面接著五個或更多字元的描述。

```
aws ec2 describe-snapshots --filters Name=description,Values=database????
```



## Example 範例：根據日期篩選

使用時 AWS CLI，您可以使用 JMESPath 來篩選使用運算式的結果。#####[describe-snapshots](#)##### (# 2020-03-31 ##) ##### AWS ## (# 123456789012 ##) #####  
#####如果您未指定擁有者，結果會包含所有公有快照。

```
aws ec2 describe-snapshots --filters Name=owner-id,Values=123456789012 --query "Snapshots[?(StartTime<='2020-03-31')].[SnapshotId]" --output text
```

下列命令會顯示指定日期範圍中所建立的所有快照的 ID。

```
aws ec2 describe-snapshots --filters Name=owner-id,Values=123456789012 --query "Snapshots[?(StartTime>='2019-01-01') && (StartTime<='2019-12-31')].[SnapshotId]" --output text
```

## 根據標籤篩選

如需如何根據標籤篩選資源清單的範例，請參閱[透過命令列使用標籤](#)。

## 使用 Amazon EC2 全域檢視來檢視跨區域資源

Amazon EC2 全球檢視可讓您在單一主控台中同時檢視和搜尋單一區域或跨多個 AWS 區域的 Amazon EC2 和 Amazon VPC 資源。如需詳細資訊，請參閱 [Amazon EC2 全域檢視](#)。

## Amazon EC2 全域檢視

透過 Amazon EC2 全域檢視，您可以在單一主控台中跨越單一 AWS 區域或多個多個區域檢視一些 Amazon EC2 和 Amazon VPC 資源。Amazon EC2 全域檢視還提供 global search (全域搜尋) 功能，讓您可以同時搜尋多個區域的特定資源或特定資源類型。

Amazon EC2 全域檢視不允許您以任何方式修改資源。

### 支援的資源

您可以使用 Amazon EC2 全域檢視，檢視已啟用您 AWS 帳戶 的所有區域的下列資源的全域摘要。

- Auto Scaling 群組
- DHCP 選項集

- 輸出限定網際網路閘道
- 彈性 IP
- 端點服務
- 執行個體
- 網際網路閘道
- 受管理的字首清單
- NAT 閘道
- 網路 ACL
- 網路介面
- 路由表
- 安全群組
- 子網
- 磁碟區
- VPC
- VPC 端點
- VPC 對等連線

## 所需的許可

使用者必須具備下列許可才能使用 Amazon EC2 全域檢視。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "autoscaling:DescribeAutoScalingGroups",
 "ec2:DescribeRegions",
 "ec2:DescribeDhcpOptions",
 "ec2:DescribeEgressOnlyInternetGateways",
 "ec2:DescribeAddresses",
 "ec2:DescribeVpcEndpointServices",
 "ec2:DescribeInstances",
 "ec2:DescribeInternetGateways",
```

```
"ec2:DescribePrefixLists",
"ec2:DescribeNatGateways",
"ec2:DescribeNetworkAcls",
"ec2:DescribeNetworkInterfaces",
"ec2:DescribeRouteTables",
"ec2:DescribeSecurityGroups",
"ec2:DescribeSubnets",
"ec2:DescribeVolumes",
"ec2:DescribeVpcs",
"ec2:DescribeVpcEndpoints",
"ec2:DescribeVpcPeeringConnections"
],
"Resource": "*"
}]
}
```

## 使用 Amazon EC2 全域檢視

在 <https://console.aws.amazon.com/ec2globalview/home> 開啟 Amazon EC2 全域檢視主控台。

### Important

您不能使用 Firefox 中的私有視窗來存取 Amazon EC2 Global View。

主控台包含下列項目：

- Region explorer (區域總管) - 此索引標籤包含下列部分：

- 摘要：提供所有區域資源的高階概觀。

「啟用區域」表示您 AWS 帳戶已啟用的「區域」數目。其餘欄位表示您目前在這些區域中擁有的資源數目。選擇任一連結，即可檢視所有區域中該類型的資源。例如，如果 Instances (執行個體) 標籤下的連結是 29 in 10 Regions (10 個區域中的 29 個)，則表示您目前跨 10 個區域擁有 29 個執行個體。選擇該連結以檢視全部 29 個執行個體的清單。

- 資源區域計數：列出所有 AWS 區域 (包括未啟用您帳戶的區域)，並針對每個區域提供每種資源類型的總計。

選擇區域名稱，以檢視該特定區域所有類型的所有資源。例如，選擇 Africa (Cape Town) af-south-1 (非洲 (開普敦) af-south-1) 以檢視該區域中的所有 VPC、子網路、執行個體、安全群組、磁碟區和 Auto Scaling 群組。或者，選取一個區域，然後選取 View resources for selected Region (檢視所選區域的資源)。

選擇特定區域中特定資源類型的值，以僅檢視該區域中該類型的資源。例如，選擇非洲 (開普敦) af-south-1 以僅檢視該區域中的執行個體。

- Global search (全域搜尋) - 此索引標籤可讓您跨單一區域或多個區域搜尋特定資源或特定資源類型。它也可讓您檢視特定資源的詳細資訊。

若要搜尋資源，請在方格前面的欄位中輸入搜尋條件。您可以依區域、資源類型以及指派給資源的標籤進行搜尋。

若要檢視特定資源的詳細資訊，請在方格中選取該資源。您也可以選擇要在該各個主控台中開啟的資源 ID。例如，選擇執行個體 ID，以在 Amazon EC2 主控台中開啟執行個體，或選擇子網 ID，以在 Amazon VPC 主控台中開啟子網。

#### Tip

如果您只使用特定區域或資源類型，則可以自訂 Amazon EC2 全域檢視，以僅顯示特定區域和資源類型。若要自訂顯示的區域和資源類型，請在導覽面板中選擇設定，然後在資源和區域索引標籤上，選取您不想在 Amazon EC2 全域檢視中顯示的區域和資源類型。

## 標記您的 Amazon EC2 資源

為協助您管理您的執行個體、映像和其他 Amazon EC2 資源，您可將您自己的中繼資料，以標籤的形式指派給每個資源。標籤可讓您以不同的方式對 AWS 資源進行分類，例如，依目的、擁有者或環境。當您有許多相同類型的資源時，這將會很有用，因為—您可以依據先前指派的標籤，快速識別特定的資源。本主題說明標籤並示範如何建立它們。

#### Warning

許多不同 API 呼叫傳回了標籤金鑰和其值。拒絕存取 DescribeTags 不會自動拒絕存取其他 API 傳回的標籤。根據最佳實務，建議您不要在標籤中包含敏感資料。

### 內容

- [標籤基本概念](#)
- [標記您的 資源](#)
- [標籤限制](#)

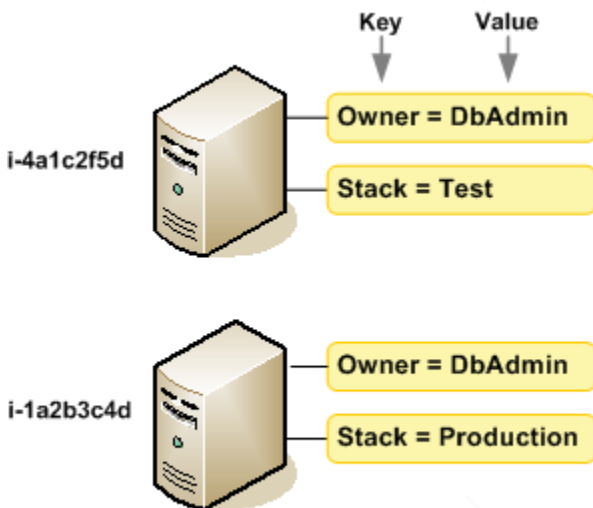
- [標籤與存取管理](#)
- [標記您的資源以便計費](#)
- [透過主控台使用標籤](#)
- [透過命令列使用標籤](#)
- [使用執行個體中繼資料中的執行個體標籤](#)
- [使用將標籤新增至資源 CloudFormation](#)

## 標籤基本概念

標籤是指派給 AWS 資源的標籤。每個標籤皆包含由您定義的一個金鑰與一個選用值。

標籤可讓您以不同的方式對 AWS 資源進行分類，例如，依目的、擁有者或環境。例如，您可以為您帳戶的 Amazon EC2 執行個體定義一組標籤，協助您追蹤各個執行個體的擁有者與堆疊層級。

下圖說明標籤建立的運作方式。在此範例中，您已在每個執行個體上指派兩個標籤 – 一個標籤的金鑰為 Owner，另一個標籤的金鑰為 Stack。每個標籤也都有一個關聯值。



我們建議您為每種資源類型建立符合您需求的標籤金鑰。使用一致的標籤金鑰組可讓您更輕鬆管理您的資源。您可以根據您新增的標籤搜尋和篩選資源。如需有關如何實作有效資源標記策略的詳細資訊，請參閱[標記最佳做法 AWS 白皮書](#)。

標籤對 Amazon EC2 來說不具有任何語意意義，並會嚴格解譯為字元字串。此外，標籤不會自動指派給您的資源。您可以編輯標籤金鑰和值，並且可以隨時從資源移除標籤。您可以將標籤的值設為空白字

串，但您無法將標籤的值設為 Null。若您將與現有標籤具有相同鍵的標籤新增到該資源，則新值會覆寫舊值。如果您刪除資源，也會刪除任何該資源的標籤。

### Note

刪除資源之後，其標籤可能會在主控制台、API 和 CLI 輸出中保持短時可見。這些標籤將逐漸取消與資源的關聯，並永久刪除。

## 標記您的 資源

您可以為您帳戶中現有的大多數 Amazon EC2 資源新增標籤。如下[資料表](#)列出支援標籤建立的資源。

如果您使用的是 Amazon EC2 主控台，可以使用相關資源畫面上的「標籤」索引標籤將標籤套用至資源，也可以使用 AWS Resource Groups 主控台內的標籤編輯器。有些資源畫面可讓您在建立資源時指定資源的標籤；例如，具有 Name 鍵和您指定之值的標籤。在大多數的案例中，主控台會立即在建立資源後套用標籤 (而非在資源建立過程時)。主控台可能會根據 Name 標籤整理資源，但此標籤對 Amazon EC2 服務來說不具有任何語意意義。

如果您使用的是 Amazon EC2 API AWS CLI、或 AWS 開發套件，則可以使用 CreateTags EC2 API 動作將標籤套用至現有資源。此外，有些資源建立動作可讓您在建立資源時指定資源的標籤。若標籤無法在資源建立時套用，我們會轉返資源建立程序。這可確保資源不是具有標籤建立，就是不會建立，因此無論何時都不會有不具有標籤的資源。藉由在建立時為資源建立標籤，您可以消除在資源建立後執行自訂標籤指令碼的必要。如需有關讓使用者在建立時為資源加上標籤的詳細資訊，請參閱 [在建立期間授予標籤資源的許可](#)。

下表說明可以標記的 Amazon EC2 資源，以及可以使用 Amazon EC2 API、或 AWS 開發套件在建立時標記的資源。AWS CLI

### Amazon EC2 資源的標記支援

資源	支援標籤	支援在建立時標記
AFI	是	是
AMI	是	是
Bundle 任務	否	否
Capacity Reservation	是	是

資源	支援標籤	支援在建立時標記
電信業者閘道	是	是
用戶端 VPN 端點	是	是
用戶端 VPN 路由	否	否
客戶閘道	是	是
Dedicated Host	是	是
專用執行個體保留	是	是
DHCP 選項	是	是
EBS 快照	是	是
EBS 磁碟區	是	是
EC2 Fleet	是	是
出口限定網際網路閘道	是	是
彈性 IP 地址	是	是
Elastic Graphics 加速器	是	否
執行個體	是	是
執行個體事件時段	是	是
執行個體存放區磁碟區	N/A	N/A
網際網路閘道	是	是
IP 地址集區 (BYOIP)	是	是
金鑰對 (Key pair)	是	是
啟動範本	是	是

資源	支援標籤	支援在建立時標記
啟動範本版本	否	否
本機閘道	是	否
本機閘道路由表	是	否
本機閘道虛擬介面	是	否
本機閘道虛擬介面群組	是	否
本機閘道路由表 VPC 關聯	是	是
本機閘道路由表虛擬介面群組關聯	是	否
NAT 閘道	是	是
網路 ACL	是	是
網路介面	是	是
置放群組	是	是
字首清單	是	是
Reserved Instance	是	否
預留執行個體清單	否	否
路由表	是	是
Spot Fleet 請求	是	是
Spot 執行個體請求	是	是
安全群組	是	是
安全群組規則	是	否
子網	是	是



資源	支援標籤	支援在建立時標記
流量鏡像篩選條件	是	是
流量鏡像工作階段	是	是
流量鏡像目標	是	是
Transit gateway	是	是
Transit Gateway 多點傳送網域	是	是
Transit gateway 路由表	是	是
Transit gateway VPC 連接	是	是
虛擬私有閘道	是	是
VPC	是	是
VPC 端點	是	是
VPC 端點服務	是	是
VPC 端點服務組態	是	是
VPC 流程日誌	是	是
VPC 對等連線	是	是
VPN 連接	是	是

您可以在 Amazon EC2 主控台中使用 Amazon EC2 [啟動執行個體精靈](#)，在建立過程為執行個體、磁碟區、Elastic Graphics、網路介面和 Spot 執行個體請求建立標籤。您可以使用磁碟區畫面在建立時為 EBS 磁碟區建立標籤，或使用快照畫面為 EBS 快照建立標籤。或者，您也可以使用資源建立的 Amazon EC2 API (例如，[RunInstances](#)) 在建立資源時套用標籤。

您可以在您的 IAM 政策中將標籤式的資源層級許可套用到支援在建立時新增標籤的 Amazon EC2 API 動作，以實作可在建立時為資源建立標籤之使用者和群組的細微控制。您的資源從建立時便已獲得適當保全，由於標籤會立即套用到您的資源，因此控制使用資源的任何標籤式資源層級許可都會立即生效。

您可以更準確的追蹤和報告您的資源。您可以強制新資源使用標籤，並控制哪些標籤金鑰和值會在您的資源上設定。

您也可以將資源層級許可套用到 IAM 政策中的 `CreateTags` 和 `DeleteTags` Amazon EC2 API 動作，控制哪些標籤金鑰和值會在您現有的資源上設定。如需詳細資訊，請參閱 [範例：標籤資源](#)。

如需為您的資源建立標籤以便計費的詳細資訊，請參閱 AWS Billing 使用者指南中的 [Using cost allocation tags](#) (使用成本分配標籤)。

## 標籤限制

以下基本限制適用於標籤：

- 每一資源最多標籤數 - 50
- 對於每一個資源，每個標籤金鑰必須是唯一的，且每個標籤金鑰只能有一個值。
- 索引鍵長度上限 - 128 個 UTF-8 Unicode 字元
- 值的長度上限 - 256 個 UTF-8 Unicode 字元
- 允許的字元
  - 儘管 EC2 允許其標籤中的任何字符，但其他 AWS 服務的限制性更強。所有 AWS 服務中允許的字元包括：字母 (a-z,A-Z)、數字 (0-9) 和 UTF-8 中可表示的空格，以及下列字元：+ - = . \_ : / @.
  - 若啟用執行個體中繼資料中的執行個體標籤，則執行個體標籤鍵值只能使用字母 (a-z、A-Z)、數字 (0-9) 和下列字元：+ - = . , \_ : @。執行個體標籤鍵不可含有空格或 /，而且不能僅包含 . (一個句點)、.. (兩個句點) 或 \_index。如需詳細資訊，請參閱 [使用執行個體中繼資料中的執行個體標籤](#)。
- 標籤金鑰與值皆區分大小寫。
- 前aws:綴保留供 AWS 使用。如果標籤具有此字首的標籤金鑰，則您無法編輯或刪除標籤的金鑰或值。具 aws: 字首的標籤，不算在受資源限制的標籤計數內。

您無法僅根據標籤終止、停止或刪除資源。您必須指定資源識別符。例如，若要刪除您套用稱為 DeleteMe 標籤金鑰的快照，您必須搭配快照的資源識別符 (例如 DeleteSnapshots) 使用 snap-1234567890abcdef0 動作。

當您標記公開或共用資源時，您指派的標籤僅適用於您的 AWS 帳戶；沒有其他 AWS 帳戶可以存取這些標籤。對於以標籤為基礎的共用資源存取控制，每個 AWS 帳號都必須指派自己的標籤集，以控制對資源的存取。

您無法為所有資源新增標籤。如需詳細資訊，請參閱 [Amazon EC2 資源的標記支援](#)。

## 標籤與存取管理

如果您使用的是 AWS Identity and Access Management (IAM)，您可以控制 AWS 帳戶中哪些使用者有權建立、編輯或刪除標籤。如需詳細資訊，請參閱 [在建立期間授予標籤資源的許可](#)。

您也可以使用資源標籤來實作以屬性為基礎的控制 (ABAC)。您可以建立 IAM 政策，允許根據資源的標籤進行操作。如需詳細資訊，請參閱 [使用資源標籤控制對 EC2 資源的存取](#)。

## 標記您的資源以便計費

您可以使用標籤來組織帳 AWS 單，以反映自己的成本結構。為此，請註冊以獲取包含標籤鍵值的 AWS 帳戶帳單。如需有關使用標籤設定成本分配報告的詳細資訊，請參閱 AWS Billing 使用者指南中的 [每月成本分配報告](#)。若想要查看合併資源的成本，您可根據具有相同標籤金鑰值的資源來整理您的帳單資訊。例如，您可以使用特定應用程式名稱來標記數個資源，然後整理帳單資訊以查看該應用程式跨數項服務的總成本。如需詳細資訊，請參閱 AWS Billing 使用者指南中的 [使用成本分配標籤](#)。

### Note

若您才剛啟用報告，目前月份的資料會在 24 小時之後提供檢視。

成本分配標籤可用來表示哪些資源產生成本，但刪除或停用這些資源不一定能降低成本。例如，由另一個快照參考的快照資料會進行保留，即使包含原始資料的快照遭到刪除也不例外。如需詳細資訊，請參閱 AWS Billing 使用者指南中的 [Amazon Elastic Block Store 磁碟區及快照](#)。

### Note

已標記的彈性 IP 地址，不會顯示在您的成本分配報告中。

## 透過主控台使用標籤

您可以使用 Amazon EC2 主控台顯示單個資源的標籤，並一次從一個資源中套用或移除標籤。

您可以使用 AWS Resource Groups 主控台內的標籤編輯器來顯示所有區域中所有 Amazon EC2 資源的標籤。可以根據資源和資源類型來檢視標籤，也可以檢視與特定標籤相關聯的資源類型。您可以從多個資源和多個資源類型中一次套用或移除標籤。標籤編輯器提供集中、統一的方式來建立和管理標籤。若要取得更多資訊，請參閱 [〈標記 AWS 資源使用指南〉](#)。

## 任務

- [顯示標籤](#)
- [新增和刪除個別資源上的標籤](#)
- [新增和刪除多個資源的標籤](#)
- [在您啟動執行個體時新增標籤](#)
- [使用標籤篩選資源清單](#)

## 顯示標籤

您可以在 Amazon EC2 主控台中顯示單個資源的標籤。若要顯示所有資源的標籤，請使用 AWS Resource Groups 主控台內的標籤編輯器。

### 顯示單個資源的標籤

當您在 Amazon EC2 主控台中選取資源限定頁面時，即會顯示那些資源的清單。例如，若您從導覽窗格選取 Instances (執行個體)，主控台即會顯示您的 Amazon EC2 執行個體。當您從這些清單中的其中一個選取資源時 (例如，一個執行個體)，若資源支援標籤，您便可以檢視和管理其標籤。在大多數的資源頁面上，您可以選擇 Tags (標籤) 索引標籤來檢視標籤。

可以為資源清單新增資料欄，以顯示具有相同金鑰的標籤的所有值。您可以使用此資料欄來根據標籤排序和篩選資源清單。

### New console

若要為資源清單新增資料欄以顯示標籤

1. 在 EC2 主控台中，選擇畫面右上角的偏好設定齒輪圖示。
2. 在偏好設定對話方塊中，針對為資料欄加上標籤 (左下角)，選取其中一個標籤索引鍵，然後選擇確認。

### Old console

有兩種方式可為資源清單新增資料行以顯示您的標籤：

- 在 Tags (標籤) 標籤上，選取 Show Column (顯示資料行)。新的資料行即會新增到主控台。
- 選取 Show/Hide Columns (顯示/隱藏資料行) 齒輪狀圖示，並在 Show/Hide Columns (顯示/隱藏資料行) 對話方塊中，在 Your Tag Keys (您的標籤金鑰) 下方選取標籤金鑰。

## 顯示多個資源的標籤

可以使用 [AWS Resource Groups 主控台](#) 中的標籤編輯器，跨多個資源顯示標籤。若要取得更多資訊，請參閱〈[標記 AWS 資源使用指南](#)〉。

## 新增和刪除個別資源上的標籤

您可以直接從資源的頁面管理個別資源的標籤。

### 為個別資源新增標籤

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 請在導覽列中，選取要標記的資源所在的區域。如需詳細資訊，請參閱 [資源位置](#)。
3. 在導覽窗格中，選取資源類型 (例如，Instances (執行個體))。
4. 從資源清單選取資源，然後選取 Tags (標籤) 標籤。
5. 選擇管理標籤，然後選擇新增新的標籤。輸入標籤的金鑰和值。為每個要新增的其他標籤重新選擇新增新的標籤。當您完成新增標籤的作業時，請選擇 Save (儲存)。

### 從個別資源刪除標籤

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 請在導覽列中，選取要取消標記的資源所在的區域。如需詳細資訊，請參閱 [資源位置](#)。
3. 在導覽窗格中，選擇資源類型 (例如，Instances (執行個體))。
4. 從資源清單選取資源，然後選取 Tags (標籤) 標籤。
5. 選擇 Manage tags (管理標籤)。對於每個要移除的標籤，請選擇移除。當您完成移除標籤的作業時，請選擇 Save (儲存)。

## 新增和刪除多個資源的標籤

### 為多個資源新增標籤

1. 在「AWS Resource Groups」主控台中開啟「標籤編輯器」，位於 <https://console.aws.amazon.com/resource-groups/tag-editor>。
2. 針對區域，請選取要標記的資源所在的一個或多個區域。
3. 對於資源類型，請選取要標記的資源類型 (例如，AWS::EC2::Instance)。
4. 選擇搜尋資源。

5. 在資源搜尋結果下，選取要標記之每項資源旁的核取方塊。
6. 選擇管理所選資源的標籤。
7. 在編輯所有選取資源的標籤下，選擇新增標籤，然後輸入新的標籤金鑰和值。為每個要新增的其他標籤重新選擇 Add tag (新增標籤)。

#### Note

若您使用與現有標籤相同的標籤金鑰新增新的標籤，新的標籤會覆寫現有的標籤。

8. 選擇檢閱和套用標籤變更。
9. 選擇 Apply changes to all selected (套用變更到所有已選項目)。

### 從多個資源中移除標籤

1. 在「AWS Resource Groups」主控台中開啟「標籤編輯器」，位於 <https://console.aws.amazon.com/resource-groups/tag-editor>。
2. 針對區域，請選取要取消標記的資源所在的區域。
3. 對於資源類型，請選取要取消標記的資源類型 (例如，AWS::EC2::Instance)。
4. 選擇搜尋資源。
5. 在資源搜尋結果下，選取要取消標記的每個資源旁的核取方塊。
6. 選擇管理所選資源的標籤。
7. 在編輯所有選取資源的標籤下，選擇要移除之標籤旁邊的移除標籤。
8. 選擇檢閱和套用標籤變更。
9. 選擇 Apply changes to all selected (套用變更到所有已選項目)。

### 在您啟動執行個體時新增標籤

#### New console

若要使用啟動執行個體精靈新增標籤

1. 從導覽列，選取執行個體的區域。這項選擇非常重要，因為有些 Amazon EC2 資源可在不同區域間共享，其他則無法。選取符合您的需求的區域。如需詳細資訊，請參閱 [資源位置](#)。
2. 選擇啟動執行個體。

3. 在 Name and tags (名稱和標籤) 下，您可以輸入執行個體的描述性名稱，並指定標籤。

執行個體名稱是一個標籤，其中鍵是 Name (名稱)，而值是您指定的名稱。您可以標記執行個體、磁碟區、Elastic Graphics，以及網路介面。對於 Spot 執行個體，您只能標記 Spot 執行個體請求。

指定執行個體名稱和其他標籤是選用的。

- 對於 Name (名稱)，輸入執行個體的描述性名稱。如果您未指定名稱，則可以透過其 ID 來標識執行個體，該 ID 將在您啟動執行個體時自動產生。
  - 若要新增其他標籤，請選擇 Add additional tags (新增其他標籤)。選取 Add tag (新增標籤)，然後輸入鍵和值，然後選取要標記的資源類型。為每個要新增的其他標籤重新選擇 Add tag (新增標籤)。
4. 在 Application and OS Images (Amazon Machine Image) (應用程式和作業系統映像 (Amazon Machine Image)) 下，選取執行個體的作業系統 (OS) 及 AMI。如需詳細資訊，請參閱 [應用程式和作業系統映像 \(Amazon Machine Image\)](#)。
  5. 在 Key pair (login) (金鑰對 (登入)) 下，針對 Key pair name (金鑰對名稱)，選擇現有的金鑰對或建立新的金鑰對。
  6. 您可以將所有其他欄位保留為預設值，或為您想要的執行個體組態選擇特定值。如需欄位的相關資訊，請參閱 [使用定義的參數啟動執行個體](#)。
  7. 在 Summary (摘要) 面板中，檢閱您的設定，然後選擇 Launch instance (啟動執行個體)。

## Old console

若要使用啟動執行個體精靈新增標籤

1. 從導覽列，選取執行個體的區域。這項選擇非常重要，因為有些 Amazon EC2 資源可在不同區域間共享，其他則無法。選取符合您的需求的區域。如需詳細資訊，請參閱 [資源位置](#)。
2. 選擇 Launch Instance (啟動執行個體)。
3. Choose an Amazon Machine Image (AMI) (選擇 Amazon Machine Image (AMI)) 頁面會顯示基本組態清單，稱為 Amazon Machine Image (AMI)。選取要使用的 AMI，然後選取 Select (選取)。如需詳細資訊，請參閱 [尋找 AMI](#)。
4. 在 Configure Instance Details (設定執行個體詳細資訊) 頁面上，根據需求設定執行個體的設定，然後選擇 Next: Add Storage (下一步：新增儲存體)。
5. 在 Add Storage (新增儲存體) 頁面上，您可以為您的執行個體指定額外的儲存體磁碟區。完成時，選擇 Next: Add Tags (下一步：新增標籤)。

6. 在 Add Tags (新增標籤) 頁面上，指定執行個體、磁碟區或兩者的標籤。選擇 Add another tag (新增另一個標籤) 以為您的執行個體新增超過一個標籤。當完成時，請選擇 Next: Configure Security Group (下一步：設定安全群組)。
7. 在 Configure Security Group (設定安全群組) 頁面上，您可以從您擁有的現有安全群組中選擇，或是讓精靈為您建立新的安全群組。完成時，選擇 Review and Launch (檢閱及啟動)。
8. 檢閱您的設定。當您對您的選項感到滿意時，請選取 Launch (啟動)。選取現有的金鑰對或建立新的金鑰對，選取確認核取方塊，然後選取 Launch Instances (啟動執行個體)。

## 使用標籤篩選資源清單

您可以根據一或多個標籤金鑰和標籤值篩選您的資源清單。

在 Amazon EC2 主控台中按標籤篩選資源清單

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選取資源類型 (例如，Instances (執行個體))。
3. 選擇搜尋欄位。
4. 在清單中的標籤下，選擇標籤金鑰。
5. 從清單中選擇對應的標籤值。
6. 完成後，請移除篩選器。

如需有關在 Amazon EC2 主控台中使用篩選條件的詳細資訊，請參閱 [列出與篩選您的資源](#)。

使用標籤編輯器按標籤來篩選多個區域中的多個資源

您可以使用 AWS Resource Groups 主控台內的標籤編輯器，依標籤篩選多個區域中的多個資源。如需詳細資訊，請參閱《標記 AWS 資源使用者指南》中的 [搜尋要標記的資源](#)。

## 透過命令列使用標籤

建立 EC2 資源時，您可以使用 create 命令的標籤規格參數，將標籤新增至許多 EC2 資源。您可以使用資源的描述命令來檢視資源的標籤。您也可以使用下列命令為現有資源新增、更新或刪除標籤。

任務	AWS CLI	AWS Tools for Windows PowerShell
新增或覆寫一或多個標籤	<a href="#">create-tags</a>	<a href="#">New-EC2Tag</a>



任務	AWS CLI	AWS Tools for Windows PowerShell
刪除一或多個標籤	<a href="#">delete-tags</a>	<a href="#">Remove-EC2Tag</a>
說明一或多個標籤	<a href="#">describe-tags</a>	<a href="#">Get-EC2Tag</a>

## 任務

- [在建立資源時新增標籤](#)
- [將標籤新增到現有資源](#)
- [描述標籤資源](#)

## 在建立資源時新增標籤

下列範例示範如何在您建立資源時套用標籤。

### Note

您在命令列輸入 JSON 格式參數的方式會因您的作業系統而異。

- Linux、macOS 或 Unix 和視窗 PowerShell — 使用單引號 (') 括住 JSON 資料結構。
- Windows – 請在搭配 Windows 命令列使用命令時忽略單引號。

如需詳細資訊，請參閱 [AWS CLI 的 Specifying parameter values \(指定參數值\)](#)。

### Example 範例：啟動執行個體並為執行個體和磁碟區套用標籤

下列 [run-instances](#) 命令會啟動執行個體，並將含有金鑰 **webserver** 和值 **production** 的標籤套用至執行個體。命令也會為任何建立的 EBS 磁碟區 (在此範例為根磁碟區) 套用金鑰為 **cost-center**，值為 **cc123** 的標籤。

```
aws ec2 run-instances \
 --image-id ami-abc12345 \
 --count 1 \
 --instance-type t2.micro \
 --key-name MyKeyPair \
 --tag-specifications 'ResourceType=instance,Tags=[{Key=webserver,Value=production}]' \
 --tag-specifications 'ResourceType=ebs,Tags=[{Key=cost-center,Value=cc123}]'
```

```
--subnet-id subnet-6e7f829e \
--tag-specifications
'ResourceType=instance,Tags=[{Key=webserver,Value=production}]'
'ResourceType=volume,Tags=[{Key=cost-center,Value=cc123}]'
```

您可以在啟動時為執行個體和磁碟區套用相同的標籤金鑰和值。下列命令會啟動執行個體，並為執行個體和任何建立的 EBS 磁碟區套用鍵為 **cost-center**，值為 **cc123** 的標籤。

```
aws ec2 run-instances \
 --image-id ami-abc12345 \
 --count 1 \
 --instance-type t2.micro \
 --key-name MyKeyPair \
 --subnet-id subnet-6e7f829e \
 --tag-specifications 'ResourceType=instance,Tags=[{Key=cost-center,Value=cc123}]'
 'ResourceType=volume,Tags=[{Key=cost-center,Value=cc123}]'
```

Example 範例：建立磁碟區及套用標籤

下列 [create-volume](#) 命令會建立磁碟區，並套用兩個標籤：**purpose=production** 和 **cost-center=cc123**。

```
aws ec2 create-volume \
 --availability-zone us-east-1a \
 --volume-type gp2 \
 --size 80 \
 --tag-specifications 'ResourceType=volume,Tags=[{Key=purpose,Value=production},
{Key=cost-center,Value=cc123}]'
```

將標籤新增到現有資源

下面的例子說明如何使用 [create-tags](#) 命令將標籤新增到現有資源。

Example 範例：將標籤新增至資源

此範例會將標籤 **Stack=production** 新增至指定的映像，或覆寫 AMI 的現有標籤，其中標籤金鑰為 **Stack**。如果命令成功，則不會傳回任何輸出。

```
aws ec2 create-tags \
 --resources ami-78a54011 \
 --tags Key=Stack,Value=production
```

## Example 範例：將標籤新增至多個資源

此範例會為 AMI 和執行個體新增 (或覆寫) 兩個標籤。其中一個標籤只包含金鑰 (**webserver**)，沒有值 (我們會將值設定為空字串)。其他標籤包含金鑰 (**stack**) 與值 (**Production**)。如果命令成功，則不會傳回任何輸出。

```
aws ec2 create-tags \
 --resources ami-1a2b3c4d i-1234567890abcdef0 \
 --tags Key=webserver,Value= Key=stack,Value=Production
```

## Example 範例：新增具有特殊字元的標籤

此範例會將標籤 **[Group]=test** 新增至執行個體。中括號 ([ 和 ]) 是特殊字元，必須將其逸出。

如果您使用 Linux 或 OS X，若要逸出特殊字元，請以雙引號 (") 括住具有特殊字元的元素，然後使用單引號 (') 括住整個金鑰和值結構。

```
aws ec2 create-tags \
 --resources i-1234567890abcdef0 \
 --tags 'Key="[Group]",Value=test'
```

如果您使用 Windows，若要逸出特殊字元，請以雙引號 (") 括住具有特殊字元的元素，然後在每個雙引號字元前面加上反斜線 (\)，如下所示：

```
aws ec2 create-tags ^
 --resources i-1234567890abcdef0 ^
 --tags Key="[Group]",Value=test
```

如果您使用 Windows PowerShell，若要逸出特殊字元，請以雙引號 (") 括住具有特殊字元的值，在每個雙引號字元前加上反斜線 (\)，然後以單引號 (') 括住整個索引鍵和值結構，如下所示：

```
aws ec2 create-tags `
 --resources i-1234567890abcdef0 `
 --tags 'Key="[Group]",Value=test'
```

## 描述標籤資源

下列範例說明如何將篩選與 [describe-instances](#) 搭配使用，以檢視具有特定標籤的執行個體。所有 EC2 描述命令都會使用此語法，在單一資源類型中依標籤篩選。或者，您可以使用 [describe-tags](#) 命令，在 EC2 資源類型中依標籤篩選。

## Example 範例：描述具有指定標籤金鑰的執行個體

下列命令會描述具有 **Stack** 標籤的執行個體，無論標籤的值為何。

```
aws ec2 describe-instances \
 --filters Name=tag-key,Values=Stack
```

## Example 範例：描述具有指定標籤的執行個體

下列命令會描述具有 **Stack=production** 標籤的執行個體。

```
aws ec2 describe-instances \
 --filters Name=tag:Stack,Values=production
```

## Example 範例：描述具有指定標籤值的執行個體

下列命令會描述具有值為 **production** 標籤的執行個體，無論標籤金鑰為何。

```
aws ec2 describe-instances \
 --filters Name=tag-value,Values=production
```

## Example 範例：使用指定的標籤描述所有 EC2 資源

以下命令描述帶有標籤 **Stack=Test** 的所有 EC2 資源。

```
aws ec2 describe-tags \
 --filters Name=key,Values=Stack Name=value,Values=Test
```

## 使用執行個體中繼資料中的執行個體標籤

您可以從執行個體中繼資料存取執行個體的標籤。透過從執行個體中繼資料存取標籤，您無需再使用 `DescribeInstances` 或 `DescribeTags` API 呼叫來擷取標籤資訊，從而減少每秒的 API 交易數，並可讓您的標籤擷取隨控制的執行個體數量進行擴展。此外，在執行個體上執行的本機程序可以直接從執行個體中繼資料檢視執行個體的標籤資訊。

根據預設，不可從執行個體中繼資料取得標籤；您必須明確允許存取。您可以在執行個體啟動時允許存取，或在執行中或已停止執行個體上啟動後允許存取。您也可以透過在啟動範本中指定此許可來允許標籤的存取。使用範本啟動的執行個體允許存取執行個體中繼資料中的標籤。

若新增或移除執行個體標籤，會在執行個體執行期間更新執行個體中繼資料，而無需停止然後啟動執行個體。

## 主題

- [允許存取執行個體中繼資料中的標籤](#)
- [關閉對執行個體中繼資料中標籤的存取](#)
- [檢視是否允許存取執行個體中繼資料中的標籤](#)
- [從執行個體中繼資料擷取標籤](#)

## 允許存取執行個體中繼資料中的標籤

根據預設，無法存取執行個體中繼資料中的執行個體標籤。對於每個執行個體，您必須使用下列其中一種方法來明確允許存取。

允許使用主控台存取執行個體中繼資料中的標籤

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體，然後依次選取 Actions (動作)、Instance settings (執行個體設定)、Allow tags in instance metadata (允許執行個體中繼資料中的標籤)。
4. 要允許存取執行個體中繼資料中的標籤，請選取 Allow (允許) 核取方塊。
5. 選擇儲存。

使用 AWS CLI 允許在啟動時存取執行個體中繼資料中的標籤

使用 [run-instances](#) 命令並將 InstanceMetadataTags 設定為 enabled。

```
aws ec2 run-instances \
 --image-id ami-0abcdef1234567890 \
 --instance-type c3.large \
 ...
 --metadata-options "InstanceMetadataTags=enabled"
```

使用 AWS CLI 允許存取執行中或已停止執行個體上的的執行個體中繼資料中的標籤

使用 [modify-instance-metadata-options](#) 命令，並將 `--instance-metadata-tags` 設定為 enabled。

```
aws ec2 modify-instance-metadata-options \
 --instance-id i-123456789example \
 --instance-metadata-tags enabled
```

```
--instance-metadata-tags enabled
```

## 關閉對執行個體中繼資料中標籤的存取

若要關閉對執行個體中繼資料中的中繼資料標籤的存取，請使用下列其中一種方法。您不需要在啟動時關閉對執行個體中繼資料上的執行個體標籤的存取，因為根據預設，它處於關閉狀態。

使用主控台關閉對執行個體中繼資料中標籤的存取

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體，然後依次選取 Actions (動作)、Instance settings (執行個體設定)、Allow tags in instance metadata (允許執行個體中繼資料中的標籤)。
4. 要允許存取執行個體中繼資料中的標籤，請選擇 Allow (允許) 核取方塊。
5. 選擇儲存。

若要關閉執行個體中繼資料中繼資料中標籤的存取權 AWS CLI

使用 [modify-instance-metadata-options](#) 命令，並將 `--instance-metadata-tags` 設定為 `disabled`。

```
aws ec2 modify-instance-metadata-options \
 --instance-id i-123456789example \
 --instance-metadata-tags disabled
```

## 檢視是否允許存取執行個體中繼資料中的標籤

對於每個執行個體，您可以使用 Amazon EC2 主控台或 AWS CLI 檢視是否允許從執行個體中繼資料存取執行個體標籤。

若要檢視是否允許使用主控台存取執行個體中繼資料中的標籤

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取一個執行個體。
3. 在 Details (詳細資訊) 標籤上，核取 Allow tags in instance metadata (允許執行個體中繼資料中的標籤) 欄位。如果值為 Enabled (已啟用)，則允許執行個體中繼資料中的標籤。如果值為 Disabled (已停用)，則不允許執行個體中繼資料中的標籤。

若要檢視是否允許存取執行個體中繼資料中的標籤，請使用 AWS CLI

使用 [describe-instances](#) 命令並指定執行個體 ID。

```
aws ec2 describe-instances \
 --instance-ids i-1234567890abcdef0
```

以下範例輸出因空間不足而被截斷。此 "InstanceMetadataTags" 參數會指出是否允許執行個體中繼資料中的標籤。如果值為 `enabled`，則允許執行個體中繼資料中的標籤。如果值為 `disabled`，則不允許執行個體中繼資料中的標籤。

```
{
 "Reservations": [
 {
 "Groups": [],
 "Instances": [
 {
 "AmiLaunchIndex": 0,
 "ImageId": "ami-0abcdef1234567890",
 "InstanceId": "i-1234567890abcdef0",
 ...
 }
],
 "MetadataOptions": {
 "State": "applied",
 "HttpTokens": "optional",
 "HttpPutResponseHopLimit": 1,
 "HttpEndpoint": "enabled",
 "HttpProtocolIpv6": "disabled",
 "InstanceMetadataTags": "enabled"
 },
 ...
 }
]
}
```

## 從執行個體中繼資料擷取標籤

如果允許存取執行個體中繼資料中的執行個體標籤，則可以從執行個體中繼資料存取 `tags/instance` 類別。如需如何從執行個體中繼資料擷取標籤的範例，請參閱 [取得執行個體的執行個體標籤](#)。

## 使用將標籤新增至資源 CloudFormation

對於 Amazon EC2 資源類型，您可以使用 `Tags` 或 `TagSpecifications` 屬性來指定標籤。

下列範例會將標籤加入**Stack=Production**至[AWS::EC2::Instance](#)使用其Tags屬性。

Example 範例：YAML 中的標籤

```
Tags:
 - Key: "Stack"
 Value: "Production"
```

Example 範例：JSON 中的標籤

```
"Tags": [
 {
 "Key": "Stack",
 "Value": "Production"
 }
]
```

下列範例會使用其TagSpecifications性質**Stack=Production**將標籤加入至[AWS::EC2::LaunchTemplate LaunchTemplateData](#)。

Example 範例：TagSpecifications 在 YAML 中

```
TagSpecifications:
 - ResourceType: "instance"
 Tags:
 - Key: "Stack"
 Value: "Production"
```

Example 範例：TagSpecifications 在 JSON 中

```
"TagSpecifications": [
 {
 "ResourceType": "instance",
 "Tags": [
 {
 "Key": "Stack",
 "Value": "Production"
 }
]
 }
]
```



# Amazon EC2 服務配額

Amazon EC2 提供不同的資源供您使用。這些資源包含映像、執行個體、磁碟區和快照。當您建立時 AWS 帳戶，我們會針對每個區域設定這些資源的預設配額 (也稱為限制)。例如，您在區域內可啟動的執行個體數目有上限。因此，舉例來說，如果您在美國西部 (奧勒岡) 區域中啟動執行個體，請求不能讓您的用量超過該區域內的執行個體上限。

Service Quotas 主控台是一個集中的位置，您可以在其中檢視和管理 AWS 服務配額，並要求提高您使用的許多資源配額。使用我們提供的配額資訊來管理您的 AWS 基礎架構。計劃在您需要的時候，先行請求提高配額。

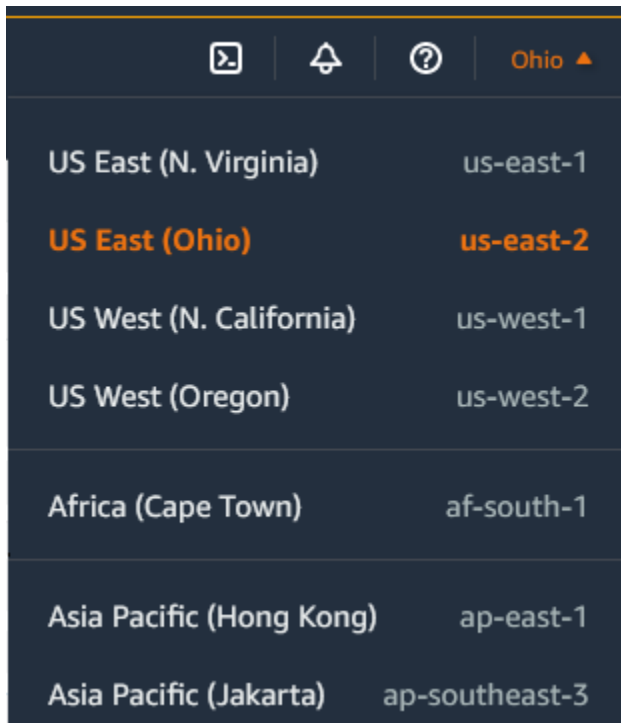
如需詳細資訊，請參閱中的 [Amazon EC2 端點和配額](#) 以及 [Amazon Web Services 一般參考](#)

## 檢視您目前的配額

可以使用 Service Quotas 主控台檢視每個區域的配額。

使用 Service Quotas 主控台檢視目前的配額

1. 開啟 Service Quotas 主控台，網址為 <https://console.aws.amazon.com/servicequotas/home/services/ec2/quotas/>。
2. 從導覽列 (位於畫面頂部) 中，選取一個區域。



3. 使用篩選欄位，按照資源名稱篩選清單。例如，輸入 **On-Demand** 找出隨需執行個體的配額。

4. 若要檢視詳細資訊，請選擇配額名稱以開啟配額的詳細資訊頁面。

## 請求增加

您可以請求增加每個區域的配額。

使用 Service Quotas 主控台請求提高

1. 開啟 Service Quotas 主控台，網址為 <https://console.aws.amazon.com/servicequotas/home/services/ec2/quotas/>。
2. 從導覽列 (位於畫面頂部) 中，選取一個區域。
3. 使用篩選欄位，按照資源名稱篩選清單。例如，輸入 **On-Demand** 找出隨需執行個體的配額。
4. 如果配額可調整，請選取配額，然後選擇請求提高配額。
5. 對於變更配額值，輸入新配額值。
6. 選擇請求。
7. 若要檢視主控台中任何擱置或最近解決的請求，請從導覽窗格中選擇儀表板。對於擱置的請求，請選擇請求狀態以開啟請求回條。請求的初始狀態為 Pending (待定)。狀態變更為「要求配額」後，您會看到案例編號與 AWS Support。選擇案例編號，為請求開啟票證。

如需詳細資訊，包括如何使用 AWS CLI 或 SDK 要求提高配額，請參閱《Service Quotas 使用者指南》中的[要求增加配額](#)。

## 使用通訊埠 25 傳送的電子郵件限制

在所有執行個體上，根據預設，Amazon EC2 會限制連接埠 25 上針對公有 IP 地址的輸出流量。您可以請求移除此限制。如需詳細資訊，請參閱[如何從 Amazon EC2 執行個體或 Lambda 函數移除連接埠 25 的限制？](#)

### Note

此限制不會適用於在連接埠 25 上傳送至下列地址的輸出流量：

- 存在來源網路介面的 VPC 主 CIDR 區塊中的 IP 地址。
- 在 [RFC 1918](#)、[RFC 6598](#) 和 [RFC 4193](#) 中定義的 CIDR 內的 IP 地址。

# 對 EC2 執行個體進行故障診斷

下列程序和秘訣可協助您疑難排解 Amazon EC2 執行個體的問題。

## 目錄

- [Windows 執行個體的常見問題](#)
- [Windows 執行個體的常見訊息](#)
- [針對執行個體啟動問題進行故障診斷](#)
- [連線到 Linux 執行個體的疑難排解](#)
- [疑難排解連接至您的 Windows 執行個體](#)
- [重設遺失或過期的 Windows 管理員密碼](#)
- [無法連線故障診斷執行個體](#)
- [停用執行個體時的疑難排解](#)
- [執行個體終止的故障診斷 \(關機\)](#)
- [使用失敗狀態檢查進行 Linux 執行個體](#)
- [對 Linux 執行個體從錯誤的磁碟區開機](#)
- [疑難排解 Windows 執行個體的系統問題](#)
- [使用適用於 Linux 的 EC2Rescue](#)
- [使用 EC2Rescue for Windows Server](#)
- [亞馬遜 EC2 執行個體的 Amazon EC2 序列主控台](#)
- [傳送診斷中斷 \(僅適用於進階使用者\)](#)

## Windows 執行個體的常見問題

下列的故障診斷秘訣可協助您解決 EC2 Windows Server 執行個體的相關問題。

### 問題

- [EBS 磁碟區不是在 Windows Server 2016 和 2019 和更新版本上初始化](#)
- [在目錄服務還原模式 \(DSRM\) 中開機 EC2 Windows 執行個體](#)
- [執行個體遺失網路連線能力或排程任務未如預期執行](#)
- [無法取得主控台輸出](#)

- [網路不提供 Windows Server 2012 R2](#)
- [磁碟簽章衝突](#)

## EBS 磁碟區不是在 Windows Server 2016 和 2019 和更新版本上初始化

從適用於 Windows Server 2016 和 2019 和更新版本的 Amazon Machine Image (AMI) 建立的執行個體使用 EC2Launch v1 代理程式處理各種啟動任務，包括初始化 EBS 磁碟區。根據預設，EC2Launch v1 不會初始化輔助磁碟區。然而，您可以設定 EC2Launch v1 自動初始化這些磁碟，如下所示。

### 將磁碟機代號映射到磁碟區

1. 連線至執行個體，以設定 C:\ProgramData\Amazon\EC2-Windows\Launch\Config\DriveLetterMappingConfig.json 檔案並在文字編輯器中開啟。
2. 指定磁碟區設定如下：

```
{
 "driveLetterMapping": [
 {
 "volumeName": "sample volume",
 "driveLetter": "H"
 }
]
}
```

3. 儲存您的變更並結束檔案。
4. 開啟視窗，PowerShell 並使用下列命令來執行 EC2Launch v1 指令碼，以初始化磁碟：

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeDisks.ps1
```

若要在每次執行個體開機時初始化磁碟，請新增 -Schedule 標記，如下所示：

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeDisks.ps1 -Schedule
```

EC2Launch v1 代理程式可以執行執行個體初始化指令碼，例如與 InitializeInstance.ps1 指令碼平行處理的 initializeDisks.ps1。如果 InitializeInstance.ps1 指令碼會重新啟動執行個體，它可能會中斷執行個體啟動時執行的其他排程任務。為了避免任何潛在的衝突，我們建議您將邏輯新增至 initializeDisks.ps1 指令碼，以確保執行個體初始化已先完成。

**Note**

如果 EC2Launch 指令碼未初始化磁碟區，請確定磁碟區已上線。如果磁碟區離線，請執行下列命令以使所有磁碟上線。

```
PS C:\> Get-Disk | Where-Object IsOffline -Eq $True | Set-Disk -IsOffline $False
```

## 在目錄服務還原模式 (DSRM) 中開機 EC2 Windows 執行個體

如果執行 Microsoft Active Directory 的執行個體發生系統故障或其他重大問題，您可在稱為目錄服務還原模式(DSRM) 的特殊版本安全模式中開機此執行個體，執行故障診斷。在 DSRM 中，您可修復或復原 Active Directory。

### DSRM 的磁碟機支援

啟用 DSRM 和在執行個體中開機的方式取決於執行個體執行的磁碟機。在 EC2 主控台中，您可在系統日誌中檢閱執行個體的磁碟機版本詳細資訊。下表顯示 DSRM 支援的驅動程式。

驅動程式版本	是否支援 DSRM ?	後續步驟
Citrix PV 5.9	否	從備份還原執行個體。您無法啟用 DSRM。
AWS PV 光伏	否	雖然此驅動程式不支援 DSRM，但您仍然可以分離根磁碟區與無法連線的執行個體、拍攝磁碟區快照或從磁碟區建立 AMI，然後將它連接到同一可用區域的另一個執行個體做為輔助磁碟區。然後即可啟用 DSRM (如本節中所述)。
AWS PV 7.2.2 及更高版本	是	分離根磁碟區、將它連接到其他執行個體，然後啟用 DSRM (如本節中所述)。
增強型聯網	是	分離根磁碟區、將它連接到其他執行個體，然後啟用 DSRM (如本節中所述)。

如需如何啟用增強型網路功能的相關資訊，請參閱[the section called “彈性網路轉接器 \(ENA\)”](#)。如需升級 AWS PV 驅動程式的相關資訊，請參閱[升級 Windows 執行個體上的 PV 驅](#)

## 設定執行個體在 DSRM 中開機

EC2 Windows 執行個體在作業系統執行前沒有網路連線。因此，您無法按鍵盤的 F8 按鈕選取開機選項。您必須使用下列其中一個程序，在 DSRM 中開機 EC2 Windows Server 執行個體。

如果您懷疑 Active Directory 已毀損，但執行個體仍在執行，您可設定此執行個體使用 System Configuration (系統組態) 對話方塊或命令提示在 DSRM 中開機。

使用 System Configuration (系統組態) 對話方塊在 DSRM 中開機線上執行個體

1. 在 Run (執行) 對話方塊中輸入 `msconfig`，然後按 Enter。
2. 選擇 Boot (開機) 標籤。
3. 在 Boot options (開機選項) 下選擇 Safe boot (安全開機)。
4. 選擇 Active Directory repair (Active Directory 修復)，然後選擇 OK (確定)。系統會提示您重新開機伺服器。

使用命令列在 DSRM 中開機線上執行個體

從命令提示視窗執行下列命令：

```
bcdedit /set safeboot dsrepair
```

如果執行個體離線且無法連線，您必須分離根磁碟區，然後將它連接到其他執行個體，以啟用 DSRM 模式。

在 DSRM 中開機離線執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 找出並選取受影響的執行個體。選擇 Instance state (執行個體狀態)、Stop instance (停止執行個體)。
4. 選擇 Launch instances (啟動執行個體)，並在與受影響執行個體相同的可用區域中，建立暫時性的執行個體。選擇使用不同 Windows 版本的執行個體類型。例如，如果您的執行個體是視窗伺服器 2016 年，請選擇視窗伺服器 2019 執行個體。

**⚠ Important**

若您沒有在與受影響的執行個體相同的可用區域內建立執行個體，您將無法將受影響執行個體的根磁碟區連接到新的執行個體。

5. 在導覽窗格中，選擇 Volumes (磁碟區)。
6. 尋找受影響執行個體的根磁碟區。[分離](#)磁碟區，再將其[連接](#)至先前建立的暫時性執行個體。使用預設裝置名稱 (xvdf) 連接它。
7. 使用遠端桌面連接至暫時性執行個體，然後使用 Disk Management 公用程式將[磁碟區變成可用](#)。
8. 開啟命令提示並執行下列命令。使用您剛連接之輔助磁碟區的實際磁碟機代號取代 D：

```
bcdedit /store D:\Boot\BCD /set {default} safeboot dsrepair
```

9. 在磁碟管理公用程式中，選擇先前連接的磁碟機，開啟內容 (按右鍵) 選單，然後選擇 Offline (離線)。
10. 在 EC2 主控台中，將受影響的磁碟區與暫時性執行個體分離，然後將其重新連接至裝置名稱為 /dev/sda1 的執行個體。您必須指定此裝置名稱，將磁碟區指定為根磁碟區。
11. [啟動](#)實例。
12. 在執行個體通過 EC2 主控台的運作狀態檢查後，使用遠端桌面連線到執行個體並確認它在 DSRM 模式中開機。
13. (選用) 刪除或停止您在此程序中建立的暫時性執行個體。

## 執行個體遺失網路連線能力或排程任務未如預期執行

如果您重新啟動您的執行個體但它遺失網路連線能力，可能是執行個體的時間發生錯誤。

Windows 執行個體預設使用國際標準時間 (UTC)。如果您將執行個體的時間設至不同的時區然後重新啟動，就會發生時間位移，且執行個體會暫時遺失 IP 地址。執行個體最後會在重新取得網路連線能力，但這可能需要數小時。執行個體重新取得網路連線能力所需要的時間視 UTC 和其他時間之間的差異而定。

此相同的時間問題也會導致排程任務未如預期執行。在本例中，因為執行個體的時間不正確，所以排程的任務未如預期執行。

若要永久使用 UTC 以外的時區，您必須設定 RealTimeIsUniversal 登錄機碼。沒有這個機碼，執行個體會重新啟動後使用 UTC。

## 解決造成網路連線能力遺失的時間問題

1. 請務必執行建議的 PV 驅動程式。如需詳細資訊，請參閱 [the section called “升級 PV 驅動程式”](#)。
2. 確認下列登錄機碼存在且已設定為 1：HKEY\_LOCAL\_MACHINE\ 系統\ CurrentControl 設定\ 控制 \ 資訊\ TimeZone RealTime IsUniversal

## 無法取得主控台輸出

若為 Windows 執行個體，執行個體主控台會顯示在 Windows 開機程序期間執行的任務輸出。如果 Windows 成功開機，最後記錄的訊息為 Windows is Ready to use。您也可以在主控台中顯示事件記錄訊息，但根據您的 Windows 版本，預設情況下可能不會啟用此功能。如需詳細資訊，請參閱 [the section called “設定 Windows 啟動代理程式”](#)。

若要使用 Amazon EC2 主控台取得執行個體的主控台輸出，請選取執行個體，然後選擇 Actions (動作)、Monitor and troubleshoot (監視和故障診斷)、Get system log (取得系統記錄檔)。要使用命令行獲取控制台輸出，請使用以下命令之一：[獲取控制台輸出](#) ( ) 或 ( ) AWS CLI。[Get-EC2ConsoleOutput](#) AWS Tools for Windows PowerShell

若是執行 Windows Server 2012 R2 或更舊版本的執行個體，如果主控台輸出是空的，可能表示 EC2Config 服務發生問題，例如設定錯誤的組態檔或 Windows 無法正確開機。若要修復此問題，請下載並安裝最新版的 EC2Config。如需詳細資訊，請參閱 [the section called “安裝 EC2Config”](#)。

## 網路不提供 Windows Server 2012 R2

如需疑難排解無法在網路上使用的 Windows 伺服器 2012 R2 執行個體的相關資訊，請參閱 [Windows 伺服器 2012 R2 在執行個體重新開機後中斷網路和儲存區連線](#)。

## 磁碟簽章衝突

您可以使用 [EC2Rescue for Windows Server](#)，檢查並解決磁碟簽章衝突。或者，您可以執行下列步驟，手動解決磁碟簽章問題。

### Warning

下列程序說明如何使用登錄編輯程式來編輯 Windows 登錄檔。如果您不熟悉 Windows 登錄檔或如何使用登錄編輯程式來安全地進行變更，請參閱 [設定登錄檔](#)。

1. 開啟命令提示，輸入 regedit.exe，然後按 Enter。



2. 在登錄編輯程式中，從內容功能表中選擇 HKEY\_LOCAL\_MACHINE (按一下右鍵)，然後選擇 Find (尋找)。
3. 輸入 Windows Boot Manager，然後選擇 Find Next (尋找下一個)。
4. 選擇名為 11000001 的金鑰。此金鑰是您在上一個步驟中找到的金鑰的同級。
5. 在右窗格中，選擇 Element，然後從內容功能表中選擇 Modify (修改)(按一下右鍵)。
6. 在資料中的位移為 0x38 處找到四位元組的磁碟簽章。這是開機組態資料庫簽章 (BCD)。反轉位元組以建立磁碟簽章，然後寫下來。例如，下列資料所代表的磁碟簽章是 E9EB3AA5：

```
...
0030 00 00 00 00 01 00 00 00
0038 A5 3A EB E9 00 00 00 00
0040 00 00 00 00 00 00 00 00
...
```

7. 在 [命令提示字元] 視窗中，執行下列命令以啟動 Microsoft DiskPart。

```
diskpart
```

8. 執行指select disk DiskPart 令並指定發生磁碟簽章衝突的磁碟區的磁碟編號。

#### Tip

若要查看發生磁碟簽章衝突的磁碟區的磁碟編號，請使用磁碟管理公用程式。開啟命令提示，輸入 `compmgmt.msc`，然後按 Enter。在左側導覽面板中，按兩下磁碟管理。在磁碟管理公用程式中，查看發生磁碟簽章衝突的離線磁碟區的磁碟編號。

```
DISKPART> select disk 1
Disk 1 is now the selected disk.
```

9. 運行以下 DiskPart 命令以獲取磁盤簽名。

```
DISKPART> uniqueid disk
Disk ID: 0C764FA8
```

10. 如果上一步中顯示的磁碟簽章與您先前寫下的磁碟簽章不符，請使用下列 DiskPart 命令來變更磁碟簽章，使其符合：

```
DISKPART> uniqueid disk id=E9EB3AA5
```

## Windows 執行個體的常見訊息

本節包含根據常見訊息協助您故障診斷問題的秘訣。

### 訊息

- [「未提供密碼」](#)
- [「尚未提供密碼」](#)
- [「無法擷取 Windows 密碼」](#)
- [等待中繼資料服務](#)
- [無法啟動 Windows](#)
- [Windows 不是正版 \(0x80070005\)](#)
- [無終端機伺服器的授權伺服器可提供授權](#)
- [「部分設定是由您的組織所管理」](#)

### 「未提供密碼」

若要使用遠端桌面連線到 Windows 執行個體，您必須指定帳戶和密碼。提供的帳戶和密碼是以您啟動執行個體所用的 AMI 為基礎。您可擷取管理員帳戶自動產生的密碼，或使用 AMI 建立來源之原始執行個體中使用的帳戶和密碼。

您可以為使用自訂 Windows AMI 啟動的執行個體產生管理員帳戶密碼。要產生密碼，您需要在建立 AMI 之前在作業系統中配置一些設定。如需詳細資訊，請參閱 [創建一個 Amazon EBS-backed AMI](#)。

如果您的 Windows 執行個體未設定產生隨機密碼，當您使用主控台擷取自動產生的密碼時，就會收到以下訊息：

```
Password is not available.
The instance was launched from a custom AMI, or the default password has changed. A
password cannot be retrieved for this instance. If you have forgotten your password,
you can
reset it using the Amazon EC2 configuration service. For more information, see
Passwords for a
```

```
Windows Server instance.
```

檢查執行個體的主控制台輸出，查看您啟動執行個體所用的 AMI，建立時是否停用產生密碼。如果停用產生密碼，主控台輸出會包含以下內容：

```
Ec2SetPassword: Disabled
```

如果停用產生密碼而您不記得原始執行個體的密碼，您可重設此執行個體的密碼。如需詳細資訊，請參閱 [重設遺失或過期的 Windows 管理員密碼](#)。

## 「尚未提供密碼」

若要使用遠端桌面連線到 Windows 執行個體，您必須指定帳戶和密碼。提供的帳戶和密碼是以您啟動執行個體所用的 AMI 為基礎。您可擷取管理員帳戶自動產生的密碼，或使用 AMI 建立來源之原始執行個體中使用的帳戶和密碼。

您應該會在幾分鐘內收到密碼。如果未提供密碼，當您使用主控台擷取自動產生的密碼時，就會收到以下訊息：

```
Password not available yet.
Please wait at least 4 minutes after launching an instance before trying to retrieve
the
auto-generated password.
```

如果超過四分鐘仍未收到密碼，則可能是執行個體的啟動代理程式未設定為產生密碼。請檢查主控台輸出是否為空以確認。如需詳細資訊，請參閱 [無法取得主控台輸出](#)。

同時確認用於存取管理入口網站的 AWS Identity and Access Management (IAM) 帳戶是否具有允許的 `ec2:GetPasswordData` 動作。如需 IAM 許可的詳細資訊，請參閱 [什麼是 IAM?](#)

## 「無法擷取 Windows 密碼」

若要擷取管理員帳戶的自動產生密碼，您必須使用啟動執行個體時所指定之金鑰對的私有金鑰。如果您在啟動執行個體時未指定金鑰對，就會收到下列訊息。

```
Cannot retrieve Windows password
```

您可以使用相同的 AMI，確定指定金鑰對，終止此執行個體並啟動新的執行個體。

## 等待中繼資料服務

Windows 執行個體必須先從自己的執行個體中繼資料取得資訊，才能自行啟動。根據預設，WaitForMetaDataAvailable 設定會確定 EC2Config 服務會先等待執行個體中繼資料變成可存取，才繼續開機程序。如需詳細資訊，請參閱 [使用執行個體中繼資料](#)。

如果執行個體的中繼資料連線能力測試失敗，請嘗試下列方法解決此問題。

- 檢查您 VPC 的 CIDR 區塊。如果在 IP 地址範圍由 224.0.0.0 到 255.255.255.255 (類別 D 和類別 E IP 地址範圍) 的 VPC 中啟動執行個體，Windows 執行個體即無法正確開機。這些是保留的 IP 地址範圍，不應該指派給主機裝置。我們建議您使用來自私有 (非可公開路由) IP 地址範圍 (如 [RFC 1918](#) 中所指定) 的 CIDR 區塊來建立 VPC。
- 系統可能已以靜態 IP 地址設定。請嘗試[建立網路介面](#)並[連接至執行個體](#)。
- 在您無法連線之 Windows 執行個體上啟用 DHCP
  1. 停止受影響的執行個體並分離其根磁碟區。
  2. 啟動與受影響執行個體同一可用區域中的暫時執行個體。

### Warning

如果您的暫時執行個體與原始執行個體以相同的 AMI 為基礎，則您必須完成額外的步驟，否則您無法在還原原始執行個體磁碟區之後將其啟動，原因是磁碟簽名衝突。或者，請為暫時執行個體選擇不同的 AMI。例如，如果原始執行個體使用 AWS 視窗 AMI 適用於視窗伺服器 2016 年，請使用 AWS 視窗 AMI 啟動暫存執行個體。

3. 從受影響的執行個體將根磁碟區連接到此暫時執行個體。連線至暫時執行個體、開啟 Disk Management (磁碟管理) 公用程式，然後讓磁碟機上線。
4. 從暫時執行個體，開啟 Regedit 並選取 HKEY\_LOCAL\_MACHINE。從 File (檔案) 功能表中，選擇 Load Hive (載入 Hive)。選取磁碟機、開啟檔案 Windows\System32\config\SYSTEM，然後在提示時指定金鑰名稱 (您可使用任何名稱)。
5. 選取剛載入的機碼並導覽至 ControlSet001\Services\Tcpip\Parameters\Interfaces。每個網路介面均按 GUID 列出。選取正確的網路介面。如已停用 DHCP 並指派了靜態 IP 地址，EnableDHCP 設為 0。若要啟用 DHCP，請將 EnableDHCP 設為 1，然後刪除下列機碼，如果它們存在：NameServer、SubnetMask、IPAddress 和 DefaultGateway。再次選取該金鑰，然後從 File (檔案) 功能表中選擇 Unload Hive (取消載入 Hive)。

**Note**

如有多個網路介面，您需要找出正確的介面來啟用 DHCP。若要找出正確的網路介面，請檢閱下列機碼值：NameServer、SubnetMask、IPAddress 和 DefaultGateway。這些值會顯示前一個執行個體的靜態組態。

6. (選用) 如已啟用 DHCP，您可能沒有中繼資料服務的路由。更新 EC2Config 可解決此問題。
  - a. [下載](#)並安裝最新版的 EC2Config 服務。如需安裝此服務的詳細資訊，請參閱 [the section called “安裝 EC2Config”](#)。
  - b. 將 .zip 檔案中的檔案解壓縮到所連接之磁碟機上的 Temp 目錄中。
  - c. 開啟 Regedit 並選取 HKEY\_LOCAL\_MACHINE。從 File (檔案) 功能表中，選擇 Load Hive (載入 Hive)。選取磁碟機、開啟檔案 Windows\System32\config\SOFTWARE，然後在提示時指定金鑰名稱 (您可使用任何名稱)。
  - d. 選取剛載入的機碼並導覽至 Microsoft\Windows\CurrentVersion。選取 RunOnce 機碼。(如果此機碼不存在，請在 CurrentVersion 按一下滑鼠右鍵、指向 New (新增)、選取 Key (機碼) 並命名機碼 RunOnce。) 按一下滑鼠右鍵、指向 New (新增)，然後選取 String Value (字串值)。在名稱輸入 Ec2Install，在資料輸入 C:\Temp\Ec2Install.exe -q。
  - e. 再次選取該金鑰，然後從 File (檔案) 功能表中選擇 Unload Hive (取消載入 Hive)。
7. (選用) 如果您的暫時執行個體與原始執行個體以相同的 AMI 為基礎，則您必須完成下列步驟，否則您無法在還原原始執行個體磁碟區之後將其啟動，原因是磁碟簽名衝突。

**Warning**

下列程序說明如何使用登錄編輯程式來編輯 Windows 登錄檔。如果您不熟悉 Windows 登錄檔或如何使用登錄編輯程式來安全地進行變更，請參閱 [設定登錄檔](#)。

- a. 開啟命令提示，輸入 regedit.exe，然後按 Enter。
- b. 在登錄編輯程式中，從內容功能表中選擇 HKEY\_LOCAL\_MACHINE (按一下右鍵)，然後選擇 Find (尋找)。
- c. 輸入 Windows Boot Manager，然後選擇 Find Next (尋找下一個)。
- d. 選擇名為 11000001 的金鑰。此金鑰是您在上一個步驟中找到的金鑰的同級。
- e. 在右窗格中，選擇 Element，然後從內容功能表中選擇 Modify (修改)(按一下右鍵)。

- f. 在資料中的位移為 0x38 處找到四位元組的磁碟簽章。反轉位元組以建立磁碟簽章，然後寫下來。例如，下列資料所代表的磁碟簽章是 E9EB3AA5：

```
...
0030 00 00 00 00 01 00 00 00
0038 A5 3A EB E9 00 00 00 00
0040 00 00 00 00 00 00 00 00
...
```

- g. 在 [命令提示字元] 視窗中，執行下列命令以啟動 Microsoft DiskPart。

```
diskpart
```

- h. 執行下列 DiskPart 命令以選取磁碟區。(可以使用磁碟管理公用程式，確認磁碟編號為 1)。

```
DISKPART> select disk 1

Disk 1 is now the selected disk.
```

- i. 運行以下 DiskPart 命令以獲取磁盤簽名。

```
DISKPART> uniqueid disk

Disk ID: 0C764FA8
```

- j. 如果上一步中顯示的磁碟簽章與您先前寫下的 BCD 的磁碟簽章不符，請使用下列 DiskPart 命令來變更磁碟簽章，使其符合：

```
DISKPART> uniqueid disk id=E9EB3AA5
```

8. 使用 Disk Management (磁碟管理) 公用程式讓磁碟機離線。

#### Note

如果暫時執行個體執行的作業系統與受影響執行個體的相同，磁碟機會自動離線，因此您不需要手動將其離線。

9. 分離磁碟區與暫時執行個體。如果不再需要使用此暫時執行個體，您可終止它。
10. 將磁碟區連接為 /dev/sda1 來還原受影響執行個體的根磁碟區。
11. 啟動受影響的執行個體。

如果您已連線到此執行個體，請由此執行個體開啟網際網路瀏覽器，然後輸入下列中繼資料伺服器的 URL：

```
http://169.254.169.254/latest/meta-data/
```

如果無法聯絡中繼資料伺服器，請嘗試下列方法解決問題：

- [下載](#)並安裝最新版的 EC2Config 服務。如需安裝此服務的詳細資訊，請參閱 [the section called “安裝 EC2Config”](#)。
- 檢查 Windows 執行個體是否正在執行 RedHat 光伏驅動程式。如果是，請更新至 Citrix PV 驅動程式。如需詳細資訊，請參閱 [the section called “升級 PV 驅動程式”](#)。
- 確認防火牆、IPSec 和代理伺服器設定不會封鎖中繼資料服務 (169.254.169.254) 或 AWS KMS 伺服器的傳出流量 (地址指定在 TargetKMSServer 的 C:\Program Files\Amazon\Ec2ConfigService\Settings\ActivationSettings.xml 元素中)。
- 使用下列命令確認您有中繼資料服務的路由 (169.254.169.254)。

```
route print
```

- 檢查是否有可能影響您執行個體可用區域的網路問題。請前往 <http://status.aws.amazon.com/>。

## 無法啟動 Windows

視窗執行個體使用視窗 AWS KMS 啟用。您可以收到以下訊息：A problem occurred when Windows tried to activate. Error Code 0xC004F074，如果您的執行個體無法連線到 AWS KMS 伺服器。Windows 必須每 180 天啟動一次。EC2Config 會嘗試在啟動期間到期之前聯絡 AWS KMS 伺服器，以確保 Windows 保持啟動狀態。

如果您發生 Windows 啟用問題，請使用下列程序予以解決。

EC2Config (Windows Server 2012 R2 AMI 及舊版)

1. [下載](#)並安裝最新版的 EC2Config 服務。如需安裝此服務的詳細資訊，請參閱 [the section called “安裝 EC2Config”](#)。
2. 登入執行個體並開啟下列檔案：C:\Program Files\Amazon\Ec2ConfigService\Settings\config.xml。
3. 在檔案中找到 EC2 WindowsActivate 外掛程config.xml式。將狀態變更為 Enabled (已啟用) 並儲存變更。

4. 在 Windows 服務嵌入式管理單元中，重新啟動 EC2Config 服務或重新開機執行個體。

如果這無法解決啟用問題，請遵循下列其他步驟。

1. 設定 AWS KMS 目標：C:\> slmgr.vbs /skms 169.254.169.250:1688
2. 啟動 Windows：C:\> slmgr.vbs /ato

#### EC2Launch (Windows Server 2016 AMI 及更新版本)

1. 從具有管理權限的 PowerShell 提示中，匯入 EC2Launch 模組：

```
PS C:\> Import-Module "C:\ProgramData\Amazon\EC2-Windows\Launch\Module\Ec2Launch.psd1"
```

2. 呼叫新增路由函數以查看新路由清單：

```
PS C:\> Add-Routes
```

3. 調用 Set-ActivationSettings 函數：

```
PS C:\> Set-Activationsettings
```

4. 接著執行下列指令碼以啟動 Windows：

```
PS C:\> cscript "${env:SYSTEMROOT}\system32\slmgr.vbs" /ato
```

針對 EC2Config 和 EC2Launch，如果您仍收到啟用錯誤，請確認下列資訊。

- 確認您具有通往 AWS KMS 伺服器的路由。開啟 C:\Program Files\Amazon\Ec2ConfigService\Settings\ActivationSettings.xml 並找到 TargetKMSServer 元素。執行下列命令，並檢查是否列出這些 AWS KMS 伺服器的位址。

```
route print
```

- 確認已設定 AWS KMS 用戶端金鑰。執行下列命令並檢查輸出。

```
C:\Windows\System32\slmgr.vbs /dlv
```



如果輸出包含「錯誤：找不到產品金鑰」，表示未設定 AWS KMS 用戶端金鑰。如果未設定 AWS KMS 用戶端金鑰，請依照以下 Microsoft 文章：用戶端[安裝金鑰所述查詢AWS KMS 用戶端金鑰](#)，然後執行下列命令來設定 AWS KMS 用戶端金鑰。

```
C:\Windows\System32\slmgr.vbs /ipk client_key
```

- 確認系統的時間和時區是否正確。如果您使用的是 UTC 以外的時區，請新增下列登錄機碼並將 1 其設定，以確保時間正確：HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Control\TimeZoneInformation\RealTimeIsUniversal。
- 如已啟用 Windows 防火牆，請使用下列命令暫時停用它。

```
netsh advfirewall set allprofiles state off
```

## Windows 不是正版 (0x80070005)

視窗執行個體使用視窗 AWS KMS 啟用。如果執行個體無法完成啟用程序，會回報所用 Windows 不是正版。

請嘗試[無法啟動 Windows](#)的建議。

## 無終端機伺服器的授權伺服器可提供授權

Windows Server 預設透過遠端桌面可同時授權兩名使用者使用。如果您需要透過遠端桌面提供兩位以上的使用者同時存取您的 Windows 執行個體，您可購買遠端桌面服務客戶存取授權 (CAL) 並安裝遠端桌面工作階段主機和遠端桌面工作階段授權伺服器角色。

檢查是否有下列問題：

- 您已超過可同時執行的 RDP 工作階段上限。
- 您已安裝 Windows 遠端桌面服務角色。
- 授權已過期。如果授權已過期，您就無法以使用者身分連線到您的 Windows 執行個體。您可以嘗試下列方法：
  - 使用 /admin 參數從命令列連線至執行個體，例如：

```
mstsc /v:instance /admin
```

如需詳細資訊，請參閱下列 Microsoft 文章：[Access Remote Desktop Via Command Line](#)。

- 停止執行個體、分離其 Amazon EBS 磁碟區，然後將它們連接到同一可用區域的另一個執行個體，以復原您的資料。

## 「部分設定是由您的組織所管理」

從最新 Windows Server AMI 啟動的執行個體可能顯示 Windows Update 對話方塊訊息，其中指出「部分設定是由您的組織所管理」。此訊息是因為 Windows Server 中的變更而出現，不影響 Windows Update 的行為或造成您無法管理更新設定。

### 如何移除警告

1. 開啟 `gpedit.msc` 並導覽至電腦組態 > 系統管理範本 > Windows 元件 > Windows 更新。編輯設定自動更新，然後將其設為已啟用。
2. 在命令提示字元中，使用 `gpupdate /force` 更新群組政策。
3. 關閉「Windows Update 設定」再重新開啟。您將看到上述訊息表示您的設定正由您的組織所管理，後面接著「我們將自動下載更新，付費連線的情況除外 (因為可能需要支付費用)。在該種情況下，我們會自動下載保持 Windows 順暢運作所需的這些更新。」
4. 返回 `gpedit.msc`，然後將群組政策設回尚未設定。再次執行 `gpupdate /force`。
5. 關閉命令提示字元並等待幾分鐘。
6. 重新開啟「Windows Update 設定」。您應該不會看到「部分設定是由您的組織所管理」這個訊息。

## 針對執行個體啟動問題進行故障診斷

下列問題會導致您無法啟動執行個體。

### 啟動問題

- [無效裝置名稱](#)
- [超過執行個體限制](#)
- [執行個體容量不足](#)
- [目前並不支援要求的組態。請查看文件以了解支援的組態。](#)
- [立刻終止執行個體](#)
- [許可不足](#)

- [視窗啟動後不久的 CPU 使用率過高 \(僅限 Windows 執行個體\)](#)

## 無效裝置名稱

### 描述

當您嘗試啟動新的執行個體時，發生 Invalid device name *device\_name* 錯誤。

### 原因

如果您在嘗試啟動執行個體時收到此錯誤，表示在請求中為一個或多個磁碟區指定的裝置名稱具有無效的裝置名稱。可能的原因包括：

- 選取的 AMI 可能正在使用裝置名稱。
- 裝置名稱可能會保留給根磁碟區。
- 裝置名稱可能會用於請求中的其他磁碟區。
- 裝置名稱可能對作業系統無效。

### 解決方案

要解決問題：

- 請確保您選取的 AMI 中未使用裝置名稱。執行下列命令以檢視 AMI 使用的裝置名稱。

```
aws ec2 describe-images --image-id ami_id --query
'Images[*].BlockDeviceMappings[].DeviceName'
```

- 請確保您沒有使用為根磁碟區保留的裝置名稱。如需詳細資訊，請參閱 [可用裝置名稱](#)。
- 請確保請求中指定的每個磁碟區都有唯一的裝置名稱。
- 請確保您指定的裝置名稱格式正確。如需詳細資訊，請參閱 [可用裝置名稱](#)。

## 超過執行個體限制

### 描述

當您嘗試啟動新的執行個體或重新啟動已停止的執行個體時，發生 InstanceLimitExceeded 錯誤。

## 原因

當您嘗試啟動新的執行個體或重新啟動已停止的執行個體時，如果發生 `InstanceLimitExceeded` 錯誤，就表示已達到您在區域中可啟動的執行個體數目限制。建立 AWS 帳戶時，我們會針對每個區域可執行的執行個體數量設定預設限制。

## 解決方案

您可以要求提高每一區域的執行個體限制。如需詳細資訊，請參閱 [Amazon EC2 服務配額](#)。

## 執行個體容量不足

### 描述

當您嘗試啟動新的執行個體或重新啟動已停止的執行個體時，發生 `InsufficientInstanceCapacity` 錯誤。

### 原因

當您嘗試啟動執行個體或重新啟動已停止的執行個體時，如果發生此錯誤，就表示 AWS 目前可用的隨需容量不足，無法滿足您的請求。

### 解決方案

若要解決問題，請嘗試下列方法：

- 等候幾分鐘，然後再次提交您的請求；容量會頻繁轉移。
- 以降低的執行個體數提交新請求。例如，如果您要提出一次啟動 15 個執行個體的請求，請改嘗試提出 3 個啟動 5 個執行個體的請求，或 15 個啟動 1 個執行個體的請求。
- 如果啟動執行個體，請提交新的請求，而不要指定可用區域。
- 如果啟動執行個體，請使用不同的執行個體類型 (您可以在後面的階段調整大小) 來提交新的請求。如需詳細資訊，請參閱 [變更執行個體類型](#)。
- 如果透過叢集置放群組來啟動執行個體，可能會出現容量不足的錯誤。如需詳細資訊，請參閱 [使用置放群組](#)。

目前並不支援要求的組態。請查看文件以了解支援的組態。

### 描述

當您嘗試啟動新的執行個體時，由於不支援執行個體組態，因此發生 `Unsupported` 錯誤。

## 原因

錯誤訊息會提供其他詳細資訊。例如，指定的區域或可用區域可能不支援執行個體類型或執行個體購買選項。

## 解決方案

請嘗試不同的執行個體組態。若要搜尋符合您需求的執行個體類型，請參閱 [尋找 Amazon EC2 執行個體類型](#)。

## 立刻終止執行個體

### 描述

您的執行個體會從 pending 狀態進入 terminated 狀態。

### 原因

下列是執行個體為什麼可能會立即終止的幾個原因：

- 您已超過 EBS 磁碟區限制。如需詳細資訊，請參閱 [執行個體磁碟區限制](#)。
- EBS 快照已損毀。
- 根 EBS 磁碟區已加密，且您沒有存取 KMS 金鑰進行解密的許可。
- AMI 的區塊型設備映射中指定的快照已加密，且您沒有存取 KMS 金鑰進行解密的許可，或您無法存取 KMS 金鑰來加密還原的磁碟區。
- 您用來啟動執行個體的執行個體存放區後端 AMI 缺少必要部分 (an image.part.xx 檔案)。

如需詳細資訊，請使用下列其中一種方法來查明終止原因。

使用 Amazon EC2 主控台查明終止原因

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)，然後選取該執行個體。
3. 在第一個索引標籤上，找出狀態轉換原因旁的的原因。

若要取得終止原因，請使用 AWS Command Line Interface

1. 使用 [describe-instances](#) 命令並指定執行個體 ID。

```
aws ec2 describe-instances --instance-id instance_id
```

2. 檢閱命令傳回的 JSON 回應，並記下 StateReason 回應元素中的值。

下列程式碼區塊顯示 StateReason 回應元素的範例。

```
"StateReason": {
 "Message": "Client.VolumeLimitExceeded: Volume limit exceeded",
 "Code": "Server.InternalError"
},
```

要獲取終止原因 AWS CloudTrail

如需詳細資訊，請參閱AWS CloudTrail 使用指南中的[檢視具有 CloudTrail 事件歷程記錄](#)的事件。

## 解決方案

根據終止原因，採取下列其中一項動作：

- **Client.VolumeLimitExceeded: Volume limit exceeded** — 刪除未使用的磁碟區。您可以[提交申請](#)來增加磁碟區容量上限。
- **Client.InternalError: Client error on launch**— 確定您具有存取 AWS KMS keys 用於解密和加密磁碟區的所需權限。如需詳細資訊，請參閱 AWS Key Management Service 開發人員指南中的 [在 AWS KMS中使用金鑰政策](#)。

## 許可不足

### 描述

當您嘗試啟動新的執行個體時發生 "*errorMessage*": "You are not authorized to perform this operation." 錯誤，並且啟動失敗。

### 原因

如果在嘗試啟動執行個體時發生此錯誤，表示您沒有啟動執行個體所需的 IAM 許可。

可能缺少的許可包括：

- `ec2:RunInstances`
- `iam:PassRole`

可能還缺少其他許可。如需啟動執行個體所需的許可清單，請參閱 [範例：使用 EC2 啟動執行個體精靈](#) 和 [啟動執行個體 \(RunInstances\)](#) 下方的 IAM 政策範例。

## 解決方案

要解決問題：

- 如果您以 IAM 使用者身分提出請求，請確認您具有以下許可：
  - `ec2:RunInstances` 使用萬用字元資源 (「\*」)
  - `iam:PassRole` 與資源符合角色 ARN (例如，`arn:aws:iam::999999999999:role/ExampleRoleName`)
- 如果沒有上述許可，請編輯與 IAM 角色或使用者關聯的 [IAM 政策](#)，以新增缺少的必要許可。

如果您的問題尚未解決，且繼續收到啟動失敗錯誤，可以解碼錯誤中包含的授權失敗訊息。解碼的訊息包含 IAM 政策中缺少的許可。[如需詳細資訊，請參閱 EC2 執行個體啟動期間收到 "UnauthorizedOperation" 錯誤後如何解碼授權失敗訊息？](#)

## 視窗啟動後不久的 CPU 使用率過高 (僅限 Windows 執行個體)

### Note

此疑難排解提示僅適用於 Windows 執行個體。

如果 Windows Update 設為 Check for updates but let me choose whether to download and install them (檢查是否有更新，但讓我選擇是否下載並安裝) (預設的執行個體設定)，此項檢查會使用執行個體 50 - 99% 的 CPU。如果此 CPU 耗用造成您應用程式的問題，您可在 Control Panel (控制面板) 手動變更 Windows Update 設定，或在 Amazon EC2 user data (Amazon EC2 使用者資料) 欄位中使用下列指令碼：

```
reg add "HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\WindowsUpdate\Auto Update" /v AUOptions /t REG_DWORD /d 3 /f net stop wuauclt net start wuauclt
```

當您執行此指令碼時，請指定 /d 的值。預設值為 3。可能的值包括以下：

1. 絕不檢查更新
2. 檢查是否有更新，但讓我選擇是否下載並安裝
3. 下載更新，但讓我選擇是否安裝
4. 自動安裝更新

您修改執行個體的使用者資料之後，就可以執行它。[如需詳細資訊，請參閱啟動時在 Windows 執行個體上執行命令。](#)

## 連線到 Linux 執行個體的疑難排解

下列資訊和常見錯誤可協助您疑難排解連線到執行個體的問題。

### 連線問題

- [連線問題的常見原因](#)
- [連線到執行個體時發生錯誤：連線時間逾時](#)
- [錯誤：無法載入金鑰 ... 預期為：任一私有金鑰](#)
- [錯誤：伺服器無法辨識使用者金鑰](#)
- [錯誤：許可遭拒或 \[執行個體\] 連接埠 22 已關閉連線](#)
- [錯誤：未受保護的私有金鑰檔案](#)
- [錯誤：私有金鑰的開頭必須為「-----BEGIN RSA PRIVATE KEY-----」，結尾為「-----END RSA PRIVATE KEY-----」](#)
- [錯誤：伺服器拒絕了我們的金鑰，或是沒有可用的支援驗證方法](#)
- [無法對執行個體執行 Ping 動作](#)
- [錯誤：伺服器意外關閉網路連線](#)
- [錯誤：EC2 Instance Connect 的主機金鑰驗證失敗](#)
- [無法使用 EC2 Instance Connect 連線至 Ubuntu 執行個體](#)
- [我的私有金鑰遺失。如何連線到我的 Linux 執行個體？](#)

### 連線問題的常見原因

建議您透過確認已正確執行下列任務來開始疑難排解執行個體的連線問題。



## 確認執行個體的使用者名稱

使用您的使用者帳戶的使用者名稱或用於啟動執行個體的 AMI 預設使用者名稱，可連線至執行個體。

- 取得使用者帳戶的使用者名稱。

如需如何建立使用者帳戶的詳細資訊，請參閱[管理 Linux 執行個體上的系統使用者](#)。

- 取得您用於啟動執行個體的 AMI 的預設使用者名稱：

用於啟動執行個體的 AMI	預設使用者名稱
AL2023	ec2-user
Amazon Linux 2	
Amazon Linux	
CentOS	centos 或 ec2-user
Debian	admin
Fedora	fedora 或 ec2-user
RHEL	ec2-user 或 root
SUSE	ec2-user 或 root
Ubuntu	ubuntu
Oracle	ec2-user
Bitnami	bitnami
Rocky Linux	rocky
其他	檢查 AMI 提供者

## 確認您的安全性群組規則允許流量

請確保與執行個體關聯的安全群組允許來自您的 IP 地址的傳入 SSH 流量。VPC 的預設安全群組根據預設並不允許傳入 SSH 流量。啟動執行個體精靈建立的安全群組根據預設會啟用 SSH 流量。如

需將傳入 SSH 流量規則新增至 Linux 執行個體的步驟，請參閱 [從您電腦連線到執行個體的規則](#)。  
如需驗證步驟，請參閱 [連線到執行個體時發生錯誤：連線時間逾時](#)。

## 確認您的執行個體已就緒

啟動執行個體之後，執行個體可能需要幾分鐘的時間準備就緒讓您連接。檢查您的執行個體，以確保它正在執行並通過其狀態檢查。

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取您的執行個體。
3. 請確認下列內容：
  - a. 在 Instance state (執行個體狀態) 欄中，驗證您的執行個體處於 running 狀態。
  - b. 在 Status check (狀態檢查) 欄中，驗證您的執行個體已通過兩項狀態檢查。

## 確認已滿足連線的所有先決條件

確保您備妥連線所需的所有資訊。如需詳細資訊，請參閱 [連接至您的 Linux 執行個體](#)。

如需連線類型的特定先決條件，例如 SSH、EC2 Instance Connect、OpenSSH、PuTTY 等，請參閱下列選項。

### Linux 或 macOS X

如果您的本機電腦作業系統是 Linux 或 macOS X，則請檢查以下連線選項的特定先決條件：

- [SSH 用戶端](#)
- [EC2 執行個體連線](#)
- [AWS Systems Manager 會話管理器](#)

### Windows

如果您的本機電腦作業系統是 Windows，則請檢查以下連線選項的特定先決條件：

- [OpenSSH](#)
- [PuTTY](#)
- [AWS Systems Manager 會話管理器](#)
- [適用於 Linux 的 Windows 子系統](#)

## 連線到執行個體時發生錯誤：連線時間逾時

如果您試著連線至執行個體，並收到錯誤訊息 `Network error: Connection timed out` 或 `Error connecting to [instance], reason: -> Connection timed out: connect`，則請嘗試下列動作：

檢查安全群組規則。

您需要安全群組規則，來允許透過適當的連接埠，從本機電腦公有 IPv4 地址傳入流量。

1. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取您的執行個體。
3. 在主控台頁面底部的 Security (安全) 標籤上，檢查 Inbound rules (傳入規則) 下方所選執行個體已生效的規則清單。
  - 如果是 Linux 執行個體，請確認規則存在，此規則會允許從本機電腦傳送流量到連接埠 22 (SSH)。
  - 如果是 Windows 執行個體，請確認規則存在，此規則會允許從本機電腦傳送流量到連接埠 3389 (RDP)。

如果安全群組不具有規則來允許從本機電腦傳入流量，請在安全群組中新增規則。如需詳細資訊，請參閱 [從您電腦連線到執行個體的規則](#)。

4. 針對允許傳入流量的規則，請檢查 Source (來源) 欄位。如果該值是單一 IP 地址，而且 IP 地址不是靜態的，則每次重新啟動電腦時都會指派新的 IP 地址。這將導致規則不會包括您電腦的 IP 地址流量。如果您的電腦位於企業網路中，或者如果您是透過網際網路服務供應商 (ISP) 連線，或者如果您的電腦 IP 地址是動態地址並且在每次重新啟動電腦時都會變更，則 IP 地址可能不是靜態地址。若要確保您的安全群組規則允許從本機電腦傳入流量，請指定用戶端電腦使用的 IP 地址範圍，而不是為 Source (來源) 指定單一 IP 地址。

如需安全群組規則的詳細資訊，請參閱 Amazon VPC 使用者指南 中的 [安全群組規則](#)。

查看子網的路由表。

您需要一個路由路徑，來將目的地位於 VPC 之外的所有流量，傳送到 VPC 的網際網路閘道。

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取您的執行個體。

3. 在 Networking (網路) 標籤上，請記下 VPC ID 和 Subnet ID (子網 ID) 的值。
4. 在 <https://console.aws.amazon.com/vpc/> 開啟 Amazon VPC 主控台。
5. 在導覽窗格中，選擇 Internet Gateways (網際網路閘道)。請確認已經有網際網路閘道連結到 VPC。否則請選擇 Create internet gateway (建立網際網路閘道)、輸入網際網路閘道的名稱，然後選擇 Create internet gateway (建立網際網路閘道)。然後，針對您建立的網際網路閘道，選取 Actions (動作)、Attach to VPC (連接至 VPC)、選取您的 VPC，然後選取 Attach internet gateway (連接網際網路閘道)，以將網際網路閘道連接至您的 VPC。
6. 在導覽窗格中，選取 Subnets (子網)，然後選取子網。
7. 在 Route Table (路由表) 索引標籤中，確認有包含  $0.0.0.0/0$  的路由做為目的地，以及有 VPC 的網際網路閘道做為目標。如果您是使用其 IPv6 地址來連線到執行個體，請確認有針對所有 IPv6 流量 ( $::/0$ ) 的路由，來指向網際網路閘道。若否，請執行下列操作：
  - a. 選擇路由表 ID (rtb-xxxxxxx) 以導覽至路由表。
  - b. 在 Routes (路由) 標籤中，選擇 Edit routes (編輯路由)。選擇 Add route (新增路由)，使用  $0.0.0.0/0$  做為目的地，並以網際網路閘道為目標。若是 IPv6，請選擇 Add route (新增路由)，使用  $::/0$  做為目的地，並以網際網路閘道為目標。
  - c. 選擇 Save routes (儲存路由)。

查看子網的網路存取控制清單 (ACL)。

網路 ACL 必須允許連接埠 22 (適用於 Linux 執行個體) 或連接埠 3389 (適用於 Windows 執行個體) 上的本機 IP 地址的傳入流量。它也必須允許傳送到暫時連接埠 (1024-65535) 的傳出流量。

1. 在 <https://console.aws.amazon.com/vpc/> 開啟 Amazon VPC 主控台。
2. 在導覽窗格中，選擇 Subnets (子網)。
3. 選取您的子網。
4. 在 Network ACL (網路 ACL) 索引標籤中，對於 Inbound rules (傳入規則)，請確認這些規則允許透過所需的連接埠，從您的電腦傳入流量。否則，請刪除或修改會阻止流量的規則。
5. 對於 Outbound Rules (傳出規則)，確認這些規則允許透過暫時連接埠將流量傳出到電腦。否則，請刪除或修改會阻止流量的規則。

如果您的電腦位於企業網路中

請詢問網路管理員，了解內部的防火牆是否允許透過 22 號 (適用於 Linux 執行個體) 或 3389 號埠 (適用於 Windows 執行個體)，來傳入流量到您的電腦或從您的電腦傳出。

如果您的電腦上具有防火牆，請確認此防火牆是否允許透過 22 號 (適用於 Linux 執行個體) 或 3389 號埠 (適用於 Windows 執行個體)，來傳入流量到您的電腦或從您的電腦傳出。

確認執行個體是否具備公有 IPv4 地址。

如果沒有，您可建立彈性 IP 地址與您執行個體的關聯。如需詳細資訊，請參閱 [彈性 IP 地址](#)。

檢查執行個體上的 CPU 負載，伺服器可能會過載。

AWS 自動提供 Amazon CloudWatch 指標和執行個體狀態等資料，您可以使用這些資料來查看執行個體上有多少 CPU 負載，並在必要時調整負載的處理方式。如需詳細資訊，請參閱 [使用監控執行個體 CloudWatch](#)。

- 如果負載會變動不定，您可以利用 [Auto Scaling](#) 和 [Elastic Load Balancing](#) 功能，來自動擴展或縮減執行個體的規模。
- 如果負載穩定增加，您可以改用較大容量的執行個體類型。如需詳細資訊，請參閱 [變更執行個體類型](#)。

若要使用 IPv6 地址來連線到執行個體，請檢察下列項目：

- 與子網具有關聯的路由表，必須包含讓 IPv6 流量 (:::/0) 傳到網際網路閘道的路由路徑。
- 安全群組規則必須允許從本機的 IPv6 地址，透過適當的通訊埠傳入流量 (Linux 為 22 號埠；Windows 為 3389 號埠)。
- 網路 ACL 必須允許外傳和傳入的 IPv6 流量。
- 如果從舊版的 AMI 啟動執行個體，則執行個體可能未針對 DHCPv6 設定 (在網路介面上不會自動辨識 IPv6 地址)。如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的在執行個體上設定 IPv6。
- 您的本機電腦必須擁有 IPv6 地址，且必須設定以使用 IPv6。

## 錯誤：無法載入金鑰 ... 預期為：任一私有金鑰

如果您嘗試連接到您的執行個體並收到錯誤訊息 `unable to load key ... Expecting: ANY PRIVATE KEY`，則表示存放私有金鑰的檔案設定不正確。如果私有金鑰檔案以 `.pem` 結尾，則表示可能仍設定不正確。私有金鑰檔案設定錯誤的可能原因是憑證遺失。

如果私有金鑰檔案設定不正確，請依照以下步驟解決錯誤

1. 建立新的金鑰對。如需詳細資訊，請參閱 [使用 Amazon EC2 建立金鑰對](#)。

**Note**

或者，您也可以使用第三方工具來建立新的金鑰對。如需詳細資訊，請參閱 [使用第三方工具來建立金鑰對](#)，然後將公有金鑰匯入 Amazon EC2。

2. 將新的金鑰對新增到執行個體。如需詳細資訊，請參閱 [我的私有金鑰遺失。如何連線到我的 Linux 執行個體？](#)。
3. 使用新的金鑰對連線至執行個體。

## 錯誤：伺服器無法辨識使用者金鑰

若如果使用 SSH 連線到執行個體

- 使用 `ssh -vvv`，在連線時取得三重 verbose (詳細模式) 的除錯資訊：

```
ssh -vvv -i path/key-pair-name.pem instance-user-name@ec2-203-0-113-25.compute-1.amazonaws.com
```

下列的輸出範例，示範了在嘗試使用伺服器無法辨識的金鑰來連線到執行個體時，可能會顯示的訊息：

```
open/ANT/myusername/.ssh/known_hosts).
debug2: bits set: 504/1024
debug1: ssh_rsa_verify: signature correct
debug2: kex_derive_keys
debug2: set_newkeys: mode 1
debug1: SSH2_MSG_NEWKEYS sent
debug1: expecting SSH2_MSG_NEWKEYS
debug2: set_newkeys: mode 0
debug1: SSH2_MSG_NEWKEYS received
debug1: Roaming not allowed by server
debug1: SSH2_MSG_SERVICE_REQUEST sent
debug2: service_accept: ssh-userauth
debug1: SSH2_MSG_SERVICE_ACCEPT received
debug2: key: boguspem.pem ((nil))
debug1: Authentications that can continue: publickey
debug3: start over, passed a different list publickey
debug3: preferred gssapi-keyex,gssapi-with-mic,publickey,keyboard-
interactive,password
```

```

debug3: authmethod_lookup publickey
debug3: remaining preferred: keyboard-interactive,password
debug3: authmethod_is_enabled publickey
debug1: Next authentication method: publickey
debug1: Trying private key: boguspem.pem
debug1: read PEM private key done: type RSA
debug3: sign_and_send_pubkey: RSA 9c:4c:bc:0c:d0:5c:c7:92:6c:8e:9b:16:e4:43:d8:b2
debug2: we sent a publickey packet, wait for reply
debug1: Authentications that can continue: publickey
debug2: we did not send a packet, disable method
debug1: No more authentication methods to try.
Permission denied (publickey).

```

### 如果使用 PuTTY 連線到您的執行個體

- 確認私有金鑰 (.pem) 檔案是否已轉換為 PuTTY (.ppk) 可以辨識的格式。如需有關轉換私有金鑰的詳細資訊，請參閱[使用 PuTTY 從 Windows 連線至 Linux 執行個體](#)。

#### Note

在 PuTTYgen 中，載入私有金鑰，然後選取 Save Private Key (儲存私有金鑰)，而非 Generate (產生)。

- 確認您在連線時使用了適當的 AMI 使用者名稱。在 PuTTY Configuration (PuTTY 組態) 視窗的 Host name (主機名稱) 中，輸入使用者的名稱。

用於啟動執行個體的 AMI	預設使用者名稱
AL2023	ec2-user
Amazon Linux 2	
Amazon Linux	
CentOS	centos 或 ec2-user
Debian	admin
Fedora	fedora 或 ec2-user

用於啟動執行個體的 AMI	預設使用者名稱
RHEL	ec2-user 或 root
SUSE	ec2-user 或 root
Ubuntu	ubuntu
Oracle	ec2-user
Bitnami	bitnami
Rocky Linux	rocky
其他	檢查 AMI 提供者

- 確認已建立傳入安全群組的規則，以允許從適當的連接埠傳入流量。如需詳細資訊，請參閱 [從您電腦連線到執行個體的規則](#)。

## 錯誤：許可遭拒或 [執行個體] 連接埠 22 已關閉連線

若使用 SSH 連線至執行個體，並出現下列任一錯誤：Host key not found in [directory]、Permission denied (publickey)、Authentication failed, permission denied 或 Connection closed by [instance] port 22，請確認您在連線時使用了適當的 AMI 使用者名稱，且已為執行個體指定正確的私有金鑰 (.pem) 檔案。

適合的使用者名稱如下：

用於啟動執行個體的 AMI	預設使用者名稱
AL2023	ec2-user
Amazon Linux 2	
Amazon Linux	
CentOS	centos 或 ec2-user
Debian	admin



用於啟動執行個體的 AMI	預設使用者名稱
Fedora	fedora 或 ec2-user
RHEL	ec2-user 或 root
SUSE	ec2-user 或 root
Ubuntu	ubuntu
Oracle	ec2-user
Bitnami	bitnami
Rocky Linux	rocky
其他	檢查 AMI 提供者

例如，若要使用 SSH 用戶端來連接至 Amazon Linux 執行個體，請使用下列命令：

```
ssh -i /path/key-pair-name.pem instance-user-name@ec2-203-0-113-25.compute-1.amazonaws.com
```

請確認在啟動執行個體時，所使用的私有金鑰檔案可對應到您所選取的金鑰對。

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取您的執行個體。
3. 在 Details (詳細資訊) 標籤上，驗證 Instance details (執行個體詳細資訊) 下方中 Key pair name (金鑰對名稱) 的值。
4. 如果在啟動執行個體時，您並未指定金鑰對，可以終止該執行個體並啟動新的執行個體，來確保指定金鑰對。如果這是一直在使用的執行個體，但是您已經沒有金鑰對的 .pem 檔案，可以用新的金鑰對來取代舊的。如需詳細資訊，請參閱 [我的私有金鑰遺失。如何連線到我的 Linux 執行個體？](#)。

如果產生自己的金鑰對，請確定金鑰產生器已經過設定，可產生 RSA 金鑰。DSA 金鑰是不被接受的。

如果收到 Permission denied (publickey) 錯誤，而上述的所有情況皆不適用 (例如，先前可以連線)，可能是對執行個體家目錄的權限已變更。對 `/home/instance-user-name/.ssh/authorized_keys` 的權限必須限定為只指派給擁有者。

### 在執行個體上確認權限

1. 停止執行個體並分離根磁碟區。如需詳細資訊，請參閱 [停止並啟動 Amazon EC2 執行個體](#)。
2. 在目前執行個體所在的同一個可用區域中，啟動暫時執行個體 (使用與目前執行個體所用 AMI 類似的或相同的 AMI)，然後將根磁碟區連結到暫時執行個體。
3. 連線到暫時執行個體、建立掛載點，然後掛載已連結的磁碟區。
4. 從暫時執行個體，檢查已連結磁碟區 `/home/instance-user-name/` 目錄的權限。必要時，請依照下列方式來調整權限：

```
[ec2-user ~]$ chmod 600 mount_point/home/instance-user-name/.ssh/authorized_keys
```

```
[ec2-user ~]$ chmod 700 mount_point/home/instance-user-name/.ssh
```

```
[ec2-user ~]$ chmod 700 mount_point/home/instance-user-name
```

5. 取消掛載磁碟區、將該磁碟區從暫時執行個體分離，然後再重新連結到原始執行個體。請務必為根磁碟區指定正確的裝置名稱，例如 `/dev/xvda`。
6. 啟動執行個體。如果不再需要暫時執行個體，您可以將其終止。

## 錯誤：未受保護的私有金鑰檔案

私有金鑰檔案必須受到保護，不讓其他任何使用者讀取或寫入。如果私有金鑰可以讓除了您以外的任何人讀取或寫入，SSH 會忽略您的金鑰，而且會出現下列的警告訊息。

```

@@
@ WARNING: UNPROTECTED PRIVATE KEY FILE! @
@@
Permissions 0777 for '.ssh/my_private_key.pem' are too open.
It is required that your private key files are NOT accessible by others.
This private key will be ignored.
bad permissions: ignore key: .ssh/my_private_key.pem
Permission denied (publickey).

```

如果在您試著登入執行個體時，出現類似的訊息，請檢驗該錯誤訊息的第一行，來確認是否針對執行個體使用了正確的公有金鑰。上列的範例使用私有金鑰 `.ssh/my_private_key.pem`，此金鑰具有 0777 的檔案權限，可讓所有人讀取或寫入此檔案。這種權限等級非常地不安全，因此 SSH 會忽略此金鑰。

如果從 macOS 或 Linux 連線，請執行下列命令來修正此錯誤，替換私有金鑰檔案的路徑。

```
[ec2-user ~]$ chmod 0400 .ssh/my_private_key.pem
```

如果從 Windows 連線，請在您的本機電腦上執行下列步驟。

1. 導覽至 `.pem` 檔案。
2. 在 `.pem` 檔案上按一下右鍵，然後選取 Properties (屬性)。
3. 選擇 Security (安全) 標籤。
4. 選取 Advanced (進階)。
5. 請確認您是檔案的擁有者。如果不是，請將擁有者變更為您的使用者名稱。
6. 選取 Disable inheritance (停用繼承) 和 Remove all inherited permissions from this object (移除此物件的所有繼承許可)。
7. 選取 Add (新增)、Select a principal (選取委託人)，輸入您的使用者名稱，然後選取 OK (確定)。
8. 從 Permission Entry (許可項目) 視窗中，授予 Read (讀取) 許可，然後選取 OK (確定)。
9. 按一下 Apply (套用)，以確保儲存所有設定。
10. 選取 OK (確定) 關閉 Advanced Security Settings (進階安全設定) 視窗。
11. 選取 OK (確定) 關閉 Properties (屬性) 視窗。
12. 您應該能夠透過 SSH 從 Windows 連線至 Linux 執行個體。

在 Windows 命令提示中執行下列命令。

1. 在命令提示中，導覽至 `.pem` 檔案的檔案路徑位置。
2. 執行下列命令，重設並移除明確許可：

```
icacls.exe $path /reset
```

3. 執行下列命令，對目前使用者授予讀取許可：

```
icacls.exe $path /GRANT:R "$($env:USERNAME):(R)"
```

4. 執行下列命令以停用繼承並移除繼承的許可。

```
icacls.exe $path /inheritance:r
```

5. 您應該能夠透過 SSH 從 Windows 連線至 Linux 執行個體。

**錯誤：**私有金鑰的開頭必須為「-----BEGIN RSA PRIVATE KEY-----」，結尾為「-----END RSA PRIVATE KEY-----」

如果您使用第三方工具 (例如 ssh-keygen) 來建立 RSA 金鑰對，則它會以 OpenSSH 金鑰格式產生私有金鑰。當您連線到您的執行個體時，如果以 OpenSSH 格式使用私有金鑰來解密密碼，則您會得到錯誤 Private key must begin with "-----BEGIN RSA PRIVATE KEY-----" and end with "-----END RSA PRIVATE KEY-----"。

若要解決錯誤，私有金鑰必須採用 PEM 格式。使用下列命令，以 PEM 格式建立私有金鑰：

```
ssh-keygen -m PEM
```

**錯誤：**伺服器拒絕了我們的金鑰，或是沒有可用的支援驗證方法

如果使用 PuTTY 來連線到執行個體，並出現下列任一錯誤：Error: Server refused our key (錯誤：伺服器拒絕了我們的金鑰) 或 Error: No supported authentication methods available (錯誤：沒有可用的支援驗證方法)，請確認您在連線時使用了適當的 AMI 使用者名稱。在 PuTTY Configuration (PuTTY 組態) 視窗的 User name (使用者名稱) 中輸入使用者名稱。

適合的使用者名稱如下：

用於啟動執行個體的 AMI	預設使用者名稱
AL2023	ec2-user
Amazon Linux 2	
Amazon Linux	
CentOS	centos 或 ec2-user
Debian	admin

用於啟動執行個體的 AMI	預設使用者名稱
Fedora	fedora 或 ec2-user
RHEL	ec2-user 或 root
SUSE	ec2-user 或 root
Ubuntu	ubuntu
Oracle	ec2-user
Bitnami	bitnami
Rocky Linux	rocky
其他	檢查 AMI 提供者

您也應該確認：

- 使用的 PuTTY 是最新版。如需詳細資訊，請參閱 [PuTTY 網頁](#)。
- 您的私有金鑰 (.pem) 檔案是否已正確轉換為 PuTTY (.ppk) 可以辨識的格式。如需有關轉換私有金鑰的詳細資訊，請參閱 [使用 PuTTY 從 Windows 連線至 Linux 執行個體](#)。

## 無法對執行個體執行 Ping 動作

ping 命令是一種 ICMP 傳輸流量 — 如果無法對執行個體執行 ping 命令，請確定傳入安全群組的規則，可允許 ICMP 傳輸流量接收從所有來源，或是從您用來發出指令的電腦或執行個體，所傳送的 Echo Request 訊息。

如果無法從執行個體發出 ping 指令，請確定安全群組外傳的規則，可允許 ICMP 傳輸流量，來讓 Echo Request 訊息傳到所有目的地，或是傳到您想要對其執行 ping 的主機。

Ping 命令也可能會遭到防火牆封鎖，或因為網路延遲或硬體問題而逾時。如需進一步疑難排解的協助，請洽詢您的區域網路或系統管理員。

## 錯誤：伺服器意外關閉網路連線

如果您使用 PuTTY 連接到執行個體，並且收到錯誤「伺服器意外關閉網路連線」，請驗證您在 PuTTY 組態的連線頁面上是否已啟用保持連線，以避免中斷連線。部分伺服器在指定的期間內未收到任何資料時會中斷連接用戶端。將 Seconds between keepalives (保持連線之間的秒數) 設為 59 秒。

如果您在啟用保持連線之後仍遇到問題，請在 PuTTY 組態的連線頁面上嘗試停用 Nagle 的演算法。

## 錯誤：EC2 Instance Connect 的主機金鑰驗證失敗

如果您輪替執行個體主機金鑰，新的主機金鑰不會自動上傳到 AWS 受信任的主機金鑰資料庫。因此當您嘗試使用 EC2 Instance Connect 以瀏覽器為基礎的用戶端連線到您的執行個體時，這會導致主機金鑰驗證失敗，而無法連線到您的執行個體。

若要解決此錯誤，您必須在 `eic_harvest_hostkeys` 執行個體上執行指令碼，以將新的主機金鑰上傳至 EC2 Instance Connect。指令碼位於 Amazon Linux 2 執行個體上的 `/opt/aws/bin/` 和 Ubuntu 執行個體上的 `/usr/share/ec2-instance-connect/`。

### Amazon Linux 2

解決 Amazon Linux 2 執行個體上的主機金鑰驗證失敗錯誤

1. 使用 SSH 連線到您的執行個體。

您可以使用 EC2 Instance Connect CLI 或在您啟動執行個體時指派至該執行個體的 SSH 金鑰對，並使用您在啟動執行個體所用的 AMI 預設使用者名稱，來進行連線。Amazon Linux 2 的預設使用者名稱為 `ec2-user`。

例如，如果執行個體的啟動方式是 Amazon Linux 2，執行個體的公有 DNS 名稱是 `ec2-a-b-c-d.us-west-2.compute.amazonaws.com`，金鑰對是 `my_ec2_private_key.pem`，則使用以下命令來透過 SSH 連接至執行個體：

```
$ ssh -i my_ec2_private_key.pem ec2-user@ec2-a-b-c-d.us-west-2.compute.amazonaws.com
```

如需連線至執行個體的詳細資訊，請參閱[使用 SSH 從 Linux 或 macOS 連線至 Linux 執行個體](#)。

2. 導覽至以下資料夾。

```
[ec2-user ~]$ cd /opt/aws/bin/
```

3. 在執行個體上執行以下命令。

```
[ec2-user ~]$./eic_harvest_hostkeys
```

請注意，呼叫成功時不會產生輸出。

您現在可以使用 EC2 Instance Connect 以瀏覽器為基礎的用戶端連線至執行個體。

## Ubuntu

### 解決 Ubuntu 執行個體上的主機金鑰驗證失敗錯誤

1. 使用 SSH 連線到您的執行個體。

您可以使用 EC2 Instance Connect CLI 或在您啟動執行個體時指派至該執行個體的 SSH 金鑰對，並使用您在啟動執行個體所用的 AMI 預設使用者名稱，來進行連線。若是 Ubuntu，預設使用者名稱為 ubuntu。

例如，如果執行個體的啟動方式是 Ubuntu，執行個體的公有 DNS 名稱是 `ec2-a-b-c-d.us-west-2.compute.amazonaws.com`，金鑰對是 `my_ec2_private_key.pem`，則使用以下命令來透過 SSH 連接至執行個體：

```
$ ssh -i my_ec2_private_key.pem ubuntu@ec2-a-b-c-d.us-west-2.compute.amazonaws.com
```

如需連線至執行個體的詳細資訊，請參閱[使用 SSH 從 Linux 或 macOS 連線至 Linux 執行個體](#)。

2. 導覽至以下資料夾。

```
[ec2-user ~]$ cd /usr/share/ec2-instance-connect/
```

3. 在執行個體上執行以下命令。

```
[ec2-user ~]$./eic_harvest_hostkeys
```

請注意，呼叫成功時不會產生輸出。

您現在可以使用 EC2 Instance Connect 以瀏覽器為基礎的用戶端連線至執行個體。

## 無法使用 EC2 Instance Connect 連線至 Ubuntu 執行個體

如果您使用 EC2 Instance Connect 連線至 Ubuntu 執行個體，並且在嘗試連線時出現錯誤，可以使用以下資訊嘗試修復問題。

### 可能的原因

該執行個體的 `ec2-instance-connect` 套件並非最新版本。

### 解決方案

如下所示將該執行個體的 `ec2-instance-connect` 套件更新至最新版本：

1. 使用 EC2 Instance Connect 以外的方法[連線](#)至您的執行個體。
2. 在您的執行個體上執行以下命令，將 `ec2-instance-connect` 套件更新至最新版本。

```
apt update && apt upgrade
```

## 我的私有金鑰遺失。如何連線到我的 Linux 執行個體？

如果遺失了 EBS 後端執行個體的私有金鑰，您可以重新獲得對執行個體的存取權限。您必須停止執行個體、中斷其根磁碟區的連結，並將該磁碟區做為資料磁碟區連接至其他執行個體、使用新的公有金鑰修改 `authorized_keys` 檔案、將磁碟區移回到原始的執行個體，然後重新啟動執行個體。關於啟動、連線到和停用執行個體，詳細資訊請參閱 [執行個體生命週期](#)。

僅具有 EBS 根磁碟區的執行個體支援此程序。如果根裝置是執行個體存放區磁碟區，則無法使用此程序來重新獲取執行個體的存取權；您必須具備私有金鑰，才能連線到執行個體。若要確定執行個體的根裝置類型，請開啟 Amazon EC2 主控台，選擇執行個體，選取執行個體，選擇儲存索引標籤，然後在根裝置詳細資訊區段中，檢查根裝置類型的值。

此值為 EBS 或 INSTANCE-STORE。

除了下列步驟之外，遺失私有金鑰時還有其他方法可以連線到 Linux 執行個體。如需詳細資訊，請參閱 [如果我在初次啟動後遺失了 SSH 金鑰對，該如何連接到 Amazon EC2 執行個體？](#)

使用不同的金鑰對以連線到 EBS 後端執行個體的步驟



- [步驟 1：建立新的金鑰對](#)
- [步驟 2：取得關於原始執行個體及其根磁碟區的資訊](#)
- [步驟 3：停止原始執行個體](#)
- [步驟 4：啟動暫時執行個體](#)
- [步驟 5：將根磁碟區與原始執行個體分離，並將其連接到暫時執行個體](#)
- [步驟 6：在掛載到暫時執行個體的原始磁碟區上，將新的公有金鑰新增至的 `authorized\_keys`](#)
- [步驟 7：取消掛載原始磁碟區，並將該磁碟區從暫時執行個體分離，然後再重新連接到原始執行個體](#)
- [步驟 8：使用新的金鑰對連線到原始執行個體](#)
- [步驟 9：清理](#)

## 步驟 1：建立新的金鑰對

使用 Amazon EC2 主控台或第三方工具來產生新的金鑰對。如果希望新金鑰對的名稱，和已遺失私有金鑰的相同，您必須先刪除現有的金鑰對。如需建立新金鑰對的詳細資訊，請參閱[使用 Amazon EC2 建立金鑰對](#)或[使用第三方工具來建立金鑰對](#)，然後將公有金鑰匯入 Amazon EC2。

## 步驟 2：取得關於原始執行個體及其根磁碟區的資訊

請儲存下列資訊，完成此程序將會需要用到這些資訊。

取得原始執行個體的相關資訊

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中選取 Instances (執行個體)，然後選取您要連線的執行個體。(我們會將此執行個體稱為「原始」執行個體。)
3. 在 Details (詳細資訊) 索引標籤上記下執行個體 ID 和 AMI ID。
4. 在網路索引標籤上記下「可用區域」。
5. 在 Storage (儲存) 索引標籤下的根裝置名稱，記下根磁碟區的裝置名稱 (例如：`/dev/xvda`)。接著，在區塊型儲存裝置下找出這個裝置名稱，並記下磁碟區 ID (例如：`vol-0a1234b5678c910de`)。

## 步驟 3：停止原始執行個體

選擇 Instance state (執行個體狀態)、Stop instance (停止執行個體)。如果此選項已停用，則執行個體已停止或其根設備是執行個體存放磁碟區。

**⚠ Warning**

停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區的資料，請將資料備份至持久性儲存。

## 步驟 4：啟動暫時執行個體

### New console

#### 啟動暫時執行個體

1. 在導覽窗格中，選擇 Instances (執行個體)，然後選擇 Launch instances (啟動執行個體)。
2. 在 Name and tags (名稱和標籤) 區段，針對 Name (名稱)，輸入 Temporary (暫時)。
3. 在 Application and OS Images (應用程式和作業系統映像) 區段，選取您用來啟動原始執行個體的同一個 AMI。如果此 AMI 無法使用，您可以從已停止的執行個體，建立能夠使用的 AMI。如需詳細資訊，請參閱 [創建一個 Amazon EBS-backed AMI](#)。
4. 在 Instance type (執行個體類型) 區段，保留預設執行個體類型。
5. 在 Key pair (金鑰對) 區段，針對 Key pair name (金鑰對名稱)，選取要使用的現有金鑰對或建立新的金鑰對。
6. 在 Network settings (網路設定) 區段，選取 Edit (編輯)，然後針對 Subnet (子網)，選取與原始執行個體相同可用區域中的子網。
7. 在 Summary (摘要) 面板中，選擇 Launch (啟動)。

### Old console

選擇 Launch Instance (啟動執行個體)，接著再使用 Launch Wizard，透過下列選項來啟動 temporary (暫時) 執行個體：

- 在 Choose an AMI (選取 AMI) 頁面上，選取您之前用來啟動原始執行個體的同一個 AMI。如果此 AMI 無法使用，您可以從已停止的執行個體，建立能夠使用的 AMI。如需詳細資訊，請參閱 [創建一個 Amazon EBS-backed AMI](#)。
- 在 Choose an Instance Type (選取執行個體類型) 頁面上，保留精靈為您選取的預設執行個體類型。
- 在 Configure Instance Details (設定執行個體詳細資訊) 頁面中，指定相同的可用區域做為原始執行個體。如果在 VPC 中啟動執行個體，請選取這個可用區域中的子網。

- 在 Add Tags (新增標籤) 頁面上，將 Name=Temporary 標籤加入執行個體，來表示這是暫時執行個體。
- 在 Review (檢閱) 頁面上，選擇 Launch (啟動)。選擇您在步驟 1 建立的金鑰對，然後選擇 Launch Instances (啟動執行個體)。

## 步驟 5：將根磁碟區與原始執行個體分離，並將其連接到暫時執行個體

1. 在導覽窗格中選取 Volumes (磁碟區)，接著再選取原始執行個體的根裝置磁碟區 (您已在先前的步驟中記下其磁碟區 ID)。選取 Actions (動作)、Detach Volume (分離磁碟區)，然後選取 Detach (分離)。等待磁碟區的狀態變成 available。(您可能需要選取 Refresh (重新整理) 圖示。)
2. 在已選取磁碟區的狀態下，選取 Actions (動作)，然後再選取 Attach volume (連接磁碟區)。選取暫時執行個體的執行個體 ID、記下 Device name (裝置名稱) 中的指定裝置名稱 (例如：/dev/sdf)，接著再選取 Attach volume (連接磁碟區)。

### Note

如果您從 AWS Marketplace AMI 啟動原始執行個體，且磁碟區包含 AWS Marketplace 代碼，則必須先停止暫存執行個體，然後才能連接磁碟區。

## 步驟 6：在掛載到暫時執行個體的原始磁碟區上，將新的公有金鑰新增至的 `authorized_keys`

1. 連線到暫時執行個體。
2. 從暫時執行個體掛載已連結到執行個體的磁碟區，如此您就能存取其檔案系統。例如，如果裝置名為 /dev/sdf，請使用下列的指令來將磁碟區掛載為 /mnt/tempvol。

### Note

在執行個體上所顯示的裝置名稱可能會有不同。例如，掛載為 /dev/sdf 的裝置，在執行個體上可能會顯示為 /dev/xvdf。Red Hat 的某些版本 (或是其變體版本，例如 CentOS) 甚至可能會將結尾的字母增加 4 個字元，其中 /dev/sdf 會變成 /dev/xvdk。

- a. 使用 `lsblk` 命令來查看磁碟區是否已分割。

```
[ec2-user ~]$ lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
xvda 202:0 0 8G 0 disk
##xvda1 202:1 0 8G 0 part /
xvdf 202:80 0 101G 0 disk
##xvdf1 202:81 0 101G 0 part
xvdg 202:96 0 30G 0 disk
```

在上述的範例中，`/dev/xvda` 和 `/dev/xvdf` 為分割的磁碟區，而 `/dev/xvdg` 則不是。如果磁碟區已分割，則在接下來的步驟中，您會掛載 (`/dev/xvdf1`) 而非原始裝置 (`/dev/xvdf`)。

- b. 建立暫時的目錄來掛載磁碟區。

```
[ec2-user ~]$ sudo mkdir /mnt/tempvol
```

- c. 使用您先前找出的磁碟區名稱或裝置名稱，將磁碟區 (或分割磁碟區) 掛載至暫時掛載點。必要命令取決於作業系統的檔案系統。請注意，在執行個體上所顯示的設備名稱可能會有不同。如需詳細資訊，請參閱步驟 6 中的 [note](#)。

- Amazon Linux、Ubuntu 和 Debian

```
[ec2-user ~]$ sudo mount /dev/xvdf1 /mnt/tempvol
```

- Amazon Linux 2、CentOS、SUSE Linux 12 和 RHEL 7.x

```
[ec2-user ~]$ sudo mount -o nouuid /dev/xvdf1 /mnt/tempvol
```

#### Note

如果您收到檔案系統毀損的錯誤，請執行下列命令，使用 `fsck` 公用程式來檢查檔案系統並修復任何問題：

```
[ec2-user ~]$ sudo fsck /dev/xvdf1
```

3. 從暫時執行個體使用下列的指令，用暫時執行個體的 `authorized_keys` 所產生的新公有金鑰，來更新已掛載磁碟區上的 `authorized_keys`。

**⚠ Important**

下列的範例使用 Amazon Linux 使用者名稱 `ec2-user`。您可以需要換成不同的使用者名稱，例如適用於 Ubuntu 執行個體的 `ubuntu`。

```
[ec2-user ~]$ cp .ssh/authorized_keys /mnt/tempvol/home/ec2-user/.ssh/authorized_keys
```

如果此項複製動作成功，您就可以進行下一個步驟。

(選用) 否則，如果您沒有編輯 `/mnt/tempvol` 中的檔案的許可，必須使用 `sudo` 來更新檔案，然後檢查對於該檔案的許可，以確認您是否能夠登入原始執行個體。使用下列命令以檢查對檔案的許可。

```
[ec2-user ~]$ sudo ls -l /mnt/tempvol/home/ec2-user/.ssh
total 4
-rw----- 1 222 500 398 Sep 13 22:54 authorized_keys
```

在本範例輸出中，`222` 為使用者 ID，而 `500` 為群組 ID。接下來，請使用 `sudo` 來重新執行先前失敗的複製命令。

```
[ec2-user ~]$ sudo cp .ssh/authorized_keys /mnt/tempvol/home/ec2-user/.ssh/authorized_keys
```

再次執行下列命令，以確定許可是否已變更。

```
[ec2-user ~]$ sudo ls -l /mnt/tempvol/home/ec2-user/.ssh
```

如果使用者 ID 和群組 ID 已變更，請使用下列命令來還原這些 ID。

```
[ec2-user ~]$ sudo chown 222:500 /mnt/tempvol/home/ec2-user/.ssh/authorized_keys
```

## 步驟 7：取消掛載原始磁碟區，並將該磁碟區從暫時執行個體分離，然後再重新連接到原始執行個體

1. 從暫時執行個體取消掛載已連結的磁碟區，如此您就能將該磁碟區重新連結到原始執行個體。例如，使用以下命令來取消掛載位於 `/mnt/tempvol` 的磁碟區。

```
[ec2-user ~]$ sudo umount /mnt/tempvol
```

2. 將磁碟區從暫時執行個體分離 (您在上一步已卸載該磁碟區)：從 Amazon EC2 主控台，選取導覽窗格中的 Volumes (磁碟區)，選取原始執行個體的根裝置磁碟區 (您在上一步已記下磁碟區 ID)，選取 Actions (動作)、Detach volume (分離磁碟區)，接著再選取 Detach (分離)。等待磁碟區的狀態變成 available。(您可能需要選取 Refresh (重新整理) 圖示。)
3. 將磁碟區重新連接到原始執行個體：在已選取磁碟區的狀態下，選取 Actions (動作)、Attach volume (連接磁碟區)。選取原始執行個體的執行個體 ID、指定您先前在 [步驟 2](#) 中為原始根裝置連接 (`/dev/sda1` 或 `/dev/xvda`) 記錄的裝置名稱，然後選取 Attach volume (連接磁碟區)。

### Important

如果您並未指定與原始連結相同的裝置名稱，則無法啟動原始執行個體。Amazon EC2 預期根裝置磁碟區位於 `sda1` 或 `/dev/xvda`。

## 步驟 8：使用新的金鑰對連線到原始執行個體

選取原始執行個體，然後依序選取 Instance state (執行個體狀態)、Start instance (啟動執行個體)。在執行個體進入 `running` 狀態之後，您可以使用新金鑰對的私有金鑰檔案，來連線到該執行個體。

### Note

如果新金鑰對的名稱和對應的私有金鑰檔案，與原始金鑰對的名稱不同，請務必在連線到執行個體時，指定新私有金鑰檔案的名稱。

## 步驟 9：清理

(選用) 如果不需再使用時，您可以終止暫時執行個體。選取暫時執行個體，再依序選取 Instance state (執行個體狀態)、Terminate instance (終止執行個體)。

# 疑難排解連接至您的 Windows 執行個體

下列資訊和常見錯誤可協助您疑難排解連線至 Windows 執行個體的問題。

## 連線問題

- [遠端桌面無法連線到遠端電腦](#)
- [使用 macOS RDP 用戶端時發生錯誤](#)
- [RDP 顯示黑色螢幕而非桌面](#)
- [無法使用非管理員的使用者從遠端登入至執行個體](#)
- [使用的遠程桌面問題故障 AWS Systems Manager](#)
- [使用遠端登錄在 EC2 執行個體上啟用遠端桌面](#)
- [我的私有金鑰遺失。如何連接到我的 Windows 執行個體？](#)

## 遠端桌面無法連線到遠端電腦

嘗試下列方法解決執行個體連線相關問題：

- 確認您使用的是正確的公有 DNS 主機名稱。(在 Amazon EC2 主控台，選取執行個體，接著在詳細資訊窗格中勾選 Public DNS (IPv4) (公有 DNS (IPv4))。) 如果您的執行個體是在 VPC 中，而您找不到公有 DNS 名稱，您必須啟用 DNS 主機名稱。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的 [VPC 的 DNS 屬性](#)。
- 確認您的執行個體有公有的 IPv4 地址。如果沒有，您可建立彈性 IP 地址與您執行個體的關聯。如需詳細資訊，請參閱 [彈性 IP 地址](#)。
- 若要使用 IPv6 地址連線到您的執行個體，請確認您的本機電腦有 IPv6 地址，且設定使用 IPv6。如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的在執行個體上設定 IPv6。
- 確認您的安全群組有允許 RDP 存取的規則。
- 如果您已複製密碼但收到錯誤 Your credentials did not work，請嘗試在提示時手動輸入密碼。有可能是您在複製密碼時少了一個字元或多了一個空白字元。
- 請確認執行個體已通過狀態檢查。如需詳細資訊，請參閱 [您的執行個體的狀態檢查](#) 及 [the section called “Linux 上的狀態檢查失敗”](#)。
- 確認子網路的路由表有路由可將目標 VPC 外的所有流量傳送至 VPC 的網際網路閘道。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的 [建立自訂路由表](#) (網際網路閘道)。
- 確認 Windows 防火牆或其他防火牆軟體未封鎖執行個體的 RDP 流量。我們建議您停用 Windows 防火牆，並使用安全群組規則控制對您執行個體的存取。您可以使用 [AWSSupport-TroubleshootRDP](#)

以 [disable the Windows Firewall profiles using SSM Agent](#) : 若要在未設定 AWS Systems Manager、使 [AWS Support-ExecuteEC2Rescue](#) 用或使用下列手動步驟的 Windows 執行個體上停用 Windows 防火牆：

## 手動步驟

1. 停止受影響的執行個體並分離其根磁碟區。
2. 啟動與受影響執行個體同一可用區域中的暫時執行個體。

### Warning

如果您的暫時執行個體與原始執行個體以相同的 AMI 為基礎，則您必須完成額外的步驟，否則您無法在還原原始執行個體磁碟區之後將其啟動，原因是磁碟簽名衝突。或者，請為暫時執行個體選擇不同的 AMI。例如，如果原始執行個體使用 AWS 視窗 AMI 適用於視窗伺服器 2016 年，請使用 AWS 視窗 AMI 啟動暫存執行個體。

3. 從受影響的執行個體將根磁碟區連接到此暫時執行個體。連線至暫時執行個體、開啟 Disk Management (磁碟管理) 公用程式，然後讓磁碟機上線。
4. 開啟 Regedit 並選取 HKEY\_LOCAL\_MACHINE。從 File (檔案) 功能表中，選擇 Load Hive (載入 Hive)。選取磁碟機、開啟檔案 Windows\System32\config\SYSTEM，然後在提示時指定金鑰名稱 (您可使用任何名稱)。
5. 選擇剛載入的機碼並導覽至 ControlSet001\Services\SharedAccess\Parameters\FirewallPolicy。為名稱格式為 xxxxProfile 的每一個金鑰選取金鑰，並將 EnableFirewall 從 1 變更為 0。再次選取該金鑰，然後從 File (檔案) 功能表中選擇 Unload Hive (取消載入 Hive)。
6. (選用) 如果您的暫時執行個體與原始執行個體以相同的 AMI 為基礎，則您必須完成下列步驟，否則您無法在還原原始執行個體磁碟區之後將其啟動，原因是磁碟簽名衝突。

### Warning

下列程序說明如何使用登錄編輯程式來編輯 Windows 登錄檔。如果您不熟悉 Windows 登錄檔或如何使用登錄編輯程式來安全地進行變更，請參閱 [設定登錄檔](#)。

- a. 開啟命令提示，輸入 regedit.exe，然後按 Enter。



- b. 在登錄編輯程式中，從內容功能表中選擇 HKEY\_LOCAL\_MACHINE (按一下右鍵)，然後選擇 Find (尋找)。
- c. 輸入 Windows Boot Manager，然後選擇 Find Next (尋找下一個)。
- d. 選擇名為 11000001 的金鑰。此金鑰是您在上一個步驟中找到的金鑰的同級。
- e. 在右窗格中，選擇 Element，然後從內容功能表中選擇 Modify (修改)(按一下右鍵)。
- f. 在資料中的位移為 0x38 處找到四位元組的磁碟簽章。反轉位元組以建立磁碟簽章，然後寫下來。例如，下列資料所代表的磁碟簽章是 E9EB3AA5：

```
...
0030 00 00 00 00 01 00 00 00
0038 A5 3A EB E9 00 00 00 00
0040 00 00 00 00 00 00 00 00
...
```

- g. 在 [命令提示字元] 視窗中，執行下列命令以啟動 Microsoft DiskPart。

```
diskpart
```

- h. 執行下列 DiskPart 命令以選取磁碟區。(可以使用磁碟管理公用程式，確認磁碟編號為 1)。

```
DISKPART> select disk 1

Disk 1 is now the selected disk.
```

- i. 運行以下 DiskPart 命令以獲取磁盤簽名。

```
DISKPART> uniqueid disk

Disk ID: 0C764FA8
```

- j. 如果上一步中顯示的磁碟簽章與您先前寫下的 BCD 的磁碟簽章不符，請使用下列 DiskPart 命令來變更磁碟簽章，使其符合：

```
DISKPART> uniqueid disk id=E9EB3AA5
```

7. 使用 Disk Management (磁碟管理) 公用程式讓磁碟機離線。

**Note**

如果暫時執行個體執行的作業系統與受影響執行個體的相同，磁碟機會自動離線，因此您不需要手動將其離線。

8. 分離磁碟區與暫時執行個體。如果不再需要使用此暫時執行個體，您可終止它。
  9. 將受影響執行個體的根磁碟區連接為 `/dev/sda1` 來還原它。
  10. 啟動實例。
- 在不屬於 Active Directory 域的執行個體上，確認網路層級身分驗證已停用 (使用 [AWSSupport-TroubleshootRDP](#) 以 [disable NLA](#))。
  - 確認遠端桌面服務 (TermService) 啟動類型為 [自動]，且服務已啟動 ([AWSSupport-TroubleshootRDP](#) 使用 [enable and start the RDP service](#))。
  - 確認您連線至正確的遠端桌面通訊協定連接埠，預設為 3389 (使用 [AWSSupport-TroubleshootRDP](#) 以 [read the current RDP port](#) 和 [change it back to 3389](#))。
  - 確認您的執行個體上允許遠端桌面連線 (使用 [AWSSupport-TroubleshootRDP](#) 以 [enable Remote Desktop connections](#))。
  - 請確認密碼尚未過期。如果密碼已過期，您可以重設它。如需詳細資訊，請參閱 [重設遺失或過期的 Windows 管理員密碼](#)。
  - 如果您嘗試使用在此執行個體上建立的使用者進行連線，但收到錯誤 `The user cannot connect to the server due to insufficient access privileges`，請確認您已授權使用者可在本機登入。如需詳細資訊，請參閱 [授予成員登入本機的權限](#)。
  - 如果您嘗試的同時 RDP 工作階段超過允許上限，您的工作階段會終止並顯示訊息：`Your Remote Desktop Services session has ended. Another user connected to the remote computer, so your connection was lost.`。根據預設，您的執行個體可以有兩個同時 RDP 工作階段。

## 使用 macOS RDP 用戶端時發生錯誤

如果您從 Microsoft 網站使用遠端桌面連線用戶端連線到 Windows 伺服器執行個體，您可能會收到下列錯誤：

```
Remote Desktop Connection cannot verify the identity of the computer that you want to connect to.
```

從 Mac App Store 下載 Microsoft Remote Desktop 應用程式，然後使用該應用程式連線到執行個體。

## RDP 顯示黑色螢幕而非桌面

請嘗試下列方法解決此問題：

- 如需其他資訊，請參閱主控台輸出。若要使用 Amazon EC2 主控台取得執行個體的主控台輸出，請選取執行個體，然後選擇 Actions (動作)、Monitor and troubleshoot (監視和故障診斷)、Get system log (取得系統記錄檔)。
- 確認您執行的是最新版的 RDP 用戶端。
- 嘗試 RDP 用戶端的預設設定。如需詳細資訊，請參閱 [遠端工作階段環境](#)。
- 如果您使用的是遠端桌面連線，請嘗試使用 /admin 選項啟動它，如下所示。

```
mstsc /v:instance /admin
```

- 如果伺服器執行全螢幕應用程式，可能已停止回應。請使用 Ctrl+Shift+Esc 啟動 Windows 任務管理員，然後結束應用程式。
- 如果伺服器過度使用，可能已停止回應。若要使用 Amazon EC2 主控台監控執行個體，請選取該執行個體，然後選取 Monitoring (監控) 索引標籤。如果您需要將執行個體類型變更為較大的大小，請參閱 [變更執行個體類型](#)。

## 無法使用非管理員的使用者從遠端登入至執行個體

如果您無法使用非管理員帳戶的使用者從遠端登入 Windows 執行個體，請確定您已授予該使用者從本機登入的權限。請參閱 [授予使用者或群組權限，以從本機登入網域中的網域控制器](#)。

## 使用的遠程桌面問題故障 AWS Systems Manager

您可以使用 AWS Systems Manager 來疑難排解使用 RDP 連線至 Windows 執行個體的問題。

### AWSSupport-疑難排解

AWSSupport-疑難排解 TrDP 自動化文件可讓使用者檢查或修改可能影響遠端桌面通訊協定 (RDP) 連線的目標執行個體上的一般設定，例如 RDP 連接埠、網路層驗證 (NLA) 和 Windows 防火牆設定檔。根據預設，此文件會讀取和輸出這些設定的值。

AWSSupport-疑難排解 TrDP 自動化文件可與啟用 (受管執行個體) 搭配使用的 EC2 執行個體、現場部署執行個體和虛擬機器 AWS Systems Manager (VM) 搭配使用。此外，它也可以與未啟用搭配

Systems Manager 使用的 Windows 伺服器之 EC2 執行個體搭配使用。如需有關啟用執行個體以搭配使用的資訊 AWS Systems Manager，請參閱《使 AWS Systems Manager 用指南》中的 [受管節點](#)。

若要使用疑難 AWSSupport 排解 TrDP 文件進行疑難排解

1. 登入 [Systems Manager 主控台](#)。
2. 確認您位於受損執行個體所在的同一區域。
3. 從左側導覽窗格中選擇 Documents (文件)。
4. 在 Owned by Amazon (由 Amazon 所有) 標籤中，在搜尋欄位中輸入 AWSSupport-TroubleshootRDP。當 AWSSupport-TroubleshootRDP 文件出現時，請選取它。
5. 選擇 Execute automation (執行自動化)。
6. 對於 Execution Mode (執行模式)，選擇 Simple execution (簡易執行)。
7. 針對輸入參數 InstanceId，啟用顯示互動式執行個體選擇器。
8. 選擇 Amazon EC2 執行個體。
9. 檢閱 [範例](#)，然後選擇 Execute (執行)。
10. 若要監控執行進度，對於 Execution status (執行狀態)，等待狀態從 Pending (待定) 變成 Success (成功)。展開輸出以檢視結果。若要檢視個別步驟的輸出，請在 Executed Steps (執行的步驟) 中從 Step ID (步驟 ID) 選擇一個項目。

### AWSSupport-疑難排解範例

下列範例說明如何使用 AWSSupport-疑難排解方式完成常見的疑難排解工作。您可以使用範例 AWS CLI [start-automation-execution](#) 指令或提供的連結 AWS Management Console。

Example 範例：檢查目前 RDP 狀態

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP" --parameters "InstanceId=instance_id, Action=Custom" --region region_code
```

AWS Systems Manager 控制台：

```
https://console.aws.amazon.com/systems-manager/automation/execute/AWSSupport-TroubleshootRDP?region=region#documentVersion=$LATEST
```

## Example 範例：停用 Windows 防火牆

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP"
 --parameters "InstanceId=instance_id, Action=Custom, Firewall=Disable" --
 region region_code
```

AWS Systems Manager 控制台：

```
https://console.aws.amazon.com/systems-manager/automation/execute/AWSSupport-
TroubleshootRDP?region=region_code#documentVersion=$LATEST&Firewall=Disable
```

## Example 範例：停用網路層級身分驗證

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP"
 --parameters "InstanceId=instance_id, Action=Custom, NLASettingAction=Disable" --
 region region_code
```

AWS Systems Manager 控制台：

```
https://console.aws.amazon.com/systems-manager/automation/execute/AWSSupport-
TroubleshootRDP?region=region_code#documentVersion
```

## Example 範例：將 RDP 服務啟動類型設為自動並啟動 RDP 服務

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP"
 --parameters "InstanceId=instance_id, Action=Custom, RDPServiceStartupType=Auto,
 RDPServiceAction=Start" --region region_code
```

AWS Systems Manager 控制台：

```
https://console.aws.amazon.com/systems-manager/automation/execute/
AWSSupport-TroubleshootRDP?region=region_code#documentVersion=
$LATEST&RDPServiceStartupType=Auto&RDPServiceAction=Start
```

## Example 範例：還原預設 RDP 連接埠 (3389)

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP"
 --parameters "InstanceId=instance_id, Action=Custom, RDPPortAction=Modify" --
 region region_code
```

AWS Systems Manager 控制台：

```
https://console.aws.amazon.com/systems-manager/automation/execute/AWSSupport-
 TroubleshootRDP?region=region_code#documentVersion=$LATEST&RDPPortAction=Modify
```

## Example 範例：允許遠端連線

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP"
 --parameters "InstanceId=instance_id, Action=Custom, RemoteConnections=Enable" --
 region region_code
```

AWS Systems Manager 控制台：

```
https://console.aws.amazon.com/systems-manager/automation/execute/AWSSupport-
 TroubleshootRDP?region=region_code#documentVersion=$LATEST&RemoteConnections=Enable
```

## AWSSupport-執行 2 救援

AWSSupport-Executeec2Rescue 自動化文件使用適用於 Windows 伺服器的 EC2Rescue 來自動疑難排解和還原 EC2 執行個體連線和 RDP 問題。如需詳細資訊，請參閱[在無法觸達的執行個體上執行 EC2Rescue 工具](#)。

AWSSupport-ExecuteEC2Rescue 自動化文件需要停止並重新啟動執行個體。Systems Manager 會停止執行個體並建立 Amazon Machine Image (AMI)。存放在執行個體存放磁碟區的資料會遺失。如果您不是使用彈性 IP 地址，則公有 IP 地址會變更。如需詳細資訊，請參閱《AWS Systems Manager 使用者指南》中的[在無法觸達的執行個體上執行 EC2Rescue 工具](#)。

若要使用-ExecuteC2Rescue 文件進 AWSSupport行疑難排解

1. 開啟 [Systems Manager 主控台](#)。
2. 確認您位於受損 Amazon EC2 執行個體所在的同一區域。

3. 在導覽面板中，選擇文件。
4. 搜尋並選取 `AWSsupport-ExecuteEC2Rescue` 文件，然後選擇執行自動化。
5. 在 Execution Mode (執行模式) 中，選擇 Simple execution (簡易執行)。
6. 在「輸入參數」區段中，針對 `UnreachableInstanceID`，輸入無法存取之執行個體的 Amazon EC2 執行個體 ID。
7. (選擇性) 若要收集作業系統日誌以進行 Amazon EC2 執行個體疑難排解 `LogDestination`，請輸入 Amazon 簡單儲存服務 (Amazon S3) 儲存貯體名稱。日誌會自動上傳至指定的儲存貯體。
8. 選擇 Execute (執行)。
9. 若要監控執行進度，請在 Execution status (執行狀態) 中，等待狀態從 Pending (待定) 變成 Success (成功)。展開輸出以檢視結果。若要檢視個別步驟的輸出，請在 Executed Steps (執行的步驟) 中選擇 Step ID (步驟 ID)。

## 使用遠端登錄在 EC2 執行個體上啟用遠端桌面

如果您無法存取的執行個體不是由 AWS 系統管理員工作階段管理員所管理，您可以使用遠端登錄來啟用遠端桌面。

1. 從 EC2 控制台，停止無法連線的執行個體。
2. 將無法連線之執行個體的根磁碟區，連接到相同可用區域中的另一個執行個體以做為儲存磁碟區。如果同一可用區域中沒有可連線的執行個體，請啟動一個執行個體。記下無法連線的執行個體上根磁碟區的裝置名稱。
3. 在可連線的執行個體上，開啟「磁碟管理」。在「命令提示」視窗中執行下列命令，即可執行這項操作。

```
diskmgmt.msc
```

4. 以滑鼠右鍵按一下來自無法連線之執行個體的新連接磁碟區，然後選擇線上。
5. 開啟 Windows 登錄編輯程式。在「命令提示」視窗中執行下列命令，即可執行這項操作。

```
regedit
```

6. 在「登錄編輯程式」中，選擇 `HKEY_LOCAL_MACHINE`，然後依序選擇檔案、載入 Hive。
7. 選取連接磁碟區的磁碟機，導覽至 `\Windows\System32\config\`，選取 `SYSTEM`，然後選擇 Open (開啟)。
8. 對於 Key Name (鍵值名稱)，請輸入 `hive` 的唯一名稱，並選擇 OK (確定)。

9. 請先備份登錄 hive，再對登錄進行任何變更。
  - a. 在「登錄編輯程式」主控台樹狀目錄中，選取您載入的 Hive : HKEY\_LOCAL\_MACHINE\*your-key-name* .
  - b. 依序選擇檔案、匯出。
  - c. 在「匯出登錄檔案」對話方塊中，選擇您要儲存備份複本的位置，然後在 File name (檔案名稱) 欄位中輸入備份檔案的名稱。
  - d. 選擇 Save (儲存)。
10. 在「登錄編輯程式」中，導覽至 HKEY\_LOCAL\_MACHINE\*your key name*\ControlSet001\Control\Terminal Server，然後在詳細資訊窗格中按兩下 fDenyTSConnections。
11. 在 Edit DWORD (編輯 DWORD) 值方塊中，在 Value data (值資料) 欄位中輸入 0。
12. 選擇 OK (確定)。

 Note

如果 Value data (值資料) 欄位中的值為 1，則執行個體將拒絕遠端桌面連線。值 0 允許遠端桌面連線。

13. 在「登錄編輯程式」中，選擇 HKEY\_LOCAL\_MACHINE\*your-key-name*，然後依序選擇檔案、卸載 Hive。
14. 關閉「登錄編輯程式」和「磁碟管理」。
15. 在 EC2 主控台中，將磁碟區與可連線的執行個體中斷連接，然後再重新連接到無法連線的執行個體。將磁碟區連接到無法連線的執行個體時，請在裝置欄位中輸入您稍早儲存的裝置名稱。
16. 重新啟動無法連線的執行個體。

## 我的私有金鑰遺失。如何連接到我的 Windows 執行個體？

當您連線到新啟動的 Windows 執行個體時，使用您在啟動執行個體時所指定金鑰對中的私有金鑰，將管理員帳戶的密碼解密。

如果您遺失了管理員密碼，而且也不再擁有私有金鑰，則必須重設密碼或建立新的執行個體。如需詳細資訊，請參閱 [重設遺失或過期的 Windows 管理員密碼](#)。如需使用 Systems Manager 重設密碼之步驟的文件，請參閱《AWS Systems Manager 使用者指南》中的 [在 EC2 執行個體上重設密碼和 SSH 金鑰](#)。



# 重設遺失或過期的 Windows 管理員密碼

## Note

本節僅適用於 Windows 執行個體。

如果您因為 Windows 管理員密碼遺失或過期，而無法再存取 Windows Amazon EC2 執行個體，您可以重設密碼。

## Note

自動 AWS Systems Manager 化文件會自動套用重設本機管理員密碼所需的手動步驟。如需詳細資訊，請參閱 AWS Systems Manager 使用者指南中的 [重設 EC2 執行個體上的密碼和 SSH 金鑰](#)。

手動重設管理員密碼的方法需使用 EC2Launch v2、EC2Config 或 EC2Launch。

- 對於包含 EC2Launch v2 代理程式的所有支援 Windows AMI，請使用 EC2Launch v2。
- 如為 Windows Server 2016 之前的 Windows AMI，請用 EC2Config 服務。
- 如為 Windows Server 2016 和更新版本 AMI，請用 EC2Launch 服務。

這些程序還說明在遺失用來建立執行個體的金鑰對時如何連線至執行個體。Amazon EC2 使用公有金鑰來加密資料片段 (例如密碼)，而私有金鑰則用於解密資料。公有金鑰和私有金鑰稱為金鑰對。在 Windows 執行個體下用金鑰對取得管理員密碼，然後使用 RDP 登入。

## Note

如果您已停用執行個體上的本機管理員帳戶，且已為 Systems Manager 設定您的執行個體，您也可以使用 EC2Rescue 和執行命令來重新啟用和重設本機管理員密碼。如需詳細資訊，請參閱 [使用 EC2Rescue 搭配 Systems Manager 執行命令的 Windows 伺服器](#)。

## 目錄

- [使用 EC2Launch v2 重設 Windows 管理員密碼](#)

- [使用 EC2Config 重設 Windows 管理員密碼](#)
- [使用 EC2Launch 重設 Windows 管理員密碼](#)

## 使用 EC2Launch v2 重設 Windows 管理員密碼

如果您遺失了 Windows 管理員密碼，且使用包含 EC2Launch v2 代理程式的已支援 Windows AMI，您可以使用 EC2Launch v2 產生新密碼。

如果您使用的是不包含 EC2Launch v2 代理程式的 Windows Server 2016 或更新版本 AMI，請參閱 [使用 EC2Launch 重設 Windows 管理員密碼](#)。

如果您使用的是不包含 EC2Launch v2 代理程式的 Windows Server 2016 之前的 Windows Server AMI，請參閱 [使用 EC2Config 重設 Windows 管理員密碼](#)。

### Note

如果您已停用執行個體上的本機管理員帳戶，且已為 Systems Manager 設定您的執行個體，您也可以使用 EC2Rescue 和執行命令來重新啟用和重設本機管理員密碼。如需詳細資訊，請參閱 [使用 EC2Rescue 搭配 Systems Manager 執行命令的 Windows 伺服器](#)。

### Note

自動 AWS Systems Manager 化文件會自動套用重設本機管理員密碼所需的手動步驟。如需詳細資訊，請參閱 AWS Systems Manager 使用者指南中的 [重設 EC2 執行個體上的密碼和 SSH 金鑰](#)。

若要使用 EC2Launch v2 來重設 Windows 管理員密碼，您需要執行下列步驟：

- [步驟 1：確認 EC2Launch v2 代理程式正在執行](#)
- [步驟 2：將根磁碟區從執行個體中分離](#)
- [步驟 3：將磁碟區連接至暫時執行個體](#)
- [步驟 4：刪除 .run-once 檔案](#)
- [步驟 5：重新啟動原始執行個體](#)

## 步驟 1：確認 EC2Launch v2 代理程式正在執行

嘗試重設管理員密碼之前，請確認 EC2Launch v2 代理程式已安裝且正在執行中。您在本節稍後會用 EC2Launch v2 代理程式重設管理員密碼。

若要確認 EC2Launch v2 代理程式正在執行

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中選擇「Instances (執行個體)」，然後選擇需要重設密碼的執行個體。此執行個體即為本程序中的原始執行個體。
3. 選擇 Actions (動作)、Monitor and troubleshoot (監視和故障診斷)、Get system log (取得系統日誌)。
4. 找出 EC2 Launch 項目，例如 Launch: EC2Launch v2 service v2.0.124。如果您看到此項目，表示 EC2Launch v2 服務正在執行中。

如果系統日誌輸出為空白，或如果 EC2Launch v2 代理程式未執行，則請用執行個體主控台螢幕擷取畫面服務為執行個體進行故障診斷。如需詳細資訊，請參閱 [擷取無法連線執行個體的螢幕擷取畫面](#)。

## 步驟 2：將根磁碟區從執行個體中分離

如果存放密碼的磁碟區連接至執行個體做為根磁碟區，則您無法使用 EC2Launch v2 來重設管理員密碼。您必須將磁碟區從原始執行個體中分離，才能將磁碟區連接至暫時執行個體做為次要磁碟區。

若要將根磁碟區從執行個體中分離

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取需要重設密碼的執行個體，然後選擇執行個體狀態、停止執行個體。執行個體的状态變更為 Stopped (已停止) 之後，請繼續下一個步驟。
4. (選用) 如果您擁有在啟動此執行個體時已指定的私有金鑰，請繼續下一個步驟。否則，請使用下列步驟將執行個體取代為使用新金鑰對啟動的新執行個體。
  - a. 使用 Amazon EC2 主控台來建立新金鑰對。若要使新金鑰對的名稱與已遺失私有金鑰的金鑰對的名稱相同，您必須先刪除現有金鑰對。
  - b. 選取要取代的執行個體。請注意執行個體的執行個體類型、VPC、子網路、安全群組和 IAM 角色。

- c. 選擇 Actions (動作)、Image and templates (映像和範本)、Create image (建立映像)。輸入映像的名稱和說明，然後選擇 Create image (建立映像)。在導覽窗格中，選擇 AMIs (AMI)。在將映像狀態變更為 available (可用) 之後，繼續下一個步驟。
  - d. 選取映像，選擇 Actions (動作)，然後選擇 Launch (啟動)。
  - e. 完成精靈，選取相同的執行個體類型、VPC、子網路、安全群組和 IAM 角色作為要取代的執行個體，然後選擇 Launch (啟動)。
  - f. 出現提示時，請選擇您為新執行個體建立的金鑰對，選取確認核取方塊，然後選擇 Launch Instances (啟動執行個體)。
  - g. (選用) 如果原始執行個體有關聯的 Elastic IP 地址，將其傳輸至新執行個體。如果原始執行個體除了根磁碟區之外還有 EBS 磁碟區，請將它們傳輸至新執行個體。
5. 將根磁碟區從原始執行個體中分離，如下所示：
    - a. 選取原始執行個體並選擇儲存空間索引標籤。記下根裝置名稱下的根裝置名稱。在 [封鎖裝置] 底下找到具有此裝置名稱的磁碟區，並記下磁碟區 ID。
    - b. 在導覽窗格中，選擇 Volumes (磁碟區)。
    - c. 在磁碟區清單中，選取您記下為根裝置的磁碟區，然後選擇 [動作] > [卸離磁碟區]。在將磁碟區狀態變更為 available (可用) 後，繼續下一步驟。
  6. 如果您建立了新的執行個體來取代原始執行個體，您可以立即終止原始執行個體。它不再需要。對於此程序的其餘部分，原始例證的所有參照都會套用至您建立的新例證。

### 步驟 3：將磁碟區連接至暫時執行個體

接下來，啟動暫時執行個體，並將磁碟區連接至此執行個體做為次要磁碟區。這是您用來修改組態檔案的執行個體。

#### 啟動暫時執行個體並連接磁碟區

1. 啟動暫時執行個體，如下所示：
  - a. 在導覽窗格中，選擇 Instances (執行個體)，選擇 Launch instances (啟動執行個體)，然後選擇 AMI。

**⚠ Important**

若要避免磁碟簽章衝突，您必須為不同版本的 Windows 選取 AMI。例如，如果原始執行個體執行 Windows Server 2019，則使用適用於 Windows Server 2016 的基本 AMI 啟動暫時執行個體。

- b. 保留預設執行個體類型，然後選擇 Next: Configure Instance Details (下一步：設定執行個體的詳細資訊)。
- c. 在 Configure Instance Details (設定執行個體詳細資訊) 頁面中，對於 Subnet (子網路)，選取與原始執行個體相同的可用區域，然後選擇 Review and Launch (檢閱和啟動)。

**⚠ Important**

暫時執行個體與原始執行個體必須位於相同可用區域。如果您的暫時執行個體位於不同的可用區域，則無法將原始執行個體的根磁碟區與它連接。

- d. 在 Review Instance Launch (檢閱執行個體啟動) 頁面，選擇 Launch (啟動)。
  - e. 出現提示時，建立新金鑰對，將其下載到電腦上的安全位置，然後選擇 Launch Instances (啟動執行個體)。
2. 將磁碟區連接到暫時執行個體作為輔助磁碟區，如下所示：
    - a. 在導覽窗格中，選擇 Volumes (磁碟區)，選取從原始執行個體分離的磁碟區，然後選擇 Actions (動作)、Attach Volume (連接磁碟區)。
    - b. 在 Attach Volume (連接磁碟區) 對話方塊中，對於 Instances (執行個體)，開始輸入暫時執行個體的名稱或 ID，然後從清單中選取執行個體。
    - c. 對於 Device (裝置)，輸入 **xvdf** (如果尚未出現在此)，選擇 Attach (連接)。

## 步驟 4：刪除 .run-once 檔案

現在必須從附接至執行個體的離線磁碟區刪除 .run-once 檔案。這會指示 EC2Launch v2 以 once 頻率執行所有工作，其中包括設定管理員密碼。您連接之輔助磁碟區中的檔案路徑會類似 D:\ProgramData\Amazon\EC2Launch\state\.run-once。

若要刪除 .run-once 檔案

1. 開啟磁碟管理公用程式，然後使用下列指示將磁碟機連線：[讓 Amazon EBS 磁碟區可供使用](#)。

2. 在已連線的磁碟中找到 `.run-once` 檔案。
3. 刪除 `.run-once` 檔案。

**⚠ Important**

設定為執行一次的任何指令碼都會由此動作觸發。

## 步驟 5：重新啟動原始執行個體

刪除 `.run-once` 檔案之後，將磁碟區重新連接至原始執行個體做為根磁碟區，然後使用金鑰對來連接至執行個體，以擷取管理員密碼。

1. 將磁碟區重新連接到原始執行個體，如下所示：
  - a. 在導覽窗格中，選擇 Volumes (磁碟區)，選取從暫時執行個體分離的磁碟區，然後選擇 Actions (動作)、Attach Volume (連接磁碟區)。
  - b. 在 Attach Volume (連接磁碟區) 對話方塊中，對於 Instances (執行個體)，開始輸入您的原始執行個體的名稱或 ID，然後選取執行個體。
  - c. 對於 Device (裝置)，輸入 `/dev/sda1`。
  - d. 選擇 Attach (連接)。在磁碟區狀態變更為 `in-use` 之後，繼續下一個步驟。
2. 在導覽窗格中，選擇 Instances (執行個體)。選取原始執行個體，然後依序選擇 Instance state (執行個體狀態)、Start instance (啟動執行個體)。在執行個體狀態變更為 `Running` 之後，繼續下一個步驟。
3. 使用新金鑰對的私有金鑰擷取新的 Windows 系統管理員密碼，然後連線到執行個體。如需詳細資訊，請參閱 [連線至 Windows 執行個體](#)。

**⚠ Important**

先停止執行個體，接著再啟動後，該執行個體將得到新的公有 IP 地址。務必使用目前的公有 DNS 名稱連接至執行個體。如需詳細資訊，請參閱 [執行個體生命週期](#)。

4. (選用) 如果不再使用暫時執行個體，您可以終止它。選取暫時執行個體，再依序選擇 Instance State (執行個體狀態)、Terminate instance (終止執行個體)。

## 使用 EC2Config 重設 Windows 管理員密碼

如果您遺失了 Windows 管理員密碼，且使用 Windows Server 2016 之前的 Windows AMI，您可以使用 EC2Config 代理程式產生新密碼。

如果您使用的是 Windows Server 2016 或更新版本 AMI，請參閱 [使用 EC2Launch 重設 Windows 管理員密碼](#)，或您可透過 [EC2Rescue 工具](#) 以利用 EC2Launch 服務產生新密碼。

### Note

如果您已停用執行個體上的本機管理員帳戶，且已為 Systems Manager 設定您的執行個體，您也可以使用 EC2Rescue 和執行命令來重新啟用和重設本機管理員密碼。如需詳細資訊，請參閱 [使用 EC2Rescue 搭配 Systems Manager 執行命令的 Windows 伺服器](#)。

### Note

自動 AWS Systems Manager 化文件會自動套用重設本機管理員密碼所需的手動步驟。如需詳細資訊，請參閱 AWS Systems Manager 使用者指南中的 [重設 EC2 執行個體上的密碼和 SSH 金鑰](#)。

若要使用 EC2Config 來重設 Windows 管理員密碼，您需要執行下列步驟：

- [步驟 1：確認 EC2Config 服務在執行中](#)
- [步驟 2：將根磁碟區從執行個體中分離](#)
- [步驟 3：將磁碟區連接至暫時執行個體](#)
- [步驟 4：修改組態檔案](#)
- [步驟 5：重新啟動原始執行個體](#)

### 步驟 1：確認 EC2Config 服務在執行中

嘗試重設管理員密碼之前，請確認 EC2Config 服務已安裝且正執行中。您在本節稍後會用 EC2Config 服務重設管理員密碼。

確認 EC2Config 服務在執行中

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。

2. 在導覽窗格中選擇「Instances (執行個體)」，然後選擇需要重設密碼的執行個體。此執行個體即為本程序中的原始執行個體。
3. (新主控台) 選擇 Actions (動作)、Monitor and troubleshoot (監視和故障診斷)、Get system log (取得系統日誌)。

(舊主控台) 選擇 Actions (動作)、System Settings (系統設定)、Get System Log (取得系統日誌)。

4. 找出 EC2 Agent 項目，例如：EC2 Agent: Ec2Config service v3.18.1118。如果找到此項目，表示 EC2Config 服務正在執行中。

如果系統日誌輸出為空白，或如果 EC2Config 服務未執行，請用執行個體主控台螢幕擷取畫面服務為執行個體進行故障診斷。如需詳細資訊，請參閱 [擷取無法連線執行個體的螢幕擷取畫面](#)。

## 步驟 2：將根磁碟區從執行個體中分離

如果存放密碼的磁碟區連接至執行個體做為根磁碟區，則您無法使用 EC2Config 來重設管理員密碼。您必須將磁碟區從原始執行個體中分離，才能將磁碟區連接至暫時執行個體做為次要磁碟區。

若要將根磁碟區從執行個體中分離

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取需要重設密碼的執行個體，然後選擇執行個體狀態、停止執行個體。執行個體的狀態變更為 Stopped (已停止) 之後，請繼續下一個步驟。
4. (選用) 如果您擁有在啟動此執行個體時已指定的私有金鑰，請繼續下一個步驟。否則，請使用下列步驟將執行個體取代為使用新金鑰對啟動的新執行個體。
  - a. 使用 Amazon EC2 主控台來建立新金鑰對。若要使新金鑰對的名稱與已遺失私有金鑰的金鑰對的名稱相同，您必須先刪除現有金鑰對。
  - b. 選取要取代的執行個體。請注意執行個體的執行個體類型、VPC、子網路、安全群組和 IAM 角色。
  - c. 選擇 Actions (動作)、Image and templates (映像和範本)、Create image (建立映像)。輸入映像的名稱和說明，然後選擇 Create image (建立映像)。在導覽窗格中，選擇 AMIs (AMI)。在將映像狀態變更為 available (可用) 之後，繼續下一個步驟。
  - d. 選取映像，選擇 Actions (動作)，然後選擇 Launch (啟動)。
  - e. 完成精靈，選取相同的執行個體類型、VPC、子網路、安全群組和 IAM 角色作為要取代的執行個體，然後選擇 Launch (啟動)。



- f. 出現提示時，請選擇您為新執行個體建立的金鑰對，選取確認核取方塊，然後選擇 Launch Instances (啟動執行個體)。
  - g. (選用) 如果原始執行個體有關聯的 Elastic IP 地址，將其傳輸至新執行個體。如果原始執行個體除了根磁碟區之外還有 EBS 磁碟區，請將它們傳輸至新執行個體。
5. 將根磁碟區從原始執行個體中分離，如下所示：
    - a. 選取原始執行個體並選擇儲存空間索引標籤。記下根裝置名稱下的根裝置名稱。在 [封鎖裝置] 底下找到具有此裝置名稱的磁碟區，並記下磁碟區 ID。
    - b. 在導覽窗格中，選擇 Volumes (磁碟區)。
    - c. 在磁碟區清單中，選取您記下為根裝置的磁碟區，然後選擇 [動作] > [卸離磁碟區]。在將磁碟區狀態變更為 available (可用) 後，繼續下一步驟。
  6. 如果您建立了新的執行個體來取代原始執行個體，您可以立即終止原始執行個體。它不再需要。對於此程序的其餘部分，原始例證的所有參照都會套用至您建立的新例證。

### 步驟 3：將磁碟區連接至暫時執行個體

接下來，啟動暫時執行個體，並將磁碟區連接至此執行個體做為次要磁碟區。這是您用來修改組態檔案的執行個體。

#### 啟動暫時執行個體並連接磁碟區

1. 啟動暫時執行個體，如下所示：
  - a. 在導覽窗格中，選擇 Instances (執行個體)，選擇 Launch instances (啟動執行個體)，然後選擇 AMI。

#### Important

若要避免磁碟簽章衝突，您必須為不同版本的 Windows 選取 AMI。例如，如果原始執行個體執行 Windows Server 2019，則使用適用於 Windows Server 2016 的基本 AMI 啟動暫時執行個體。

- b. 保留預設執行個體類型，然後選擇 Next: Configure Instance Details (下一步：設定執行個體的詳細資訊)。
- c. 在 Configure Instance Details (設定執行個體詳細資訊) 頁面中，對於 Subnet (子網路)，選取與原始執行個體相同的可用區域，然後選擇 Review and Launch (檢閱和啟動)。

**⚠ Important**

暫時執行個體與原始執行個體必須位於相同可用區域。如果您的暫時執行個體位於不同的可用區域，則無法將原始執行個體的根磁碟區與它連接。

- d. 在 Review Instance Launch (檢閱執行個體啟動) 頁面，選擇 Launch (啟動)。
  - e. 出現提示時，建立新金鑰對，將其下載到電腦上的安全位置，然後選擇 Launch Instances (啟動執行個體)。
2. 將磁碟區連接到暫時執行個體作為輔助磁碟區，如下所示：
    - a. 在導覽窗格中，選擇 Volumes (磁碟區)，選取從原始執行個體分離的磁碟區，然後選擇 Actions (動作)、Attach Volume (連接磁碟區)。
    - b. 在 Attach Volume (連接磁碟區) 對話方塊中，對於 Instances (執行個體)，開始輸入暫時執行個體的名稱或 ID，然後從清單中選取執行個體。
    - c. 對於 Device (裝置)，輸入 **xvdf** (如果尚未出現在此)，選擇 Attach (連接)。

#### 步驟 4：修改組態檔案

將磁碟區連接至暫時執行個體做為次要磁碟區之後，請在組態檔案中修改 Ec2SetPassword 外掛程式。

##### 修改組態檔案

1. 從暫時執行個體，修改次要磁碟區上的組態檔案，如下所示：
  - a. 啟動並連接至暫時執行個體。
  - b. 使用下列指示將磁碟機上線：[讓 Amazon EBS 磁碟區可供使用](#)。
  - c. 導覽至次要磁碟區，然後使用「記事本」等文字編輯器開啟 `\Program Files\Amazon\Ec2ConfigService\Settings\config.xml`。
  - d. 在檔案頂端，找出名稱為 Ec2SetPassword 的外掛程式，如螢幕擷取畫面所示。將狀態從 Disabled 變更為 Enabled 並儲存檔案。

```
<?xml version="1.0" standalone="yes"?>
<Ec2ConfigurationSettings>
 <Plugins>
 <Plugin>
 <Name>Ec2SetPassword</Name>
 <State>Disabled</State>
 </Plugin>
 <Plugin>
 <Name>Ec2SetComputerName</Name>
 <State>Disabled</State>
 </Plugin>
 <Plugin>
 <Name>Ec2InitializeDrives</Name>
 <State>Enabled</State>
 </Plugin>
 <Plugin>
 <Name>Ec2EventLog</Name>
 <State>Disabled</State>
 </Plugin>
 <Plugin>
 <Name>Ec2ConfigureRDP</Name>
 <State>Disabled</State>
 </Plugin>
 <Plugin>
 <Name>Ec2OutputRDPcert</Name>
 <State>Enabled</State>
 </Plugin>
 <Plugin>
 <Name>Ec2SetDriveLetter</Name>
 <State>Enabled</State>
 </Plugin>
 <Plugin>

```

2. 修改組態檔案之後，將次要磁碟區從暫時執行個體中分離，如下所示：
  - a. 使用 Disk Management (磁碟管理) 公用程式讓磁碟區離線。
  - b. 中斷與暫時執行個體的連線，並返回 Amazon EC2 主控台。
  - c. 在導覽窗格中選擇 Volumes (磁碟區)、選取磁碟區，然後選擇 Actions (動作)、Detach Volume (分離磁碟區)。在將磁碟區狀態變更為 available (可用) 後，繼續下一步驟。

## 步驟 5：重新啟動原始執行個體

修改組態檔案之後，將磁碟區重新連接至原始執行個體做為根磁碟區，然後使用金鑰對來連接至執行個體，以擷取管理員密碼。

1. 將磁碟區重新連接到原始執行個體，如下所示：
  - a. 在導覽窗格中，選擇 Volumes (磁碟區)，選取從暫時執行個體分離的磁碟區，然後選擇 Actions (動作)、Attach Volume (連接磁碟區)。

- b. 在 Attach Volume (連接磁碟區) 對話方塊中，對於 Instances (執行個體)，開始輸入您的原始執行個體的名稱或 ID，然後選取執行個體。
  - c. 對於 Device (裝置)，輸入 `/dev/sda1`。
  - d. 選擇 Attach (連接)。在磁碟區狀態變更為 in-use 之後，繼續下一個步驟。
2. 在導覽窗格中，選擇 Instances (執行個體)。選取原始執行個體，然後依序選擇 Instance state (執行個體狀態)、Start instance (啟動執行個體)。在執行個體狀態變更為 Running 之後，繼續下一個步驟。
  3. 使用新金鑰對的私有金鑰擷取新的 Windows 系統管理員密碼，然後連線到執行個體。如需詳細資訊，請參閱 [連線至 Windows 執行個體](#)。

#### Important

先停止執行個體，接著再啟動後，該執行個體將得到新的公有 IP 地址。務必使用目前的公有 DNS 名稱連接至執行個體。如需詳細資訊，請參閱 [執行個體生命週期](#)。

4. (選用) 如果不再使用暫時執行個體，您可以終止它。選取暫時執行個體，再依序選擇 Instance State (執行個體狀態)、Terminate instance (終止執行個體)。

## 使用 EC2Launch 重設 Windows 管理員密碼

如果您遺失了 Windows 管理員密碼，且使用 Windows Server 2016 或更新版本 AMI，您可透過 [EC2Rescue 工具](#) 以利用 EC2Launch 服務產生新密碼。

如果您使用的是不包含 EC2Launch v2 代理程式的 Windows Server 2016 或更新版本 AMI，則可以使用 EC2Launch v2 產生新密碼。

如果使用 Windows Server 2016 之前的 Windows Server AMI，請參閱 [使用 EC2Config 重設 Windows 管理員密碼](#)。

#### Warning

停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區的資料，請將資料備份至持久性儲存。

**Note**

如果您已停用執行個體上的本機管理員帳戶，且已為 Systems Manager 設定您的執行個體，您也可以使用 EC2Rescue 和執行命令來重新啟用和重設本機管理員密碼。如需詳細資訊，請參閱[使用 EC2Rescue 搭配 Systems Manager 執行命令的 Windows 伺服器](#)。

**Note**

自動 AWS Systems Manager 化文件會自動套用重設本機管理員密碼所需的手動步驟。如需詳細資訊，請參閱 AWS Systems Manager 使用者指南中的[重設 EC2 執行個體上的密碼和 SSH 金鑰](#)。

若要使用 EC2Launch 來重設 Windows 管理員密碼，您需要執行下列步驟：

- [步驟 1：將根磁碟區從執行個體中分離](#)
- [步驟 2：將磁碟區連接至暫時執行個體](#)
- [步驟 3：重設管理員密碼](#)
- [步驟 4：重新啟動原始執行個體](#)

### 步驟 1：將根磁碟區從執行個體中分離

如果存放密碼的磁碟區連接至執行個體做為根磁碟區，則您無法使用 EC2Launch 來重設管理員密碼。您必須將磁碟區從原始執行個體中分離，才能將磁碟區連接至暫時執行個體做為次要磁碟區。

若要將根磁碟區從執行個體中分離

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 選取需要重設密碼的執行個體，然後選擇執行個體狀態、停止執行個體。執行個體的状态變更為 Stopped (已停止) 之後，請繼續下一個步驟。
4. (選用) 如果您擁有在啟動此執行個體時已指定的私有金鑰，請繼續下一個步驟。否則，請使用下列步驟將執行個體取代為使用新金鑰對啟動的新執行個體。
  - a. 使用 Amazon EC2 主控台來建立新金鑰對。若要使新金鑰對的名稱與已遺失私有金鑰的金鑰對的名稱相同，您必須先刪除現有金鑰對。

- b. 選取要取代的執行個體。請注意執行個體的執行個體類型、VPC、子網路、安全群組和 IAM 角色。
  - c. 選擇 Actions (動作)、Image and templates (映像和範本)、Create image (建立映像)。輸入映像的名稱和說明，然後選擇 Create image (建立映像)。在導覽窗格中，選擇 AMIs (AMI)。在將映像狀態變更為 available (可用) 之後，繼續下一個步驟。
  - d. 選取映像，選擇 Actions (動作)，然後選擇 Launch (啟動)。
  - e. 完成精靈，選取相同的執行個體類型、VPC、子網路、安全群組和 IAM 角色作為要取代的執行個體，然後選擇 Launch (啟動)。
  - f. 出現提示時，請選擇您為新執行個體建立的金鑰對，選取確認核取方塊，然後選擇 Launch Instances (啟動執行個體)。
  - g. (選用) 如果原始執行個體有關聯的 Elastic IP 地址，將其傳輸至新執行個體。如果原始執行個體除了根磁碟區之外還有 EBS 磁碟區，請將它們傳輸至新執行個體。
5. 將根磁碟區從原始執行個體中分離，如下所示：
    - a. 選取原始執行個體並選擇儲存空間索引標籤。記下根裝置名稱下的根裝置名稱。在 [封鎖裝置] 底下找到具有此裝置名稱的磁碟區，並記下磁碟區 ID。
    - b. 在導覽窗格中，選擇 Volumes (磁碟區)。
    - c. 在磁碟區清單中，選取您記下為根裝置的磁碟區，然後選擇 [動作] > [卸離磁碟區]。在將磁碟區狀態變更為 available (可用) 後，繼續下一步驟。
  6. 如果您建立了新的執行個體來取代原始執行個體，您可以立即終止原始執行個體。它不再需要。對於此程序的其餘部分，原始例證的所有參照都會套用至您建立的新例證。

## 步驟 2：將磁碟區連接至暫時執行個體

接下來，啟動暫時執行個體，並將磁碟區連接至此執行個體做為次要磁碟區。這是您用來執行 EC2Launch 的執行個體。

### 啟動暫時執行個體並連接磁碟區

1. 啟動暫時執行個體，如下所示：
  - a. 在導覽窗格中，選擇 Instances (執行個體)，選擇 Launch instances (啟動執行個體)，然後選擇 AMI。

**⚠ Important**

若要避免磁碟簽章衝突，您必須為不同版本的 Windows 選取 AMI。例如，如果原始執行個體執行 Windows Server 2019，則使用適用於 Windows Server 2016 的基本 AMI 啟動暫時執行個體。

- b. 保留預設執行個體類型，然後選擇 Next: Configure Instance Details (下一步：設定執行個體的詳細資訊)。
- c. 在 Configure Instance Details (設定執行個體詳細資訊) 頁面中，對於 Subnet (子網路)，選取與原始執行個體相同的可用區域，然後選擇 Review and Launch (檢閱和啟動)。

**⚠ Important**

暫時執行個體與原始執行個體必須位於相同可用區域。如果您的暫時執行個體位於不同的可用區域，則無法將原始執行個體的根磁碟區與它連接。

- d. 在 Review Instance Launch (檢閱執行個體啟動) 頁面，選擇 Launch (啟動)。
  - e. 出現提示時，建立新金鑰對，將其下載到電腦上的安全位置，然後選擇 Launch Instances (啟動執行個體)。
2. 將磁碟區連接到暫時執行個體作為輔助磁碟區，如下所示：
    - a. 在導覽窗格中，選擇 Volumes (磁碟區)，選取從原始執行個體分離的磁碟區，然後選擇 Actions (動作)、Attach Volume (連接磁碟區)。
    - b. 在 Attach Volume (連接磁碟區) 對話方塊中，對於 Instances (執行個體)，開始輸入暫時執行個體的名稱或 ID，然後從清單中選取執行個體。
    - c. 對於 Device (裝置)，輸入 `xvdf` (如果尚未出現在此)，選擇 Attach (連接)。

### 步驟 3：重設管理員密碼

接下來，連接至暫時執行個體，並使用 EC2Launch 來重設管理員密碼。

#### 重設管理員密碼

1. 連接至暫時執行個體，並在執行個體上使用 EC2Rescue for Windows Server 工具重設管理員密碼，如下所示：

- a. 下載 [EC2Rescue for Windows Server](#) zip 檔案，解壓縮內容，然後執行 EC2Rescue.exe (EC2Rescue.exe)。
  - b. 在 License Agreement (授權合約) 畫面上閱讀授權合約，如果接受條款，請選擇 I Agree (我同意)。
  - c. 在 Welcome to EC2Rescue for Windows Server (歡迎使用 EC2Rescue for Windows Server) 畫面上，選擇 Next (下一步)。
  - d. 在 Select mode (選取模式) 畫面上，選擇 Offline instance (離線執行個體)。
  - e. 在 Select a disk (選取磁碟) 畫面上選取 xvdf 設備，然後選擇 Next (下一步)。
  - f. 確認選取磁碟，然後選擇 Yes (是)。
  - g. 載入磁碟區之後，選擇 OK (確定)。
  - h. 在 Select Offline Instance Option (選取離線執行個體選項) 畫面上，選擇 Diagnose and Rescue (診斷與救援)。
  - i. 在 Summary (摘要) 畫面上檢閱資訊，然後選擇 Next (下一步)。
  - j. 在 Detected possible issues (偵測到的可能問題) 畫面上，選取 Reset Administrator Password (重設管理員密碼)，然後選擇 Next (下一步)。
  - k. 在 Confirm (確認) 畫面上選擇 Rescue (救援)、OK (確定)。
  - l. 在 Done (完成) 畫面上選擇 Finish (完成)。
  - m. 關閉 EC2Rescue for Windows Server 工具，中斷與暫時執行個體的連線，然後返回 Amazon EC2 主控台。
2. 將次要 (xvdf) 磁碟區從暫時執行個體中分離，如下所示：
    - a. 在導覽窗格中選擇 Instances (執行個體)，然後選取暫時執行個體。
    - b. 在暫時執行個體的「Storage (儲存)」標籤上，記下列為 xvdf 的 EBS 磁碟區識別碼。
    - c. 在導覽窗格中，選擇 Volumes (磁碟區)。
    - d. 在磁碟區清單中，請勾選之前步驟中所記下的磁碟區，並選擇 Actions (動作)、Detach Volume (分離磁碟區)。在將磁碟區狀態變更為 available (可用) 後，繼續下一步驟。

#### 步驟 4：重新啟動原始執行個體

使用 EC2Launch 來重設管理員密碼之後，將磁碟區重新連接至原始執行個體做為根磁碟區，然後使用金鑰對來連接至執行個體，以擷取管理員密碼。



## 重新啟動原始執行個體

1. 將磁碟區重新連接到原始執行個體，如下所示：
  - a. 在導覽窗格中，選擇 Volumes (磁碟區)，選取從暫時執行個體分離的磁碟區，然後選擇 Actions (動作)、Attach Volume (連接磁碟區)。
  - b. 在 Attach Volume (連接磁碟區) 對話方塊中，對於 Instances (執行個體)，開始輸入您的原始執行個體的名稱或 ID，然後選取執行個體。
  - c. 對於 Device (裝置)，輸入 `/dev/sda1`。
  - d. 選擇 Attach (連接)。在磁碟區狀態變更為 in-use 之後，繼續下一個步驟。
2. 在導覽窗格中，選擇 Instances (執行個體)。選取原始執行個體，然後依序選擇 Instance state (執行個體狀態)、Start instance (啟動執行個體)。在執行個體狀態變更為 Running 之後，繼續下一個步驟。
3. 使用新金鑰對的私有金鑰擷取新的 Windows 系統管理員密碼，然後連線到執行個體。如需詳細資訊，請參閱 [連線至 Windows 執行個體](#)。
4. (選用) 如果不再使用暫時執行個體，您可以終止它。選取暫時執行個體，再依序選擇 Instance State (執行個體狀態)、Terminate instance (終止執行個體)。

## 無法連線故障診斷執行個體

您可以使用下列方法對無法連線的 Amazon EC2 執行個體進行疑難排解。

### 目錄

- [重新啟動執行個體](#)
- [執行個體主控台輸出](#)
- [擷取無法連線執行個體的螢幕擷取畫面](#)
- [Windows 執行個體的常見螢幕擷](#)
- [當主機電腦故障時的執行個體復原](#)

## 重新啟動執行個體

對於故障診斷和一般的執行個體管理，能夠重新啟動無法連線的執行個體很重要。

如同您可以按下重設鈕來重設電腦，您也可以使用 Amazon EC2 主控台、CLI 或 API，來重設 EC2 執行個體。如需詳細資訊，請參閱 [重新啟動您的執行個體](#)。

## 執行個體主控台輸出

主控台輸出是診斷問題的一項寶貴工具。在針對核心問題和服務組態問題進行故障診斷時 (這些問題可能會造成執行個體終止，或是在其 SSH 協助程式能夠啟動之前較無法連線)，這項工具尤其實用。

- Linux 執行個體 — 執行個體主控台輸出會顯示通常會在連接至電腦的實體監視器上顯示的確切主控台輸出。主控台輸出會傳回經過緩衝的資訊，這些資訊會在執行個體轉換狀態 (啟動、停止、重新啟動和終止) 之後不久發布。已發布的輸出不會連續更新，只有在可能具有最高價值時才會更新。
- Windows 執行個體 — 執行個體主控台輸出包含最後三個系統事件記錄錯誤。

在執行個體生命週期內，您隨時可以選擇擷取最新的序列主控台輸出。只有在 [AWS Nitro 系統上建置的執行個體上](#) 才支援此選項。不支援透過 Amazon EC2 主控台來使用此選項。

### Note

只會存放最近發布的 64 KB 輸出資料，在最近一次發布後至少 1 小時內，都能使用這些資料。

只有執行個體才能存取主控台輸出資料。

使用下列方法之一來獲取主控台輸出。

### Console

無法取得主控台輸出

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽窗格中選擇 (執行個體)。
3. 選取執行個體，然後再依序選擇動作、監控和故障診斷和取得系統記錄檔。

### Command line

無法取得主控台輸出

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱 [存取 Amazon EC2](#)。

- [get-console-output](#) (AWS CLI)
- [Get-EC2ConsoleOutput](#) (AWS Tools for Windows PowerShell)

## 擷取無法連線執行個體的螢幕擷取畫面

如果您無法連線至執行個體，您可以擷取執行個體的螢幕擷取畫面，並以映像檔的形式檢視。此影像可讓您看到執行個體的状态，有利於更快進行故障診斷。

您可以在執行個體執行中或執行個體當機後產生螢幕擷取畫面。產生的影像格式為 JPG，大小不超過 100 kb。螢幕擷取畫面不收取資料傳輸費用。

### 限制

下列項目不支援此功能：

- 裸機執行個體 (\*.metal 類型的執行個體)
- 執行個體正在使用 NVIDIA GRID 驅動程式
- [以 ARM 為基礎的重力子處理器提供支援的執行個體](#)
- 視窗執行個體 AWS Outposts

### 支援的區域

此功能在以下 區域可用：

- US East (N. Virginia) Region
- 美國東部 (俄亥俄) 區域
- 美國西部 (加利佛尼亞北部) 區域
- 美國西部 (奧勒岡) 區域
- 非洲 (開普敦) 區域
- 亞太區域 (香港) 區域
- 亞太區域 (海德拉巴)
- 亞太區域 (雅加達)
- 亞太區域 (墨爾本) 區域
- 亞太 (孟買) 區域
- 亞太 (大阪) 區域
- 亞太 (首爾) 區域
- 亞太區域 (新加坡) 區域
- 亞太 (雪梨) 區域

- 亞太 (東京) 區域
- 加拿大 (中部) 區域
- 加拿大西部 (卡加利) 區域
- 中國 (北京) 區域
- 中國 (寧夏) 區域
- 歐洲 (法蘭克福) 區域
- 歐洲 (愛爾蘭) 區域
- 歐洲 (倫敦) 區域
- Europe (Milan) Region
- 歐洲 (巴黎) 區域
- 歐洲 (西班牙) 區域
- 歐洲 (斯德哥爾摩) 區域
- 歐洲 (蘇黎世) 區域
- 以色列 (特拉維夫) 區域
- 南美洲 (聖保羅) 區域
- Middle East (Bahrain) Region
- 中東 (阿拉伯聯合大公國) 區域

## Console

### 取得執行個體的螢幕擷取畫面

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽窗格中選擇 (執行個體)。
3. 選取要擷取的執行個體。
4. 選擇 動作、監視和故障診斷、取得執行個體螢幕截圖。
5. 選擇 下載，或在要下載的影像上按一下滑鼠右鍵並儲存。

## Command line

### 擷取執行個體的螢幕擷取畫面

您可以使用下列其中一個命令。傳回的內容為 base64 編碼。如需有關這些命令列介面的詳細資訊，請參閱[存取 Amazon EC2](#)。

- [取得-主控台-螢幕截圖](#) (AWS CLI)
- [GetConsole螢幕擷取畫面](#) (Amazon EC2 查詢 API)

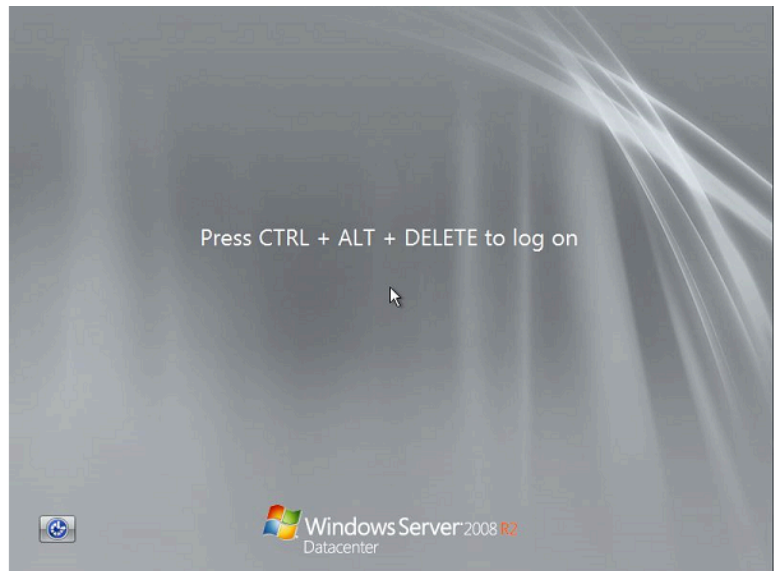
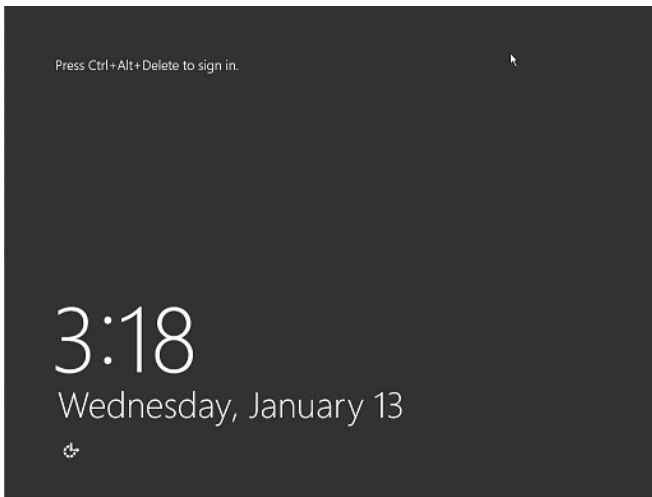
## Windows 執行個體的常見螢幕擷

您可根據服務傳回的螢幕截取畫面，使用下列資訊協助您故障診斷無法連線的 Windows 執行個體。

- [登入畫面 \(Ctrl+Alt+Delete\)](#)
- [復原主控台螢幕](#)
- [Windows 開機管理程式畫面](#)
- [Sysprep 畫面](#)
- [就緒畫面](#)
- [Windows Update 畫面](#)
- [Chkdsk](#)

### 登入畫面 (Ctrl+Alt+Delete)

主控台螢幕截取畫面服務傳回下列內容。



如果在登入期間無法連線某個執行個體，可能是您的網路組態或 Windows 遠端桌面服務發生問題。如有程序正在佔用大量的 CPU，執行個體也可能不回應。

### 網路組態

請使用下列資訊來確認您的 AWS、Microsoft Windows 和本機 (或內部部署) 網路組態並未封鎖對執行個體的存取。

### AWS 網路組態

組態	確認
安全群組組態	確認連接埠 3389 已開放給您的安全群組。確認您連線到正確的公有 IP 地址。如果執行個體與彈性 IP 地址無關聯，則公有 IP 地址會在執行個體停止/啟動後變更。如需詳細資訊，請參閱 <a href="#">遠端桌面無法連線到遠端電腦</a> 。
VPC 組態 (網路 ACL)	確認您 Amazon VPC 的存取控制清單 (ACL) 未封鎖存取。如需相關資訊，請參閱《Amazon VPC 使用者指南》中的 <a href="#">網路 ACL</a> 。
VPN 組態	如果您使用虛擬私有網路 (VPN) 連線至您的 VPC，請確認 VPN 通道的連線能力。如需詳細資訊，請參閱 <a href="#">如何故障診斷 VPN 對 Amazon VPC 的通道連線？</a>

## Windows 網路組態

組態	確認
Windows 防火牆	確認 Windows 防火牆未封鎖對您執行個體的連線。如遠端桌面故障診斷一節 <a href="#">遠端桌面無法連線到遠端電腦</a> 第 7 項所述，停用 Windows 防火牆。
進階 TCP/IP 組態 (使用靜態 IP)	因為您設定了靜態 IP 地址，所以執行個體可能不回應。若為 VPC，請 <a href="#">建立網路介面</a> 並 <a href="#">連接至執行個體</a> 。

## 本機或內部部署的網路組態

確認本機網路組態未封鎖存取。嘗試連線到與無法連線之執行個體同一 VPC 的另一個執行個體。如果您無法存取另一個執行個體，請與您的本機網路管理員合作，判斷本機政策是否限制存取。

## 遠端桌面服務問題

如果在登入期間無法連線執行個體，可能是執行個體上的遠端桌面服務 (RDS) 發生問題。

 Tip

您可以使用 `AWSsupport-TroubleshootRDP` 執行手冊，檢查和修改可能影響遠端桌面通訊協定 (RDP) 連線的各種設定。如需詳細資訊，請參閱 AWS Systems Manager Automation Runbook 參考中的 [AWSsupport-TroubleshootRDP](#)。

## 遠端桌面服務組態

組態	確認
RDS 執行中	確認 RDS 是否正在執行個體上執行。使用 Microsoft Management Console (MMC) 服務嵌入式管理單元連線至執行個體 ( <code>services.msc</code> )。在服務清單中，確認 Remote Desktop Services (遠端桌面服務) 為 Running (執行中)。如果不是，請啟動它，然後將啟動類型設為 Automatic (自動)。如果使用服務嵌入式管理單元無法連線至執行個體，請分離根磁碟區與執行個體、

組態	確認
	拍攝磁碟區快照或從磁碟區建立 AMI、將原始磁碟區連接到同一可用區域的另一個執行個體做為輔助磁碟區，然後修改 <a href="#">Start</a> 登錄機碼。完成後，重新將根磁碟區連接到原始執行個體。
RDS 已啟用	即使服務已啟動，仍可能停用。分離根磁碟區與執行個體、拍攝磁碟區快照或從磁碟區建立 AMI、將原始磁碟區連接到相同可用區域的另一個執行個體做為輔助磁碟區，然後如 <a href="#">使用遠端登錄在 EC2 執行個體上啟用遠端桌面</a> 所述，修改 Terminal Server 登錄機碼來啟用服務。  完成後，重新將根磁碟區連接到原始執行個體。

## 高 CPU 使用率

使用 Amazon 檢查執行個體上的 CPU 使用率 (最大值) 指標。CloudWatch 如果 CPUUtilization (Maximum) 的值很高，請等 CPU 下降再嘗試連線。CPU 使用率高的可能的原因如下：

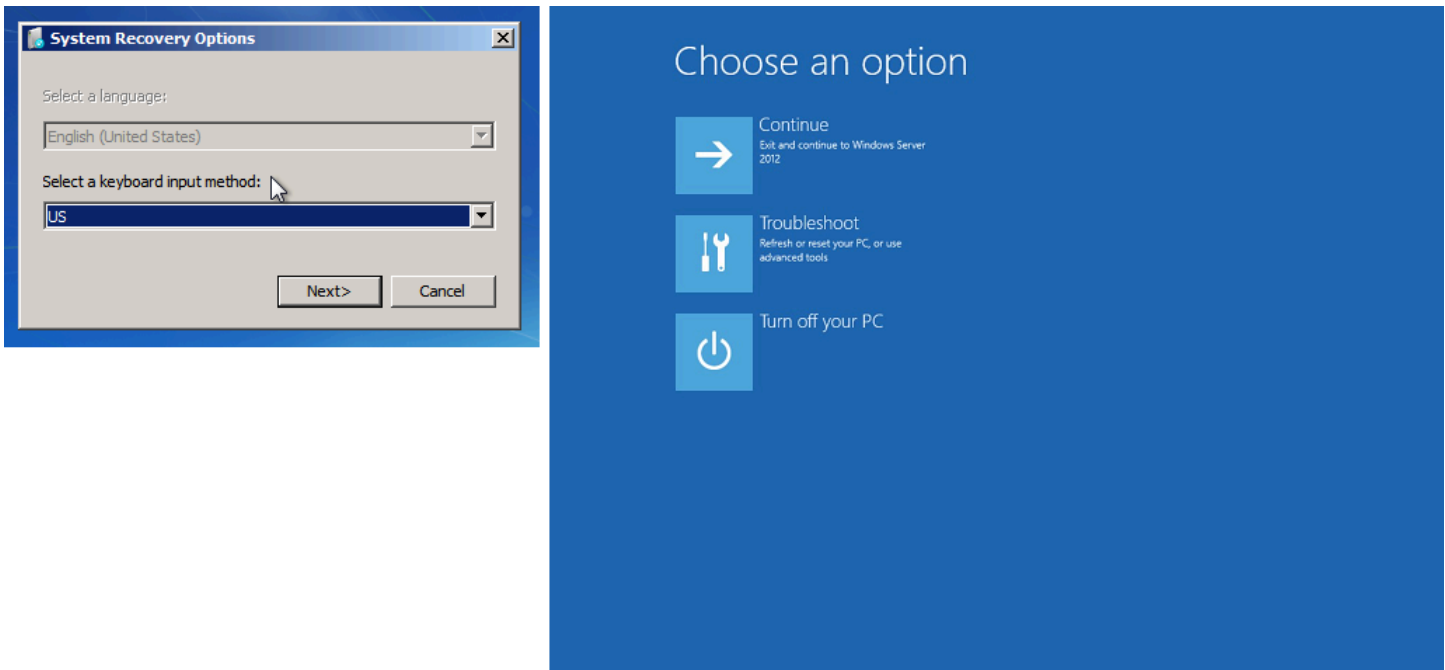
- Windows 更新
- 安全軟體掃描
- 自訂啟動指令碼
- 任務排程

如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的 [取得特定資源的統計資料](#)。如需其他的故障診斷秘訣，請參閱 [視窗啟動後不久的 CPU 使用率過高 \(僅限 Windows 執行個體\)](#)。

## 復原主控台螢幕

主控台螢幕截取畫面服務傳回下列內容。





作業系統可能在復原主控台中開機，然後卡在這個狀態，如果 `bootstatuspolicy` 不是設為 `ignoreallfailures`。請使用下列程序將 `bootstatuspolicy` 組態變更為 `ignoreallfailures`。

根據預設，由提供的公用 Windows AMI 的原則組態設定會設定 AWS 為 `ignoreallfailures`。

1. 停止無法連線的執行個體。
2. 建立根磁碟區的快照。根磁碟區連接至此執行個體做為 `/dev/sda1`。

分離根磁碟區與無法連線的執行個體、拍攝磁碟區快照或從磁碟區建立 AMI，然後將它連接到同一可用區域的另一個執行個體做為輔助磁碟區。

#### Warning

如果您的暫時執行個體與原始執行個體使用相同的 AMI 啟動，則您必須完成額外的步驟，否則您無法在還原原始執行個體根磁碟區之後將其啟動，原因是磁碟簽名衝突。如果您必須是使用同一 AMI 建立暫時執行個體，為避免發生磁碟簽章衝突，請完成 [磁碟簽章衝突](#) 中的步驟。

或者，請為暫時執行個體選擇不同的 AMI。例如，如果原始執行個體使用 AMI 適用於視窗伺服器 2016 年，請使用 AMI 啟動暫存執行個體 (適用於視窗伺服器 2019 年)。

3. 登入執行個體，從命令提示執行下列命令以將 `bootstatuspolicy` 組態變更為 `ignoreallfailures`。

```
bcdedit /store Drive Letter:\boot\bcd /set {default} bootstatuspolicy
ignoreallfailures
```

4. 將磁碟區重新連接至無法連線的執行個體，然後再次啟動執行個體。

## Windows 開機管理程式畫面

主控台螢幕截取畫面服務傳回下列內容。

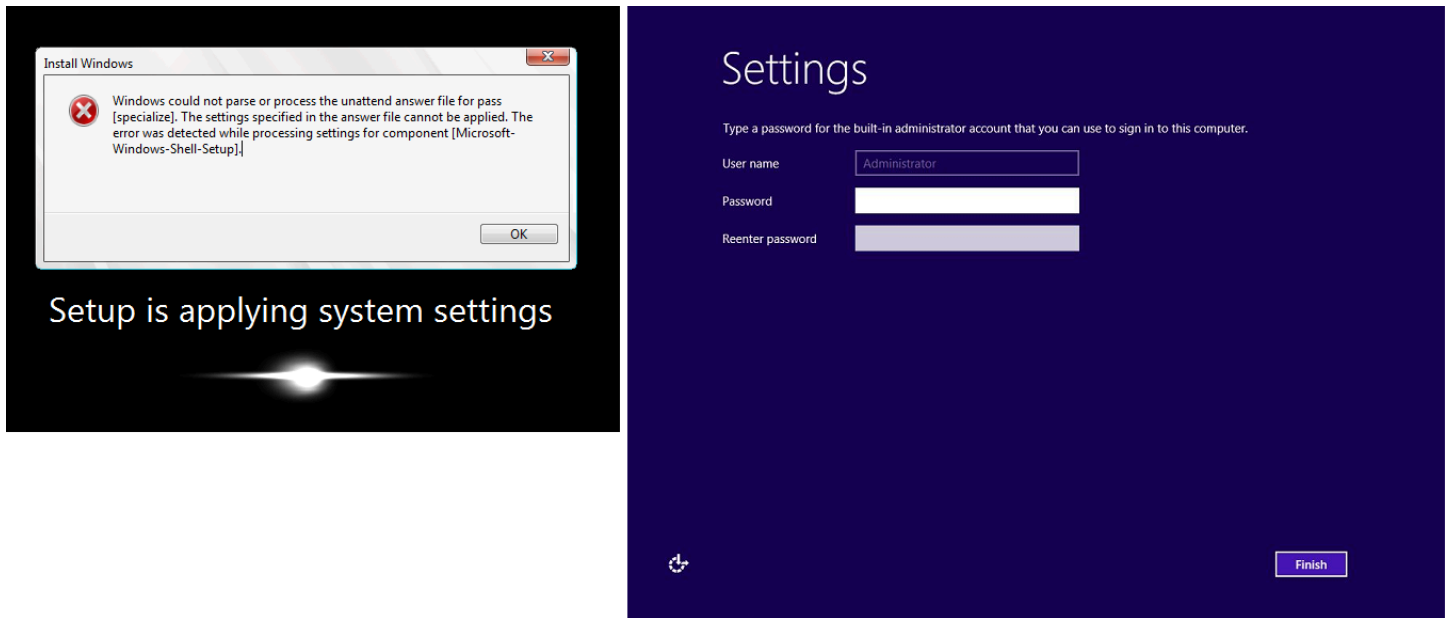
```
Windows Boot Manager
Windows failed to start. A recent hardware or software change might be the
cause. To fix the problem:
1. Insert your Windows installation disc and restart your computer.
2. Choose your language settings, and then click "Next."
3. Click "Repair your computer."
If you do not have this disc, contact your system administrator or computer
manufacturer for assistance.
File: \Boot\BCD
Status: 0xc000000f
Info: The Boot Configuration Data for your PC is missing or contains
errors.
```

```
windows Boot Manager
Windows failed to start. A recent hardware or software change might be the
cause. To fix the problem:
1. Insert your windows installation disc and restart your computer.
2. Choose your language settings, and then click "Next."
3. Click "Repair your computer."
If you do not have this disc, contact your system administrator or computer
manufacturer for assistance.
File: \Windows\system32\drivers\intelide.sys
Status: 0xc000000f
Info: Windows failed to load because a critical system driver is
missing, or corrupt.
ENTER=Continue ESC=Exit
```

作業系統的系統檔案及/或登錄發生重大損毀。當執行個體卡在這個狀態時，您應該從最近的備份 AMI 復原執行個體或啟動替代執行個體。如果您需要存取此執行個體的資料，請分離所有根磁碟區與無法連線的執行個體、拍攝這些磁碟區的快照或從這些磁碟區建立 AMI，然後將它們連接到同一可用區域的另一個執行個體做為輔助磁碟區。

## Sysprep 畫面

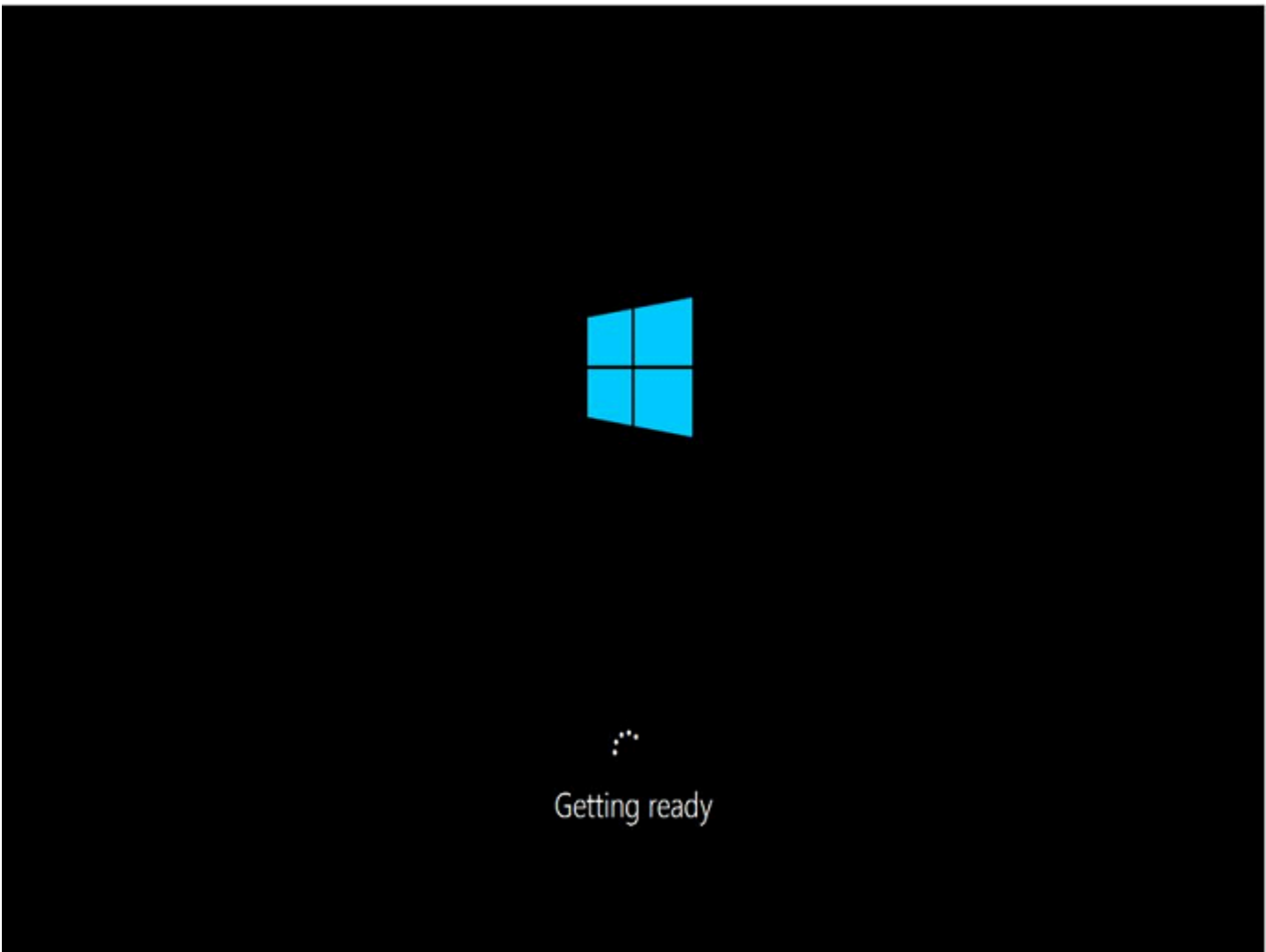
主控台螢幕截取畫面服務傳回下列內容。



如果您未使用 EC2Config 服務呼叫 sysprep 或如果作業系統在執行 Sysprep 時失敗，您可能會看到這個畫面。您可以使用 [EC2Rescue](#) 來重設密碼。否則，請參閱 [使用視窗系統建立 AMI](#)。

## 就緒畫面

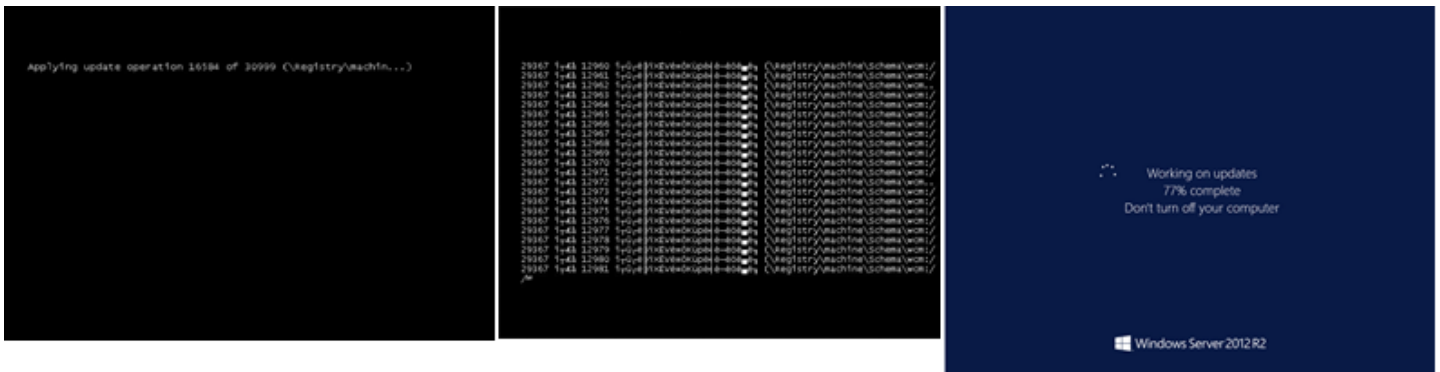
主控台螢幕截取畫面服務傳回下列內容。



重複更新執行個體主控台螢幕擷取畫面服務的畫面，以確認進度環是否旋轉。如果進度環不停旋轉，請等候作業系統啟動。您也可以使用 Amazon CloudWatch 查看作業系統是否處於作業中狀態，檢查執行個體上的 CPU 使用率 (最大值) 指標。如果進度環不旋轉，則執行個體可能卡在開機程序。重新啟動執行個體。如果重新開機不能解決此問題，請從最近的備份 AMI 復原執行個體或啟動替代執行個體。如果您需要存取此執行個體上的資料，請分離根磁碟區與無法連線的執行個體、拍攝磁碟區快照或從磁碟區建立 AMI。然後將它連接到同一可用區域的另一個執行個體做為輔助磁碟區。

## Windows Update 畫面

主控台螢幕截取畫面服務傳回下列內容。



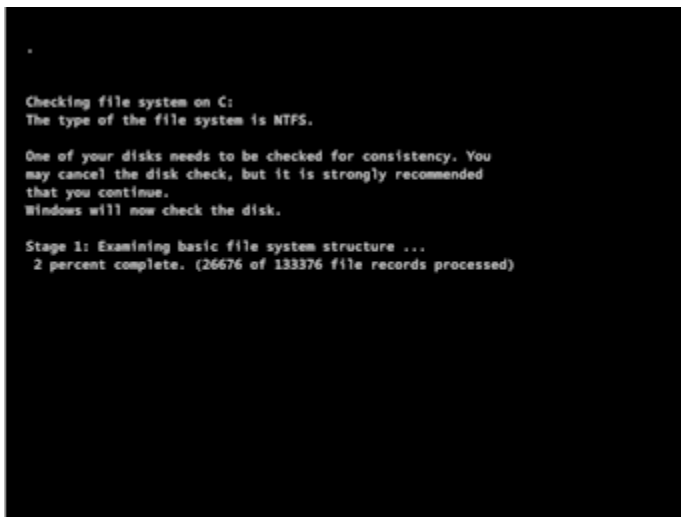
Windows Update 程序正在更新登錄。等候更新完成。不要重新開機或停止執行個體，因為這會在更新期間造成資料損毀。

### Note

Windows Update 程序在更新期間會使用伺服器的資源。如果您經常碰到此問題，請考慮使用更快的執行個體類型和更快的 EBS 磁碟區。

## Chkdsk

主控台螢幕截取畫面服務傳回下列內容。



Windows 對磁碟機使用 chkdsk 系統工具，以確認檔案系統完整性並修復邏輯檔案系統錯誤。請等候程序完成。

## 當主機電腦故障時的執行個體復原

如果底層主機電腦的硬體具有無法修復的問題，AWS 可能會排程執行執行個體停止事件。您會事先收到此類事件的電子郵件通知。

在主機電腦故障時復原其上執行的 Amazon EBS 後端執行個體

1. 將執行個體存放磁碟區上的所有重要資料備份到 Amazon EBS 或 Amazon S3。
2. 停止執行個體。
3. 啟動實例。
4. 復原所有重要的資料。

如需詳細資訊，請參閱 [停止並啟動 Amazon EC2 執行個體](#)。

在主機電腦故障時復原其上執行的執行個體存放區後端執行個體

1. 從執行個體建立 AMI。
2. 將映像上傳到 Amazon S3。
3. 將重要的資料備份到 Amazon EBS 或 Amazon S3。
4. 終止執行個體。
5. 從 AMI 啟動新的執行個體。
6. 將所有重要的資料還原到新的執行個體上。

## 停用執行個體時的疑難排解

如果停止 Amazon EBS 後端執行個體，而該執行個體似乎凍結於 `stopping` 狀態，這可能是底層主機電腦的問題。

當執行個體處於 `stopping` 狀態或 `running` 以外的任何狀態時，將不會收取執行個體使用費。當執行個體處於 `running` 狀態時，您只需支付執行個體使用費。

## 強制停止執行個體

請使用主控台或 AWS CLI 來強制停止執行個體。

**Note**

只有在執行個體為 `stopping` 狀態時，您才可以使用主控台強制停止執行個體。不論執行個體狀態為何 (`shutting-down` 和 `terminated` 除外)，您都可使用 AWS CLI 強制停用執行個體。

## Console

### 使用主控台強制停止執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)，然後選取凍結的執行個體。
3. 選擇「Instance state (執行個體狀態)」，「Force stop instance (強制停止執行個體)」，「Stop (停止)」。

請注意，唯有在您的執行個體處於 `stopping` 狀態的情況下，才能在主控台使用 Force stop instance (強制停止執行個體)。如果您的執行個體處於另一個狀態 (除了 `shutting-down` 和 `terminated`)，您可以使用 AWS CLI 來強制停止執行個體。

## AWS CLI

### 若要使用強制停止執行個體 AWS CLI

如下所示使用 [stop-instances](#) 命令和 `--force` 選項：

```
aws ec2 stop-instances --instance-ids i-0123ab456c789d01e --force
```

如果經過 10 分鐘後，執行個體仍未停止，請在 [AWS re:Post](#) 上發布請求協助的訊息。請在訊息中註明執行個體 ID，並說明您已採取的步驟，以利加速解決問題。或者，如果您已購買支援方案，可到 [Support 中心](#) 建立技術支援案例。

## 建立替代執行個體

在您等待來自 [AWS re:Post](#) 或 [Support 中心](#) 協助的同時，請建立替換執行個體，來試著解決問題。建立已凍結執行個體的 AMI，然後使用新的 AMI 來啟動新的執行個體。

### ⚠ Important

如果只註冊[系統狀態檢查](#)，則建議建立替代執行個體，因為執行個體狀態檢查將導致 AMI 複製到故障作業系統的確切複本上。確認狀態訊息後，建立 AMI 並使用新的 AMI 來啟動新執行個體。

## Console

使用主控台來建立替換執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)，然後選取凍結的執行個體。
3. 選擇 Actions (動作)、Image and templates (映像和範本)、Create image (建立映像)。
4. 在「Create image (建立映像)」頁面，執行下列動作：
  - a. 輸入 AMI 的名稱與說明。
  - b. 選擇 No reboot (不重新啟動)。
  - c. 選擇建立映像。

如需詳細資訊，請參閱 [the section called “從執行個體建立 AMI”](#)。

5. 從 AMI 啟動新的執行個體，並確認新的執行個體可正常運作。
6. 選取凍結的執行個體，然後依序選擇「Actions (動作)」、「Instance state (執行個體狀態)」、「Terminate instance (終止執行個體)」。如果該執行個體在終止時也出現凍結的狀況，Amazon EC2 會自動在幾小時內強制其終止。

## AWS CLI

使用 CLI 來建立替換執行個體

1. 利用 [create-image](#) (AWS CLI) 命令和 `--no-reboot` 選項，從凍結的執行個體建立 AMI，如下所示：

```
aws ec2 create-image --instance-id i-0123ab456c789d01e --name "AMI" --
description "AMI for replacement instance" --no-reboot
```

2. 利用 [run-instances](#) (AWS CLI) 命令，從 AMI 啟動新的執行個體，如下所示：



```
aws ec2 run-instances --image-id ami-1a2b3c4d --count 1 --instance-type c3.large
--key-name MyKeyPair --security-groups MySecurityGroup
```

3. 確認新的執行個體可正常運作。
4. 利用 [terminate-instances](#) (AWS CLI) 命令來終止凍結的執行個體，如下所示：

```
aws ec2 terminate-instances --instance-ids i-1234567890abcdef0
```

如果無法如先前步驟中的說明，從執行個體建立 AMI，您可以依照下列方式來建立替換執行個體：

(替代選項) 使用主控台來建立替換執行個體

1. 選取執行個體，然後依序選擇 Description (說明)、Block devices (區塊型儲存設備)。選取每個磁碟區，並記下其磁碟區 ID。請務必註明哪一個磁碟區是根磁碟區。
2. 在導覽窗格中，選擇 Volumes (磁碟區)。針對執行個體選取每個磁碟區，然後依序選擇 Actions (動作)、Create Snapshot (建立快照)。
3. 在導覽窗格中，選擇 Snapshots (快照)。選擇您剛建立的快照，然後依序選擇 Actions (動作)、Create Volume (建立磁碟區)。
4. 使用與凍結執行個體相同的作業系統來啟動執行個體。請寫下其根磁碟區的磁碟區 ID 和裝置名稱。
5. 在導覽窗格中，選擇「Instances (執行個體)」，選取剛才啟動的執行個體，然後依序選擇「Instance state (執行個體狀態)」、「Stop instance (停止執行個體)」。
6. 在導覽窗格中選擇 Volumes (磁碟區)，選取已停止執行個體的根磁碟區，然後依序選擇 Actions (動作)、Detach Volume (分離磁碟區)。
7. 選取從凍結執行個體建立的根磁碟區，然後依序選擇「Actions (動作)」、「Attach Volume (連結磁碟區)」，接著將該磁碟區連結到新的執行個體，並做為其根磁碟區 (使用您之前記下的裝置名稱)。將其他所有非根磁碟區的磁碟區，連結到執行個體。
8. 在導覽窗格中，選擇 Instances (執行個體)，然後選取替換個體。選擇 Instance state (執行個體狀態)、Start instance (啟動執行個體)。確認執行個體可正常運作。
9. 選取凍結的執行個體，然後依序選擇 Instance state (執行個體狀態)、Terminate instance (終止執行個體)。如果該執行個體在終止時也出現凍結的狀況，Amazon EC2 會自動在幾小時內強制其終止。

## 執行個體終止的故障診斷 (關機)

當執行個體未處於 `running` 狀態時，將不會向您收取任何執行個體使用費。換句話說，當您終止執行個體時，只要其狀態變更為 `shutting-down`，即刻起就會停止收取該執行個體所產生的費用。

### 立刻終止執行個體

若干問題可能會導致執行個體在啟動時立即終止。如需詳細資訊，請參閱「[立刻終止執行個體](#)」。

### 延遲的執行個體終止

如果執行個體處於 `shutting-down` 狀態超過幾分鐘，可能是該執行個體正在執行的關閉指令碼造成了延遲。

另一個可能的原因是底層主機電腦的問題。如果執行個體停留在 `shutting-down` 狀態好幾個小時，Amazon EC2 會將其視為凍結的執行個體，並強制加以終止。

如果執行個體似乎在終止時凍結，而且已經持續超過好幾個小時，請在 [AWS re:Post](#) 發佈求助訊息。請在訊息中註明執行個體 ID，並說明您已採取的步驟，以利加速解決問題。或者，如果您已購買支援方案，可到 [Support 中心](#) 建立技術支援案例。

### 仍顯示已終止的執行個體

在執行個體終止後，仍會短暫顯示，然後才會遭到刪除。狀態會顯示為 `terminated`。如果經過好幾個小時仍未刪除此項目，請聯絡 Support 中心。

### 錯誤：執行個體可能無法終止。修改其「禁用ApiTermination」實例屬性

如果您嘗試終止執行個體並收到 `The instance instance_id may not be terminated. Modify its 'disableApiTermination' instance attribute` 錯誤訊息，則表示執行個體已啟用終止保護。終止保護可防止執行個體被意外終止。如需詳細資訊，請參閱 [啟用終止保護](#)。

您必須先停用終止保護，才能終止執行個體。

若要使用 Amazon EC2 主控台停用終止保護，請選取執行個體，然後選擇動作、執行個體設定、變更終止保護。

若要使用停用終止保護 AWS CLI，請使用下列命令。

```
aws ec2 modify-instance-attribute --instance-id instance_id --no-disable-api-termination
```

## 執行個體自動啟動或終止

以下行為通常表示您使用了 Amazon EC2 Auto Scaling、EC2 Fleet 或 Spot Fleet，以根據您所定義的條件來自動擴展運算資源。

- 您終止某個執行個體，且新執行個體自動啟動。
- 您啟動某個執行個體，且其中一個執行個體自動終止。
- 您停止某個執行個體，該執行個體終止，且新執行個體自動啟動。

若要停止自動擴展，請參閱 [Amazon EC2 Auto Scaling 使用者指南](#)、[EC2 Fleet](#) 或 [建立 Spot Fleet 請求](#)。

## 使用失敗狀態檢查進行 Linux 執行個體

### Note

本主題僅適用於 Linux 執行個體。

下列資訊可協助您排解 Linux 執行個體無法進行狀態檢查時的問題。首先判斷應用程式是否出現任何問題。如果確認執行個體並未如預期執行應用程式，請檢閱狀態檢查資訊和系統日誌。

關於導致狀態檢查失敗的問題範例，請參閱 [您的執行個體的狀態檢查](#)。

### 內容

- [檢閱狀態檢查資訊](#)
- [擷取系統日誌](#)
- [解決 Linux 執行個體的系統記錄錯誤](#)
- [記憶體不足：終止程序](#)
- [錯誤：mmu\\_update 失敗 \(記憶體管理更新作業失敗\)](#)
- [I/O 錯誤 \(區塊型儲存設備故障\)](#)
- [I/O 錯誤：非本機也非遠端磁碟 \(故障的分散式區塊型儲存設備\)](#)
- [request\\_module：失控迴圈 modprobe \(在舊版 Linux 上建立舊核心 modprobe 的迴圈\)](#)
- [「嚴重：核心版本太舊」和「fsck：嘗試開啟 /dev 時，找不到此等檔案或目錄」\(核心與 AMI 不符\)](#)

- [「致命：無法載入 /lib/ 模組」或「BusyBox」\(遺失核心模組\)](#)
- [錯誤：無效的核心 \(與 EC2 不相容的核心\)](#)
- [fsck：嘗試開啟時找不到此等檔案或目錄... \(找不到檔案系統\)](#)
- [掛載檔案作業系統的一般錯誤 \(掛載失敗\)](#)
- [VFS：無法在未知的區塊上掛載根 fs \(根檔案系統不符\)](#)
- [錯誤：無法判定根設備的主/次編號... \(根檔案系統/裝置不相符\)](#)
- [XENBUS：裝置缺少驅動程式...](#)
- [...未進行檢查的天數，強制進行檢查 \(需要進行檔案系統檢查\)](#)
- [fsck 凍結於結束狀態... \(缺少裝置\)](#)
- [GRUB 提示 \(grubdom>\)](#)
- [帶出界面 eth0：裝置 eth0 的 MAC 地址和預期的不同，略過。\(將 MAC 地址直接寫入程式\)](#)
- [無法載入 SELinux 政策。機器正處於強制執行模式。現在停止中。\(SELinux 組態設定不當\)](#)
- [XENBUS：連線到裝置的作業逾時 \(Xenbus 逾時\)](#)

## 檢閱狀態檢查資訊

使用 Amazon EC2 主控台來調查狀態不健全的執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取您的執行個體。
3. 在詳細資訊窗格中，請選擇狀態和警示，以檢視所有系統狀態檢查和執行個體狀態檢查的個別結果。

如果未通過系統狀態檢查，您可以嘗試下列其中一個選項：

- 建立執行個體復原警示。如需詳細資訊，請參閱 [建立警示以停止、終止、重新啟動或復原執行個體](#)。
- 如果您將執行個體類型變更為 N [AWS Nitro System 上建置的執行個體](#)，如果您從沒有必要 ENA 和 NVMe 驅動程式的執行個體移轉，狀態檢查會失敗。如需詳細資訊，請參閱 [變更執行個體類型的相容性](#)。
- 如果是使用 Amazon EBS 後端 AMI 的執行個體，請停止和重新啟動執行個體。
- 如果是使用執行個體存放區後端 AMI 的執行個體，請終止執行個體，並啟動替代用的執行個體。
- 請等候 Amazon EC2 解決問題。

- 將您的問題發佈至 [AWS re:Post](#)。
- 如果您的執行個體在 Auto Scaling 群組中，則 Amazon EC2 Auto Scaling 服務會自動啟動替換的執行個體。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南中的 [Auto Scaling 執行個體的運作狀態檢查](#)。
- 擷取系統日誌並尋找錯誤。

## 擷取系統日誌

如果執行個體狀態檢查未通過，您可以重新啟動執行個體和擷取系統日誌。日誌可能會顯示錯誤，有助於對發生的問題進行故障診斷。重新啟動作業會清除日誌中不必要的資訊。

### 重新啟動執行個體和擷取系統日誌

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 Instances (執行個體)，然後選取您的執行個體。
3. 選擇 Instance state (執行個體狀態)、Reboot instance (重新啟動執行個體)。執行個體重新啟動可能需要幾分鐘的時間。
4. 確認問題是否仍持續存在；在某些情況中，重新啟動也許可以解決問題。
5. 當執行個體處於 running 狀態時，選擇 Actions (動作)、Monitor and troubleshoot (監視和疑難排解)、Get system log (取得系統日誌檔)。
6. 檢閱畫面上所顯示的日誌，然後使用下列的已知系統日誌錯誤陳述清單，來進行問題的故障診斷。
7. 如果問題無法解決，您可以在 [AWS re:Post](#) 中發佈您的問題。

## 解決 Linux 執行個體的系統記錄錯誤

對於執行個體狀態檢查失敗的 Linux 執行個體 (例如執行個體可連接性檢查)，請確認您是否遵循上述步驟擷取系統記錄。下列清單包含了一些常見的系統日誌錯誤和建議的動作，您可以針對每項錯誤採取這些動作，來解決問題。

### Memory Errors (記憶體錯誤)

- [記憶體不足：終止程序](#)
- [錯誤：mmu\\_update 失敗 \(記憶體管理更新作業失敗\)](#)

### Device Errors (裝置錯誤)

- [I/O 錯誤 \(區塊型儲存設備故障\)](#)
- [I/O 錯誤：非本機也非遠端磁碟 \(故障的分散式區塊型儲存設備\)](#)

### Kernel Errors (核心錯誤)

- [request\\_module：失控迴圈 modprobe \(在舊版 Linux 上建立舊核心 modprobe 的迴圈\)](#)
- [「嚴重：核心版本太舊」和「fsck：嘗試開啟 /dev 時，找不到此等檔案或目錄」\(核心與 AMI 不符\)](#)
- [「致命：無法載入 /lib/ 模組」或「BusyBox」\(遺失核心模組\)](#)
- [錯誤：無效的核心 \(與 EC2 不相容的核心\)](#)

### File System Errors (檔案系統錯誤)

- [fsck：嘗試開啟時找不到此等檔案或目錄... \(找不到檔案系統\)](#)
- [掛載檔案作業系統的一般錯誤 \(掛載失敗\)](#)
- [VFS：無法在未知的區塊上掛載根 fs \(根檔案系統不符\)](#)
- [錯誤：無法判定根設備的主/次編號... \(根檔案系統/裝置不相符\)](#)
- [XENBUS：裝置缺少驅動程式...](#)
- [...未進行檢查的天數，強制進行檢查 \(需要進行檔案系統檢查\)](#)
- [fsck 凍結於結束狀態... \(缺少裝置\)](#)

### Operating System Errors (作業系統錯誤)

- [GRUB 提示 \(grubdom>\)](#)
- [帶出界面 eth0：裝置 eth0 的 MAC 地址和預期的不同，略過。\(將 MAC 地址直接寫入程式\)](#)
- [無法載入 SELinux 政策。機器正處於強制執行模式。現在停止中。\(SELinux 組態設定不當\)](#)
- [XENBUS：連線到裝置的作業逾時 \(Xenbus 逾時\)](#)

## 記憶體不足：終止程序

系統記錄項目會指出 out-of-memory 錯誤，類似如下所示。

```
[115879.769795] Out of memory: kill process 20273 (httpd) score 1285879
or a child
```

```
[115879.769795] Killed process 1917 (php-cgi) vsz:467184kB, anon-
rss:101196kB, file-rss:204kB
```

## 可能的原因

### 記憶體用盡

### 建議動作

如果是此種執行個體類型	執行此作業
Amazon EBS 後端	<p>請執行下列其中一項：</p> <ul style="list-style-type: none"> <li>請停止執行個體，並修改執行個體以使用不同的執行個體類型，然後再次啟動該執行個體。例如，較大容量或記憶體最佳化的執行個體類型。</li> <li>重新啟動執行個體，以讓執行個體回到健全狀態。除非您變更執行個體類型，否則問題可能會再次發生。</li> </ul>
執行個體存放區後端	<p>請執行下列其中一項：</p> <ul style="list-style-type: none"> <li>終止執行個體並啟動新的執行個體，指定不同的執行個體類型。例如，較大容量或記憶體最佳化的執行個體類型。</li> <li>重新啟動執行個體，以讓執行個體回到健全狀態。除非您變更執行個體類型，否則問題可能會再次發生。</li> </ul>

## 錯誤：mmu\_update 失敗 (記憶體管理更新作業失敗)

系統日誌記錄項目會顯示記憶體管理更新作業的失敗，類似於下列所示記錄：

```
...
Press `ESC' to enter the menu... 0 [H[J Booting 'Amazon Linux 2011.09
(2.6.35.14-95.38.amzn1.i686)'
```

```
root (hd0)

Filesystem type is ext2fs, using whole disk

kernel /boot/vmlinuz-2.6.35.14-95.38.amzn1.i686 root=LABEL=/ console=hvc0 LANG=
en_US.UTF-8 KEYTABLE=us

initrd /boot/initramfs-2.6.35.14-95.38.amzn1.i686.img

ERROR: mmu_update failed with rc=-22
```

## 可能的原因

Amazon Linux 的問題

## 建議動作

請在 [開發人員論壇](#) 上發表您的問題，或是聯絡 [AWS Support](#)。

## I/O 錯誤 (區塊型儲存設備故障)

系統日誌記錄項目會顯示輸入/輸出錯誤，類似於下列的範例：

```
[9943662.053217] end_request: I/O error, dev sde, sector 52428288
[9943664.191262] end_request: I/O error, dev sde, sector 52428168
[9943664.191285] Buffer I/O error on device md0, logical block 209713024
[9943664.191297] Buffer I/O error on device md0, logical block 209713025
[9943664.191304] Buffer I/O error on device md0, logical block 209713026
[9943664.191310] Buffer I/O error on device md0, logical block 209713027
[9943664.191317] Buffer I/O error on device md0, logical block 209713028
[9943664.191324] Buffer I/O error on device md0, logical block 209713029
[9943664.191332] Buffer I/O error on device md0, logical block 209713030
[9943664.191339] Buffer I/O error on device md0, logical block 209713031
[9943664.191581] end_request: I/O error, dev sde, sector 52428280
[9943664.191590] Buffer I/O error on device md0, logical block 209713136
[9943664.191597] Buffer I/O error on device md0, logical block 209713137
[9943664.191767] end_request: I/O error, dev sde, sector 52428288
[9943664.191970] end_request: I/O error, dev sde, sector 52428288
[9943664.192143] end_request: I/O error, dev sde, sector 52428288
[9943664.192949] end_request: I/O error, dev sde, sector 52428288
```



```
[9943664.193112] end_request: I/O error, dev sde, sector 52428288
[9943664.193266] end_request: I/O error, dev sde, sector 52428288
...
```

## 可能的原因

執行個體類型	可能的原因
Amazon EBS 後端	故障的 Amazon EBS 磁碟區
執行個體存放區後端	故障的實體磁碟機

## 建議動作

如果是此種執行個體類型	執行此作業
Amazon EBS 後端	<p>請執行下列步驟：</p> <ol style="list-style-type: none"> <li>1. 停止執行個體。</li> <li>2. 分離磁碟區。</li> <li>3. 試著將磁碟區復原。</li> </ol> <div data-bbox="867 1167 1510 1486" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px; margin: 10px 0;"> <p><b>Note</b></p> <p>經常性地建立 Amazon EBS 磁碟區快照，是建議的理想做法。這可以大幅降低因為故障而造成資料遺失的風險。</p> </div> <ol style="list-style-type: none"> <li>4. 將磁碟區重新連結到執行個體。</li> <li>5. 啟動實例。</li> </ol>
執行個體存放區後端	<p>終止執行個體並啟動新的執行個體。</p> <div data-bbox="829 1713 1508 1885" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px; margin: 10px 0;"> <p><b>Note</b></p> <p>資料無法復原。請從備份資料復原。</p> </div>

如果是此種執行個體類型	執行此作業
	<p><b>Note</b></p> <p>理想的做法是使用 Amazon S3 或 Amazon EBS 來進行備份。執行個體存放磁碟區會與單一主機或單一磁碟的故障直接連動。</p>

## I/O 錯誤：非本機也非遠端磁碟 (故障的分散式區塊型儲存設備)

系統日誌記錄項目會顯示裝置上的輸入/輸出錯誤，類似於下列的範例：

```

...
block drbd1: Local I/O failed in request_timer_fn. Detaching...

Aborting journal on device drbd1-8.

block drbd1: I/O ERROR: neither local nor remote disk

Buffer I/O error on device drbd1, logical block 557056

lost page write due to I/O error on drbd1

JBD2: I/O error detected when updating journal superblock for drbd1-8.
```

### 可能的原因

執行個體類型	可能的原因
Amazon EBS 後端	故障的 Amazon EBS 磁碟區
執行個體存放區後端	故障的實體磁碟機

### 建議動作

終止執行個體並啟動新的執行個體。

如果是 Amazon EBS 後端執行個體，您可以藉由從最近的快照建立映像，來回復資料。在快照建立後所新增的任何資料，皆無法復原。

## request\_module : 失控迴圈 modprobe (在舊版 Linux 上建立舊核心 modprobe 的迴圈)

系統日誌記錄項目會顯示此狀況，類似於下列所示。使用不穩定的或舊版的 Linux 核心 (例如 2.6.16-xenU)，可能會在啟動時造成無止盡的循環狀況。

```
Linux version 2.6.16-xenU (builder@xenbat.amazonsa) (gcc version 4.0.1
20050727 (Red Hat 4.0.1-5)) #1 SMP Mon May 28 03:41:49 SAST 2007
```

```
BIOS-provided physical RAM map:
```

```
Xen: 0000000000000000 - 0000000026700000 (usable)
```

```
0MB HIGHMEM available.
```

```
...
```

```
request_module: runaway loop modprobe binfmt-464c
```

```
request_module: runaway loop modprobe binfmt-464c
```

```
request_module: runaway loop modprobe binfmt-464c
```

```
request_module: runaway loop modprobe binfmt-464c
```

```
request_module: runaway loop modprobe binfmt-464c
```

### 建議動作

如果是此種執行個體類型	執行此作業
Amazon EBS 後端	<p>請使用下列其中一個選項，來使用較新版的核心 (採用 GRUB 或靜態的核心)：</p> <p>選項 1：終止執行個體並啟動新的執行個體，指定 <code>-kernel</code> 和 <code>-ramdisk</code> 參數。</p> <p>選項 2：</p>

如果是此種執行個體類型	執行此作業
	<ol style="list-style-type: none"> <li>1. 停止執行個體。</li> <li>2. 修改核心和記憶體虛擬磁碟 (ramdisk) 的屬性，以使用較新版的核心。</li> <li>3. 啟動實例。</li> </ol>
執行個體存放區後端	終止執行個體並啟動新的執行個體，指定 <code>-kernel</code> 和 <code>-ramdisk</code> 參數。

「嚴重：核心版本太舊」和「fsck：嘗試開啟 /dev 時，找不到此等檔案或目錄」(核心與 AMI 不符)

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
Linux version 2.6.16.33-xenU (root@dom0-0-50-45-1-a4-ee.z-2.aes0.internal)
(gcc version 4.1.1 20070105 (Red Hat 4.1.1-52)) #2 SMP Wed Aug 15 17:27:36 SAST 2007
...
FATAL: kernel too old
Kernel panic - not syncing: Attempted to kill init!
```

## 可能的原因

不相容的核心和使用者空間

## 建議動作

如果是此種執行個體類型	執行此作業
Amazon EBS 後端	<p>請執行下列步驟：</p> <ol style="list-style-type: none"> <li>1. 停止執行個體。</li> <li>2. 修改組態，以使用較新版的核心。</li> <li>3. 啟動實例。</li> </ol>
執行個體存放區後端	<p>請執行下列步驟：</p> <ol style="list-style-type: none"> <li>1. 建立 AMI，此 AMI 會使用較新版的核心。</li> </ol>

如果是此種執行個體類型	執行此作業
	<ol style="list-style-type: none"> <li>2. 終止執行個體。</li> <li>3. 從您所建立的 AMI 啟動新的執行個體。</li> </ol>

## 「致命：無法載入 /lib/ 模組」或「BusyBox」(遺失核心模組)

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
[0.370415] Freeing unused kernel memory: 1716k freed
Loading, please wait...
WARNING: Couldn't open directory /lib/modules/2.6.34-4-virtual: No such file or
directory
FATAL: Could not open /lib/modules/2.6.34-4-virtual/modules.dep.temp for writing: No
such file or directory
FATAL: Could not load /lib/modules/2.6.34-4-virtual/modules.dep: No such file or
directory
Couldn't get a file descriptor referring to the console
Begin: Loading essential drivers... ...
FATAL: Could not load /lib/modules/2.6.34-4-virtual/modules.dep: No such file or
directory
FATAL: Could not load /lib/modules/2.6.34-4-virtual/modules.dep: No such file or
directory
Done.
Begin: Running /scripts/init-premount ...
Done.
Begin: Mounting root file system... ...
Begin: Running /scripts/local-top ...
Done.
Begin: Waiting for root file system... ...
Done.
Gave up waiting for root device. Common problems:
- Boot args (cat /proc/cmdline)
- Check rootdelay= (did the system wait long enough?)
- Check root= (did the system wait for the right device?)
- Missing modules (cat /proc/modules; ls /dev)
FATAL: Could not load /lib/modules/2.6.34-4-virtual/modules.dep: No such file or
directory
FATAL: Could not load /lib/modules/2.6.34-4-virtual/modules.dep: No such file or
directory
ALERT! /dev/sda1 does not exist. Dropping to a shell!
```

```
BusyBox v1.13.3 (Ubuntu 1:1.13.3-1ubuntu5) built-in shell (ash)
Enter 'help' for a list of built-in commands.

(initramfs)
```

## 可能的原因

這項問題可能是下列一種或多種狀況所造成：

- 缺少記憶體虛擬磁碟 (ramdisk)
- 記憶體虛擬磁碟 (ramdisk) 缺少正確的模組
- Amazon EBS 根磁碟區未正確地連結為 `/dev/sda1`

## 建議動作

如果是此種執行個體類型	執行此作業
Amazon EBS 後端	<p>請執行下列步驟：</p> <ol style="list-style-type: none"><li>1. 為 Amazon EBS 磁碟區選擇修正過的記憶體虛擬磁碟 (ramdisk)。</li><li>2. 停止執行個體。</li><li>3. 先分離然後修復磁碟區。</li><li>4. 將磁碟區連結到執行個體。</li><li>5. 啟動實例。</li><li>6. 修改 AMI 以使用修正過的記憶體虛擬磁碟 (ramdisk)。</li></ol>
執行個體存放區後端	<p>請執行下列步驟：</p> <ol style="list-style-type: none"><li>1. 終止執行個體，並使用正確的記憶體虛擬磁碟 (ramdisk) 來啟動新的執行個體。</li><li>2. 使用正確的記憶體虛擬磁碟 (ramdisk) 來建立新的 AMI。</li></ol>

## 錯誤：無效的核心 (與 EC2 不相容的核心)

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
...
root (hd0)

 Filesystem type is ext2fs, using whole disk

kernel /vmlinuz root=/dev/sda1 ro

initrd /initrd.img

ERROR Invalid kernel: elf_xen_note_check: ERROR: Will only load images
built for the generic loader or Linux images
xc_dom_parse_image returned -1

Error 9: Unknown boot failure

 Booting 'Fallback'

root (hd0)

 Filesystem type is ext2fs, using whole disk

kernel /vmlinuz.old root=/dev/sda1 ro

Error 15: File not found
```

### 可能的原因

這項問題可能是下列狀況中的其中一種或兩種同時造成：

- GRUB 不支援隨附的核心
- 備用核心不存在

### 建議動作

如果是此種執行個體類型	執行此作業
Amazon EBS 後端	請執行下列步驟：

如果是此種執行個體類型	執行此作業
	<ol style="list-style-type: none"> <li>1. 停止執行個體。</li> <li>2. 換成可正常運作的核心。</li> <li>3. 安裝備用核心。</li> <li>4. 透過修正核心來修改 AMI。</li> </ol>
執行個體存放區後端	<p>請執行下列步驟：</p> <ol style="list-style-type: none"> <li>1. 終止執行個體，並使用正確的核心來啟動新的執行個體。</li> <li>2. 使用正確的核心來建立 AMI。</li> <li>3. (選用) 透過 <a href="#">AWS Support</a> 來尋求資料復原的技術協助。</li> </ol>

## fsck：嘗試開啟時找不到此等檔案或目錄... (找不到檔案系統)

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```

Welcome to Fedora
Press 'I' to enter interactive startup.
Setting clock : Wed Oct 26 05:52:05 EDT 2011 [OK]

Starting udev: [OK]

Setting hostname localhost: [OK]

No devices found
Setting up Logical Volume Management: File descriptor 7 left open
 No volume groups found
[OK]

Checking filesystems
Checking all file systems.
[/sbin/fsck.ext3 (1) -- /] fsck.ext3 -a /dev/sda1
/dev/sda1: clean, 82081/1310720 files, 2141116/2621440 blocks
[/sbin/fsck.ext3 (1) -- /mnt/dbbackups] fsck.ext3 -a /dev/sdh
fsck.ext3: No such file or directory while trying to open /dev/sdh

```



```
/dev/sdh:
```

```
The superblock could not be read or does not describe a correct ext2
filesystem. If the device is valid and it really contains an ext2
filesystem (and not swap or ufs or something else), then the superblock
is corrupt, and you might try running e2fsck with an alternate superblock:
```

```
e2fsck -b 8193 <device>
```

```
[FAILED]
```

```
*** An error occurred during the file system check.
*** Dropping you to a shell; the system will reboot
*** when you leave the shell.
Give root password for maintenance
(or type Control-D to continue):
```

## 可能的原因

- ramdisk 檔案系統定義 `/etc/fstab` 中存在錯誤
- `/etc/fstab` 中設定錯誤的檔案系統定義
- 找不到磁碟機/磁碟機故障

## 建議動作

如果是此種執行個體類型	執行此作業
Amazon EBS 後端	<p>請執行下列步驟：</p> <ol style="list-style-type: none"> <li>1. 停止執行個體、分離根磁碟區、修復/修改 <code>/etc/fstab</code> 磁碟區、將磁碟區連結到執行個體，以及啟動執行個體。</li> <li>2. 修正 ramdisk，以納入修改過的 <code>/etc/fstab</code> (如有適用)。</li> <li>3. 修改 AMI，以使用較新版的 ramdisk。</li> </ol> <p><code>fstab</code> 中的第 6 個欄位定義了掛載的可用性要求 – 非 0 值暗示 <code>fsck</code> 將在該磁碟區上完成，而且必須成功。在 Amazon EC2 中使用此欄位可</p>

如果是此種執行個體類型	執行此作業
	能會出現問題，因為故障通常會產生互動式主控台提示，而 Amazon EC2 中目前並未提供此提示功能。使用此功能時請特別注意，並閱讀 <code>fstab</code> 的 Linux 手冊頁。
執行個體存放區後端	請執行下列步驟： <ol style="list-style-type: none"><li>1. 終止執行個體並啟動新的執行個體。</li><li>2. 分離所有錯誤的 Amazon EBS 磁碟區，並重新啟動執行個體。</li><li>3. (選用) 透過 <a href="#">AWS Support</a> 來尋求資料復原的技術協助。</li></ol>

## 掛載檔案作業系統的一般錯誤 (掛載失敗)

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
Loading xenblk.ko module
xen-vbd: registered block device major 8

Loading ehci-hcd.ko module
Loading ohci-hcd.ko module
Loading uhci-hcd.ko module
USB Universal Host Controller Interface driver v3.0

Loading mbcache.ko module
Loading jbd.ko module
Loading ext3.ko module
Creating root device.
Mounting root filesystem.
kjournald starting. Commit interval 5 seconds

EXT3-fs: mounted filesystem with ordered data mode.

Setting up other filesystems.
Setting up new root fs
no fstab.sys, mounting internal defaults
Switching to new root and running init.
```

```

unmounting old /dev
unmounting old /proc
unmounting old /sys
mountall:/proc: unable to mount: Device or resource busy
mountall:/proc/self/mountinfo: No such file or directory
mountall: root filesystem isn't mounted
init: mountall main process (221) terminated with status 1

```

*General error mounting filesystems.*

```

A maintenance shell will now be started.
CONTROL-D will terminate this shell and re-try.
Press enter for maintenance
(or type Control-D to continue):

```

## 可能的原因

執行個體類型	可能的原因
Amazon EBS 後端	<ul style="list-style-type: none"> <li>• 已分離或故障的 Amazon EBS 磁碟區。</li> <li>• 已毀損的檔案系統。</li> <li>• 不匹配的 ramdisk 和 AMI 組合 (例如 Debian ramdisk 搭配 SUSE AMI)。</li> </ul>
執行個體存放區後端	<ul style="list-style-type: none"> <li>• 故障的磁碟機。</li> <li>• 已毀損的檔案系統。</li> <li>• 不匹配的 ramdisk 和組合 (例如 Debian ramdisk 搭配 SUSE AMI)。</li> </ul>

## 建議動作

如果是此種執行個體類型	執行此作業
Amazon EBS 後端	<p>請執行下列步驟：</p> <ol style="list-style-type: none"> <li>1. 停止執行個體。</li> <li>2. 分離根 磁碟區。</li> </ol>

如果是此種執行個體類型	執行此作業
	<ol style="list-style-type: none"> <li>3. 將根磁碟區連結到已知可正常運作的執行個體。</li> <li>4. 執行檔案系統檢查 (fsck -a /dev/...)。</li> <li>5. 修正所有錯誤。</li> <li>6. 將磁碟區從已知可正常運作的執行個體分離。</li> <li>7. 將磁碟區連結到已停止的執行個體。</li> <li>8. 啟動實例。</li> <li>9. 重新檢查執行個體的狀態。</li> </ol>
執行個體存放區後端	<p>請嘗試執行下列其中一項操作：</p> <ul style="list-style-type: none"> <li>• 啟動新的執行個體。</li> <li>• (選用) 透過 <a href="#">AWS Support</a> 來尋求資料復原的技術協助。</li> </ul>

## VFS：無法在未知的區塊上掛載根 fs (根檔案系統不符)

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
Linux version 2.6.16-xenU (builder@xenbat.amazonsa) (gcc version 4.0.1
20050727 (Red Hat 4.0.1-5)) #1 SMP Mon May 28 03:41:49 SAST 2007
...
Kernel command line: root=/dev/sda1 ro 4
...
Registering block device major 8
...
Kernel panic - not syncing: VFS: Unable to mount root fs on unknown-block(8,1)
```

### 可能的原因

執行個體類型	可能的原因
Amazon EBS 後端	<ul style="list-style-type: none"> <li>• 未正確地連結裝置。</li> <li>• 根設備未連結到正確的裝置點。</li> </ul>

執行個體類型	可能的原因
	<ul style="list-style-type: none"> <li>檔案系統未採用預期的格式。</li> <li>使用舊版的核心 (例如 2.6.16-XenU)。</li> <li>在執行個體上最近更新了核心 (更新作業失敗或出現錯誤)</li> </ul>
執行個體存放區後端	硬體裝置故障。

## 建議動作

如果是此種執行個體類型	執行此作業
Amazon EBS 後端	<p>請執行下列其中一項：</p> <ul style="list-style-type: none"> <li>停止並重新啟動執行個體。</li> <li>修改根磁碟區，以掛載於正確的裝置點 (可能是 /dev/sda1 而非 /dev/sda)。</li> <li>停止並進行修改，以使用現代化的核心。</li> <li>請參考您 Linux 版本的文件，以查看已知的更新錯誤。變更或重新安裝核心。</li> </ul>
執行個體存放區後端	終止執行個體，並使用現代化的核心來啟動新的執行個體。

## 錯誤：無法判定根設備的主/次編號... (根檔案系統/裝置不相符)

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
...
XENBUS: Device with no driver: device/vif/0
XENBUS: Device with no driver: device/vbd/2048
drivers/rtc/hctosys.c: unable to open rtc device (rtc0)
Initializing network drop monitor service
Freeing unused kernel memory: 508k freed
:: Starting udevd...
done.
```

```

:: Running Hook [udev]
:: Triggering uevents...<30>udev[65]: starting version 173
done.
Waiting 10 seconds for device /dev/xvda1 ...
Root device '/dev/xvda1' doesn't exist. Attempting to create it.
ERROR: Unable to determine major/minor number of root device '/dev/xvda1'.
You are being dropped to a recovery shell
 Type 'exit' to try and continue booting
sh: can't access tty; job control turned off
[ramfs /]#

```

## 可能的原因

- 找不到虛擬區塊型儲存設備驅動程式，或是該驅動程式設定不正確
- 裝置列舉衝突 (sda 對 xvda 或 sda 而非 sda1)
- 所選的執行個體核心不正確

## 建議動作

如果是此種執行個體類型	執行此作業
Amazon EBS 後端	請執行下列步驟： <ol style="list-style-type: none"> <li>1. 停止執行個體。</li> <li>2. 分離磁碟區。</li> <li>3. 修正裝置映射問題。</li> <li>4. 啟動實例。</li> <li>5. 修正 AMI 以解決裝置映射問題。</li> </ol>
執行個體存放區後端	請執行下列步驟： <ol style="list-style-type: none"> <li>1. 透過適當的修正來建立新的 AMI (正確地對應區塊型儲存設備)。</li> <li>2. 終止執行個體，並從您所建立的 AMI 來啟動新的執行個體。</li> </ol>

## XENBUS : 裝置缺少驅動程式...

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
XENBUS: Device with no driver: device/vbd/2048
drivers/rtc/hctosys.c: unable to open rtc device (rtc0)
Initializing network drop monitor service
Freeing unused kernel memory: 508k freed
:: Starting udevd...
done.
:: Running Hook [udev]
:: Triggering uevents...<30>udev[65]: starting version 173
done.
Waiting 10 seconds for device /dev/xvda1 ...
Root device '/dev/xvda1' doesn't exist. Attempting to create it.
ERROR: Unable to determine major/minor number of root device '/dev/xvda1'.
You are being dropped to a recovery shell
 Type 'exit' to try and continue booting
sh: can't access tty; job control turned off
[ramfs /]#
```

### 可能的原因

- 找不到虛擬區塊型儲存設備驅動程式，或是該驅動程式設定不正確
- 裝置列舉衝突 (sda 對 xvda)
- 所選的執行個體核心不正確

### 建議動作

如果是此種執行個體類型	執行此作業
Amazon EBS 後端	請執行下列步驟： <ol style="list-style-type: none"><li>1. 停止執行個體。</li><li>2. 分離磁碟區。</li><li>3. 修正裝置映射問題。</li><li>4. 啟動實例。</li><li>5. 修正 AMI 以解決裝置映射問題。</li></ol>

如果是此種執行個體類型	執行此作業
執行個體存放區後端	請執行下列步驟： <ol style="list-style-type: none"><li>1. 透過適當的修正來建立 AMI (正確地對應區塊型儲存設備)。</li><li>2. 終止執行個體，並使用您所建立的 AMI 來啟動新的執行個體。</li></ol>

## ...未進行檢查的天數，強制進行檢查 (需要進行檔案系統檢查)

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
...
Checking filesystems
Checking all file systems.
[/sbin/fsck.ext3 (1) -- /] fsck.ext3 -a /dev/sda1
/dev/sda1 has gone 361 days without being checked, check forced
```

### 可能的原因

已超過檔案系統檢查時間；強制進行檔案系統檢查。

### 建議動作

- 請等候檔案系統檢查完成。視根檔案系統的大小而定，檔案系統檢查可能需要一段長時間才會完成。
- 使用 tune2fs 或檔案系統適用的工具，來修改檔案系統，以移除強制進行檔案系統檢查 (fsck) 的功能。

## fsck 凍結於結束狀態... (缺少裝置)

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
Cleaning up ifupdown....
Loading kernel modules...done.
...
Activating lvm and md swap...done.
```



```
Checking file systems...fsck from util-linux-ng 2.16.2
/sbin/fsck.xfs: /dev/sdh does not exist
fsck died with exit status 8
[31mfailed (code 8).[39;49m
```

## 可能的原因

- Ramdisk 尋找缺少的磁碟機
- 強制進行檔案系統一致性檢查
- 磁碟機故障或已分離

## 建議動作

如果是此種執行個體類型	執行此作業
Amazon EBS 後端	<p>請嘗試執行下列其中一項或多項動作來解決問題：</p> <ul style="list-style-type: none"> <li>• 停止執行個體、將磁碟區連結到執行中的現有執行個體。</li> <li>• 手動執行一致性檢查。</li> <li>• 修正 ramdisk，以加入相關的公用程式。</li> <li>• 修改檔案系統調校參數，以移除一致性要求 (不建議)。</li> </ul>
執行個體存放區後端	<p>請嘗試執行下列其中一項或多項動作來解決問題：</p> <ul style="list-style-type: none"> <li>• 將 ramdisk 與正確的工具重新綁定為套件。</li> <li>• 修改檔案系統調校參數，以移除一致性要求 (不建議)。</li> <li>• 終止執行個體並啟動新的執行個體。</li> <li>• (選用) 透過 <a href="#">AWS Support</a> 來尋求資料復原的技術協助。</li> </ul>

## GRUB 提示 (grubdom>)

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
GNU GRUB version 0.97 (629760K lower / 0K upper memory)
```

```
[Minimal BASH-like line editing is supported. For
the first word, TAB lists possible command
completions. Anywhere else TAB lists the possible
completions of a device/filename.]
```


```
grubdom>
```

### 可能的原因

執行個體類型	可能的原因
Amazon EBS 後端	<ul style="list-style-type: none"><li>找不到 GRUB 組態檔案。</li><li>使用了不正確的 GRUB 映像，預期 GRUB 組態檔案儲存於不同的位置。</li><li>使用了不支援的檔案系統來存放 GRUB 組態檔案 (例如，將根檔案系統轉換為 GRUB 先前版本不支援的類型)。</li></ul>
執行個體存放區後端	<ul style="list-style-type: none"><li>找不到 GRUB 組態檔案。</li><li>使用了不正確的 GRUB 映像，預期 GRUB 組態檔案儲存於不同的位置。</li><li>使用了不支援的檔案系統來存放 GRUB 組態檔案 (例如，將根檔案系統轉換為 GRUB 先前版本不支援的類型)。</li></ul>

## 建議動作

如果是此種執行個體類型	執行此作業
Amazon EBS 後端	<p data-bbox="829 338 1398 373">選項 1：修改 AMI 並重新啟動執行個體：</p> <ol data-bbox="829 422 1490 814" style="list-style-type: none"><li>1. 修改來源 AMI，以在標準位置建立 GRUB 組態檔案 (/boot/grub/menu.lst)。</li><li>2. 確認您的 GRUB 版本支援底層檔案系統的類型，必要時請更新 GRUB。</li><li>3. 選擇適合的 GRUB 映像 (hd0-第 1 個磁碟機或 hd00 – 第 1 個磁碟機，第 1 個分割區)。</li><li>4. 終止執行個體，並使用您所建立的 AMI 來啟動新的執行個體。</li></ol> <p data-bbox="829 890 1260 926">選項 2：修改現有的執行個體：</p> <ol data-bbox="829 974 1507 1713" style="list-style-type: none"><li>1. 停止執行個體。</li><li>2. 分離根檔案系統。</li><li>3. 將根檔案系統連結到已知可正常運作的執行個體。</li><li>4. 掛載檔案系統。</li><li>5. 建立 GRUB 組態檔案。</li><li>6. 確認您的 GRUB 版本支援底層檔案系統的類型，必要時請更新 GRUB。</li><li>7. 分離檔案系統。</li><li>8. 連結到原始執行個體。</li><li>9. 修改核心屬性，以使用適合的 GRUB 映像 (第 1 個磁碟機或第 1 個磁碟機上的第 1 個分割區)。</li><li>10. 啟動實例。</li></ol>
執行個體存放區後端	選項 1：修改 AMI 並重新啟動執行個體：

如果是此種執行個體類型	執行此作業
	<ol style="list-style-type: none"> <li>1. 使用儲存於標準位置的 GRUB 組態檔案 (/boot/grub/menu.lst)，來建立新的 AMI。</li> <li>2. 選擇適合的 GRUB 映像 (hd0-第 1 個磁碟機或 hd00 – 第 1 個磁碟機，第 1 個分割區)。</li> <li>3. 確認您的 GRUB 版本支援底層檔案系統的類型，必要時請更新 GRUB。</li> <li>4. 終止執行個體，並使用您所建立的 AMI 來啟動新的執行個體。</li> </ol> <p>選項 2：終止執行個體，並啟動新的執行個體 (指定正確的核心)。</p> <div style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px; margin-top: 10px;"> <p> <b>Note</b></p> <p>若要從現有的執行個體復原資料，請聯絡 <a href="#">AWS Support</a>。</p> </div>

帶出界面 eth0：裝置 eth0 的 MAC 地址和預期的不同，略過。(將 MAC 地址直接寫入程式)

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
...
Bringing up loopback interface: [OK]

Bringing up interface eth0: Device eth0 has different MAC address than expected,
ignoring.
[FAILED]

Starting auditd: [OK]
```

## 可能的原因

在 AMI 組態中有直接寫入程式的界面 MAC

## 建議動作

如果是此種執行個體類型	執行此作業
Amazon EBS 後端	<p>請執行下列其中一項：</p> <ul style="list-style-type: none"> <li>• 修改 AMI，以移除直接寫入程式的項目，並重新啟動執行個體。</li> <li>• 修改執行個體，以移除直接寫入程式的 MAC 位址。</li> </ul> <p>或是</p> <p>請執行下列步驟：</p> <ol style="list-style-type: none"> <li>1. 停止執行個體。</li> <li>2. 分離根 磁碟區。</li> <li>3. 將磁碟區連結到另一個執行個體，並修改該磁碟區，以移除直接寫入程式的 MAC 位址。</li> <li>4. 將磁碟區連結到原始執行個體。</li> <li>5. 啟動實例。</li> </ol>
執行個體存放區後端	<p>請執行下列其中一項：</p> <ul style="list-style-type: none"> <li>• 修改執行個體，以移除直接寫入程式的 MAC 位址。</li> <li>• 終止執行個體並啟動新的執行個體。</li> </ul>

無法載入 SELinux 政策。機器正處於強制執行模式。現在停止中。(SELinux 組態設定不當)

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
audit(1313445102.626:2): enforcing=1 old_enforcing=0 auid=4294967295
Unable to load SELinux Policy. Machine is in enforcing mode. Halting now.
Kernel panic - not syncing: Attempted to kill init!
```

## 可能的原因

SELinux 已在啟用時發生錯誤：

- GRUB 不支援隨附的核心
- 備用核心不存在

## 建議動作

如果是此種執行個體類型	執行此作業
Amazon EBS 後端	<p>請執行下列步驟：</p> <ol style="list-style-type: none"><li>1. 停止已故障的執行個體。</li><li>2. 分離已故障執行個體的根磁碟區。</li><li>3. 將根磁碟區連結到另一個執行中的 Linux 執行個體 (之後稱為復原執行個體)。</li><li>4. 連線到復原執行個體，並掛載已故障執行個體的根磁碟區。</li><li>5. 將已掛載根磁碟區上的 SELinux 停用。不同 Linux 版本的此項程序也會有所差異，如需詳細資訊，請參閱特定作業系統的文件。</li></ol> <div data-bbox="867 1247 1508 1709" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px;"><p> <b>Note</b></p><p>在某些系統上，您可以透過設定 SELINUX=disabled 檔案中的 <code>/mount_point /etc/sysconfig/selinux</code>，來停用 SELinux，其中 <code>mount_point</code> 是您在復原執行個體上掛載磁碟區的位置。</p></div> <ol style="list-style-type: none"><li>6. 取消掛載根磁碟區、將該磁碟區從復原執行個體分離，然後再重新連結到原始執行個體。</li><li>7. 啟動實例。</li></ol>

如果是此種執行個體類型	執行此作業
執行個體存放區後端	<p>請執行下列步驟：</p> <ol style="list-style-type: none"> <li>1. 終止執行個體並啟動新的執行個體。</li> <li>2. (選用) 透過 <a href="#">AWS Support</a> 來尋求資料復原的技術協助。</li> </ol>

## XENBUS：連線到裝置的作業逾時 (Xenbus 逾時)

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
Linux version 2.6.16-xenU (builder@xenbat.amazonsa) (gcc version 4.0.1
20050727 (Red Hat 4.0.1-5)) #1 SMP Mon May 28 03:41:49 SAST 2007
...
XENBUS: Timeout connecting to devices!
...
Kernel panic - not syncing: No init found. Try passing init= option to kernel.
```

### 可能的原因

- 區塊型儲存設備未連線到執行個體
- 執行個體使用舊版的執行個體核心

### 建議動作

如果是此種執行個體類型	執行此作業
Amazon EBS 後端	<p>請執行下列其中一項：</p> <ul style="list-style-type: none"> <li>• 修改 AMI 和執行個體，以使用現代化的核心並重新啟動執行個體。</li> <li>• 重新啟動執行個體。</li> </ul>
執行個體存放區後端	<p>請執行下列其中一項：</p> <ul style="list-style-type: none"> <li>• 終止執行個體。</li> </ul>

如果是此種執行個體類型	執行此作業
	<ul style="list-style-type: none"> <li>修改 AMI，以使用現代化的核心，並使用此 AMI 來啟動新的執行個體。</li> </ul>

## 對 Linux 執行個體從錯誤的磁碟區開機

### Note

此疑難排解主題僅適用於 Linux 執行個體。

在某些情況中，成為您執行個體根磁碟區的磁碟區，可能並非連結到 `/dev/xvda` 或 `/dev/sda` 的磁碟區。這可能是因為您已經將另一個執行個體的根磁碟區，或是從根磁碟區快照所建立的磁碟區，連結到包含現有根磁碟區的執行個體。

這是 Linux 中初始記憶體虛擬磁碟 (ramdisk) 的運作方式所造成的。ramdisk 會選擇在 `/` 中定義為 `/etc/fstab` 的磁碟區，而在某些發行版本中，這取決於附加至磁碟區分割區的標籤。具體來說，您會發現 `/etc/fstab` 看起來類似於下列的內容：

```
LABEL=/ / ext4 defaults,noatime 1 1
tmpfs /dev/shm tmpfs defaults 0 0
devpts /dev/pts devpts gid=5,mode=620 0 0
sysfs /sys sysfs defaults 0 0
proc /proc proc defaults 0 0
```

如果查看這兩個磁碟區的標籤，會發現皆包含 `/` 標籤：

```
[ec2-user ~]$ sudo e2label /dev/xvda1
/
[ec2-user ~]$ sudo e2label /dev/xvdf1
/
```

在此範例中，最後可能是由 `/dev/xvdf1` 成為根設備，您的執行個體會在初始的 ramdisk 執行之後，透過此根設備啟動，而非透過您原本要從中啟動的 `/dev/xvda1` 磁碟區來啟動。為了解決這項問題，請使用同樣的 `e2label` 命令，針對您不想從其啟動的已連結磁碟區，來變更其標籤。

在某些情況中，在 `/etc/fstab` 中指定 UUID 可以解決這項問題。不過，如果兩個磁碟區都來自於相同的快照，或次要的磁碟區是從主要磁碟區的快照建立，兩個磁碟區就會共用一個 UUID。



```
[ec2-user ~]$ sudo blkid
/dev/xvda1: LABEL="/" UUID=73947a77-ddbe-4dc7-bd8f-3fe0bc840778 TYPE="ext4"
PARTLABEL="Linux" PARTUUID=d55925ee-72c8-41e7-b514-7084e28f7334
/dev/xvdf1: LABEL="old/" UUID=73947a77-ddbe-4dc7-bd8f-3fe0bc840778 TYPE="ext4"
PARTLABEL="Linux" PARTUUID=d55925ee-72c8-41e7-b514-7084e28f7334
```

## 更改已連結 ext4 磁碟區的標籤

1. 使用 `e2label` 命令，將磁碟區的標籤變更為除了 `/` 以外的名稱。

```
[ec2-user ~]$ sudo e2label /dev/xvdf1 old/
```

2. 確認磁碟區已具有新的標籤。

```
[ec2-user ~]$ sudo e2label /dev/xvdf1
old/
```

## 更改已連結 xfs 磁碟區的標籤

- 使用 `xfs_admin` 命令，將磁碟區的標籤變更為除了 `/` 以外的名稱。

```
[ec2-user ~]$ sudo xfs_admin -L old/ /dev/xvdf1
writing all SBs
new label = "old/"
```

在如上述變更磁碟區標籤之後，您應該就能夠重新啟動執行個體，而且讓初始 ramdisk 在執行個體啟動時，選擇正確的磁碟區。

### Important

如果您打算分離具有新標籤的磁碟區，並將該磁碟區傳回給另一個執行個體做為根磁碟區使用，則您必須再次執行上述的程序，以將磁碟區的標籤改回原來的值。否則，其他的執行個體就不會啟動，因為 ramdisk 找不到具有 `/` 標籤的磁碟區。

## 疑難排解 Windows 執行個體的系統問題

### Note

此疑難排解主題僅適用於 Windows 執行個體。

如果您在映像準備期間發生問題或收到錯誤訊息，請檢閱下列日誌。日誌位置取決於您使用 Sysprep 執行的是 EC2Config、EC2Launch v1 或 EC2Launch v2。

- %WINDIR%\Panther\Unattendgc (EC2Config、EC2Launch v1 和 EC2Launch v2)
- %WINDIR%\System32\Sysprep\Panther (EC2Config、EC2Launch v1 和 EC2Launch v2)
- C:\Program Files\Amazon\Ec2ConfigService\Logs\Ec2ConfigLog.txt (僅限 EC2Config)
- C:\ProgramData\Amazon\Ec2Config\Logs (僅限 EC2Config)
- C:\ProgramData\Amazon\EC2-Windows\Launch\Log\EC2Launch.log (僅限 EC2Launch v1)
- %ProgramData%\Amazon\EC2Launch\log\agent.log (僅限 EC2Launch v2)

如果您在使用 Sysprep 準備映像期間收到錯誤訊息，可能會無法連接 OS。若要檢閱日誌檔案，您必須停止執行個體，並將其根磁碟區連接到另一個運作狀態良好的執行個體做為輔助磁碟區，然後於次要磁碟區檢閱前述的日誌。如需依名稱排列日誌檔用途的詳細資訊，請參閱 Microsoft 文件中的 [Windows 安裝程式相關日誌檔](#)。

如果您在 Unattendgc 日誌檔案中找到錯誤，請使用 [Microsoft Error Lookup Tool](#) 取得錯誤的其他詳細資訊。Unattendgc 日誌檔案中所報告的下列問題，通常是因為執行個體上有一或多個毀損的使用者描述檔而導致的結果：

```
Error [Shell Unattend] _FindLatestProfile failed (0x80070003) [gle=0x00000003]
Error [Shell Unattend] CopyProfile failed (0x80070003) [gle=0x00000003]
```

有兩個選項可解決此問題：

### 選項 1

在執行個體上使用 Regedit，搜尋下列機碼。確認沒有已刪除之使用者的描述檔登錄機碼。

```
[HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion
\ProfileList\
```

## 選項 2

1. 編輯相關檔案，如下所示：
  - Windows Server 2012 R2 和舊版 — 編輯 EC2Config 回應檔案 (C:\Program Files\Amazon\Ec2ConfigService\sysprep2008.xml)。
  - Windows Server 2016 和 2019 — 編輯 unattend.xml 回應檔案 (C:\ProgramData\Amazon\EC2-Windows\Launch\Sysprep\Unattend.xml)。
  - Windows Server 2022 — 編輯 unattend.xml 回應檔案 (C:\ProgramData\Amazon\EC2Launch\sysprep\unattend.xml)。
2. 將 `<CopyProfile>>true</CopyProfile>` 變更為 `<CopyProfile>>false</CopyProfile>`。
3. 重新執行 Sysprep。請注意，此組態變更將會在 Sysprep 完成之後，刪除內建管理員使用者描述檔。

## 使用適用於 Linux 的 EC2Rescue

適用於 Linux 的 EC2Rescue 是一種開放原始碼工具 easy-to-use，可在 Amazon EC2 Linux 執行個體上執行，使用其超過 100 個模組的程式庫來診斷和疑難排解常見問題。一些適用於 Linux 的 EC2Rescue 一般使用案例包括搜集 syslog 和套件管理工具日誌、收集資源使用資料和診斷/修復已知問題核心參數和常見 OpenSSH 問題。

AWS Support-TroubleshootSSH Runbook 安裝了適用於 Linux 的 EC2Rescue，然後使用該工具來檢查或嘗試修正阻止透過 SSH 遠端連線至 Linux 機器的常見問題。如需詳細資訊，並執行此自動化，請參閱 [AWS Support-TroubleshootSSH](#)。

如果您使用的是 Windows 執行個體，請參閱 [the section called “EC2Rescue for Windows Server”](#)。

### 目錄

- [安裝適用於 Linux 的 EC2Rescue](#)
- [使用適用於 Linux 的 EC2Rescue](#)
- [開發 EC2Rescue 模組](#)

## 安裝 適用於 Linux 的 EC2Rescue

適用於 Linux 的 EC2Rescue 工具可安裝在符合下列要件的 Amazon EC2 Linux 執行個體上。

### 必要條件

- 支援的作業系統：
  - Amazon Linux 2
  - Amazon Linux 2016.09+
  - SUSE Linux Enterprise Server 12+
  - RHEL 7+
  - Ubuntu 16.04+
- 軟體要求：
  - Python 2.7.9+ 或 3.2+

AWS Support-TroubleshootSSH Runbook 安裝了 適用於 Linux 的 EC2Rescue，然後使用該工具來檢查或嘗試修正阻止透過 SSH 遠端連線至 Linux 機器的常見問題。如需詳細資訊，並執行此自動化，請參閱 [AWS Support-TroubleshootSSH](#)。

如果您的系統具有所需的 Python 版本，則您可以安裝標準組建。否則，您可以安裝隨附的組建，其中包括最小的 Python 複本。

### 安裝標準組建

1. 從工作中的 Linux 執行個體下載 [適用於 Linux 的 EC2Rescue](#) 工具：

```
curl -O https://s3.amazonaws.com/ec2rescuelinix/ec2r1.tgz
```

2. (選用) 在繼續之前，您可以選擇確認 適用於 Linux 的 EC2Rescue 安裝檔案的簽章。如需詳細資訊，請參閱 [\(選用\) 確認 適用於 Linux 的 EC2Rescue 的簽章](#)。

3. 下載 sha256 雜湊檔案：

```
curl -O https://s3.amazonaws.com/ec2rescuelinix/ec2r1.tgz.sha256
```

4. 確認 tarball 的完整性：

```
sha256sum -c ec2r1.tgz.sha256
```

## 5. 解壓縮 tarball :

```
tar -xzvf ec2r1.tgz
```

## 6. 列出說明檔確認安裝成功 :

```
cd ec2r1-<version_number>
./ec2r1 help
```

## 安裝隨附的組建

如需下載和限制清單的連結，請參閱 github 上的 [適用於 Linux 的 EC2Rescue](#)。

## (選用) 確認 適用於 Linux 的 EC2Rescue 的簽章

下列是驗證用於以 Linux 為基礎之作業系統的 適用於 Linux 的 EC2Rescue 套件有效性的建議程序。

當您從網際網路下載應用程式時，建議您驗證軟體發佈者的身分，並檢查應用程式在發佈之後並未遭更改或損毀。如此可保護您，避免安裝到包含病毒或其他惡意程式碼的應用程式版本。

如果在執行此主題中的步驟後，您判斷 適用於 Linux 的 EC2Rescue 的軟體已遭更改或損毀，請勿執行安裝檔案。反之，請聯絡 Amazon Web Services。

適用於以 Linux 為基礎之作業系統的 適用於 Linux 的 EC2Rescue 檔案使用 GnuPG 簽章，這是一種用於安全數位簽章的 Pretty Good Privacy (OpenPGP) 標準的開放原始碼實作。GnuPG ( 也稱為 GPG ) 通過數字簽名提供身份驗證和完整性檢查。AWS 會發佈公開金鑰和簽章，供您用來驗證下載的 EC2Rescue 適用於 Linux 套件。如需有關 PGP 和 GnuPG (GPG) 的詳細資訊，請參閱 <http://www.gnupg.org>。

第一步是與軟體發佈者建立信任。下載軟體發佈者的公開金鑰，檢查公開金鑰的擁有者是否為聲稱的擁有者，然後將公開金鑰新增至您的 keyring。您的 keyring 是一組已知的公開金鑰。在您建立公開金鑰的真實性之後，即可用它來驗證應用程式的簽章。

### 工作

- [安裝 GPG 工具](#)
- [驗證和匯入公開金鑰](#)
- [驗證套件的簽章](#)

## 安裝 GPG 工具

如果您的作業系統是 Linux 或 Unix，那麼 GPG 工具可能已安裝。若要測試工具是否已安裝在您的系統，請在命令提示字元中輸入 `gpg2`。如果 GPG 工具已安裝，您會看到 GPG 命令提示字元。如果 GPG 工具未安裝，您會看到表示找不到該命令的錯誤。您可以從儲存庫安裝 GnuPG 套件。

在以 Debian 為基礎的 Linux 上安裝 GPG 工具

- 從終端機執行下列命令：

```
apt-get install gnupg2
```

在以 Red Hat 為基礎的 Linux 上安裝 GPG 工具

- 從終端機執行下列命令：

```
yum install gnupg2
```

## 驗證和匯入公開金鑰

此程序的下一個步驟是驗證適用於 Linux 的 EC2Rescue 公開金鑰，並將其做為信任的金鑰新增至 GPG keyring。

驗證和匯入適用於 Linux 的 EC2Rescue 公開金鑰

- 在命令提示中，使用以下命令以取得我們公開 GPG 建置金鑰的複本：

```
curl -O https://s3.amazonaws.com/ec2rescuelinux/ec2r1.key
```

- 在您儲存 `ec2r1.key` 的目錄的命令提示字元中，使用下列命令將適用於 Linux 的 EC2Rescue 公開金鑰匯入至您的 keyring：

```
gpg2 --import ec2r1.key
```

此命令會傳回類似以下的結果：

```
gpg: /home/ec2-user/.gnupg/trustdb.gpg: trustdb created
```

```
gpg: key 2FAE2A1C: public key "ec2autodiag@amazon.com <EC2 Rescue for Linux>"
imported
gpg: Total number processed: 1
gpg: imported: 1 (RSA: 1)
```

## 驗證套件的簽章

在您安裝 GPG 工具、驗證及匯入適用於 Linux 的 EC2Rescue 公有金鑰，以及驗證適用於 Linux 的 EC2Rescue 公有金鑰可信任之後，您就可以驗證適用於 Linux 的 EC2Rescue 安裝指令碼的簽章。

### 驗證適用於 Linux 的 EC2Rescue 安裝指令碼簽章

1. 在命令提示字元上，執行以下命令以下載安裝指令碼的簽章檔案：

```
curl -O https://s3.amazonaws.com/ec2rescuelineux/ec2r1.tgz.sig
```

2. 在您儲存 ec2r1.tgz.sig 與適用於 Linux 的 EC2Rescue 安裝檔案的目錄中，在命令提示字元上執行以下命令以驗證簽章。兩個檔案都必須存在。

```
gpg2 --verify ./ec2r1.tgz.sig
```

輸出應類似以下所示：

```
gpg: Signature made Thu 12 Jul 2018 01:57:51 AM UTC using RSA key ID 6991ED45
gpg: Good signature from "ec2autodiag@amazon.com <EC2 Rescue for Linux>"
gpg: WARNING: This key is not certified with a trusted signature!
gpg: There is no indication that the signature belongs to the owner.
Primary key fingerprint: E528 BCC9 0DBF 5AFA 0F6C C36A F780 4843 2FAE 2A1C
Subkey fingerprint: 966B 0D27 85E9 AECC 1146 7A9D 8851 1153 6991 ED45
```

如果輸出包含 Good signature from "ec2autodiag@amazon.com <EC2 Rescue for Linux>" 片語，表示簽章已成功驗證，您可以繼續執行適用於 Linux 的 EC2Rescue 安裝指令碼。

如果輸出包含 BAD signature 片語，請檢查您是否已正確執执行程序。如果您持續收到此回應，請聯絡 Amazon Web Services 並且不要執行您先前下載的安裝檔。

以下是您可能會看到的警告的詳細資訊：

- **WARNING:** This key is not certified with a trusted signature! There is no indication that the signature belongs to the owner.這是指您個人對於您所擁有適用於適用於 Linux 的 EC2Rescue 的真實公有金鑰的信任。在理想世界中，您請會前往 Amazon Web Services 辦公室並獲得該金鑰。不過，通常您會從網站下載。在此情況下，該網站是 Amazon Web Services 網站。
- gpg2: no ultimately trusted keys found. 這表示該特定金鑰未獲得您 (或您信任的其他人) 的「最終信任」。

如需詳細資訊，請參閱 <http://www.gnupg.org>。

## 使用適用於 Linux 的 EC2Rescue

下列是您可執行以開始使用本工具的常用任務。

### 工作

- [執行適用於 Linux 的 EC2Rescue](#)
- [上傳結果](#)
- [建立備份](#)
- [取得說明](#)

## 執行適用於 Linux 的 EC2Rescue

您可以執行適用於 Linux 的 EC2Rescue，如下列範例所示。

Example 範例：執行所有模組

若要執行所有模組，請執行不含任何選項的適用於 Linux 的 EC2Rescue：

```
./ec2r1 run
```

部分模組需要根存取權。如果您不是根使用者，請使用 `sudo` 來執行這些模組，如下所示：

```
sudo ./ec2r1 run
```

Example 範例：執行特定模組

若要僅執行特定模組，請使用 `--only-modules` 參數：



```
./ec2r1 run --only-modules=module_name --arguments
```

例如，此命令執行 dig 模組來查詢 amazon.com 網域：

```
./ec2r1 run --only-modules=dig --domain=amazon.com
```

Example 範例：檢視結果

您可以檢視 /var/tmp/ec2r1 中的結果：

```
cat /var/tmp/ec2r1/logfile_location
```

例如，檢視 dig 模組的日誌檔：

```
cat /var/tmp/ec2r1/2017-05-11T15_39_21.893145/mod_out/run/dig.log
```

## 上傳結果

如果 AWS Support 已要求結果或要共用 S3 儲存貯體的結果，請使用適用於 Linux CLI 的 EC2Rescue 工具上傳結果。適用於 Linux 的 EC2Rescue 命令的輸出應提供您需要使用的命令。

Example 範例：將結果上傳至 AWS Support

```
./ec2r1 upload --upload-directory=/var/tmp/ec2r1/2017-05-11T15_39_21.893145 --support-url="URLProvidedByAWSSupport"
```

Example 範例：將結果上傳至 S3 儲存貯體

```
./ec2r1 upload --upload-directory=/var/tmp/ec2r1/2017-05-11T15_39_21.893145 --presigned-url="YourPresignedS3URL"
```

如需為 Amazon S3 產生預先簽章的 URL 的詳細資訊，請參閱[使用預先簽章的 URL 上傳物件](#)。

## 建立備份

使用下列命令，為您的執行個體、一或多個磁碟區或特定的裝置 ID 建立儲存貯體。

Example 範例：使用 Amazon Machine Image (AMI) 備份執行個體

```
./ec2r1 run --backup=ami
```

Example 範例：備份與執行個體相關聯的所有磁碟區

```
./ec2r1 run --backup=allvolumes
```

Example 範例：備份特定磁碟區

```
./ec2r1 run --backup=volumeID
```

## 取得說明

適用於 Linux 的 EC2Rescue 包含提供資訊的說明檔及各可用命令的語法。

Example 範例：顯示一般說明

```
./ec2r1 help
```

Example 範例：列出可用的模組

```
./ec2r1 list
```

Example 範例：顯示特定模組的說明

```
./ec2r1 help module_name
```

例如，使用下列命令來顯示 dig 模組的說明檔：

```
./ec2r1 help dig
```

## 開發 EC2Rescue 模組

模組是使用 YAML (資料序列化標準) 撰寫。模組的 YAML 檔案是由單一文件組成，代表模組及其屬性。

### 新增模組屬性

下表列出可用的模組屬性。

屬性	描述
name	模組的名稱。此名稱長度應小於或等於 18 個字元。
version	模組的版本編號。
title	模組的簡述標題。此值長度應小於或等於 50 個字元。
helptext	<p>模組的延伸描述。各行長度應小於或等於 75 個字元。如果模組使用引數 (無論必要或選用), 請以 helptext 值包含它們。</p> <p>例如 :</p> <pre>helptext: !!str     Collect output from ps for system   analysis   Consumes --times= for number of times   to repeat   Consumes --period= for time period   between repetition</pre>
placement	<p>應該執行模組的階段。支援的值為 :</p> <ul style="list-style-type: none"><li>• prediagnostic</li><li>• run</li><li>• postdiagnostic</li></ul>
語言	<p>撰寫模組程式碼的語言。支援的值為 :</p> <ul style="list-style-type: none"><li>• bash</li><li>• Python</li></ul>

屬性	描述
	<div data-bbox="829 212 1511 430" style="border: 1px solid #ccc; border-radius: 10px; padding: 10px;"> <p> <b>Note</b></p> <p>Python 程式碼必須與 Python 2.7.9+ 和 Python 3.2+ 都相容。</p> </div>
remediation	<p>指出模組是否支援修補。支援的值為 True 或 False。</p> <p>如果無此屬性，模組將預設為 False，讓其成為不支援修補之模組的選用屬性。</p>
content	完整的指令碼程式碼。
限制條件	包含限制值的物件名稱。
domain	<p>說明模組如何分組或分類的描述項。包含的模組集使用下列網域：</p> <ul style="list-style-type: none"> <li>• application</li> <li>• net</li> <li>• os</li> <li>• performance</li> </ul>
class	<p>模組執行的任務類型描述項。包含的模組集使用下列類別：</p> <ul style="list-style-type: none"> <li>• collect (收集程式的輸出)</li> <li>• diagnose (根據條件集通過/失敗)</li> <li>• gather (複製檔案並寫入特定檔案)</li> </ul>

屬性	描述
distro	此模組支援的 Linux 發行版本清單。包含的模組集使用下列發行版本： <ul style="list-style-type: none"><li>• alami (Amazon Linux)</li><li>• rhel</li><li>• ubuntu</li><li>• suse</li></ul>
必要	模組從 CLI 選項使用的必要引數。
選擇性	模組可以使用的選用引數。
software	在模組中使用的軟體可執行檔。此屬性旨在指定非預設安裝的軟體。適用於 Linux 的 EC2Rescue 邏輯在執行模組前會確保這些程式存在並可執行。
package	可執行檔的來源軟體套件。此屬性旨在提供套件有關軟體的延伸詳細資料，包括下載或取得未來資訊的 URL。
sudo	指出是否一定要有根存取才能執行模組。  您不必在模組指令碼中實作 sudo 檢查。如果此值為 true，則當執行使用者有根存取時，適用於 Linux 的 EC2Rescue 邏輯只會執行模組。
perfimpact	指出模組是否會顯著影響其執行所在環境的效能。如果此值為 true 且無 <code>--perfimpact=true</code> 引數，則跳過模組。
parallelexclusive	指定需要互斥的程式。例如，所有指定 "bpf" 的模組都依序執行。

## 新增環境變數

下表列出可用的環境變數。

環境變數	描述
EC2RL_CALLPATH	ec2rl.py 的路徑。此路徑可用以尋找 lib 目錄及使用經過廠商處理的 Python 模組。
EC2RL_WORKDIR	診斷工具的主要 tmp 目錄。 預設值：/var/tmp/ec2rl 。
EC2RL_RUNDIR	存放所有輸出的目錄。 預設值：/var/tmp/ec2rl/<date&timestamp> 。
EC2RL_GATHEREDDIR	放置所收集模組資料的根目錄。 預設值：/var/tmp/ec2rl/<date&timestamp>/mod_out/gathered/ 。
EC2RL_NET_DRIVER	執行個體上按字母順序第一個非虛擬網路界面使用的驅動程式。  範例： <ul style="list-style-type: none"> <li>• xen_netfront</li> <li>• ixgbevf</li> <li>• ena</li> </ul>
EC2RL_SUDO	如果適用於 Linux 的 EC2Rescue 執行為根，即為 true；否則為 false。
EC2RL_VIRT_TYPE	如執行個體中繼資料所提供的虛擬化類型。  範例： <ul style="list-style-type: none"> <li>• default-hvm</li> </ul>

環境變數	描述
	<ul style="list-style-type: none"> <li>default-paravirtual</li> </ul>
EC2RL_INTERFACES	系統上的界面列舉清單。值是包含名稱的字串，例如 eth0、eth1 等等。這是透過 functions.bash 產生的，僅供其來源模組使用。

## 使用 YAML 語法

建構您的模組 YAML 檔案時，應該注意下列各點：

- 三個連字號 (---) 表示文件的明確開頭。
- 從資料串流建立物件時，!ec2rlcore.module.Module 標籤會告訴 YAML 剖析器要呼叫哪個建構函數。您可於 module.py 檔案中找到建構函數。
- !!str 標籤會告訴 YAML 剖析器不要嘗試判斷資料類型，改將內容解譯為字串常值。
- 管道字元 (|) 會告訴 YAML 剖析器此值為常值型純量。在本例中，剖析器包含所有空格。這對模組而言很重要，因為會保留縮排和換行字元。
- YAML 標準會縮排兩格，如下列範例所示。請確定您的指令碼保持標準縮排 (例如，Python 四格)，然後在模組檔案中將完整內容縮排兩格。

## 範例模組

範例一 (mod.d/ps.yaml)：

```

--- !ec2rlcore.module.Module
Module document. Translates directly into an almost-complete Module object
name: !!str ps
path: !!str
version: !!str 1.0
title: !!str Collect output from ps for system analysis
helptext: !!str |
 Collect output from ps for system analysis
 Requires --times= for number of times to repeat
 Requires --period= for time period between repetition
placement: !!str run
package:
 - !!str
language: !!str bash

```

```
content: !!str |
#!/bin/bash
error_trap()
{
 printf "%0.s=" {1..80}
 echo -e "\nERROR: "$BASH_COMMAND" exited with an error on line ${BASH_LINENO[0]}"
 exit 0
}
trap error_trap ERR

read-in shared function
source functions.bash
echo "I will collect ps output from this $EC2RL_DISTRO box for $times times every
$period seconds."
for i in $(seq 1 $times); do
 ps auxww
 sleep $period
done
constraint:
requires_ec2: !!str False
domain: !!str performance
class: !!str collect
distro: !!str alami ubuntu rhel suse
required: !!str period times
optional: !!str
software: !!str
sudo: !!str False
perfimpact: !!str False
parallelexclusive: !!str
```

## 使用 EC2Rescue for Windows Server

EC2Rescue 適用於 Windows 伺服器是您在 Amazon EC2 Windows 伺服器執行個體上執行的 easy-to-use 工具，用於診斷和疑難排解可能的問題。收集日誌檔案和故障診斷問題以及主動搜尋可能發生問題的區域，極其重要。它甚至可以從其他執行個體檢查 Amazon EBS 根磁碟區，並收集相關日誌以故障診斷使用該磁碟區的 Windows Server 執行個體。

EC2Rescue for Windows Server 有兩個不同的模組：從所有不同來源收集資料的資料收集器模組，以及剖析根據一系列預先定義規則所收集之資料找出問題並提供建議的分析器模組。

EC2Rescue 適用於視窗伺服器工具僅在執行視窗伺服器 2012 及更新版本的 Amazon EC2 執行個體上執行。當此工具啟動時，它會檢查自己是否在 Amazon EC2 執行個體上執行。



AWSSupport-ExecuteEC2Rescue Runbook 使用 EC2Rescue 工具進行疑難排解，並在可能的情況下修正指定 EC2 執行個體的常見連線問題。如需詳細資訊，並執行此自動化操作，請參閱 [AWSSupport-ExecuteEC2Rescue](#)。

如果您使用的是 Linux 執行個體，請參閱 [the section called “EC2Rescue for Linux”](#)。

## 目錄

- [使用 EC2Rescue for Windows Server GUI](#)
- [將 EC2Rescue for Windows Server 與命令列搭配使用](#)
- [使用 EC2Rescue for Windows Server 搭配 Systems Manager 執行命令](#)

## 使用 EC2Rescue for Windows Server GUI

EC2Rescue for Windows Server 可對離線執行個體執行下列分析：

選項	描述
診斷與救援	<p>EC2Rescue for Windows Server 可以偵測並解決下列服務設定的問題：</p> <ul style="list-style-type: none"> <li>• 系統時間 <ul style="list-style-type: none"> <li>• RealTimeis通用-偵測RealTimeisUniversal 登錄機碼是否已啟用。如果停用，當時區值不是設為 UTC 時，Windows 系統時間會產生差異。</li> </ul> </li> <li>• Windows 防火牆 <ul style="list-style-type: none"> <li>• Domain networks (網域網路) - 偵測此 Windows 防火牆描述檔為啟用或停用。</li> <li>• Private networks (私有網路) - 偵測此 Windows 防火牆描述檔為啟用或停用。</li> <li>• Guest or public networks (訪客或公有網路) - 偵測此 Windows 防火牆描述檔為啟用或停用。</li> </ul> </li> </ul>

選項	描述
	<ul style="list-style-type: none"><li>• 遠端桌面<ul style="list-style-type: none"><li>• Service Start (服務啟動) - 偵測是否啟用遠端桌面服務。</li><li>• Remote Desktop Connections (遠端桌面連線) - 偵測是否啟用此服務。</li><li>• TCP Port (TCP 連接埠) - 偵測遠端桌面服務正在接聽哪一個連接埠。</li></ul></li> <li>• EC2Config (Windows Server 2012 R2 和更早的版本)<ul style="list-style-type: none"><li>• Installation (安裝) - 偵測安裝的是哪個 EC2Config 版本。</li><li>• Service Start (服務啟動) - 偵測是否啟用 EC2Config 服務。</li><li>• EC2 SetPassword-產生新的管理員密碼。</li><li>• EC2 HandleUser Data-可讓您在執行個體下次啟動時執行使用者資料指令碼。</li></ul></li> <li>• EC2Launch (Windows Server 2016 及更新版本)<ul style="list-style-type: none"><li>• Installation (安裝) - 偵測安裝的是哪個 EC2Launch 版本。</li><li>• EC2 SetPassword-產生新的管理員密碼。</li></ul></li> <li>• 網路界面<ul style="list-style-type: none"><li>• DHCP Service Startup (DHCP 服務啟動) - 偵測是否啟用 DHCP 服務。</li><li>• Ethernet detail (乙太網路詳細資訊) - 如果偵測到，顯示網路驅動程式版本的相關資訊。</li></ul></li></ul>

選項	描述
	<ul style="list-style-type: none"> <li>• DHCP on Ethernet (乙太網路上的 DHCP) - 偵測是否啟用 DHCP。</li> <li>• 磁碟簽章狀態</li> <li>• Signature on disk (磁碟上的簽章) 和 Signature on Boot Configuration Database (BCD) (開機組態資料庫 (BCD) 上的簽章)- 偵測磁碟簽章和 BCD 簽章是否相同。如果值不同，EC2Rescue 會嘗試使用 BCD 上的簽章覆寫磁碟簽章。</li> </ul>
還原	<p>執行下列動作之一：</p> <ul style="list-style-type: none"> <li>• Last Known Good Configuration (上一個已知的良好組態) - 嘗試將執行個體開機為上一個已知的可開機狀態。</li> <li>• Restore registry from backup (從備份還原登錄) - 從 <code>\Windows\System32\config\RegBack</code> 還原登錄。</li> </ul>
擷取日誌	允許您擷取執行個體的日誌用於分析。

EC2Rescue for Windows Server 可從作用中和離線的執行個體收集下列資料：

項目	描述
事件日誌	收集應用程式、系統和 EC2Config 事件日誌。
登錄	收集 SYSTEM 和 SOFTWARE Hive。
Windows Update 日誌	收集 Windows Update 產生的日誌檔案。

項目	描述
	<p> <b>Note</b></p> <p>在 Windows Server 2016 年及更新版本中，日誌會以 Event Tracing for Windows (ETW) 格式收集。</p>
Sysprep 日誌	收集 Windows 系統準備工具產生的日誌檔案。
驅動程式設定日誌	收集 Windows SetupAPI 日誌 (setupapi.dev.log 和 setupapi.setup.log )。
開機組態	收集 HKEY_LOCAL_MACHINE\BCD00000000 Hive。
記憶體傾印	收集執行個體上所有的記憶體傾印檔案。
EC2Config 檔案	收集 EC2Config 服務產生的日誌檔案。
EC2Launch 檔案	收集 EC2Launch 指令碼產生的日誌檔案。
SSM Agent 檔案	收集 SSM Agent 和修補程式管理員日誌產生的日誌檔案。
EC2 ElasticGPUs 檔案	收集與 Elastic GPU 相關的事件日誌。
ECS	收集與 Amazon ECS 相關的日誌。
CloudEndure	收集與 CloudEndure 代理程式相關的記錄檔。

EC2Rescue for Windows Server 可從作用中執行個體收集下列其他資料：

項目	描述
系統資訊	收集 MSInfo32。
群組政策結果	收集 Group Policy (群組政策) 報告。

## 分析離線的執行個體

Offline Instance (離線的執行個體) 選項對偵錯 Windows 執行個體的開機問題很有幫助。

在離線的執行個體上執行動作

1. 從工作中的 Windows Server 執行個體，下載 [EC2Rescue for Windows Server](#) 工具並解壓縮檔案。

您可以執行下列 PowerShell 命令來下載 EC2Rescue，而不必變更網際網路資源管理器增強型安全性設定 (ESC)：

```
Invoke-WebRequest https://s3.amazonaws.com/ec2rescue/windows/EC2Rescue_latest.zip -
OutFile $env:USERPROFILE\Desktop\EC2Rescue_latest.zip
```

此命令會將 EC2Rescue .zip 檔案下載到目前登入使用者的桌面。

### Note

如果您在下載檔案時收到錯誤訊息，而且您使用的是 Windows 伺服器 2016 或更早版本，則可能需要為您的 PowerShell 終端機啟用 TLS 1.2。您可以使用下列命令為目前 PowerShell 工作階段啟用 TLS 1.2，然後再試一次：

```
[Net.ServicePointManager]::SecurityProtocol =
[Net.SecurityProtocolType]::Tls12
```

2. 如果故障的執行個體尚未停止，請停止它。
3. 從故障的執行個體分離 EBS 根磁碟區，再將此磁碟區連接到已安裝 EC2Rescue for Windows Server 的工作中 Windows 執行個體。
4. 在工作中的執行個體上執行 EC2Rescue for Windows Server 工具並選擇 Offline Instance (離線的執行個體)。
5. 選取新掛載磁碟區上的磁碟，然後選擇 Next (下一步)。
6. 確認選取磁碟，然後選擇 Yes (是)。
7. 選擇要執行的離線執行個體選項並選擇 Next (下一步)。

EC2Rescue for Windows Server 工具會掃描磁碟區，並根據選取的日誌檔案收集故障診斷資訊。

## 收集作用中執行個體的資料

您可收集作用中執行個體的日誌和其他資料。

### 收集作用中執行個體的資料

1. 連接至 Windows 執行個體。
2. 將 [EC2Rescue for Windows Server](#) 工具下載到您的 Windows 執行個體並解壓縮檔案。

您可以執行下列 PowerShell 命令來下載 EC2Rescue，而不必變更網際網路資源管理器增強型安全性設定 (ESC)：

```
Invoke-WebRequest https://s3.amazonaws.com/ec2rescue/windows/EC2Rescue_latest.zip -
OutFile $env:USERPROFILE\Desktop\EC2Rescue_latest.zip
```

此命令會將 EC2Rescue .zip 檔案下載到目前登入使用者的桌面。

#### Note

如果您在下載檔案時收到錯誤訊息，而且您使用的是 Windows 伺服器 2016 或更早版本，則可能需要為您的 PowerShell 終端機啟用 TLS 1.2。您可以使用下列命令為目前 PowerShell 工作階段啟用 TLS 1.2，然後再試一次：

```
[Net.ServicePointManager]::SecurityProtocol =
[Net.SecurityProtocolType]::Tls12
```

3. 開啟 EC2Rescue for Windows Server 應用程式並接受授權合約。
4. 選擇 Next (下一步)、Current instance (目前的執行個體)、Capture logs (擷取日誌)。
5. 選取要收集的資料項目並選擇 Collect... (收集)。閱讀警告內容，然後選擇 Yes (是) 繼續。
6. 為 ZIP 檔案選擇檔案名稱及位置，然後選擇 Save (儲存)。
7. 完成 EC2Rescue for Windows Server 之後，選擇 Open Containing Folder (開啟包含的資料夾) 檢視 ZIP 檔案。
8. 選擇 Finish (完成)。

## 將 EC2Rescue for Windows Server 與命令列搭配使用

EC2Rescue for Windows Server 命令列界面 (CLI) 讓您透過編寫程式的方式執行 EC2Rescue for Windows Server 外掛程式 (稱為「動作」)。

EC2Rescue for Windows Server 工具有兩種執行模式：

- /online—這可讓您對 EC2Rescue for Windows Server 安裝所在的執行個體採取動作，例如收集日誌檔案。
- /offline:<device\_id>—這可讓您對連線到不同 Amazon EC2 Windows 執行個體的離線根磁碟區採取動作，此執行個體已安裝 EC2Rescue for Windows Server。

將 [EC2Rescue for Windows Server](#) 工具下載到您的 Windows 執行個體並解壓縮檔案。您可以使用下列命令來檢視說明檔案：

```
EC2RescueCmd.exe /help
```

EC2Rescue for Windows Server 會議主持人可在 Amazon EC2 Windows 執行個體上執行下列動作：

- [收集動作](#)
- [救援動作](#)
- [還原動作](#)

### 收集動作

#### Note


您可收集所有日誌、完整的日誌群組，或群組內的個別日誌。

EC2Rescue for Windows Server 可從作用中和離線的執行個體收集下列資料。

日誌群組	可用日誌	描述
all		收集所有可用的日誌。
eventlog	• 'Application'	收集應用程式、系統和 EC2Config 事件日誌。

日誌群組	可用日誌	描述
	<ul style="list-style-type: none"> <li>'System'</li> <li>'EC2ConfigService'</li> </ul>	
memory-dump	<ul style="list-style-type: none"> <li>'Memory Dump File'</li> <li>'Mini Dump Files'</li> </ul>	收集執行個體上所有的記憶體傾印檔案。
ec2config	<ul style="list-style-type: none"> <li>'Log Files'</li> <li>'Configuration Files'</li> </ul>	收集 EC2Config 服務產生的日誌檔案。
ec2launch	<ul style="list-style-type: none"> <li>'Logs'</li> <li>'Config'</li> </ul>	收集 EC2Launch 指令碼產生的日誌檔案。
ssm-agent	<ul style="list-style-type: none"> <li>'Log Files'</li> <li>'Patch Baseline Logs'</li> <li>'InstanceData'</li> </ul>	收集 SSM Agent 和修補程式管理員日誌產生的日誌檔案。
sysprep	'Log Files'	收集 Windows 系統準備工具產生的日誌檔案。
driver-setup	<ul style="list-style-type: none"> <li>'SetupAPI Log Files'</li> <li>'DPIInst Log File'</li> <li>'AWS PV Setup Log File'</li> </ul>	收集 Windows SetupAPI 日誌 (setupapi.dev.log 和 setupapi.setup.log )。
registry	<ul style="list-style-type: none"> <li>'SYSTEM'</li> <li>'SOFTWARE'</li> <li>'BCD'</li> </ul>	收集 SYSTEM 和 SOFTWARE Hive。
egpu	<ul style="list-style-type: none"> <li>'Event Log'</li> <li>'System Files'</li> </ul>	收集與 Elastic GPU 相關的事件日誌。
boot-config	'BCDEDIT Output'	收集 HKEY_LOCAL_MACHINE \BCD00000000 Hive。



日誌群組	可用日誌	描述
windows-update	'Log Files'	收集 Windows Update 產生的日誌檔案。  <div style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px; background-color: #e6f2ff;"> <p> <b>Note</b></p> <p>在 Windows Server 2016 年及更新版本中，日誌會以 Event Tracing for Windows (ETW) 格式收集。</p> </div>
cloudendure	<ul style="list-style-type: none"> <li>• 'Migrate Script Logs'</li> <li>• 'Driver Logs'</li> <li>• 'CloudEndure File List'</li> </ul>	收集與 CloudEndure 代理程式相關的記錄檔。

EC2Rescue for Windows Server 可從作用中執行個體收集下列其他資料。

日誌群組	可用日誌	描述
system-info	'MSInfo32 Output'	收集 MSInfo32。
gpresult	'GPResult Output'	收集 Group Policy (群組政策) 報告。

以下為可用的選項：

- /output: < 輸出 FilePath >-以 zip 格式儲存收集的記錄檔案所需的目標檔案路徑位置。
- /no-offline - 離線模式中使用的選用屬性。完成動作後不會將磁碟區設為離線。
- /no-fix-signature - 離線模式中使用的選用屬性。完成動作後不會修正可能的磁碟簽章衝突。

## 範例

以下是使用EC2Rescue for Windows Server CLI 的範例。

### 線上模式範例

收集所有可用的日誌：

```
EC2RescueCmd /accepteula /online /collect:all /output:<outputFilePath>
```

只收集特定的日誌群組：

```
EC2RescueCmd /accepteula /online /collect:ec2config /output:<outputFilePath>
```

收集日誌群組內的個別日誌：

```
EC2RescueCmd /accepteula /online /collect:'ec2config.Log Files,driver-setup.SetupAPI
Log Files' /output:<outputFilePath>
```

### 離線模式範例

收集 EBS 磁碟區中所有可用的日誌。磁碟區依 device\_id 值指定。

```
EC2RescueCmd /accepteula /offline:xvdf /collect:all /output:<outputFilePath>
```

只收集特定的日誌群組：

```
EC2RescueCmd /accepteula /offline:xvdf /collect:ec2config /output:<outputFilePath>
```

## 救援動作

EC2Rescue for Windows Server 可以偵測並解決下列服務設定的問題：

服務群組	可用動作	描述
all		
system-time	'RealTimeIsUniversal'	系統時間

服務群組	可用動作	描述
		<ul style="list-style-type: none"> <li>• RealTimeis通用-偵測RealTimeisUniversal 登錄機碼是否已啟用。如果停用，當時區值不是設為 UTC 時，Windows 系統時間會產生差異。</li> </ul>
firewall	<ul style="list-style-type: none"> <li>• 'Domain networks'</li> <li>• 'Private networks'</li> <li>• 'Guest or public networks'</li> </ul>	<p>Windows 防火牆</p> <ul style="list-style-type: none"> <li>• Domain networks (網域網路) - 偵測此 Windows 防火牆描述檔為啟用或停用。</li> <li>• Private networks (私有網路) - 偵測此 Windows 防火牆描述檔為啟用或停用。</li> <li>• Guest or public networks (訪客或公有網路) - 偵測此 Windows 防火牆描述檔為啟用或停用。</li> </ul>
rdp	<ul style="list-style-type: none"> <li>• 'Service Start'</li> <li>• 'Remote Desktop Connections'</li> <li>• 'TCP Port'</li> </ul>	<p>遠端桌面</p> <ul style="list-style-type: none"> <li>• Service Start (服務啟動) - 偵測是否啟用遠端桌面服務。</li> <li>• Remote Desktop Connections (遠端桌面連線) - 偵測是否啟用此服務。</li> <li>• TCP Port (TCP 連接埠) - 偵測遠端桌面服務正在接聽哪一個連接埠。</li> </ul>

服務群組	可用動作	描述
ec2config	<ul style="list-style-type: none"> <li>'Service Start'</li> <li>'Ec2SetPassword'</li> <li>'Ec2HandleUserData'</li> </ul>	EC2Config <ul style="list-style-type: none"> <li>Service Start (服務啟動) - 偵測是否啟用 EC2Config 服務。</li> <li>EC2 SetPassword-產生新的管理員密碼。</li> <li>EC2 HandleUser Data-可讓您在執行個體下次啟動時執行使用者資料指令碼。</li> </ul>
ec2launch	'Reset Administrator Password'	產生新的 Windows 管理員密碼。
network	'DHCP Service Startup'	網路界面 <ul style="list-style-type: none"> <li>DHCP Service Startup (DHCP 服務啟動) - 偵測是否啟用 DHCP 服務。</li> </ul>

以下為可用的選項：

- /level:<level> - 動作應該觸發的檢查層級選用屬性。允許的值為：information、warning、error、all。預設會設定為 error。
- /check-only - 會產生報告但不修改離線磁碟區的選用屬性。

#### Note

如果適用於 Windows Server 的 EC2Rescue 偵測到可能的磁碟簽章衝突，即使您使用該選項，它也會在離線處理程序完成之後更正簽章。/check-only您必須使用/no-fix-signature此選項來防止修正。

- /no-offline - 在動作完成後防止將磁碟區設為離線的選用屬性。
- /no-fix-signature - 完成動作後不會修正可能磁碟簽章衝突的選用屬性。

## 救援範例

以下是使用EC2Rescue for Windows Server CLI 的範例。磁碟區使用 `device_id` 值指定。

嘗試修正磁碟區中找到的所有問題：

```
EC2RescueCmd /accepteula /offline:xvdf /rescue:all
```

嘗試修正磁碟區服務群組內的所有問題：

```
EC2RescueCmd /accepteula /offline:xvdf /rescue:firewall
```

嘗試修正磁碟區服務群組內的特定項目：

```
EC2RescueCmd /accepteula /offline:xvdf /rescue:rdp.'Service Start'
```

指定磁碟區中嘗試修正的多個問題：

```
EC2RescueCmd /accepteula /offline:xvdf /rescue:'system-time.RealTimeIsUniversal,ec2config.Service Start'
```

## 還原動作

EC2Rescue for Windows Server 可以偵測並解決下列服務設定的問題：

服務群組	可用的動作	描述
還原上一個已知的良好組態	lkgc	Last Known Good Configuration (上一個已知的良好組態) - 嘗試將執行個體開機為上一個已知的可開機狀態。
從最新的備份還原 Windows 登錄	regback	Restore registry from backup (從備份還原登錄) - 從 <code>\Windows\System32\config\RegBack</code> 還原登錄。

以下為可用的選項：

- /no-offline— 在動作完成後防止將磁碟區設為離線的選用屬性。
- /no-fix-signature— 完成動作後不會修正可能磁碟簽章衝突的選用屬性。

## 還原範例

以下是使用 EC2Rescue for Windows Server CLI 的範例。磁碟區使用 device\_id 值指定。

還原磁碟區中上一個已知的良好組態：

```
EC2RescueCmd /accepteula /offline:xvdf /restore:lkgc
```

在磁碟區上還原上一個 Windows 登錄備份：

```
EC2RescueCmd /accepteula /offline:xvdf /restore:regback
```

## 使用 EC2Rescue for Windows Server 搭配 Systems Manager 執行命令

AWS Support 提供 Systems Manager 執行命令文件，以連接啟用系統管理員的執行個體，以執行適用於 Windows 伺服器的 EC2Rescue。執行命令文件稱為 AWSSupport-RunEC2RescueForWindowsTool。

此 Systems Manager 執行命令文件會執行下列任務：

- 下載並驗證 EC2Rescue for Windows Server。
- 匯入 PowerShell 模組以簡化您與工具的互動。
- 使用提供 RescueCmd 的命令和參數執行 EC2。

Systems Manager 執行命令文件接受三個參數：

- Command (命令)—EC2Rescue for Windows Server 動作。目前允許的值為：
  - ResetAccess 重設本機管理員密碼。會重設目前執行個體的本機管理員密碼，隨機產生的密碼則以 /EC2Rescue/Password/<INSTANCE\_ID> 安全存放在參數存放區中。如果您選取此動作但不提供任何參數，則密碼會使用預設的 KMS 金鑰自動加密。或者，您可在 Parameters (參數) 中指定 KMS 金鑰 ID，使用您自己的金鑰加密密碼。
  - CollectLogs 使用此動作執行適用於視窗伺服器的 EC2Rescue。/collect:all 如果您選取此動作，Parameters 必須包含要將日誌上傳到該處的 Amazon S3 儲存貯體名稱。

- FixAll使用此動作執行適用於視窗伺服器的 EC2Rescue。/rescue:all如果您選取此動作，Parameters 必須包含要救援的區塊型儲存設備名稱。
- 參數-要為指定指令傳遞的 PowerShell 參數。

### Note

為了使ResetAccess動作正常運作，您的 Amazon EC2 執行個體必須附加下列政策，才能將加密密碼寫入參數存放區。將此政策連接到相關 IAM 角色後，請等候數分鐘再嘗試重設執行個體的密碼。

使用預設 KMS 金鑰：

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ssm:PutParameter"
],
 "Resource": [
 "arn:aws:ssm:region:account_id:parameter/EC2Rescue/
 Passwords/<instanceid>"
]
 }
]
}
```

使用自訂 KMS 金鑰：

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ssm:PutParameter"
],
 "Resource": [
 "arn:aws:ssm:region:account_id:parameter/EC2Rescue/
 Passwords/<instanceid>"
]
 }
]
}
```

```
]
 },
 {
 "Effect": "Allow",
 "Action": [
 "kms:Encrypt"
],
 "Resource": [
 "arn:aws:kms:region:account_id:key/<kmskeyid>"
]
 }
]
```

下列程序說明如何在 Amazon EC2 主控台中檢視本文件的 JSON。

#### 檢視 Systems Manager 執行命令文件的 JSON

1. 開啟位於 <https://console.aws.amazon.com/systems-manager/home> 的 Systems Manager 主控台。
2. 在導覽窗格中，展開 Shared Services (系統管理員共享服務) 並選擇 Documents (文件)。
3. 在搜尋列中，將 Owner (擁有者) 設定為 Owned by Me or Amazon (由我或 Amazon 所有) 並將 Document name prefix (文件名稱前綴) 設定為 AWSSupport-RunEC2RescueForWindowsTool。
4. 選取 AWSSupport-RunEC2RescueForWindowsTool 文件，然後選擇 Contents (內容) 並檢視 JSON。

#### 範例

以下為如何使用 AWS CLI 以 Systems Manager 執行命令文件來執行 EC2Rescue for Windows Server 的範例。若要取得有關使用傳送指令的更多資訊 AWS CLI，請參閱《指[AWS CLI 令參考](#)》。

嘗試修正離線根磁碟區內的所有已識別問題

嘗試修正在連線到 Amazon EC2 Windows 執行個體之離線根磁碟區中找到的問題：



```
aws ssm send-command --instance-ids "i-0cb2b964d3e14fd9f" --document-name "AWSSupport-RunEC2RescueForWindowsTool" --comment "EC2Rescue offline volume xvdf" --parameters "Command=FixAll, Parameters='xvdf'" --output text
```

### 收集目前 Amazon EC2 Windows 執行個體的日誌

收集目前線上 Amazon EC2 Windows 執行個體的所有日誌，然後將它們上傳至 Amazon S3 儲存貯體：

```
aws ssm send-command --instance-ids "i-0cb2b964d3e14fd9f" --document-name "AWSSupport-RunEC2RescueForWindowsTool" --comment "EC2Rescue online log collection to S3" --parameters "Command=CollectLogs, Parameters='YOURS3BUCKETNAME'" --output text
```

### 收集離線 Amazon EC2 Windows 執行個體磁碟區的日誌

收集連線到 Amazon EC2 Windows 執行個體之離線磁碟區的所有日誌，然後將它們上傳至有預先簽章 URL 的 Amazon S3：

```
aws ssm send-command --instance-ids "i-0cb2b964d3e14fd9f" --document-name "AWSSupport-RunEC2RescueForWindowsTool" --comment "EC2Rescue offline log collection to S3" --parameters "Command=CollectLogs, Parameters=\"-Offline -BlockDeviceName xvdf -S3PreSignedUrl 'YOURS3PRESIGNEDURL'\"" --output text
```

### 重設本機管理員密碼

下列範例示範您可用以重設本機管理員密碼的方法。輸出會提供參數存放區連結，您可在此找到隨機產生的安全密碼，然後以本機管理員身分使用您的 Amazon EC2 Windows 執行個體 RDP。

使用預設的 AWS KMS key 別名/aws/ssm 重設線上執行個體的本機管理員密碼：

```
aws ssm send-command --instance-ids "i-0cb2b964d3e14fd9f" --document-name "AWSSupport-RunEC2RescueForWindowsTool" --comment "EC2Rescue online password reset" --parameters "Command=ResetAccess" --output text
```

使用 KMS 金鑰 重設線上執行個體的本機管理員密碼：

```
aws ssm send-command --instance-ids "i-0cb2b964d3e14fd9f" --document-name "AWSSupport-RunEC2RescueForWindowsTool" --comment "EC2Rescue online password reset" --parameters "Command=ResetAccess, Parameters=a133dc3c-a2g4-4fc6-a873-6c0720104bf0" --output text
```

**Note**

在此範例中，KMS 金鑰 為 `a133dc3c-a2g4-4fc6-a873-6c0720104bf0`。

## 亞馬遜 EC2 執行個體的 Amazon EC2 序列主控台

使用 EC2 序列主控台，您可以存取 Amazon EC2 執行個體的序列埠，您可以使用此連接埠來疑難排解開機、網路設定和其他問題。序列主控台不需要您的執行個體具有任何網路功能。使用序列主控台，您可以為執行個體輸入命令，就好像鍵盤和監視器直接連接到執行個體的序列埠一樣。序列主控台工作階段在執行個體重新開機和停止期間會持續。在重新開機期間，您可以從一開始檢視所有的開機訊息。

依預設，無法存取序列主控台。您的組織必須授與對序列主控台的帳戶存取權，並設定 IAM 政策，以授與使用者對序列主控台的存取權限。可透過使用執行個體 ID、資源標籤和其他 IAM 控制桿，以細微層級控制序列主控台存取。如需詳細資訊，請參閱 [設定對 EC2 序列主控台的存取](#)。

您可以透過使用 EC2 主控台或 AWS CLI 存取序列主控台。

序列主控台可供免費使用。

### 主題

- [必要條件](#)
- [設定對 EC2 序列主控台的存取](#)
- [連線至 EC2 序列主控台](#)
- [終止與 EC2 序列主控台的連線](#)
- [使用 EC2 序列主控台對 Amazon EC2 執行個體進行疑難排解](#)

## 必要條件

若要連線 EC2 序列主控台和使用所選工具來進行疑難排解，必須具備下列必要條件：

- [AWS 區域](#)
- [Wavelength 區和 AWS Outposts](#)
- [本機區域](#)
- [執行個體類型](#)
- [授與存取權](#)

- [支援瀏覽器型用戶端](#)
- [執行個體狀態](#)
- [Amazon EC2 Systems Manager](#)
- [SSH 伺服器](#)
- [設定所選的疑難排解工具](#)

## AWS 區域

AWS 區域 除加拿大西部 ( 卡爾加里 ) 以外的所有支持。

## Wavelength 區和 AWS Outposts

不支援。

## 本機區域

所有本地區域皆支援。

## 執行個體類型

支援的執行個體類型：

- Linux
  - 在 Nitro 系統上建置的所有虛擬化執行個體。
  - 所有裸機執行個體，除了：
    - 一般用途：a1.metal、mac1.metal、mac2.metal
    - 加速運算：g5g.metal
    - 記憶體最佳化：u-6tb1.metal、u-9tb1.metal、u-12tb1.metal、u-18tb1.metal、u-24tb1.metal
- Windows

在 Nitro 系統上建置的所有虛擬化執行個體。在裸機執行個體上不受支援。

## 授與存取權

您必須完成設定工作，才能向 EC2 序列主控台授與存取權。如需詳細資訊，請參閱 [設定對 EC2 序列主控台的存取](#)。

## 支援瀏覽器型用戶端

若要使用以瀏覽器為基礎的用戶端連線到序列主控台，您的瀏覽器必須支援 WebSocket。如果您的瀏覽器不支援 WebSocket，請[使用您自己的金鑰和 SSH 用戶端連線到序列主控台。](#)

## 執行個體狀態

必須為 running。

若執行個體處於 pending、stopping、stopped、shutting-down，或 terminated 狀態，將無法連線到序列主控台。

如需執行個體狀態的詳細資訊，請參閱 [執行個體生命週期](#)。

## Amazon EC2 Systems Manager

如果執行個體使用 Amazon EC2 Systems Manager，則必須在執行個體上安裝 SSM Agent 3.0.854.0 版或更新版本。如需有關 SSM Agent 的資訊，請參閱 AWS Systems Manager 使用者指南中的[使用 SSM Agent](#)。

## SSHD 伺服器

您不需要在執行個體上安裝或執行 sshd 伺服器。

## 設定所選的疑難排解工具

### Linux 執行個體

若要透過序列主控台疑難排解 Linux 執行個體的問題，您可以使用 GRUB 或 SysRq。在使用這些工具之前，您必須先對要使用工具所在的每個執行個體執行設定步驟。

### 工具

- [設定 GRUB](#)
- [配置 SysRq](#)

### 設定 GRUB

您必須先將執行個體設定為透過序列主控台使用 GRUB，才能透過序列主控台使用 GRUB。

若要設定 GRUB，請根據用來啟動執行個體的 AMI 選擇下列其中一個程序。

## Amazon Linux 2

在 Amazon Linux 2 執行個體上設定 GRUB

1. [連接至您的 Linux 執行個體](#)
2. 在 `/etc/default/grub` 中新增或變更下列選項：
  - 設定 `GRUB_TIMEOUT=1`。
  - 新增 `GRUB_TERMINAL="console serial"`。
  - 新增 `GRUB_SERIAL_COMMAND="serial --speed=115200"`。

以下是 `/etc/default/grub` 的範例。您可能需要根據系統設定來變更組態。

```
GRUB_CMDLINE_LINUX_DEFAULT="console=tty0 console=ttyS0,115200n8 net.ifnames=0
 biosdevname=0 nvme_core.io_timeout=4294967295 rd.emergency=poweroff rd.shell=0"
GRUB_TIMEOUT=1
GRUB_DISABLE_RECOVERY="true"
GRUB_TERMINAL="console serial"
GRUB_SERIAL_COMMAND="serial --speed=115200"
```

3. 執行下列命令來套用更新的組態。

```
[ec2-user ~]$ sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

## Ubuntu

在 Ubuntu 執行個體上設定 GRUB

1. [連線到您的執行個體](#)。
2. 在 `/etc/default/grub.d/50-cloudimg-settings.cfg` 中新增或變更下列選項：
  - 設定 `GRUB_TIMEOUT=1`。
  - 新增 `GRUB_TIMEOUT_STYLE=menu`。
  - 新增 `GRUB_TERMINAL="console serial"`。
  - 移除 `GRUB_HIDDEN_TIMEOUT`。
  - 新增 `GRUB_SERIAL_COMMAND="serial --speed=115200"`。

以下是 `/etc/default/grub.d/50-cloudimg-settings.cfg` 的範例。您可能需要根據系統設定來變更組態。

```
Cloud Image specific Grub settings for Generic Cloud Images
CLOUD_IMG: This file was created/modified by the Cloud Image build process

Set the recordfail timeout
GRUB_RECORDFAIL_TIMEOUT=0

Do not wait on grub prompt
GRUB_TIMEOUT=1
GRUB_TIMEOUT_STYLE=menu

Set the default commandline
GRUB_CMDLINE_LINUX_DEFAULT="console=tty1 console=ttyS0
nvme_core.io_timeout=4294967295"

Set the grub console type
GRUB_TERMINAL="console serial"
GRUB_SERIAL_COMMAND="serial --speed 115200"
```

3. 執行下列命令來套用更新的組態。

```
[ec2-user ~]$ sudo update-grub
```

## RHEL

在 RHEL 執行個體上設定 GRUB

1. [連線到您的執行個體](#)。
2. 在 `/etc/default/grub` 中新增或變更下列選項：
  - 移除 `GRUB_TERMINAL_OUTPUT`。
  - 新增 `GRUB_TERMINAL="console serial"`。
  - 新增 `GRUB_SERIAL_COMMAND="serial --speed=115200"`。

以下是 `/etc/default/grub` 的範例。您可能需要根據系統設定來變更組態。

```
GRUB_TIMEOUT=1
GRUB_DISTRIBUTOR="$(sed 's, release .*$,,g' /etc/system-release)"
GRUB_DEFAULT=saved
GRUB_DISABLE_SUBMENU=true
GRUB_CMDLINE_LINUX="console=tty0 console=ttyS0,115200n8 net.ifnames=0
rd.blacklist=nouveau nvme_core.io_timeout=4294967295 crashkernel=auto"
GRUB_DISABLE_RECOVERY="true"
GRUB_ENABLE_BLSCFG=true
GRUB_TERMINAL="console serial"
GRUB_SERIAL_COMMAND="serial --speed=115200"
```

3. 執行下列命令來套用更新的組態。

```
[ec2-user ~]$ sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

## CentOS

對於使用 CentOS AMI 啟動的執行個體，預設會針對序列主控台設定 GRUB。

以下是 /etc/default/grub 的範例。您的組態可能會視您的系統設定而有所不同。

```
GRUB_TIMEOUT=1
GRUB_DISTRIBUTOR="$(sed 's, release .*$,,g' /etc/system-release)"
GRUB_DEFAULT=saved
GRUB_DISABLE_SUBMENU=true
GRUB_TERMINAL="serial console"
GRUB_SERIAL_COMMAND="serial --speed=115200"
GRUB_CMDLINE_LINUX="console=tty0 crashkernel=auto console=ttyS0,115200"
GRUB_DISABLE_RECOVERY="true"
```

## 配置 SysRq

若要設定 SysRq，請啟用目前開機週期的 SysRq 指令。若要使組態持續存在，您也可以為後續開機啟用這些 SysRq 指令。

啟用目前開機週期的所有 SysRq 指令

1. [連線到您的執行個體](#)。
2. 執行下列命令。

```
[ec2-user ~]$ sudo sysctl -w kernel.sysrq=1
```

**Note**

此設定會在下次重新開機時清除。

為後續啟動啟用所有 SysRq 指令

1. 建立文件 `/etc/sysctl.d/99-sysrq.conf` 並在您最喜歡的編輯器中開啟。

```
[ec2-user ~]$ sudo vi /etc/sysctl.d/99-sysrq.conf
```

2. 新增以下這一行。

```
kernel.sysrq=1
```

3. 重新啟動執行個體以套用變更。

```
[ec2-user ~]$ sudo reboot
```

4. 出現 login 提示時，輸入您[先前設定](#)的密碼型使用者的使用者名稱，然後按 Enter。
5. 出現 Password 提示時，輸入密碼，然後按 Enter。

## Windows 執行個體

若要透過序列主控台對 Windows 執行個體進行疑難排解，您可以使用特殊管理主控台 (SAC)。在能夠使用 SAC 之前，您必須先在每個要使用 SAC 的執行個體上啟用 SAC 和開機功能表。

### 啟用 SAC 和開機功能表

**Note**

如果您在執行個體上啟用 SAC，則依賴密碼擷取的 EC2 服務將無法從 Amazon EC2 主控台運作。Amazon EC2 上的 Windows 啟動代理程式 (EC2Config、EC2Launch v1 和 EC2Launch v2) 倚賴序列主控台執行各種任務。在執行個體上啟用 SAC 時，這些任務不會成功執行。如需



有關視窗在 Amazon EC2 啟動代理程式上的詳細資訊，請參閱[the section called “設定執行個體”](#)。如果您啟用 SAC，可以稍後停用它。如需詳細資訊，請參閱[停用 SAC 和開機選單](#)。

使用下列其中一種方法來啟用執行個體上的 SAC 和開機功能表。

## PowerShell

在 Windows 執行個體上啟用 SAC 和開機功能表

1. [Connect](#) 線至執行個體，並從提高權限的 PowerShell 命令列執行下列步驟。
2. 啟用 SAC。

```
bcdedit /ems '{current}' on
bcdedit /emssettings EMSPORT:1 EMSBAUDRATE:115200
```

3. 啟用開機功能表。

```
bcdedit /set '{bootmgr}' displaybootmenu yes
bcdedit /set '{bootmgr}' timeout 15
bcdedit /set '{bootmgr}' bootems yes
```

4. 透過重新啟動執行個體來套用更新的組態。

```
shutdown -r -t 0
```

## Command prompt

在 Windows 執行個體上啟用 SAC 和開機功能表

1. [連接](#)到您的執行個體，並從命令提示字元執行下列步驟。
2. 啟用 SAC。

```
bcdedit /ems {current} on
bcdedit /emssettings EMSPORT:1 EMSBAUDRATE:115200
```

3. 啟用開機功能表。

```
bcdedit /set {bootmgr} displaybootmenu yes
bcdedit /set {bootmgr} timeout 15
```

```
bcdedit /set {bootmgr} bootems yes
```

4. 透過重新啟動執行個體來套用更新的組態。

```
shutdown -r -t 0
```

## 設定對 EC2 序列主控台的存取

若要設定對序列主控台的存取權，您必須在帳戶層級授與序列主控台存取權，然後設定 IAM 政策以授予使用者存取權。對於 Linux 執行個體，您還必須在每個執行個體上設定以密碼為基礎的使用者，以便使用者可以使用序列主控台進行疑難排解。

在開始之前，請務必檢查[先決條件](#)。

### 主題

- [EC2 序列主控台的存取層級](#)
- [管理 EC2 序列主控台的帳戶存取](#)
- [設定 EC2 序列主控台存取的 IAM 政策](#)
- [在 Linux 執行個體上設定作業系統使用者密碼](#)

## EC2 序列主控台的存取層級

依預設，帳戶層級無法存取序列主控台。您需要在帳戶層級明確授予對序列主控台的存取權。如需詳細資訊，請參閱 [管理 EC2 序列主控台的帳戶存取](#)。

您可以使用服務控制政策 (SCP) 來允許對組織內序列主控台的存取。然後，您可以使用 IAM 政策控制存取，在使用者層級進行精細的存取控制。透過組合使用 SCP 和 IAM 政策，您擁有對序列主控台不同層級的存取控制。

### 組織層級

您可以使用服務控制政策 (SCP) 來允許組織內成員帳戶存取序列主控台。如需 SCP 的詳細資訊，請參閱 AWS Organizations 使用者指南中的[服務控制政策](#)。

### 執行個體層級

您可以使用 IAM PrincipalTag 和 ResourceTag 建構，並透過 ID 指定執行個體，來設定序列主控台存取政策。如需詳細資訊，請參閱 [設定 EC2 序列主控台存取的 IAM 政策](#)。

## 使用者層級

您可以設定 IAM 政策，以允許或拒絕指定的使用者將 SSH 公開金鑰推送至特定執行個體的序列主控台服務的權限，從而在使用者層級設定存取。如需詳細資訊，請參閱 [設定 EC2 序列主控台存取的 IAM 政策](#)。

### 作業系統層級 (僅限 Linux 執行個體)

您可以在客體作業系統層級設定使用者密碼。在某些使用案例下，這可讓您存取序列主控台。不過，若要監控日誌，您不需要以密碼為基礎的使用者。如需詳細資訊，請參閱 [在 Linux 執行個體上設定作業系統使用者密碼](#)。

## 管理 EC2 序列主控台的帳戶存取

依預設，帳戶層級無法存取序列主控台。您需要在帳戶層級明確授予對序列主控台的存取權。

### 主題

- [授予使用者管理帳戶存取的許可](#)
- [檢視序列主控台的帳戶存取狀態](#)
- [授與序列主控台的帳戶存取權](#)
- [拒絕帳戶存取序列主控台](#)

### 授予使用者管理帳戶存取的許可

若要允許使用者管理對 EC2 序列主控台的帳戶存取權，您必須授予他們所需的 IAM 許可。

下列政策會授與檢視帳戶狀態，以及允許和阻止帳戶存取 EC2 序列主控台的權限。

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:GetSerialConsoleAccessStatus",
 "ec2:EnableSerialConsoleAccess",
 "ec2:DisableSerialConsoleAccess"
],
 "Resource": "*"
 }
]
}
```

```
]
}
```

如需詳細資訊，請參閱《IAM 使用者指南》中的[建立 IAM 政策](#)。

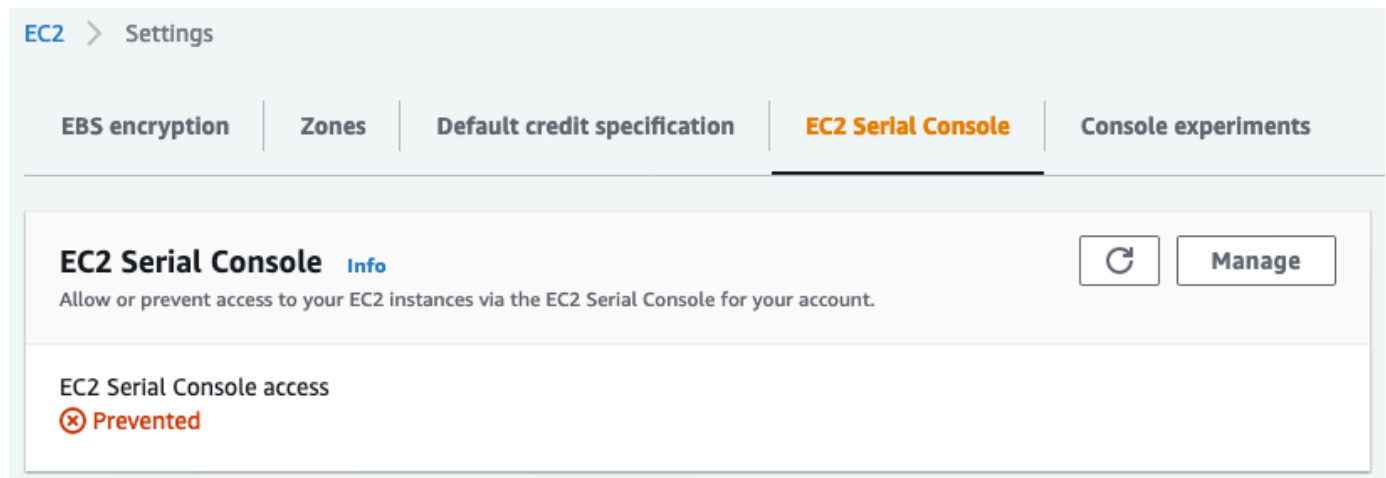
檢視序列主控台的帳戶存取狀態

檢視序列主控台 (主控台) 的帳戶存取狀態

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽窗格中，選擇 EC2 Dashboard (EC2 儀表板)。
3. 在 Account attributes (帳戶屬性) 中，選擇 EC2 Serial Console (EC2 序列主控台)。

EC2 Serial Console access (EC2 序列主控台存取) 欄位會指出是 Allowed (允許) 還是 Prevented (禁止) 帳戶存取。

下列螢幕擷取畫面顯示該帳戶被禁止使用 EC2 序列主控台。



檢視序列主控台的帳戶存取狀態 (AWS CLI)

使用 [get-serial-console-access-status](#) 命令來檢視序列主控台的帳戶存取狀態。

```
aws ec2 get-serial-console-access-status --region us-east-1
```

在下面的輸出中，true 表示允許該帳戶存取序列控制台。

```
{
 "SerialConsoleAccessEnabled": true
}
```

```
}
```

## 授與序列主控台的帳戶存取權

### 授與序列主控台 (主控台) 的帳戶存取權

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽窗格中，選擇 EC2 Dashboard (EC2 儀表板)。
3. 在 Account attributes (帳戶屬性) 中，選擇 EC2 Serial Console (EC2 序列主控台)。
4. 選擇 Manage (管理)。
5. 若要允許存取帳戶中所有執行個體的 EC2 序列主控台，請選取 Allow (允許) 核取方塊。
6. 選擇 Update (更新)。

### 授與帳戶存取序列主控台 (AWS CLI)

使用 [enable-serial-console-access](#) 命令，允許帳戶存取序列主控台。

```
aws ec2 enable-serial-console-access --region us-east-1
```

在下面的輸出中，true 表示允許該帳戶存取序列控制台。

```
{
 "SerialConsoleAccessEnabled": true
}
```

## 拒絕帳戶存取序列主控台

### 拒絕帳戶存取序列主控台 (主控台)

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在左側導覽窗格中，選擇 EC2 Dashboard (EC2 儀表板)。
3. 在 Account attributes (帳戶屬性) 中，選擇 EC2 Serial Console (EC2 序列主控台)。
4. 選擇 Manage (管理)。
5. 若要阻止存取帳戶中所有執行個體的 EC2 序列主控台，請清除 Allow (允許) 核取方塊。
6. 選擇 Update (更新)。

### 拒絕帳戶存取序列主控台 (AWS CLI)

使用 [disable-serial-console-access](#) 命令來阻止帳戶存取序列主控台。

```
aws ec2 disable-serial-console-access --region us-east-1
```

在下面的輸出中，false 表示該帳戶被拒絕存取序列控制台。

```
{
 "SerialConsoleAccessEnabled": false
}
```

## 設定 EC2 序列主控台存取的 IAM 政策

依預設，使用者無法存取序列主控台。您的組織必須設定 IAM 政策，以授予使用者所需的存取權。如需詳細資訊，請參閱《IAM 使用者指南》中的[建立 IAM 政策](#)。

若要存取序列主控台，請建立包含 `ec2-instance-connect:SendSerialConsoleSSHPublicKey` 動作的 JSON 政策文件。此動作會授予使用者將公有金鑰推送至序列主控台服務的許可，此服務會啟動序列主控台工作階段。建議限制對特定 EC2 執行個體的存取。否則，擁有此許可的所有使用者都可以連線到所有 EC2 執行個體的序列主控台。

### 範例 IAM 政策

- [明確允許存取序列主控台](#)
- [明確拒絕存取序列主控台](#)
- [使用資源標籤來控制對序列主控台的存取](#)

### 明確允許存取序列主控台

依預設，沒有人可以存取序列主控台。若要授與對序列主控台的存取權，您必須設定政策以明確允許存取。建議您設定限制存取特定執行個體的政策。

下列政策允許存取特定執行個體的序列主控台，並依其執行個體 ID 進行識別。

請注意，`DescribeInstances`、`DescribeInstanceTypes`、和 `GetSerialConsoleAccessStatus` 動作不支援資源層級許可，因此為這些動作指定所有資源，以 \* (星號) 表示。

```
{
 "Version": "2012-10-17",
 "Statement": [
```

```

 {
 "Sid": "AllowDescribeInstances",
 "Effect": "Allow",
 "Action": [
 "ec2:DescribeInstances",
 "ec2:DescribeInstanceTypes",
 "ec2:GetSerialConsoleAccessStatus"
],
 "Resource": "*"
 },
 {
 "Sid": "AllowinstanceBasedSerialConsoleAccess",
 "Effect": "Allow",
 "Action": [
 "ec2-instance-connect:SendSerialConsoleSSHPublicKey"
],
 "Resource": "arn:aws:ec2:region:account-id:instance/i-0598c7d356eba48d7"
 }
]
}

```

## 明確拒絕存取序列主控台

下列 IAM 政策允許存取所有執行個體的序列主控台 (以 \* (星號) 表示), 並明確拒絕存取特定執行個體的序列主控台 (依其 ID 進行識別)。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "AllowSerialConsoleAccess",
 "Effect": "Allow",
 "Action": [
 "ec2-instance-connect:SendSerialConsoleSSHPublicKey",
 "ec2:DescribeInstances",
 "ec2:DescribeInstanceTypes",
 "ec2:GetSerialConsoleAccessStatus"
],
 "Resource": "*"
 },
 {
 "Sid": "DenySerialConsoleAccess",
 "Effect": "Deny",

```

```

 "Action": [
 "ec2-instance-connect:SendSerialConsoleSSHPublicKey"
],
 "Resource": "arn:aws:ec2:region:account-id:instance/i-0598c7d356eba48d7"
 }
]
}

```

## 使用資源標籤來控制對序列主控台的存取

您可以使用資源標籤來控制對執行個體序列主控台的存取。

以屬性為基礎的存取控制是一種授權策略，根據可附加至使用者和 AWS 資源的標籤來定義權限。例如，下列政策允許使用者啟動執行個體的序列主控台連線，只有當該執行個體的資源標籤和主參與者的標籤具有相同的標籤鍵 `SerialConsole` 值時，才能啟動執行個體的序列主控台連線。

如需有關使用標籤來控制 AWS 資源存取權的詳細資訊，請參閱 IAM 使用者指南 [中的控制 AWS 資源存取](#)。

請注意，`DescribeInstances`、`DescribeInstanceTypes`、和 `GetSerialConsoleAccessStatus` 動作不支援資源層級許可，因此為這些動作指定所有資源，以 \* (星號) 表示。

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "AllowDescribeInstances",
 "Effect": "Allow",
 "Action": [
 "ec2:DescribeInstances",
 "ec2:DescribeInstanceTypes",
 "ec2:GetSerialConsoleAccessStatus"
],
 "Resource": "*"
 },
 {
 "Sid": "AllowTagBasedSerialConsoleAccess",
 "Effect": "Allow",
 "Action": [
 "ec2-instance-connect:SendSerialConsoleSSHPublicKey"
],
 "Resource": "arn:aws:ec2:region:account-id:instance/*",
 }
]
}

```



```
 "Condition": {
 "StringEquals": {
 "aws:ResourceTag/SerialConsole":
"${aws:PrincipalTag/SerialConsole}"
 }
 }
]
}
```

## 在 Linux 執行個體上設定作業系統使用者密碼

### Note

本節僅適用於 Linux 執行個體。

您無需密碼即可連線到序列主控台。不過，若要使用序列主控台對 Linux 執行個體進行疑難排解，執行個體必須有以密碼為基礎的 OS 使用者。

您可以為任何 OS 使用者 (包括根使用者) 設定密碼。請注意，根使用者可以修改所有檔案，而每個 OS 使用者的許可有限制。

您必須為每個要使用序列主控台的執行個體設定使用者密碼。這是每個執行個體都有的一次性要求。

### Note

下列指示僅適用於您使用 Linux AMI 所提供的 Linux AMI 啟動執行個體時，AWS 因為依預設，由 AWS 提供的 AMI 並未設定以密碼為基礎的使用者。如果您使用已設定根使用者密碼的 AMI 啟動執行個體，則可以略過這些指示。

若要在 Linux 執行個體上設定作業系統使用者密碼

1. [連線到您的執行個體](#)。您可以使用任何方法連線到執行個體，EC2 序列主控台連線方法除外。
2. 若要設定使用者的密碼，請使用 `passwd` 命令。在下列範例中，使用者是 `root`。

```
[ec2-user ~]$ sudo passwd root
```

下列為範例輸出。

```
Changing password for user root.
New password:
```

3. 出現 New password 提示時，輸入新密碼。
4. 出現提示時，重新輸入密碼。

## 連線至 EC2 序列主控台

您可以使用 Amazon EC2 主控台或透過 SSH 連線到 EC2 執行個體的序列控制台。連線到序列主控台後，您可以使用它來疑難排解開機、網路設定和其他問題。如需疑難排解的詳細資訊，請參閱 [使用 EC2 序列主控台對 Amazon EC2 執行個體進行疑難排解](#)。

### 考量事項

- 每個執行個體僅支援 1 個使用中的序列主控台連線。
- 除非您終止連線，否則序列主控台連線通常會持續 1 個小時。但是，在系統維護期間，Amazon EC2 將終止序列主控台工作階段的連線。
- 從序列主控台中斷連線後，需要 30 秒才能縮減工作階段，以便允許新的工作階段。
- 支援的序列主控台連接埠：ttyS0(Linux 執行個體) 和 COM1 (Windows 執行個體)
- 當您連線到序列主控台時，可能會發現執行個體的輸送量略有下降。

### 主題

- [使用以瀏覽器為基礎的用戶端連接](#)
- [使用您自己的金鑰和 SSH 用戶端來進行連接](#)
- [EC2 序列主控台端點和指紋](#)

## 使用以瀏覽器為基礎的用戶端連接

您可以使用以瀏覽器為基礎的用戶端連線到 EC2 執行個體的序列主控台。您可以在 Amazon EC2 主控台中選取執行個體並選擇連線到序列主控台來執行這項操作。以瀏覽器為基礎的用戶端會處理權限，並提供成功的連線。

EC2 序列控制台適用於大多數瀏覽器，並支援鍵盤和滑鼠輸入。

連線之前，請確認您已完成[事前準備](#)。

使用以瀏覽器為基礎的用戶端 (Amazon EC2 主控台) 連線到執行個體的序列埠

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇執行個體。
3. 請選取執行個體並選擇 Actions (動作)、Monitor and troubleshoot (監控和疑難排解)、EC2 Serial Console (EC2 序列主控台)、Connect (連線)。

或者，請選取執行個體並選擇 Connect (連線)、EC2 Serial Console (EC2 序列主控台)、Connect (連線)。

瀏覽器內的終端機視窗隨即開啟。

4. 按 Enter。如果傳回登入提示，表示您已連線到序列主控台。

如果螢幕保持黑色，您可以使用下列資訊來協助解決連線到序列主控台的問題：

- 檢查您是否已設定序列主控台的存取權。如需詳細資訊，請參閱 [設定對 EC2 序列主控台的存取](#)。
- (僅限 Linux 執行個體) 用 SysRq 於連線到序列主控台。SysRq 不需要您透過以瀏覽器為基礎的用戶端進行連線。如需詳細資訊，請參閱 [使用疑難排解 Linux 執行個體 SysRq](#)。
- (僅限 Linux 執行個體) 重新啟動裝置。如果您對執行個體有 SSH 存取權限，然後使用 SSH 連接到您的執行個體，並使用以下命令重新啟動 Getty。

```
[ec2-user ~]$ sudo systemctl restart serial-getty@ttyS0
```

- 將執行個體重新開機。您可以使用 SysRq (Linux 執行個體)、EC2 主控台或 AWS CLI。如需詳細資訊，請參閱 [使用疑難排解 Linux 執行個體 SysRq](#) (Linux 執行個體) 或 [重新啟動您的執行個體](#)。
5. (僅限 Linux 執行個體) 出現 **login** 提示時，輸入您 [先前設定](#) 之密碼型使用者的使用者名稱，然後按 Enter。
  6. (僅限 Linux 執行個體) 出現 Password 提示時，輸入密碼，然後按 Enter。

您現在已登入執行個體，並可使用序列主控台進行疑難排解。

## 使用您自己的金鑰和 SSH 用戶端來進行連接

當您使用序列主控台 API 時，可以使用自己的 SSH 金鑰和從您選擇的 SSH 用戶端連接執行個體。這可讓您從發佈公有的金鑰至執行個體的序列主控台功能受益。

連線之前，請確認您已完成[事前準備](#)。

使用 SSH 連線到執行個體的序列主控台

### 1. 將 SSH 公開金鑰推送到執行個體，以啟動序列控制台工作階段

使用 [send-serial-console-ssh-public-key](#) 命令將 SSH 公開金鑰推送至執行個體。這會啟動序列主控台工作階段。

如果此執行個體已經啟動序列主控台工作階段，則命令會失敗，因為您一次只能開啟一個工作階段。從序列主控台中斷連線後，需要 30 秒才能縮減工作階段，以便允許新的工作階段。

```
aws ec2-instance-connect send-serial-console-ssh-public-key \
 --instance-id i-001234a4bf70dec41EXAMPLE \
 --serial-port 0 \
 --ssh-public-key file://my_key.pub \
 --region us-east-1
```

### 2. 使用私有金鑰連線到序列主控台

在從序列主控台服務中移除公開金鑰之前，請使用 `ssh` 命令連線到序列主控台。在將其移除之前，您有 60 秒時間。

使用對應於公開金鑰的私有金鑰。

使用者名稱格式為 `instance-id.port0`，其中包含執行個體 ID 和連接埠 0。在下列範例中，使用者名為 `i-001234a4bf70dec41EXAMPLE.port0`。

每個區域的序列主控台服務端點各不相同。請參閱每個區域端點的 [EC2 序列主控台端點和指紋](#) 資料表。在下列範例中，序列主控台服務位於 `us-east-1` 區域。

```
ssh -i my_key i-001234a4bf70dec41EXAMPLE.port0@serial-console.ec2-instance-connect.us-east-1.aws
```

### 3. (選擇性) 驗證指紋

當您第一次連線到序列主控台時，系統會提示您確認指紋。您可以將序列主控台指紋與顯示進行驗證的指紋進行比較。如果這些指紋不符合，可能有人嘗試發動中間人攻擊。如果它們匹配，您可以放心地連接到序列主控台。

以下指紋適用於 `us-east-1` 區域的序列控制台服務。如需每個區域的指紋，請參閱 [EC2 序列主控台端點和指紋](#)。

```
SHA256:dXwn5ma/xadVMeBZGEru5l2gx+yI5LDiJaLUcz0FMmw
```

**Note**

指紋只會在您第一次連線到序列主控台時出現。

- 按 Enter。如果傳回提示，表示您已連線到序列主控台。

如果螢幕保持黑色，您可以使用下列資訊來協助解決連線到序列主控台的問題：

- 檢查您是否已設定序列主控台的存取權。如需詳細資訊，請參閱 [設定對 EC2 序列主控台的存取](#)。
- (僅限 Linux 執行個體) 用 SysRq 於連線到序列主控台。SysRq 不需要您透過 SSH 連線。如需詳細資訊，請參閱 [使用疑難排解 Linux 執行個體 SysRq](#)。
- (僅限 Linux 執行個體) 重新啟動裝置。如果您對執行個體有 SSH 存取權限，然後使用 SSH 連接到您的執行個體，並使用以下命令重新啟動 Getty。

```
[ec2-user ~]$ sudo systemctl restart serial-getty@ttyS0
```

- 將執行個體重新開機。您可以使用 SysRq (僅限 Linux 執行個體)、EC2 主控台或 AWS CLI。如需詳細資訊，請參閱 [使用疑難排解 Linux 執行個體 SysRq](#) (僅限 Linux 執行個體) 或 [重新啟動您的執行個體](#)。
- (僅限 Linux 執行個體) 出現 **login** 提示時，輸入您 [先前設定](#) 之密碼型使用者的使用者名稱，然後按 Enter。
  - (僅限 Linux 執行個體) 出現 Password 提示時，輸入密碼，然後按 Enter。

您現在已登入執行個體，並可使用序列主控台進行疑難排解。

## EC2 序列主控台端點和指紋

下列項目是 EC2 序列主控台的服務端點和指紋。若要以程式設計方式連線至執行個體的序列主控台，您可使用 EC2 序列主控台端點。每個 AWS 區域的 EC2 序列主控台端點和指紋都是獨一無二的。

區域名稱	區域	端點	指紋
美國東部 (俄亥俄)	us-east-2	serial-console.ec2-instance-connect.us-east-2.aws	SHA256 : Eh wPkTzRtTY 7TRZ26xbb0/高 V9JRM7m捷克0x瓦 特/D/0
美國東部 (維吉尼亞北部)	us-east-1	serial-console.ec2-instance-connect.us-east-1.aws	SHA256 : DX WN5MA/基5L2克X +YI5L 盧捷克 0 芬米 DiJa
美國西部 (加利佛尼亞北部)	us-west-1	serial-console.ec2-instance-connect.us-west-1.aws	SHA256:OH ldlcMET8u 7QLSX3jmR TRAPFHVtq byoLZBMUCqiH3Y
美國西部 (奧勒岡)	us-west-2	serial-console.ec2-instance-connect.us-west-2.aws	SHA256 : 梅西 23 比 6YG 德哈瓦 哈 1O2JVc TqKa HainqZc MwgNk
非洲 (開普敦)	af-south-1	ec2-serial-console.af-south-1.api.aws	SHA256 : RMWZ2 樓聚焦 5 日利 2K 少 量二十一百萬 VePe lgXsczo
亞太區域 (香港)	ap-east-1	ec2-serial-console.ap-east-1.api.aws	SHA256 : T0Q1LPI Z P7tkm2x 九百兆 焦 XxCho HplnAkb FsjYnifk
亞太區域 (海德拉巴)	ap-south-2	ec2-serial-console.ap-south-2.api.aws	SHA256 : 西班牙 白葡萄牙白俄羅斯 聯邦 4 號 + 作品

區域名稱	區域	端點	指紋
			YJ15DVW845 吉 ValoewAu 高清頻道
亞太區域 (雅加達)	ap-southeast-3	ec2-serial-console.ap-southeast-3.api.aws	SHA256:5 +LFNS32 基/4O0zifBx4 bzgsyfgy3O8mik ZwgrCh
亞太區域 (墨爾本)	ap-southeast-4	ec2-serial-console.ap-southeast-4.api.aws	SHA256 : 阿瓦 克 27 小時 5 克茨 什 0O7H90p0G 46WFET6ZJVM FgLvjn
亞太區域 (孟買)	ap-south-1	serial-console.ec2-instance-connect.ap-south-1.aws	SHA256: XcYmklq OBL 上半葉糖尿病 H8ISO51 瑞士皮斯姆 35bSU40 RxEg
亞太區域 (大阪)	ap-northeast-3	ec2-serial-console.ap-northeast-3.api.aws	SHA256 : Am0/jibK 9 阿克BnBuFnHr斯格 8 圖/VHFX /三英國
亞太區域 (首爾)	ap-northeast-2	serial-console.ec2-instance-connect.ap-northeast-2.aws	SHA256:FoqWXNX +DZ++GuNTztg9 PK49WYMqBX +FrcZM2dSrql
亞太區域 (新加坡)	ap-southeast-1	serial-console.ec2-instance-connect.ap-southeast-1.aws	SHA256: PLFN7WCDH X3QMwLu1G/ O8TUX7L C6L45Coy QgZua

區域名稱	區域	端點	指紋
亞太區域 (悉尼)	ap-southeast-2	serial-console.ec2-instance-connect.ap-southeast-2.aws	SHA256: Y 英国 9 瑞典克洛克祖 + CW9/VSE984CF5TGZ4 FvMw
亞太區域 (東京)	ap-northeast-1	序列控制台 .ec2 實例連接 .ap-東北 -1.aws	SHA256: RQFS 直流中央控制器 T9EM / HMRFQ + 克里奧特 5um4K OfQawew
加拿大 (中部)	ca-central-1	serial-console.ec2-instance-connect.ca-central-1.aws	SHA256 : 第二季 O6 ZwmpMwkp O6YW738 菲奧斯杜 2 吉姆莫斯 4 TyEv
中國 (北京)	cn-north-1	ec2-serial-console .cn-north-1.api.amazonwebservices.com.cn	SHA256 : 2G HVFY4H7U3+ 華富克 MeqjvSlngpg斯 28 / GGGG+Y
中國 (寧夏)	cn-northwest-1	ec2-serial-console .cn-northwest-1.api.amazonwebservices.com.cn	SHA256 : TDGrnkiQ 葉布霍 4 OdVf 斯祖阿 09V5 旅館
歐洲 (法蘭克福)	eu-central-1	serial-console.ec2-instance-connect.eu-central-1.aws	SHA256 : 阿姆夫斯/Y OL8 安茲 1 腳趾 + IcOd OIkXv 西班牙
歐洲 (愛爾蘭)	eu-west-1	serial-console.ec2-instance-connect.eu-west-1.aws	SHA256 : H2 AGAWO4 哈赫特姆 6 埃茲 3BJ7 烏季克西 2Q 扎瓦庫 6e TrHj



區域名稱	區域	端點	指紋
歐洲 (倫敦)	eu-west-2	serial-console.ec2-instance-connect.eu-west-2.aws	SHA256RnJg: A69RD5CE/AEG4Amm53I6L1ZPVS/BCV3TPW2 8
歐洲 (米蘭)	eu-south-1	ec2-serial-console.eu-south-1.api.aws	SHA256 : LC0KOV BVRXN0A7n99-乙 JnpgFy基葡萄糖 X95cuus7QK30
歐洲 (巴黎)	eu-west-3	serial-console.ec2-instance-connect.eu-west-3.aws	SHA256:q8ldnAf9pym eNe8BnFVngY3RPAr/kxswJUzfrlxeEWs
歐洲 (西班牙)	eu-south-2	ec2-serial-console.eu-south-2.api.aws	SHA256: GOCW2FLO669Q 电子游戏 6fzuz/4F4 N7T45 NxqFx ZcwoEc
歐洲 (斯德哥爾摩)	eu-north-1	serial-console.ec2-instance-connect.eu-north-1.aws	SHA256: TKG 福州 GSS3 立方 8 印度 6W2U32 EPNPKFKFK DvocDi
歐洲 (蘇黎世)	eu-central-2	ec2-serial-console.eu-central-2.api.aws	SHA256:8PX2MMF6 0N M4/ 4 瓦克富 WdCw特 QXWP6MK UlzKfw IfRz
以色列 (特拉維夫)	il-central-1	ec2-序列控制台中央-1.api.aws	SHA256 : JR 6Q8V6KNP8 + QSFQ4DJ5 NvtYy 調光譜分析儀 WGSM1S U

區域名稱	區域	端點	指紋
中東 (巴林)	me-south-1	ec2-serial-console .me-south-1.api.aws	SHA256 : NP JLLKHU2 UQ2 千 伏索克 5 倍周捷克 CBZCQC3K8 QnLd
中東 (阿拉伯聯合大公 國)	me-central-1	ec2-serial-console .me-central-1.api.aws	SHA256 : ZPB5 杜 FwPeyyk基布茲 + L0D B4比希/XZXn efSDKBVLE
南美洲 (聖保羅)	sa-east-1	序列控制台 .ec2 實例 連接 .sa-東 1.aws	SHA256NaQz: RD2+/32 乌克兰 1 伊 维埃姆 C+博特比赫 62OQDi
AWS GovCloud (美國 東部)	us-gov-east-1	serial-console.ec2 -instance-connect. us-gov-east-1.amaz onaws.com	SHA256 : 泰瑞十 九lkqnDc日大豆立 法會 38 年 + DH ZnmtebVF28
AWS GovCloud (美國 西部)	us-gov-west-1	serial-console.ec2 -instance-connect. us-gov-west-1.amaz onaws.com	SHA256: 嘉道理農 場植物園 + 紫外線 3 倍 F8 8NGO2YZLQX 5DQ OIPf Zilw

## 終止與 EC2 序列主控台的連線

如果您不再需要連線至執行個體的 EC2 序列主控台，就可以終止其連線。當您終止與序列主控台的連線時，執行個體上執行的任何 shell 工作階段都會繼續執行。如果想要結束 shell 工作階段，您必須先將其結束，然後才能終止與序列主控台的連線。

### 考量事項

- 除非您終止連線，否則序列主控台連線通常會持續 1 個小時。但是，在系統維護期間，Amazon EC2 將終止序列主控台工作階段的連線。
- 從序列主控台中斷連線後，需要 30 秒才能縮減工作階段，以便允許新的工作階段。

終止與序列主控台連線的方式取決於用戶端。

### 瀏覽器型用戶端

若要終止與序列主控台的連線，請關閉序列主控台的瀏覽器內終端機視窗。

### 標準開啟 SSH 用戶端

若要終止與序列主控台的連線，請使用下列命令關閉 SSH 連線。必須在新行之後立即執行此命令。

```
~.
```

根據您使用的 SSH 用戶端，您用來關閉 SSH 連線的命令可能會有所不同。

## 使用 EC2 序列主控台對 Amazon EC2 執行個體進行疑難排解

透過使用 EC2 序列主控台，您可以連接到執行個體的序列埠來疑難排解開機、網路設定和其他問題。

### Note

開始之前，請確定您已完成[先決條件](#)。

## Linux 執行個體

### 主題

- [使用 GRUB 疑難排解 Linux 執行個體](#)
- [使用疑難排解 Linux 執行個體 SysRq](#)

### 使用 GRUB 疑難排解 Linux 執行個體

GNU GRUB (GNU Grand 統一開機載入程式的簡稱，通常稱為 GRUB) 是大多數 Linux 作業系統的預設開機載入程式。從 GRUB 功能表中，您可以選擇要開機的核心，或修改功能表項目以變更核心的開機方式。這在疑難排解失敗的執行個體時非常有用。

GRUB 功能表會在開機過程中顯示。該功能表無法透過一般 SSH 存取，但您可以透過 EC2 序列主控台存取。

## Single user mode

單一使用者模式會以較低的執行層級啟動核心。例如，它可能掛載檔案系統，但不會啟動網路，讓您有機會執行修復執行個體所需的維護。

### 開機至單一使用者模式

1. [連線](#)到執行個體的序列主控台。
2. 使用下列命令重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

3. 在重新開機期間，當 GRUB 功能表出現時，按任意鍵停止啟動過程。
4. 在 GRUB 功能表中，使用方向鍵選取要開機的核心，然後按鍵盤上的 e。
5. 使用方向鍵將游標定位在包含核心的行上。該行以 linux 或 linux16 開頭，取決於用於啟動執行個體的 AMI。對於 Ubuntu，兩行以 linux 開頭，必須在下一個步驟中修改這兩行。
6. 在該行的末尾，添加單詞 single。

以下是 Amazon Linux 2 的範例。

```
linux /boot/vmlinuz-4.14.193-149.317.amzn2.aarch64 root=UUID=d33f9c9a-\
dadd-4499-938d-ebbf42c3e499 ro console=tty0 console=ttyS0,115200n8 net.ifname\
s=0 biosdevname=0 nvme_core.io_timeout=4294967295 rd.emergency=poweroff rd.she\
ll=0 single
```

7. 按 Ctrl+X 以開機進入單一使用者模式。
8. 出現 login 提示時，輸入您[先前設定](#)的密碼型使用者的使用者名稱，然後按 Enter。
9. 出現 Password 提示時，輸入密碼，然後按 Enter。

## Emergency mode

緊急模式與單一使用者模式類似，不同之處在於核心以最低的執行層級執行。

若要開機進入緊急模式，請遵循與單一使用者模式相同的步驟，但是在步驟 6 中，請新增字詞 emergency 而非 single。

## 使用疑難排解 Linux 執行個體 SysRq

System Request (SysRq) 密鑰有時被稱為 SysRq 「魔術」，可以用來在 shell 之外直接向內核發送命令，無論內核在做什麼，內核都會響應。例如，如果執行個體已停止回應，您可以使用 SysRq 金鑰告知核心損毀或重新啟動。有關更多信息，請參見維基百科中的[魔術 SysRq 密鑰](#)。

您可以在 EC2 序列主控台瀏覽器型用戶端或 SSH 用戶端中使用 SysRq 命令。每個用戶端傳送中斷請求的命令是不同的。

若要使用 SysRq，請根據您使用的用戶端選擇下列其中一個程序。

### Browser-based client

SysRq 在序列主控台瀏覽器型用戶端中使用

1. [連線](#)到執行個體的序列主控台。
2. 若要傳送中斷請求，請按 CTRL+0 (零)。如果您的鍵盤支援，您也可以使用「暫停」或「Break」鍵來傳送中斷請求。

```
[ec2-user ~]$ CTRL+0
```

3. 若要發出 SysRq 指令，請按鍵盤上與所需指令相對應的按鍵。例如，若要顯示指 SysRq 令清單，請按h。

```
[ec2-user ~]$ h
```

h 命令會輸出類似下列內容。

```
[1169.389495] sysrq: HELP : loglevel(0-9) reboot(b) crash(c) terminate-all-tasks(e) memory-full-oom-kill(f) kill-all-tasks(i) thaw-filestems(j) sak(k) show-backtrace-all-active-cpus(l) show-memory-usage(m) nice-all-RT-tasks(n) poweroff(o) show-registers(p) show-all-timers(q) unraw(r) sync(s) show-task-states(t) unmount(u) show-blocked-tasks(w) dump-ftrace-buffer(z)
```

### SSH client

若要 SysRq 在 SSH 用戶端中使用

1. [連線](#)到執行個體的序列主控台。

- 若要傳送中斷請求，請按下 ~B (波狀符號，後面接著大寫 B)。

```
[ec2-user ~]$ ~B
```

- 若要發出 SysRq 指令，請按鍵盤上與所需指令相對應的按鍵。例如，若要顯示指 SysRq 令清單，請按 h。

```
[ec2-user ~]$ h
```

h 命令會輸出類似下列內容。

```
[1169.389495] sysrq: HELP : loglevel(0-9) reboot(b) crash(c) terminate-all-
tasks(e) memory-full-oom-kill(f) kill-all-tasks(i) thaw-filestystems
(j) sak(k) show-backtrace-all-active-cpus(l) show-memory-usage(m) nice-all-RT-
tasks(n) poweroff(o) show-registers(p) show-all-timers(q) unraw(r
) sync(s) show-task-states(t) unmount(u) show-blocked-tasks(w) dump-ftrace-
buffer(z)
```

#### Note

根據您使用的 SSH 用戶端，您用於傳送中斷請求的命令可能會有所不同。

## Windows 執行個體

### 使用 SAC 來疑難排解您的 Windows 執行個體

Windows 的特殊管理主控台 (SAC) 功能提供了一種疑難排解 Windows 執行個體的方法。透過連線到執行個體的序列主控台並使用 SAC，您可以中斷開機程序並以安全模式啟動 Windows。

#### Note

如果您在執行個體上啟用 SAC，則依賴密碼擷取的 EC2 服務將無法從 Amazon EC2 主控台運作。Amazon EC2 上的 Windows 啟動代理程式 (EC2Config、EC2Launch v1 和 EC2Launch v2) 倚賴序列主控台執行各種任務。在執行個體上啟用 SAC 時，這些任務不會成功執行。如需有關視窗在 Amazon EC2 啟動代理程式上的詳細資訊，請參閱 [the section called “設定執行個體”](#)。如果您啟用 SAC，可以稍後停用它。如需詳細資訊，請參閱 [停用 SAC 和開機選單](#)。

## 主題

- [使用 SAC](#)
- [使用開機功能表](#)
- [停用 SAC 和開機選單](#)

## 使用 SAC

## 使用 SAC

### 1. [連接至序列主控台。](#)

如果執行個體上已啟用 SAC，序列主控台會顯示 SAC> 提示。

```
Computer is booting, SAC started and initialized.

Use the "ch -?" command for information about using channels.
Use the "?" command for general help.

SAC>?
EVENT: The CMD command is now available.
SAC_
```

### 2. 若要顯示 SAC 命令，請輸入？，然後按 Enter。

#### 預期的輸出結果

```
SAC>?
ch Channel management commands. Use ch -? for more help.
cmd Create a Command Prompt channel.
d Dump the current kernel log.
f Toggle detailed or abbreviated tlist info.
? or help Display this list.
i List all IP network numbers and their IP addresses.
i <#> <ip> <subnet> <gateway> Set IPv4 addr., subnet and gateway.
id Display the computer identification information.
k <pid> Kill the given process.
l <pid> Lower the priority of a process to the lowest possible.
lock Lock access to Command Prompt channels.
m <pid> <MB-allow> Limit the memory usage of a process to <MB-allow>.
p Toggle paging the display.
r <pid> Raise the priority of a process by one.
s Display the current time and date (24 hour clock used).
s mm/dd/yyyy hh:mm Set the current time and date (24 hour clock used).
t Tlist.
restart Restart the system immediately.
shutdown Shutdown the system immediately.
crashdump Crash the system. You must have crash dump enabled.
```

- 若要建立命令提示通道 (例如 cmd0001 或 cmd0002)，請輸入 **cmd**，然後按 Enter。
- 若要檢視命令提示通道，請按 ESC，然後按 TAB。

#### 預期的輸出結果

```
Name: Cmd0001
Description: Command
Type: VT-UTF8
Channel GUID: ef9f20a0-1287-11eb-82b0-0e4ba51872e5
Application Type GUID: 63d02271-8aa4-11d5-bccf-00b0d014a2d0

Press <esc><tab> for next channel.
Press <esc><tab>0 to return to the SAC channel.
Use any other key to view this channel.
```

- 若要切換通道，請同時按 ESC+TAB+通道號碼。例如，若要切換至 cmd0002 頻道 (如果已建立)，請按 ESC+TAB+2。
- 輸入命令提示通道所需的憑證。

```
Please enter login credentials.
Username: Administrator
Domain : .
Password: *****
```

命令提示字元與您在桌面上取得的完整功能命令 Shell 相同，但是除了它，不允許讀取已經輸出的字元。

```
Microsoft Windows [Version 10.0.17763.1457]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>diskpart

Microsoft DiskPart version 10.0.17763.1

Copyright (C) Microsoft Corporation.
On computer: EC2AMAZ-ASR4SAI

DISKPART> list disk

 Disk ### Status Size Free Dyn Gpt
 ----- -
 Disk 0 Online 30 GB 0 B
 Disk 1 Online 46 GB 46 GB

DISKPART>
```



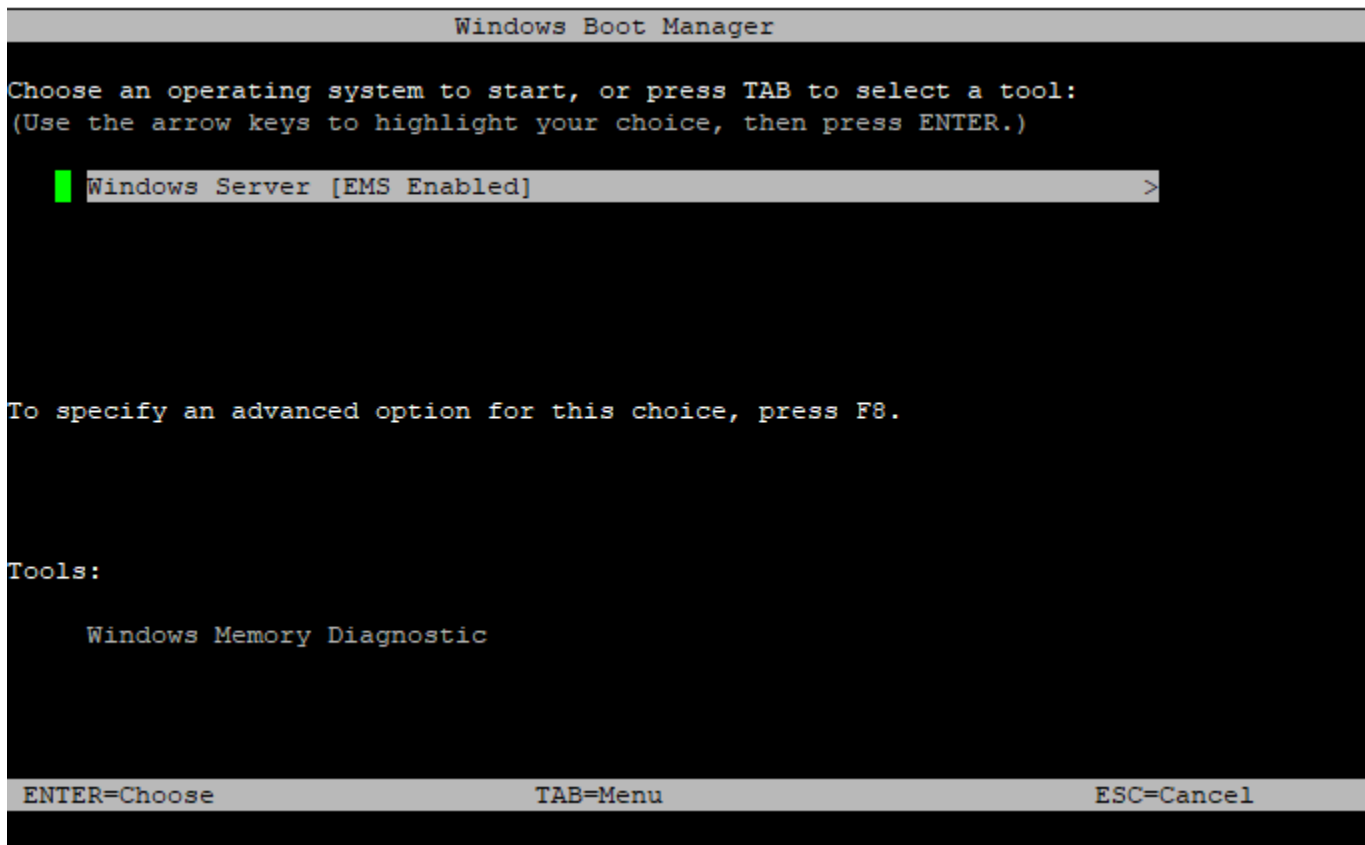
PowerShell 也可以從命令提示符使用。

請注意，您可能需要將進度偏好設定設定為無訊息模式。

```
PS C:\Windows\system32> $ProgressPreference="SilentlyContinue"
PS C:\Windows\system32> $computerInfo = Get-ComputerInfo
PS C:\Windows\system32> $computerInfo.Csprocessors[0].Name
Intel(R) Xeon(R) Platinum 8124M CPU @ 3.00GHz
PS C:\Windows\system32> $computerInfo.Csprocessors[0].Description
Intel64 Family 6 Model 85 Stepping 4
PS C:\Windows\system32> _
```

## 使用開機功能表

如果執行個體啟用了開機功能表，且透過 SSH 連線後重新啟動，您應該會看到開機功能表，如下所示。



```
Windows Boot Manager

Choose an operating system to start, or press TAB to select a tool:
(Use the arrow keys to highlight your choice, then press ENTER.)

Windows Server [EMS Enabled] >

To specify an advanced option for this choice, press F8.

Tools:

Windows Memory Diagnostic

ENTER=Choose TAB=Menu ESC=Cancel
```

## 開機功能表命令

### ENTER

啟動選取的作業系統項目。

## TAB

切換至「工具」功能表。

## ESC

取消並重新啟動執行個體。

## ESC 後跟 8

相當於按下 F8。顯示所選項目的進階選項。

## ESC 鍵 + 向左箭頭

回到初始開始功能表。

### Note

單獨的 ESC 鍵不會帶您回到主功能表，因為 Windows 正在等待查看是否正在進行逸出序列。

```
Advanced Boot Options

Choose Advanced Options for: Windows Server
(Use the arrow keys to highlight your choice.)

Repair Your Computer

Safe Mode
Safe Mode with Networking
Safe Mode with Command Prompt

Enable Boot Logging
Enable low-resolution video
Last Known Good Configuration (advanced)
Debugging Mode
Disable automatic restart on system failure
Disable Driver Signature Enforcement
Disable Early Launch Anti-Malware Driver

Start Windows Normally

Description: View a list of system recovery tools you can use to repair
startup problems, run diagnostics, or restore your system.

ENTER=Choose ESC=Cancel
```

## 停用 SAC 和開機選單

如果您啟用 SAC 和開機選單，可以稍後停用這些功能。

使用下列其中一種方法來停用執行個體上的 SAC 和開機功能表。

### PowerShell

在 Windows 執行個體上停用 SAC 和開機功能表

1. [Connect](#) 線至執行個體，並從提高權限的 PowerShell 命令列執行下列步驟。
2. 首先透過將值變更為 no 來停用開機選單。

```
bcdedit /set '{bootmgr}' displaybootmenu no
```

3. 然後透過將值變更為 off 來停用 SAC。

```
bcdedit /ems '{current}' off
```

4. 透過重新啟動執行個體來套用更新的組態。

```
shutdown -r -t 0
```

### Command prompt

在 Windows 執行個體上停用 SAC 和開機功能表

1. [連接](#)到您的執行個體，並從命令提示字元執行下列步驟。
2. 首先透過將值變更為 no 來停用開機選單。

```
bcdedit /set {bootmgr} displaybootmenu no
```

3. 然後透過將值變更為 off 來停用 SAC。

```
bcdedit /ems {current} off
```

4. 透過重新啟動執行個體來套用更新的組態。

```
shutdown -r -t 0
```

## 傳送診斷中斷 (僅適用於進階使用者)

### Warning

診斷中斷專供進階使用者使用。不正確的使用可能會對執行個體造成負面影響。將診斷中斷傳送至執行個體可能會觸發執行個體當機並重新開機，這會導致資料遺失。

您可以傳送診斷中斷給無法連線或無回應的執行個體，以手動觸發 Linux 執行個體的核心錯誤，或針對 Windows 執行個體觸發停止錯誤 (通常稱為藍色畫面錯誤)。

### Linux 執行個體

發生核心錯誤時，Linux 作業系統通常會當機並重新開機。作業系統的具體行為取決於其組態。核心錯誤也可用來促使執行個體的作業系統核心執行任務，例如產生損毀傾印檔案。然後，您可以利用損毀傾印檔案中的資訊來分析根本原因，對執行個體進行除錯。作業系統會代表執行個體本身在本機產生損毀傾印資料。

### Windows 執行個體

一般而言，Windows 作業系統在停止錯誤發生時會當機並重新開機，但具體行為取決於其組態。停止錯誤也會導致作業系統將除錯資訊 (例如核心記憶體傾印) 寫入檔案。然後，您可以利用此資訊來分析根本原因，以進行執行個體的除錯。作業系統會代表執行個體本身在本機產生記憶體傾印資料。

將診斷中斷傳送至執行個體之前，建議您先參閱作業系統的說明文件，然後進行必要的組態變更。

### 內容

- [支援的執行個體類型](#)
- [必要條件](#)
- [傳送診斷中斷](#)

## 支援的執行個體類型

所有 Nitro 型執行個體類型都支援診斷中斷，但由 AWS Graviton 處理器驅動的執行個體類型除外。如需詳細資訊，請參閱以 [AWS Nitro 系統建置的執行個體](#) 和 [AWS 重力子](#)。

## 必要條件

使用診斷中斷之前，您必須設定執行個體的作業系統。這可確保它在發生核心錯誤 (Linux 執行個體) 或停止錯誤 (Windows 執行個體) 時執行您需要的動作。

### Linux 執行個體

設定 Amazon Linux 2 在核心錯誤發生時產生損毀傾印

1. 連線到您的執行個體。
2. 安裝 kexec 和 kdump。

```
[ec2-user ~]$ sudo yum install kexec-tools -y
```

3. 設定核心保留適當的記憶體數量給次要核心。該保留的記憶體數量取決於執行個體可用的記憶體總數。使用您偏好的文字編輯器開啟 `/etc/default/grub` 檔案，找出開頭為 `GRUB_CMDLINE_LINUX_DEFAULT` 的那一行，然後依下列格式新增 `crashkernel` 參數：`crashkernel=memory_to_reserve`。例如，若要保留 160MB，請如下修改 `grub` 檔案：

```
GRUB_CMDLINE_LINUX_DEFAULT="crashkernel=160M console=tty0 console=ttyS0,115200n8
net.ifnames=0 biosdevname=0 nvme_core.io_timeout=4294967295 rd.emergency=poweroff
rd.shell=0"
GRUB_TIMEOUT=0
GRUB_DISABLE_RECOVERY="true"
```

4. 儲存變更並關閉 `grub` 檔案。
5. 重建 GRUB2 組態檔案。

```
[ec2-user ~]$ sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

6. 在以 Intel 和 AMD 處理器為架構的執行個體上，`send-diagnostic-interrupt` 命令會將 unknown non-maskable interrupt (NMI) 傳送至執行個體。您必須將核心設定為收到不明 NMI 時損毀。使用您偏好的文字編輯器開啟 `/etc/sysctl.conf` 檔案，然後新增下列命令。

```
kernel.unknown_nmi_panic=1
```

7. 重新開機並重新連接您的執行個體。
8. 確認核心已使用正確的 `crashkernel` 參數開機。

```
$ grep crashkernel /proc/cmdline
```

下列輸出範例指出組態成功。

```
BOOT_IMAGE=/boot/vmlinuz-4.14.128-112.105.amzn2.x86_64 root=UUID=a1e1011e-
e38f-408e-878b-fed395b47ad6 ro crashkernel=160M console=tty0 console=ttyS0,115200n8
net.ifnames=0 biosdevname=0 nvme_core.io_timeout=4294967295 rd.emergency=poweroff
rd.shell=0
```

## 9. 確認 kdump 服務在執行中。

```
[ec2-user ~]$ systemctl status kdump.service
```

下列輸出範例顯示 kdump 服務在執行中的結果。

```
kdump.service - Crash recovery kernel arming
Loaded: loaded (/usr/lib/systemd/system/kdump.service; enabled; vendor preset:
enabled)
Active: active (exited) since Fri 2019-05-24 23:29:13 UTC; 22s ago
Process: 2503 ExecStart=/usr/bin/kdumpctl start (code=exited, status=0/SUCCESS)
Main PID: 2503 (code=exited, status=0/SUCCESS)
```

### Note

根據預設，損毀傾印檔案會儲存至 `/var/crash/`。若要變更位置，請使用您偏好的文字編輯器來修改 `/etc/kdump.conf` 檔案。

設定 Amazon Linux 在核心錯誤發生時產生損毀傾印

1. 連線到您的執行個體。
2. 安裝 `kexec` 和 `kdump`。

```
[ec2-user ~]$ sudo yum install kexec-tools -y
```

3. 設定核心保留適當的記憶體數量給次要核心。該保留的記憶體數量取決於執行個體可用的記憶體總數。

```
$ sudo grubby --args="crashkernel=memory_to_reserve" --update-kernel=ALL
```

例如，若要保留 160MB 給損毀核心，請使用下列命令。

```
$ sudo grubby --args="crashkernel=160M" --update-kernel=ALL
```

4. 在以 Intel 和 AMD 處理器為架構的執行個體上，`send-diagnostic-interrupt` 命令會將 unknown non-maskable interrupt (NMI) 傳送至執行個體。您必須將核心設定為收到不明 NMI 時損毀。使用您偏好的文字編輯器開啟 `/etc/sysctl.conf` 檔案，然後新增下列命令。

```
kernel.unknown_nmi_panic=1
```

5. 重新開機並重新連接您的執行個體。
6. 確認核心已使用正確的 `crashkernel` 參數開機。

```
$ grep crashkernel /proc/cmdline
```

下列輸出範例指出組態成功。

```
root=LABEL=/ console=tty1 console=ttyS0 selinux=0 nvme_core.io_timeout=4294967295
LANG=en_US.UTF-8 KEYTABLE=us crashkernel=160M
```

7. 確認 `kdump` 服務在執行中。

```
[ec2-user ~]$ sudo service kdump status
```

如果服務在執行中，命令會傳回 `Kdump is operational` 回應。

#### Note

根據預設，損毀傾印檔案會儲存至 `/var/crash/`。若要變更位置，請使用您偏好的文字編輯器來修改 `/etc/kdump.conf` 檔案。

設定 SUSE Linux Enterprise、Ubuntu 或 Red Hat Enterprise Linux

在以 Intel 和 AMD 處理器為架構的執行個體上，`send-diagnostic-interrupt` 命令會將 unknown non-maskable interrupt (NMI) 傳送至執行個體。您必須透過調整作業系統的設定檔，將核心設定為在收到未知 NMI 時當機。如需如何設定核心損毀的詳細資訊，請參閱您作業系統的說明文件：

- [SUSE Linux Enterprise](#)
- [Ubuntu](#)
- [Red Hat Enterprise Linux \(RHEL\)](#)

## Windows 執行個體

設定 Windows 在停止錯誤發生時產生記憶體傾印

1. 連線到您的執行個體。
2. 開啟控制台，選擇系統、進階系統設定。
3. 在系統內容對話方塊中，選擇進階索引標籤。
4. 在啟動及修復區段中，選擇設定...
5. 在系統失敗區段中，視需要進行設定，然後選擇確定。

如需有關設定 Windows 停止錯誤的詳細資訊，請參閱 [Windows 記憶體傾印檔案選項的概觀](#)。

## 傳送診斷中斷

完成必要的組態變更後，您可以使用 AWS CLI 或 Amazon EC2 API 將診斷中斷傳送到執行個體。

### AWS CLI

將診斷中斷傳送至執行個體 (AWS CLI)

使用 [send-diagnostic-interrupt](#) 命令並指定執行個體 ID。

```
aws ec2 send-diagnostic-interrupt --instance-id i-1234567890abcdef0
```

### PowerShell

將診斷中斷傳送至執行個體 (AWS Tools for Windows PowerShell)

使用指 [Send-EC2DiagnosticInterrupt](#) 令程式並指定執行個體識別碼。



```
PS C:\> Send-EC2DiagnosticInterrupt -InstanceId i-1234567890abcdef0
```

# 文件歷史記錄

下表說明從 2019 年開始，Amazon EC2 使用者指南的重要新增內容。我們也會經常更新指南，以解決您發送給我們的反饋。

變更	描述	日期
<a href="#">EC2 執行個體類型尋找器 — 其他參數</a>	EC2 執行個體類型尋找器現在提供其他參數，讓您為工作負載指定更詳細的需求。	2024年6月5日
<a href="#">U7 至 12 TB、U7 英吋 -16 TB、烏七合二十四 TB 和三十二 TB 的 U7 執行個體</a>	新增記憶體增強型執行個體類型，配備第 4 代 Intel Xeon 可擴充處理器	2024年5月28日
<a href="#">適用於 EC2 快速啟動的新受管政策</a>	已新增EC2FastLaunchFullAccess政策，以從執行個體執行與 EC2 快速啟動功能相關的 API 動作。	2024年5月14日
<a href="#">AMI 註銷註冊保護</a>	您可以在 AMI 上開啟取消註冊保護，以防止意外或惡意刪除。	2024年4月23日
<a href="#">PTP 硬體時鐘 — 執行個體類型支援</a>	PTP 硬體時鐘現在可在 C7a、C7i、M7a、M7g、M7i、R7a 和 R7i 執行個體類型上使用。	2024年4月22日
<a href="#">新增強型網路功能的 Nitro 效能考量</a>	本頁面著重於網路考量，以協助您針對以 Nitro 為基礎的 Amazon EC2 執行個體進行效能調整。	2024年4月4日
<a href="#">適用於已啟用 VSS 的 EBS 快照的新受管原則</a>	Amazon EC2 VSS 提供新的 IAM 受管政策，您可以將其新增到執行個體設定檔角色，以	2024年3月28日

	確保您的許可保留 up-to-date 並遵循最佳實務。	
<a href="#">PTP 硬體時鐘 — 美國東部 (維吉尼亞北部)</a>	PTP 硬體時鐘現已在美國東部 (維吉尼亞北部) 區域推出。	2024年3月26日
<a href="#">將 IMDSv2 設為帳戶預設值</a>	您可以將帳戶中的所有新 EC2 執行個體啟動設定為預設使用執行個體中繼資料服務版本 2 (IMDSv2)。	2024年3月25日
<a href="#">標記從快照建立的新 Linux AMI</a>	當您從快照建立 Linux AMI 時，您可以標記新的 AMI。	2024年3月7日
<a href="#">複製時標記新的 AMI 和快照</a>	複製 AMI 時，您可以使用相同的標籤來標記新的 AMI 和新快照，也可以使用不同的標籤來標記它們。	2024年3月7日
<a href="#">移除 AWS 管理組件頁面</a>	AWS 管理組件主要與視窗伺服器 2012 年及之前版本搭配使用。這些舊版作業系統平台版本已不再受支援。若要管理和疑難排解在內部部署上 AWS 執行的伺服器群組，請參閱 <a href="#">AWS Systems Manager Fleet Manager</a> 。	2024年2月12日
<a href="#">EC2 執行個體 Connect 已預先安裝</a>	EC2 實例 Connect 現在已經預裝在 macOS 索諾瑪 14.2.1 或更高版本, macOS 文圖拉 13.6.3 或更高版本, 和 macOS 蒙特雷 12.7.2 或更高版本 AMI。	2024年1月26日
<a href="#">適用於 CentOS、macOS 和 RHEL 的 EC2 Instance Connect 支援</a>	您現在可以在受支援的 CentOS、macOS 和 RHEL AMI 上安裝 EC2 Instance Connect。	2023 年 12 月 6 日

<a href="#">C7a、C7i、R7a、R7i 和 R7iz 的休眠支援</a>	對您新啟動且在 C7a、C7i、R7a、R7i 和 R7iz 執行個體類型上執行的執行個體進行休眠。	2023 年 12 月 1 日
<a href="#">Amazon Q EC2 執行個體類型選取器</a>	Amazon Q EC2 執行個體類型選取器會考量您的使用案例、工作負載類型和 CPU 製造商偏好設定，以及排定價格和效能優先順序的方式。然後，它會使用這些資料，針對您的新工作負載，提供指導和建議最適合的 Amazon EC2 執行個體類型。	2023 年 11 月 28 日
<a href="#">EC2 免費方案</a>	您可以從 EC2 儀表板追蹤 EC2 免費方案用量。	2023 年 11 月 26 日
<a href="#">Console-to-Code</a>	Console-to-Code 可協助您開始使用自動化程式碼。主控台到程式碼會記錄您的主控台動作，然後使用生成式 AI，以您偏好的基礎架構即程式碼格式建議程式碼。您可以使用該程式碼作為起點，對其進行自訂以使其針對特定使用案例準備好投入生產。	2023 年 11 月 26 日
<a href="#">可設定的閒置連線追蹤逾時</a>	處於閒置狀態的安全群組連線可能會導致連線追蹤耗盡，並造成不追蹤連線及捨棄封包。您現在可以在彈性網路介面上設定安全群組連線追蹤的逾時 (以秒為單位)。	2023 年 11 月 17 日

<a href="#">PTP 硬體時鐘</a>	支援的執行個體現在具有精確時間協定 (PTP) 硬體時鐘。PTP 硬體時鐘支援 NTP 或直接 PTP 連線。	2023 年 11 月 16 日
<a href="#">為已啟用休眠的執行個體變更執行個體類型</a>	您現在可以為已啟用休眠的執行個體變更執行個體類型 (處於 stopped 狀態的情況下)。	2023 年 11 月 16 日
<a href="#">執行個體拓撲</a>	您可以使用 DescribeInstanceTopology API 偵測執行個體的位置，然後使用此資訊在實際上彼此較近的執行個體上執行 HPC 和 ML 任務，以最佳化 HPC 和 ML 任務。	2023 年 11 月 13 日
<a href="#">EC2 快速啟動共用 AMI 支援</a>	您現在可以在與您共用的 AMI 上啟用 EC2 快速啟動。當您在共用 AMI 上啟用 EC2 快速啟動時，會在您的帳戶中建立用於更快啟動的預先佈建快照。	2023 年 11 月 6 日
<a href="#">ML 的容量區塊</a>	現在您可以為日後預留 GPU 執行個體，以支援短時間的機器學習 (ML) 工作負載。	2023 年 10 月 31 日
<a href="#">Spot 執行個體休眠</a>	現在您可以讓 Spot 執行個體進入休眠，方法與隨需執行個體目前可使用的休眠體驗和執行個體系列相同。	2023 年 10 月 24 日
<a href="#">「封鎖 AMI 公有存取權」的預設設定</a>	預設情況下，所有新帳戶和沒有公開 AMI 的現有帳戶都會啟用「適用於 AMI 的封鎖公開存取」。	2023 年 10 月 20 日

<a href="#">Amazon EC2 全域檢視</a>	Amazon EC2 Global View 支援其他資源類型和可自訂的顯示選項。	2023 年 10 月 18 日
<a href="#">對 Ubuntu 22.04.2 LTS 的休眠支援 (Jammy Jellyfish)</a>	讓從 Ubuntu 22.04.2 LTS (Jammy Jellyfish) AMI 啟動的新啟動執行個體休眠。	2023 年 10 月 16 日
<a href="#">停用 AMI</a>	可以停用 AMI 以防止其用於執行個體啟動。	2023 年 10 月 12 日
<a href="#">附接的 EBS 狀態檢查</a>	您可以使用附接的 EBS 狀態檢查來監控附接至執行個體的 Amazon EBS 磁碟區是否可連線。	2023 年 10 月 11 日
<a href="#">對 Red Hat Enterprise Linux 9 的休眠支援</a>	讓從 Red Hat Enterprise Linux 9 AMI 啟動的新啟動執行個體休眠。	2023 年 10 月 2 日
<a href="#">對 Microsoft Windows Server 2022 的休眠支援</a>	讓從 Microsoft Windows Server 2022 AMI 啟動的新啟動執行個體休眠。	2023 年 10 月 2 日
<a href="#">AL2023 支援休眠</a>	讓從 AL2023 AMI 啟動的新啟動執行個體休眠。	2023 年 10 月 2 日
<a href="#">啟動 Spot 機群中的 Spot 執行個體中斷</a>	您可以在 Amazon EC2 主控台中選取 Spot 機群並啟動機群中的 Spot 執行個體中斷，以便您可以測試 Spot 執行個體上的應用程式如何處理中斷。	2023 年 9 月 21 日
<a href="#">封鎖 AMI 的公有存取權</a>	您可以在帳戶層級啟用封鎖 AMI 的公有存取權，以阻止任何公開 AMI 的嘗試。	2023 年 9 月 12 日

<a href="#">M7i 和 M7i-flex 休眠支援</a>	對您新啟動且在 M7i 和 M7i-flex 執行個體類型上執行的執行個體進行休眠。	2023 年 8 月 22 日
<a href="#">EC2-Classic 已遭棄用</a>	搭配 EC2-Classic，EC2 執行個體可在與其他客戶共用的單一平面網路中執行。Amazon VPC 取代 EC2-Classic。使用 Amazon VPC，您的執行個體可在邏輯上與您的 AWS 帳戶隔離的 Virtual Private Cloud (VPC) 中執行。	2023 年 8 月 8 日
<a href="#">專用主機</a>	您可以在 Outpost 上的特定硬體資產中分配專用執行個體。	2023 年 6 月 20 日
<a href="#">EC2 Instance Connect Endpoint</a>	現在可以透過 SSH 或 RDP 連線至執行個體，而不需要執行個體擁有公有 IPv4 地址。	2023 年 6 月 13 日
<a href="#">IMDS Package Analyzer</a>	您現在可以使用 IMDS Package Analyzer，識別 EC2 執行個體上的 IMDSv1 呼叫來源。	2023 年 6 月 1 日
<a href="#">EC2 序列主控台裸機執行個體</a>	EC2 序列主控台現在支援連線至所選裸機執行個體的序列埠。	2023 年 4 月 11 日
<a href="#">啟動範本配額</a>	您現在在 Service Quotas 主控台，或是使用 Service Quotas CLI，就能檢視啟動範本和啟動範本版本的配額。	2023 年 4 月 3 日
<a href="#">容量保留使用率通知</a>	AWS Health 現在，當您帳戶中「容量保留」的容量使用率降至 20% 以下時，會傳送通知。	2023 年 4 月 3 日

<a href="#">容量保留群組</a>	您現在可以將與您共用的容量保留新增到您擁有的容量保留群組。	2023 年 3 月 30 日
<a href="#">修改執行個體中繼資料選項</a>	您現在可使用 Amazon EC2 主控台修改執行個體中繼資料選項。	2023 年 3 月 20 日
<a href="#">就地 macOS 作業系統更新</a>	現在，您可以在 M1 Mac 執行個體上執行就地 Apple macOS 作業系統更新。	2023 年 3 月 14 日
<a href="#">UEFI preferred</a>	現在，您可以建立支援 Unified Extensible Firmware Interface (UEFI) 和 Legacy BIOS 開機模式的 AMI。	2023 年 3 月 3 日
<a href="#">修改適用於 IMDSv2 的 AMI</a>	修改現有 AMI，以便從 AMI 啟動的執行個體預設需要 IMDSv2。	2023 年 2 月 28 日
<a href="#">基於 Windows 虛擬化的安全性 - Credential Guard</a>	您可以在支援的 Amazon EC2 執行個體上啟用 Credential Guard，這是一個基於虛擬化的安全性 (VBS) 功能。	2023 年 1 月 31 日
<a href="#">啟動範本中的 AMI 別名</a>	您可以在啟動範本中指定 AWS Systems Manager 參數而不是 AMI ID，以避免每次 AMI ID 變更時都必須更新範本。	2023 年 1 月 19 日
<a href="#">C6i、I3en 和 M6i 休眠支援</a>	對您新啟動且在 C6i、I3en 和 M6i 執行個體類型上執行的執行個體進行休眠。	2022 年 12 月 19 日



<a href="#">預防未完成的寫入</a>	利用區塊儲存功能，預防未完成的寫入，提升 I/O 密集型關聯式資料庫工作負載的效能，並減少延遲，而不會對資料彈性產生負面影響。	2022 年 11 月 29 日
<a href="#">ENA Express</a>	使用 ENA Express 增加輸送量並將 EC2 執行個體之間的網路流量尾延遲降至最低。	2022 年 11 月 28 日
<a href="#">資源回收筒保留規則鎖定</a>	您可以鎖定保留規則，以協助保護規則免受意外或惡意的修改和刪除。	2022 年 11 月 23 日
<a href="#">複製 AMI 標籤</a>	複製 AMI 時，您可以同時複製使用者定義的 AMI 標籤。	2022 年 11 月 18 日
<a href="#">適用於存放和還原的 AMI 大小</a>	可存放至 Amazon S3 儲存貯體中和從 Amazon S3 儲存貯體還原的 AMI (壓縮之前) 大小現可達到 5,000 GB。	2022 年 11 月 16 日
<a href="#">priceCapacityOptimizedSpot 執行個體的配置策略</a>	使用 priceCapacityOptimized 分配策略的 Spot 機群會考慮價格和容量，來選取最不可能中斷且價格最低的 Spot 執行個體集區。	2022 年 11 月 10 日
<a href="#">price-capacity-optimizedSpot 執行個體的配置策略</a>	使用 price-capacity-optimized 分配策略的 EC2 機群會考慮價格和容量，來選取最不可能中斷且價格最低的 Spot 執行個體集區。	2022 年 11 月 10 日
<a href="#">取消與您的帳戶共用 AMI</a>	如果 AMI 已與您共享，AWS 帳戶而您不再希望該 AMI 與您的帳戶共享，則可以從 AMI 的啟動權限中刪除您的帳戶。	2022 年 11 月 4 日

<a href="#">轉移彈性 IP 地址</a>	您現在可以將彈性 IP 地址從一個地址傳輸 AWS 帳戶 到另一個地址。	2022 年 10 月 31 日
<a href="#">取代根磁碟區</a>	您可以使用 AMI 取代執行中執行個體的根 Amazon EBS 磁碟區。	2022 年 10 月 27 日
<a href="#">自動將執行個體連線至資料庫</a>	使用自動連線功能，將一個或多個 EC2 執行個體快速連線至 RDS 資料庫，以允許它們之間的流量。	2022 年 10 月 10 日
<a href="#">AMI 配額</a>	配額現在適用於建立及共用 AMI。	2022 年 10 月 10 日
<a href="#">設定適用於 IMDSv2 的 AMI</a>	設定 AMI，以便從 AMI 啟動的執行個體預設需要 IMDSv2。	2022 年 10 月 3 日
<a href="#">啟動 Spot 執行個體中斷</a>	您可以在 Amazon EC2 主控台中選取 Spot 執行個體並啟動中斷，以便您可以測試 Spot 執行個體上的應用程式如何處理中斷。	2022 年 9 月 26 日
<a href="#">已驗證的 AMI 供應商</a>	在 Amazon EC2 主控台中，Amazon 或已驗證的 Amazon 合作夥伴擁有的公用 AMI 標記為 Verified provider (已驗證的供應商)。	2022 年 7 月 22 日
<a href="#">放置群組於 AWS Outposts</a>	新增在 Outpost 上放置群組的主機分散策略。	2022 年 6 月 30 日

<a href="#">資源回收筒的條件索引鍵</a>	您可以使用 <code>rbn:Request/ResourceType</code> 和 <code>rbn:Attribute/ResourceType</code> 條件索引鍵，篩選資源回收筒請求的存取權。	2022 年 6 月 14 日
<a href="#">io2 Block Express 磁碟區</a>	您可以修改 io2 Block Express 磁碟區的大小和佈建 IOPS，並可將其啟用以進行快速快照還原。	2022 年 5 月 31 日
<a href="#">專用主機 AWS Outposts</a>	您可以配置 AWS Outposts 上的專用執行個體。	2022 年 5 月 31 日
<a href="#">執行個體停止保護</a>	若要避免意外停止執行個體，您可啟用執行個體的停止保護。	2022 年 5 月 24 日
<a href="#">UEFI 安全開機</a>	UEFI 安全開機建置於 Amazon EC2 的長期安全開機程序之上，並提供其他功能，協助客戶 defense-in-depth 保護軟體免於在重新開機期間持續存在的威脅。	2022 年 5 月 10 日
<a href="#">NitroTPM</a>	硝基可信平台模塊 ( NitroTPM ) 是由 AWS 硝基系統提供並符合 TPM 2.0 規範的虛擬設備。	2022 年 5 月 10 日
<a href="#">AMI 狀態變更事件</a>	現在當 AMI 狀態變更時，Amazon EC2 會產生一個事件。您可以使用 Amazon EventBridge 偵測這些事件並做出回應。	2022 年 5 月 9 日
<a href="#">描述公有金鑰</a>	您可以查詢 Amazon EC2 金鑰對的公有金鑰和建立日期。	2022 年 4 月 28 日

<a href="#">建立金鑰對</a>	您可以在建立新的金鑰對時指定金鑰格式 (PEM 或 PPK)。	2022 年 4 月 28 日
<a href="#">在啟動時掛載 Amazon FSx 檔案系統</a>	您可以使用新的啟動執行個體精靈，在啟動時掛載新的或現有的 Amazon FSx (適用於 NetApp ONTAP) 或 Amazon FSx (適用於 OpenZFS) 檔案系統。	2022 年 4 月 12 日
<a href="#">新的啟動執行個體精靈</a>	Amazon EC2 主控台中全新和改善的啟動體驗，提供更快、更輕鬆啟動 EC2 執行個體的方式。	2022 年 4 月 5 日
<a href="#">自動棄用公用 AMI</a>	在預設情況下，所有公用 AMI 的棄用日期都設定為自 AMI 建立日期起兩年。	2022 年 3 月 31 日
<a href="#">執行個體中繼資料類別： autoscaling/target-lifecycle-state</a>	使用 Auto Scaling 群組時，您可以從執行個體中繼資料中存取執行個體的目標生命週期狀態。	2022 年 3 月 24 日
<a href="#">AMI 上次啟動時間</a>	lastLaunchedTime 表示您的 AMI 上次用於啟動執行個體的時間。	2022 年 2 月 28 日
<a href="#">AMI 的資源回收筒</a>	資源回收筒可讓您還原意外刪除的 AMI。	2022 年 2 月 3 日
<a href="#">ED25519 金鑰</a>	EC2 執行個體 Connect 和 EC2 序列主控台不支援 ED25519 金鑰。	2022 年 1 月 20 日
<a href="#">用於容量保留的額外 RHEL 平台</a>	適用於按需容量保留的其他 Red Hat Enterprise Linux 平台。	2022 年 1 月 11 日

<a href="#">設定 Windows AMI 以實現更快速啟動</a>	使用預先佈建快照設定 Windows AMI，將啟動執行個體的速度提高 65%。	2022 年 1 月 10 日
<a href="#">執行個體中繼資料中的執行個體標籤</a>	您可以從執行個體中繼資料存取執行個體的標籤。	2022 年 1 月 6 日
<a href="#">集群放置群組中的容量保留</a>	可以在集群放置群組中建立容量保留。	2022 年 1 月 6 日
<a href="#">Amazon EBS 快照的資源回收筒</a>	Amazon EBS 快照的資源回收筒是一種快照復原功能，可讓您還原意外刪除的快照。	2021 年 11 月 29 日
<a href="#">Spot 機群 launch-before-terminate</a>	Spot 機群可以在新的替代 Spot 執行個體啟動之後，終止收到重新平衡通知的 Spot 執行個體。	2021 年 11 月 4 日
<a href="#">EC2 Fleet launch-before-terminate</a>	EC2 機群可以在新的替代 Spot 執行個體啟動之後，終止收到重新平衡通知的 Spot 執行個體。	2021 年 11 月 4 日
<a href="#">比較時間戳記</a>	您可以透過比較 Amazon EC2 Linux 執行個體的時間戳記來判斷事件的真實時間 ClockBound。	2021 年 11 月 2 日
<a href="#">與組織和 OU 共用 AMI</a>	您現在可以與下列 AWS 資源共用 AMI：組織和組織單位 (OU)。	2021 年 10 月 29 日
<a href="#">Spot 配置分數</a>	根據您的 Spot 容量需求取得區 AWS 域或可用區域的建議。	2021 年 10 月 27 日

<a href="#">Spot 機群的屬性型執行個體類型選取</a>	指定執行個體必須具有的屬性，然後 Amazon EC2 會識別具有這些屬性的所有執行個體類型。	2021 年 10 月 27 日
<a href="#">EC2 機群的屬性型執行個體類型選取</a>	指定執行個體必須具有的屬性，然後 Amazon EC2 會識別具有這些屬性的所有執行個體類型。	2021 年 10 月 27 日
<a href="#">隨需容量預留機群</a>	您可以使用容量預留機群，啟動一個群組或機群的容量預留。	2021 年 10 月 5 日
<a href="#">Ubuntu 20.04 LTS - Focal 的休眠支援</a>	讓從 Ubuntu 20.04 LTS - Focal AMI 啟動的新啟動執行個體休眠。	2021 年 10 月 4 日
<a href="#">EC2 Fleet 和目標隨需容量預留</a>	EC2 Fleet 可以按照 targeted 容量預留啟動隨需執行個體。	2021 年 9 月 22 日
<a href="#">專用主機上的 T3 執行個體</a>	Amazon EC2 專用主機上的 T3 執行個體支援。	2021 年 9 月 14 日
<a href="#">針對 RHEL、Fedora 和 CentOS 的休眠支援</a>	對您新啟動且從 RHEL、Fedora 和 CentOS AMI 啟動的執行個體進行休眠。	2021 年 9 月 9 日
<a href="#">Amazon EC2 全域檢視</a>	Amazon EC2 全球視圖可讓您在單一主控台中檢視跨多個區域的 VPC、子網路、執行個體、安全群組和磁碟 AWS 區。	2021 年 9 月 1 日

<a href="#">Amazon Data Lifecycle Manager 的 AMI 取代支援</a>	Amazon Data Lifecycle Manager EBS 後端 AMI 政策可能會取代 AMI。受 AWSDataLifecycleManagerServiceRoleForAMIManagement AWS 管理策略已更新以支援此功能。	2021 年 8 月 23 日
<a href="#">C5d、M5d 和 R5d 的休眠支援</a>	對您新啟動且在 C5d、M5d 和 R5d 執行個體類型上執行的執行個體進行休眠。	2021 年 8 月 19 日
<a href="#">Amazon EC2 金鑰對</a>	Amazon EC2 現已支援 Linux 和 Mac 執行個體上的 ED25519 金鑰。	2021 年 8 月 17 日
<a href="#">網路介面字首</a>	您可以自動或手動將私有 IPv4 或 IPv6 CIDR 範圍指派給網路介面。	2021 年 7 月 22 日
<a href="#">事件時段</a>	您可以為重新啟動、停止或終止 Amazon EC2 執行個體的排程事件定義自訂、每週重複的事件時段。	2021 年 7 月 15 日
<a href="#">支援安全群組規則的資源 ID 和標記</a>	您可以依資源 ID 參考安全群組規則。您也可以為安全群組規則新增標籤。	2021 年 7 月 7 日
<a href="#">取代 AMI</a>	您現在可以指定何時取代 AMI。	2021 年 6 月 11 日
<a href="#">Windows 每秒計費</a>	Amazon EC2 會按秒收取 Windows 和 SQL Server 用量的費用，最低收費為一分鐘。	2021 年 6 月 10 日
<a href="#">容量保留 AWS Outposts</a>	您現在可以在 AWS Outposts 上使用容量預留。	2021 年 5 月 24 日

<a href="#">共用容量保留</a>	您現在可以共用在 Local Zones 和 Wavelength 區域中建立的容量預留。	2021 年 5 月 24 日
<a href="#">根磁碟區更換</a>	您現在可以使用根磁碟區更換任務來取代執行中執行個體的根 EBS 磁碟區。	2021 年 4 月 22 日
<a href="#">使用 S3 存放和還原 AMI</a>	將 EBS 後端 AMI 存放在 S3 中，並從 S3 將其還原以啟用跨分割區複製 AMI。	2021 年 4 月 6 日
<a href="#">EC2 序列主控台</a>	透過建立執行個體的序列埠連線，疑難排解開機和網路連線問題。	2021 年 3 月 30 日
<a href="#">開機模式</a>	Amazon EC2 現在支援在選定 AMD 和 Intel EC2 執行個體上進行 UEFI 開機。	2021 年 3 月 22 日
<a href="#">建立反向 DNS 記錄</a>	您現在可以為彈性 IP 地址設定反向 DNS 查詢。	2021 年 2 月 3 日
<a href="#">在 AMI 建立時標記 AMI 和快照</a>	當您建立 AMI 時，可以使用相同的標籤來標記 AMI 和快照，或者可以使用不同的標籤來進行標記。	2020 年 12 月 4 日
<a href="#">使用 Amazon EventBridge 監控競價型叢集事件</a>	建立可觸發程式設計動作的 EventBridge 規則，以回應 Spot 叢集狀態變更和錯誤。	2020 年 11 月 20 日
<a href="#">使用 Amazon EventBridge 監控 EC2 叢集事件</a>	建立 EventBridge 規則以觸發程式設計動作，以回應 EC2 叢集狀態變更和錯誤。	2020 年 11 月 20 日
<a href="#">刪除 instant 機群</a>	刪除類型 instant 的 EC2 Fleet，並以單一 API 呼叫來終止機群中的所有執行個體。	2020 年 11 月 18 日



<a href="#">對 T3 和 T3a 的休眠支援</a>	對新啟動且在 T3 和 T3a 執行個體類型上執行的執行個體進行休眠。	2020 年 11 月 17 日
<a href="#">Amazon EFS Quick Create</a>	您可以使用 Amazon EFS 快速建立，在啟動時建立 Amazon EFS 檔案系統並將其掛接到執行個體。	2020 年 11 月 9 日
<a href="#">例項中繼資料類別：events/recommendations/rebalance</a>	針對執行個體發出 EC2 執行個體重新平衡建議通知的大約時間 (UTC)。	2020 年 11 月 4 日
<a href="#">EC2 執行個體重新平衡推薦</a>	當 Spot 執行個體的中斷風險提高時，會以訊號來通知您。	2020 年 11 月 4 日
<a href="#">在 Wavelength 區域中的容量保留</a>	容量預留現在可以在 Wavelength 區域中建立和使用。	2020 年 11 月 4 日
<a href="#">容量重新平衡</a>	設定 Spot Fleet 或 EC2 Fleet，以在 Amazon EC2 發出重新平衡建議時啟動替代 Spot 執行個體。	2020 年 11 月 4 日
<a href="#">對 I3、M5ad 和 R5ad 的休眠支援</a>	對您新啟動且在 I3、M5ad 和 R5ad 執行個體類型上執行的執行個體進行休眠。	2020 年 10 月 21 日
<a href="#">Spot 執行個體 vCPU 限制</a>	Spot 執行個體限制現在是根據執行中的 Spot 執行個體正在使用或將使用的 vCPU 的數量來管理的，直到完成開放請求。	2020 年 10 月 1 日
<a href="#">Local Zones 中的容量保留</a>	容量預留現在可以在 Local Zones 中建立和使用。	2020 年 9 月 30 日

<a href="#">對 M5a 和 R5a 的休眠支援</a>	對您新啟動且在 M5a 和 R5a 執行個體類型上執行的執行個體進行休眠。	2020 年 8 月 28 日
<a href="#">執行個體中繼資料提供執行個體位置和放置資料</a>	placement 類別下的新執行個體中繼資料欄位：區域、置放群組名稱、磁碟分割號碼、主機 ID 和可用區域 ID。	2020 年 8 月 24 日
<a href="#">容量保留群組</a>	您可以使用 AWS Resource Groups 建立容量保留的邏輯集合，然後將目標執行個體啟動到這些群組中。	2020 年 7 月 29 日
<a href="#">EC2Launch v2</a>	您可以使用 EC2Launch v2 在執行個體啟動期間執行任務，並在執行個體停止後再啟動或重新啟動時執行，也可以依需求執行任務。EC2Launch v2 支援所有版本的 Windows Server，並取代 EC2Launch 和 EC2Config。	2020 年 6 月 30 日
<a href="#">攜帶自有 IPv6 位址</a>	您可以將內部部署網路中的部分或全部 IPv6 位址範圍帶入您的 AWS 帳戶。	2020 年 5 月 21 日
<a href="#">使用 Systems Manager 參數啟動執行個體</a>	啟動執行個體時，您可以指定 AWS Systems Manager 參數而不是 AMI。	2020 年 5 月 5 日
<a href="#">自訂排程的事件通知</a>	您可以自訂排程事件通知，以便在電子郵件通知中包含標籤。	2020 年 5 月 4 日

<a href="#">Amazon Linux 2 Kernel Live Patching</a>	Amazon Linux 2 的 Kernel Live Patching 可讓您將安全性漏洞和重大錯誤修補程式套用至執行中的 Linux 核心，而不需要重新啟動或中斷執行中的應用程式。	2020 年 4 月 28 日
<a href="#">在專用主機上的 Windows Server</a>	您可以使用由 Amazon 提供的 Windows Server AMI 在專用主機上運行最新版本的 Windows Server。	2020 年 4 月 7 日
<a href="#">停止並啟動 Spot 執行個體</a>	停止您由 Amazon EBS 支援的 Spot 執行個體，並隨時啟動它們，而不需要依賴停止中斷行為。	2020 年 1 月 13 日
<a href="#">資源標記</a>	您可以標記輸出限定網際網路閘道、本機閘道、本機閘道路由表、本機閘道虛擬介面、本機閘道虛擬介面群組、本機閘道路由表 VPC 關聯，以及本機閘道路由表虛擬介面群組關聯。	2020 年 1 月 10 日
<a href="#">使用工作階段管理員連線至執行個體</a>	您可以從 Amazon EC2 主控台使用一個執行個體啟動工作階段管理員工作階段。	2019 年 12 月 18 日
<a href="#">專用主機和主機資源群組</a>	專用主機現在可與主機資源群組搭配使用。	2019 年 12 月 2 日
<a href="#">專用主機共用</a>	您現在可以跨 AWS 帳戶共用專用主機。	2019 年 12 月 2 日
<a href="#">帳戶層級的預設額度規格</a>	您可以在每個 AWS 區域的帳戶層次，設定每個高載績效執行環境系列的預設信用規格。	2019 年 11 月 25 日

<a href="#">執行個體類型探索</a>	您可以尋找符合您需求的執行個體類型。	2019 年 11 月 22 日
<a href="#">專用主機</a>	您現在可以將 專用主機 設定為支援執行個體系列中的多個執行個體類型。	2019 年 11 月 21 日
<a href="#">執行個體中繼資料服務第 2 版</a>	您可以使用 執行個體中繼資料服務第 2 版，這種工作階段導向的方法，可讓您請求執行個體中繼資料。	2019 年 11 月 19 日
<a href="#">Elastic Fabric Adapter</a>	Elastic Fabric Adapter 現在可與 Intel MPI 2019 Update 6 搭配使用。	2019 年 11 月 15 日
<a href="#">休眠支援隨需 Windows 執行個體</a>	您可以讓隨需 Windows 執行個體進入休眠。	2019 年 10 月 14 日
<a href="#">已將預留執行個體購買放入佇列</a>	最早可提前三年將預留執行個體購買放入佇列。	2019 年 10 月 4 日
<a href="#">診斷中斷</a>	您可以將診斷中斷傳送至無法連線或沒有回應的執行個體，以觸發核心錯誤。	2019 年 8 月 14 日
<a href="#">容量最佳化分配策略</a>	使用 EC2 Fleet 或 Spot Fleet，可從 Spot 集區以適用於所啟動之執行個體數量的最佳容量來啟動 Spot 執行個體。	2019 年 8 月 12 日
<a href="#">隨需容量保留共用</a>	您現在可以跨 AWS 帳戶共用容量保留。	2019 年 7 月 29 日
<a href="#">Elastic Fabric Adapter</a>	EFA 現在支援 Open MPI 3.1.4 和 Intel MPI 2019 Update 4。	2019 年 7 月 26 日

<a href="#">EC2 執行個體連線</a>	EC2 Instance Connect 為使用 Secure Shell (SSH) 連接到執行個體的簡單且安全方式。	2019 年 6 月 27 日
<a href="#">主機復原</a>	當專用主機上發生非預期的硬體失敗時，在新主機上自動重新啟動您的執行個體。	2019 年 6 月 5 日
<a href="#">VSS 應用程式一致性快照</a>	使用執 AWS Systems Manager 行命令，為連接到 Windows 執行個體的所有 Amazon EBS 磁碟區拍攝應用程式一致的快照。	2019 年 5 月 13 日
<a href="#">從 Windows 到 Linux 的 Microsoft SQL Server 資料庫平台重建助理</a>	將現有的 Microsoft SQL Server 工作負載從 Windows 移到 Linux 作業系統。	2019 年 5 月 8 日
<a href="#">Windows 自動升級</a>	使用執行 EC2 視窗執行個體的自動升級 AWS Systems Manager。	2019 年 5 月 6 日
<a href="#">Elastic Fabric Adapter</a>	您可以將 Elastic Fabric Adapter 連接到執行個體，以加速高效能運算 (HPC) 應用程式。	2019 年 4 月 29 日

如需 Amazon EC2 執行個體類型版本的相關資訊，請參閱 Amazon EC2 執行個體類型指南中的 [文件歷史記錄](#)。

## 2018 年及更早版本的歷史

下表說明 2018 年和更早年份 Amazon EC2 使用者指南的重要新增內容。

功能	API 版本	描述	發行日期
分區置放群組	2016-11-15	分區置放群組會將執行個體分散到邏輯分區，確保一個分區中的執行個體不會與其他分區中的執行個體共用底層硬體。如需詳細資訊，請參閱 <a href="#">分區置放群組</a> 。	2018 年 12 月 20 日
讓 EC2 Linux 執行個體進入休眠	2016-11-15	當您的 Linux 執行個體已啟用休眠並符合休眠必要條件時，您才能讓執行個體進入休眠。如需詳細資訊，請參閱 <a href="#">休眠您的 Amazon EC2 實例</a> 。	2018 年 11 月 28 日
Amazon Elastic Inference Accelerator	2016-11-15	您可以將 Amazon EI 加速器連接至執行個體，並新增 GPU 加速以降低執行深度學習推論的成本。	2018 年 11 月 28 日
Spot 主控台建議使用眾多執行個體	2016-11-15	Spot 主控台建議您依據 Spot 最佳實務使用眾多執行個體 (執行個體多角化)，以滿足應用程式所需的最低硬體規格 (vCPU、記憶體和儲存)。如需詳細資訊，請參閱 <a href="#">建立 Spot Fleet 請求</a> 。	2018 年 11 月 20 日
新的 EC2 Fleet 請求類型：instant	2016-11-15	EC2 Fleet 現可支援新的 instant 請求類型，您可用來跨執行個體類型同步佈建容量，並購買模型。instant 請求會在 API 回應中傳回啟動的執行個體，但不會採取進一步動作，讓您可以控制是否啟動執行個體以及啟動時機。如需詳細資訊，請參閱 <a href="#">EC2 Fleet 請求類型</a> 。	2018 年 11 月 14 日
Spot 節省資訊	2016-11-15	您可以檢視針對單一 Spot Fleet 或所有 Spot 執行個體使用 Spot 執行個體所節省的成本。如需詳細資訊，請參閱 <a href="#">購買 競價型執行個體省下的成本</a> 。	2018 年 11 月 5 日
主控台支援最佳化 CPU 選項	2016-11-15	當您啟動執行個體時，您可以使用 Amazon EC2 主控台最佳化 CPU 選項以符合特定工作負載或商業需求。如需詳細資訊，請參閱 <a href="#">最佳化 CPU 選項</a> 。	2018 年 10 月 31 日

功能	API 版本	描述	發行日期
主控台支援從執行個體建立啟動範本	2016-11-15	您可以使用 Amazon EC2 主控台並將執行個體做為新啟動範本的基礎，以建立新的啟動範本。如需詳細資訊，請參閱 <a href="#">建立啟動範本</a> 。	2018 年 10 月 30 日
On-Demand Capacity Reservations	2016-11-15	在特定的可用區域中，您可以為 Amazon EC2 執行個體預留任何期限的容量。這可讓您建立和管理容量保留，而不受預留執行個體 (RI) 的帳單折扣影響。如需詳細資訊，請參閱 <a href="#">On-Demand Capacity Reservations</a> 。	2018 年 10 月 25 日
使用自有 IP 地址 (BYOIP)	2016-11-15	您可以將內部部署網路中的部分或全部公用 IPv4 位址範圍移至您的 AWS 帳戶。將地址範圍帶到之後 AWS，它會以地址集區的形式出現在您的帳戶中。您可以從您的地址集區建立彈性 IP 地址，並將它用於您的 AWS 資源。如需詳細資訊，請參閱 <a href="#">在 Amazon EC2 中使用自有 IP 地址 (BYOIP)</a> 。	2018 年 10 月 23 日
在建立專用主機時套用標籤和主控台支援	2016-11-15	您可以在建立您的專用主機時套用標籤，並使用 Amazon EC2 主控台來管理您的專用主機標籤。如需詳細資訊，請參閱 <a href="#">配置專用執行個體</a> 。	2018 年 10 月 8 日
對 Spot Fleet 的排程擴展的主控台支援	2016-11-15	根據日期和時間，來增加或減少機群目前的容量。如需詳細資訊，請參閱 <a href="#">使用排程擴展來擴展 Spot Fleet</a> 。	2018 年 9 月 20 日
EC2 Fleet 的分配策略	2016-11-15	您可以指定隨需容量是由價格 (最低價格優先) 或優先順序 (最高優先順序優先) 履行。您可以指定要配置目標 Spot 容量的 Spot 集區數量。如需詳細資訊，請參閱 <a href="#">競價型執行個體的分配策略</a> 。	2018 年 7 月 26 日

功能	API 版本	描述	發行日期
Spot Fleets 的分配策略	2016-11-15	您可以指定隨需容量是由價格 (最低價格優先) 或優先順序 (最高優先順序優先) 履行。您可以指定要配置目標 Spot 容量的 Spot 集區數量。如需詳細資訊，請參閱 <a href="#">競價型執行個體的分配策略</a> 。	2018 年 7 月 26 日
自動快照生命週期	2016-11-15	您可以使用 Amazon Data Lifecycle Manager 來自動化建立和刪除 EBS 磁碟區的快照。如需詳細資訊，請參閱 <a href="#">Amazon Data Lifecycle Manager</a> 。	2018 年 7 月 12 日
啟動範本 CPU 選項	2016-11-15	當您使用命令列工具建立啟動範本時，您可以最佳化 CPU 選項來符合特定工作負載或商務需求。如需詳細資訊，請參閱 <a href="#">建立啟動範本</a> 。	2018 年 7 月 11 日
標記 專用主機	2016-11-15	您可以標記您的 專用主機。如需詳細資訊，請參閱 <a href="#">標記 專用執行個體</a> 。	2018 年 7 月 3 日
取得最新主控台輸出	2016-11-15	當您使用 <a href="#">get-con</a> AWS CLI sole-output 命令時，您可以擷取某些執行個體類型的最新主控台輸出。	2018 年 5 月 9 日
最佳化 CPU 選項	2016-11-15	當您啟動執行個體時，您可以最佳化 CPU 選項來符合特定工作負載或商業需求。如需詳細資訊，請參閱 <a href="#">最佳化 CPU 選項</a> 。	2018 年 5 月 8 日
EC2 Fleet	2016-11-15	您可以使用 EC2 Fleet 跨不同 EC2 執行個體類型和可用區域，並且跨隨需執行個體、保留執行個體和 Spot 執行個體購買模型來啟動一組執行個體。如需詳細資訊，請參閱 <a href="#">EC2 Fleet</a> 。	2018 年 5 月 2 日
Spot Fleets 中的隨需執行個體	2016-11-15	您可以在 Spot Fleet 請求中，加入對隨需容量的請求，來確保永遠有執行個體容量。如需詳細資訊，請參閱 <a href="#">Spot 叢集</a> 。	2018 年 5 月 2 日
建立時標記 EBS 快照	2016-11-15	您可以在建立期間將標籤套用至快照。	2018 年 4 月 2 日



功能	API 版本	描述	發行日期
變更置放群組	2016-11-15	您可以將執行個體移入或移出置放群組，或變更其置放群組。如需詳細資訊，請參閱 <a href="#">變更執行個體的置放群組</a> 。	2018 年 3 月 1 日
較長的資源 ID	2016-11-15	您可以為更多資源類型啟用較長的 ID 格式。如需詳細資訊，請參閱 <a href="#">資源 ID</a> 。	2018 年 2 月 9 日
網路效能提升	2016-11-15	在其他執行個體或 Amazon S3 之間傳送或接收網路流量時，集群置放群組之外的執行個體現在可享有更高的頻寬。	2018 年 1 月 24 日
標記彈性 IP 地址	2016-11-15	您可以標記彈性 IP 地址。如需詳細資訊，請參閱 <a href="#">為彈性 IP 地址套用標籤</a> 。	2017 年 12 月 21 日
Amazon Time Sync Service	2016-11-15	您可以使用 Amazon Time Sync Service 來確保執行個體的時間正確。如需詳細資訊，請參閱 <a href="#">設定亞馬遜 EC2 執行個體的時間</a> 。	2017 年 11 月 29 日
無限制 T2	2016-11-15	T2 效能不受限執行個體可隨時按需要大幅提升效能以超越基準。如需詳細資訊，請參閱 <a href="#">爆量效能執行個體</a> 。	2017 年 11 月 29 日
啟動範本	2016-11-15	啟動範本可包含用以啟動執行個體的所有或部分參數，因此您不需要在每次啟動執行個體時指定。如需詳細資訊，請參閱 <a href="#">從啟動範本啟動執行個體</a> 。	2017 年 11 月 29 日
分散置放	2016-11-15	若應用程式具有應該分開保存的少量重要執行個體，建議使用分散置放群組。如需詳細資訊，請參閱 <a href="#">分散置放群組</a> 。	2017 年 11 月 29 日
Spot 執行個體休眠	2016-11-15	Spot 服務可以在發生中斷時讓 Spot 執行個體休眠。如需詳細資訊，請參閱 <a href="#">休眠中斷的競價型執行個體</a> 。	2017 年 11 月 28 日

功能	API 版本	描述	發行日期
Spot Fleet 目標追蹤	2016-11-15	您可以為 Spot Fleet 設定目標追蹤擴展政策。如需詳細資訊，請參閱 <a href="#">使用目標追蹤政策來擴展 Spot Fleet</a> 。	2017 年 11 月 17 日
Spot Fleet 與 Elastic Load Balancing 整合	2016-11-15	您可以將一或多個負載平衡器連接至 Spot Fleet。	2017 年 11 月 10 日
合併及分割 可轉換預留執行個體	2016-11-15	您可以為新的 可轉換預留執行個體 交換 (合併) 兩個或多個 可轉換預留執行個體。您也可以使用修改程序將可轉換預留執行個體分割為較小的保留。如需詳細資訊，請參閱 <a href="#">交換 可轉換預留執行個體</a> 。	2017 年 11 月 6 日
修改 VPC 租用	2016-11-15	您可將 VPC 的執行個體租用屬性從 dedicated 變更為 default。如需詳細資訊，請參閱 <a href="#">變更 VPC 租用</a> 。	2017 年 10 月 16 日
每秒計費	2016-11-15	Amazon EC2 會按秒收取 Linux 用量的費用，最低收費為一分鐘。	2017 年 10 月 2 日
中斷時停止	2016-11-15	您可以指定當 競價型執行個體中斷時，Amazon EC2 應該停止或終止它們。如需詳細資訊，請參閱 <a href="#">Spot 執行個體中斷的行為</a> 。	2017 年 9 月 18 日
標記 NAT 閘道	2016-11-15	您可以標記 NAT 閘道。如需詳細資訊，請參閱 <a href="#">標記您的 資源</a> 。	2017 年 9 月 7 日
安全群組規則說明	2016-11-15	您可以為安全群組規則新增說明。如需詳細資訊，請參閱 <a href="#">安全群組規則</a> 。	2017 年 8 月 31 日
Elastic Graphics	2016-11-15	將 Elastic Graphics 加速器連接至您的執行個體，來加速應用程式的圖形效能。	2017 年 8 月 29 日

功能	API 版本	描述	發行日期
復原彈性 IP 地址	2016-11-15	如果您發行用於 VPC 的彈性 IP 地址，您可以將它復原。如需詳細資訊，請參閱 <a href="#">復原彈性 IP 地址</a> 。	2017 年 8 月 11 日
標記 Spot Fleet 執行個體	2016-11-15	您可以設定 Spot Fleet 以自動標記其啟動的執行個體。	2017 年 7 月 24 日
在建立期間標記資源	2016-11-15	您可以在建立期間將標籤套用至執行個體和磁碟區。如需詳細資訊，請參閱 <a href="#">標記您的資源</a> 。此外，您也可以使用標籤型資源層級許可，來控制套用的標籤。如需更多詳細資訊，請參閱 <a href="#">在建立期間授予標籤資源的許可</a> 。	2017 年 3 月 28 日
在連接的 EBS 磁碟區上執行修改	2016-11-15	有了連接至大多數 EC2 執行個體的大多數 EBS 磁碟區，您可以修改磁碟區大小、類型和 IOPS，無需分離磁碟區或停止執行個體。	2017 年 2 月 13 日
連接 IAM 角色	2016-11-15	您可以連接、分離或取代現有執行個體的 IAM 角色。如需詳細資訊，請參閱 <a href="#">Amazon EC2 的 IAM 的角色</a> 。	2017 年 2 月 9 日
專用的 競價型執行個體	2016-11-15	您可以在 Virtual Private Cloud (VPC) 中的單一租戶硬體上執行 競價型執行個體。如需詳細資訊，請參閱 <a href="#">為 競價型執行個體 指定租用</a> 。	2017 年 1 月 19 日
IPv6 支援	2016-11-15	您可以將 IPv6 CIDR 與 VPC 和子網建立關聯，並指派 IPv6 地址給 VPC 中的執行個體。如需詳細資訊，請參閱 <a href="#">Amazon EC2 執行個體 IP 地址</a> 。	2016 年 12 月 1 日
Spot Fleet 的自動擴展		您現在可以為 Spot Fleet 設定擴展政策。如需詳細資訊，請參閱 <a href="#">Spot Fleet 的自動擴展</a> 。	2016 年 9 月 1 日
彈性網路轉接器 (ENA)	2016-04-01	您現在可以將 ENA 用於增強聯網。如需詳細資訊，請參閱 <a href="#">增強的網路支援</a> 。	2016 年 6 月 28 日

功能	API 版本	描述	發行日期
檢視及修改較長 ID 的增強型支援	2016-04-01	您現在可以檢視及修改其他 IAM 使用者、IAM 角色或根使用者的較長 ID 設定。如需詳細資訊，請參閱 <a href="#">資源 ID</a> 。	2016 年 6 月 23 日
在 AWS 帳戶之間複製加密的 Amazon EBS 快照	2016-04-01	您現在可以在 AWS 帳戶之間複製加密的 EBS 快照。	2016 年 6 月 21 日
擷取執行個體主控台的螢幕擷取畫面	2015-10-01	當您對無法連線的執行個體除錯時，您現在可以取得其他資訊。如需詳細資訊，請參閱 <a href="#">擷取無法連線執行個體的螢幕擷取畫面</a> 。	2016 年 5 月 24 日
兩個新的 EBS 磁碟區類型	2015-10-01	您現在可以建立輸送量最佳化 HDD (st1) 和 Cold HDD (sc1) 磁碟區。	2016 年 4 月 19 日
為 Amazon EC2 添加了新 NetworkPacketsIn 的 NetworkPacketsOut 指標		為 Amazon EC2 添加了新 NetworkPacketsIn 的 NetworkPacketsOut 指標。如需詳細資訊，請參閱 <a href="#">執行個體指標</a> 。	2016 年 3 月 23 日
CloudWatch 競價型艦隊的指標		您現在可以取得 Spot 叢集的 CloudWatch 指標。如需詳細資訊，請參閱 <a href="#">CloudWatch 競價型艦隊的指標</a> 。	2016 年 3 月 21 日
排程執行個體	2015-10-01	排程的預留執行個體 (排程執行個體) 可讓您以指定的開始時間和期間，購買每日、每週或每月的容量保留。	2016 年 1 月 13 日
較長的資源 ID	2015-10-01	我們將逐步為一些 Amazon EC2 和 Amazon EBS 資源類型引進長度更長的 ID。在選擇期間，您可以為支援的資源類型啟用較長的 ID 格式。如需詳細資訊，請參閱 <a href="#">資源 ID</a> 。	2016 年 1 月 13 日

功能	API 版本	描述	發行日期
ClassicLink DNS 支援服務	2015-10-01	您可以為 VPC 啟用 ClassicLink DNS 支援，以便在連結 EC2-Classic 執行個體和 VPC 中執行個體之間尋址的 DNS 主機名稱解析為私有 IP 位址，而非公用 IP 位址。	2016 年 1 月 11 日
專用主機	2015-10-01	Amazon EC2 專用主機是具有專供您使用之執行個體容量的實體伺服器。如需詳細資訊，請參閱 <a href="#">Dedicated Hosts</a> 。	2015 年 11 月 23 日
Spot 執行個體持續時間	2015-10-01	您現在可以為 競價型執行個體指定持續時間。不支援 Spot 區塊 (2023 年 1 月)。	2015 年 10 月 6 日
Spot Fleet 修改請求	2015-10-01	您現在可以修改 Spot Fleet 請求的目標容量。如需詳細資訊，請參閱 <a href="#">修改 Spot Fleet 請求</a> 。	2015 年 9 月 29 日
Spot Fleet 多樣化分配策略	2015-04-15	您現在可以使用單一 Spot Fleet 請求，將 Spot 執行個體配置到多個 Spot 集區。如需詳細資訊，請參閱 <a href="#">競價型執行個體的分配策略</a> 。	2015 年 9 月 15 日
Spot Fleet 執行個體權重	2015-04-15	您現在可以定義每個執行個體類型對應用程式效能所貢獻的容量單位，並據以調整每個 Spot 集區您願意為 競價型執行個體支付的價格。如需詳細資訊，請參閱 <a href="#">Spot Fleet 執行個體權重</a> 。	2015 年 8 月 31 日
新的重新啟動警示動作和用於警示動作的新 IAM 角色		新增重新啟動警示動作和用於警示動作的新 IAM 角色。如需詳細資訊，請參閱 <a href="#">建立警示以停止、終止、重新啟動或復原執行個體</a> 。	2015 年 7 月 23 日
Spot Fleets	2015-04-15	您可以管理 Spot 執行個體集合或機群，而不是管理個別 Spot 執行個體請求。如需詳細資訊，請參閱 <a href="#">Spot 叢集</a> 。	2015 年 5 月 18 日
將彈性 IP 地址遷移至 EC2-Classic	2015-04-15	您無法將配置用於 EC2-Classic 的彈性 IP 地址遷移為在 VPC 中使用。	2015 年 5 月 15 日

功能	API 版本	描述	發行日期
將 VM 與多個磁碟匯入為 AMI	2015-03-01	VM Import 程序現在支援將 VM 與多個磁碟匯入為 AMI。如需詳細資訊，請參閱 VM Import/Export 使用者指南中的 <a href="#">使用 VM 匯入/匯出將 VM 匯入為映像</a> 。	2015 年 4 月 23 日
Systems Manager		Systems Manager 可讓您設定及管理 EC2 執行個體。	2015 年 2 月 17 日
Systems Manager for Microsoft SCVMM 1.5		您現在可以使用 Systems Manager for Microsoft SCVMM 來啟動執行個體，並將 VM 從 SCVMM 匯入 Amazon EC2。	2015 年 1 月 21 日
EC2 執行個體的自動復原功能		您可以建立 Amazon CloudWatch 警示來監控 Amazon EC2 執行個體，並在執行個體因基礎硬體故障或需要 AWS 參與修復的問題而受損時，自動復原執行個體。復原的執行個體與原始執行個體相同，包括執行個體 ID、IP 地址及所有執行個體中繼資料。  如需詳細資訊，請參閱 <a href="#">執行個體恢復</a> 。	2015 年 1 月 12 日
ClassicLink	2014-10-01	ClassicLink 可讓您將 EC2-Classic 執行個體連結至帳戶中的 VPC。您可以將 VPC 安全群組與 EC2-Classic 執行個體建立關聯，讓 EC2-Classic 執行個體可以使用私有 IP 地址與您 VPC 中的執行個體通訊。	2015 年 1 月 7 日
Spot 執行個體終止通知		防止 Spot 執行個體中斷的最佳方法就是建立容錯的應用程式架構。此外，您也可以使用 Spot 執行個體終止通知，此通知會在 Amazon EC2 必須終止 Spot 執行個體的兩分鐘前發出警告。  如需詳細資訊，請參閱 <a href="#">Spot 執行個體中斷通知</a> 。	2015 年 1 月 5 日

功能	API 版本	描述	發行日期
Systems Manager for Microsoft SCVMM		Microsoft SCVMM 的 Systems Manager 提供了一個簡單的 easy-to-use 界面，用於管理 AWS 資源，如 EC2 實例，從 Microsoft SCVMM。	2014 年 10 月 29 日
DescribeVolumes 分頁支援	2014-09-01	DescribeVolumes API 呼叫現在支援使用 MaxResults 和 NextToken 參數來分頁結果。如需詳細資訊，請參閱亞 Amazon EC2 API 參考 <a href="#">DescribeVolumes</a> 中的。	2014 年 10 月 23 日
增加了對 Amazon CloudWatch 日誌支持		您可以使用 Amazon CloudWatch Logs 監控、存放和存取執行個體或其他來源的系統、應用程式和自訂日誌檔。然後，您可以使用 Amazon CloudWatch 主控台、AWS CLI 中的 CloudWatch 日誌命令或 CloudWatch 日誌開發套件從日誌擷取關聯的 CloudWatch 日誌資料。	2014 年 7 月 10 日
新的 EC2 Service Limits (EC2 服務限制) 頁面		使用 Amazon EC2 主控台中的 EC2 Service Limits (EC2 服務限制) 頁面，檢視 Amazon EC2 及 Amazon VPC 為每個區域提供的目前資源限制。	2014 年 6 月 19 日
Amazon EBS 一般用途 SSD 磁碟區	2014-05-01	一般用途 SSD 磁碟區提供經濟實惠的儲存空間，適合絕大多數的工作負載。這些磁碟區的延遲時間不到 10 毫秒，且可在延長的時間內爆量至 3,000 IOPS，且其基礎效能為 3 IOPS/GiB。一般用途 SSD 磁碟區的大小範圍可從 1 GiB 到 1 TiB。	2014 年 6 月 16 日
AWS 管理組件		AWS 管理組件現在支援系統中心作業管理員 2012 年 R2。	2014 年 5 月 22 日

功能	API 版本	描述	發行日期
Amazon EBS encryption	2014-05-01	Amazon EBS 加密 可讓您順暢地加密 EBS 資料磁碟區和快照，無需建立及維護安全金鑰管理基礎設施。EBS 加密透過使用 AWS 受管金鑰加密資料，來啟用靜態資料安全。還可以在託管 EC2 執行個體的伺服器上進行加密，當資料在 EC2 執行個體和 EBS 儲存之間移動時，提供資料加密。	2014 年 5 月 21 日
Amazon EC2 用量報告		Amazon EC2 用量報告是一組顯示使用 EC2 的成本和用量資料之報告。	2014 年 1 月 28 日
匯入 Linux 虛擬機器	2013-10-15	VM Import 程序現在支援匯入 Linux 執行個體。如需詳細資訊，請參閱 <a href="#">VM Import/Export 使用者指南</a> 。	2013 年 12 月 16 日
的資源層級權限 RunInstances	2013-10-15	您現在可以在中建立政策，AWS Identity and Access Management 以控制 Amazon EC2 RunInstances API 動作的資源層級許可。如需詳細資訊和範例政策，請參閱 <a href="#">Amazon EC2 的 Identity and Access Management</a> 。	2013 年 11 月 20 日
從啟動執行個體 AWS Marketplace		您現在可以 AWS Marketplace 使用 Amazon EC2 啟動精靈從啟動執行個體。如需詳細資訊，請參閱 <a href="#">啟動 AWS Marketplace 執行個體</a> 。	2013 年 11 月 11 日
新的 Launch Wizard		新增並重新設計 EC2 Launch Wizard。如需詳細資訊，請參閱 <a href="#">使用舊版啟動執行個體精靈啟動執行個體</a> 。	2013 年 10 月 10 日
修改預留執行個體的例項類型	2013-10-01	您現在可以修改相同系列之 Linux 預留執行個體的執行個體類型 (例如 M1、M2、M3、C1)。如需詳細資訊，請參閱 <a href="#">修改預留執行個體</a> 。	2013 年 10 月 9 日



功能	API 版本	描述	發行日期
修改 Amazon EC2 預留執行個體	2013-08-15	您現在可以修改區域中的預留執行個體。如需更多詳細資訊，請參閱 <a href="#">修改預留執行個體</a> 。	2013 年 9 月 11 日
指派公有 IP 地址	2013-07-15	您現在可以在啟動 VPC 中的執行個體時，指派公有 IP 地址。如需詳細資訊，請參閱 <a href="#">在啟動執行個體期間指派公有 IPv4 地址</a> 。	2013 年 8 月 20 日
授予資源層級許可	2013-06-15	Amazon EC2 支援新的 Amazon Resource Name (ARN) 和條件金鑰。如需詳細資訊，請參閱 <a href="#">適用於 Amazon EC2 的 IAM 政策</a> 。	2013 年 7 月 8 日
增量式快照複本	2013-02-01	您現在可以執行增量式快照複本。	2013 年 6 月 11 日
AWS 管理組件		AWS 管理套件會連結 Amazon EC2 執行個體以及其中執行的 Windows 或 Linux 作業系統。AWS 管理包是 Microsoft 系統中心操作管理器的擴展。	2013 年 5 月 8 日
新的 Tags (標籤) 頁面		Amazon EC2 主控台中新增 Tags (標籤) 頁面。如需詳細資訊，請參閱 <a href="#">標記您的 Amazon EC2 資源</a> 。	2013 年 4 月 4 日
將 AMI 從一個區域複製到另一個區域	2013-02-01	您可以將 AMI 從一個區域複製到另一個區域，以便在多個 AWS 區域中快速輕鬆地啟動一致的執行個體。  如需詳細資訊，請參閱 <a href="#">複製 AMI</a> 。	2013 年 3 月 11 日

功能	API 版本	描述	發行日期
將執行個體啟動至預設 VPC 中	2013-02-01	您的 AWS 帳戶能夠將執行個體啟動到 EC2-Classic 或 VPC，或僅啟動到 VPC 中。region-by-region 如果您只能將執行個體啟動至 VPC 中，我們會為您建立一個預設 VPC。當您啟動執行個體時，我們會將其啟動至您的預設 VPC 中，除非您建立非預設 VPC 並在啟動執行個體時指定該非預設 VPC。	2013 年 3 月 11 日
EBS 快照複本	2012-12-01	您可以使用快照複本，建立資料的備份、建立新的 Amazon EBS 磁碟區，或建立 Amazon Machine Image (AMI)。	2012 年 12 月 17 日
更新 Provisioned IOPS SSD 磁碟區之 EBS 指標和狀態檢查	2012-10-01	更新 EBS 指標以包含 Provisioned IOPS SSD 磁碟區之兩個新指標。另外新增 Provisioned IOPS SSD 磁碟區之狀態檢查。	2012 年 11 月 20 日
Spot 執行個體請求狀態	2012-10-01	Spot 執行個體請求狀態可讓您輕鬆判斷 Spot 請求的狀態。	2012 年 10 月 14 日
Amazon EC2 預留執行個體 Marketplace	2012-08-15	預留執行個體 Marketplace 會將擁有不再需要之 Amazon EC2 預留執行個體的賣方，與想要購買額外容量的買方進行比對。經由預留執行個體 Marketplace 買賣的預留執行個體運作起來與任何其他預留執行個體相同，不同之處在於可以有低於完整標準的剩餘期限，並能以不同的價格出售。	2012 年 9 月 11 日
Provisioned IOPS SSD 適用於 Amazon EBS	2012-07-20	Provisioned IOPS SSD 磁碟區為依賴一致且快速回應時間的 I/O 密集型工作負載 (例如資料庫應用程式) 提供可預測的高效能。	2012 年 7 月 31 日

功能	API 版本	描述	發行日期
Amazon EC2 執行個體上的 IAM 角色	2012-06-01	<p>Amazon EC2 的 IAM 角色提供：</p> <ul style="list-style-type: none"> <li>• AWS 存取 Amazon EC2 執行個體上執行之應用程式的金鑰。</li> <li>• 自動輪替 Amazon EC2 執行個體上的 AWS 存取金鑰。</li> <li>• 針對在向您的 AWS 服務發出請求的 Amazon EC2 執行個體上執行的應用程式精細許可。</li> </ul>	2012 年 6 月 11 日
Spot 執行個體功能可讓您更輕鬆地開始處理可能的中斷情況。		<p>您現在可用下列方式管理 競價型執行個體：</p> <ul style="list-style-type: none"> <li>• 使用 Auto Scaling 啟動組態指定您願意為 競價型執行個體 支付的金額，並設定排程以指定願意為 競價型執行個體 支付的金額。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南中的 <a href="#">在 Auto Scaling 群組中啟動競價型執行個體</a>。</li> <li>• 啟動或終止執行個體時取得通知。</li> <li>• 使用 AWS CloudFormation 範本在包含 AWS 資源的堆疊中啟動 Spot 執行個體。</li> </ul>	2012 年 6 月 7 日
Amazon EC2 之 EC2 執行個體匯出及狀態檢查的時間戳記	2012-05-01	<p>新增將您原本匯入 EC2 的 Windows Server 執行個體匯出的支援。</p> <p>新增對執行個體狀態和系統狀態時間戳記的支援，以指出狀態檢查失敗的日期和時間。</p>	2012 年 5 月 25 日
Amazon VPC 之 EC2 執行個體匯出及執行個體和系統狀態檢查的時間戳記	2012-05-01	<p>新增將 EC2 執行個體匯出至 Citrix Xen、Microsoft Hyper-V 和 VMware vSphere 的支援。</p> <p>新增對執行個體和系統狀態檢查中時間戳記的支援。</p>	2012 年 5 月 25 日

功能	API 版本	描述	發行日期
AWS Marketplace AMI	2012-04-01	增加了對 AWS Marketplace AMI 的支持。	2012 年 4 月 19 日
預留執行個體定價方案	2011-12-15	新增說明如何利用預留執行個體定價方案內建的折扣定價之章節。	2012 年 3 月 5 日
Amazon Virtual Private Cloud 中 EC2 執行個體的彈性網絡介面 (ENI)	2011-12-01	新增 VPC 中 EC2 執行個體之彈性網絡介面 (ENI) 的相關章節。如需詳細資訊，請參閱 <a href="#">彈性網絡介面</a> 。	2011 年 12 月 21 日
新的 Amazon EC2 預留執行個體方案類型	2011-11-01	您可以從各種預留執行個體方案中選擇因應您預期使用的執行個體。	2011 年 12 月 01 日
Amazon EC2 執行個體狀態	2011-11-01	您可以檢視執行個體狀態的其他詳細資訊，包括可能對執行個體產生影響的排程事件。AWS 這些操作活動包括套用軟體更新或安全性修補程式所需的執行個體重新開機，或發生硬體問題時所需的執行個體淘汰。如需詳細資訊，請參閱 <a href="#">監控您的執行個體狀態</a> 。	2011 年 11 月 16 日
Amazon VPC 中的 競價型執行個體	2011-07-15	新增對 Amazon VPC 中 競價型執行個體支援的相關資訊。透過此更新，使用者可以在 Virtual Private Cloud (VPC) 中啟動 競價型執行個體。藉由在 VPC 中啟動 競價型執行個體，競價型執行個體的使用者就可以受益於 Amazon VPC。	2011 年 10 月 11 日
為 CLI 工具的使用者簡化 VM Import 程序	2011-07-15	VM Import 程序透過增強 ImportInstance 和 ImportVolume 功能而得以簡化，現在會在建立匯入任務之後將映像上傳至 Amazon EC2。此外，藉由引進 ResumeImport，使用者可以在任務停止位置重新啟動未完成的上傳。	2011 年 9 月 15 日

功能	API 版本	描述	發行日期
支援匯入 VHD 檔案格式		VM Import 現在可以匯入 VHD 格式的虛擬機器映像檔案。VHD 檔案格式與 Citrix Xen 和 Microsoft Hyper-V 虛擬化平台相容。在此版本中，VM Import 現在支援 RAW、VHD 和 VMDK (VMware ESX 相容) 映像格式。如需詳細資訊，請參閱 <a href="#">VM Import/Export 使用者指南</a> 。	2011 年 8 月 24 日
更新 Amazon EC2 VM Import Connector for VMware vCenter		新增 Amazon EC2 VM Import Connector for VMware vCenter 虛擬應用裝置 (連接器) 1.1 版的相關資訊。此更新包括對網際網路存取權的代理支援、更佳的錯誤處理、更精確的任務進度列，以及數種錯誤修正。	2011 年 6 月 27 日
競價型執行個體可用區域定價變更	2011-05-15	新增 競價型執行個體可用區域定價功能的相關資訊。在此版本中，我們新增了可用區域定價選項，做為您查詢 Spot 執行個體請求和 Spot 價格歷史記錄時傳回資訊的一部分。這些新增功能可讓您更輕鬆地判斷將 Spot 執行個體啟動至特定可用區域中所需的價格。	2011 年 5 月 26 日
AWS Identity and Access Management		已新增有關 AWS Identity and Access Management (IAM) 的資訊，讓使用者能夠指定使用者一般可用於 Amazon EC2 資源的 Amazon EC2 動作。如需詳細資訊，請參閱 <a href="#">Amazon EC2 的 Identity and Access Management</a> 。	2011 年 4 月 26 日
專用執行個體		專用執行個體是在您 Amazon Virtual Private Cloud (Amazon VPC) 內啟動的執行個體，這些執行個體在主機硬體層級上是實體相互隔離的。專用執行個體可讓您充分利用 Amazon VPC 和 AWS 雲端，其中包括隨需彈性佈建和僅按使用量付費，同時在硬體層級隔離 Amazon EC2 運算執行個體。如需詳細資訊，請參閱 <a href="#">專用執行個體</a> 。	2011 年 3 月 27 日

功能	API 版本	描述	發行日期
AWS 管理主控台的預留執行個體更新		AWS 管理主控台的更新可讓使用者更輕鬆地檢視其預留執行個體並購買額外的預留執行個體，包括專用預留執行個體。	2011 年 3 月 27 日
中繼資料資訊	2011-01-01	新增中繼資料的相關資訊，以反映 2011-01-01 版本中的變更。如需詳細資訊，請參閱 <a href="#">使用執行個體中繼資料</a> 和 <a href="#">執行個體中繼資料分類</a> 。	2011 年 3 月 11 日
Amazon EC2 VM Import Connector for VMware vCenter		新增 Amazon EC2 VM Import Connector for VMware vCenter 虛擬應用裝置 (連接器) 的相關資訊。此連接器是 VMware vCenter 的外掛程式，其與 VMware vSphere Client 整合，並提供可用來將 VMware 虛擬機器匯入 Amazon EC2 的圖形使用者介面。	2011 年 3 月 3 日
強制磁碟區分離		您現在可以使 AWS Management Console 用強制從執行個體分離 Amazon EBS 磁碟區。	2011 年 2 月 23 日
執行個體終止保護		您現在可以使用 AWS 管理主控台來防止執行個體終止。如需詳細資訊，請參閱 <a href="#">啟用終止保護</a> 。	2011 年 2 月 23 日
VM Import	2010-11-15	新增 VM Import 的相關資訊，該工具可讓您將虛擬機器或磁碟區匯入 Amazon EC2。如需詳細資訊，請參閱 <a href="#">VM Import/Export 使用者指南</a> 。	2010 年 12 月 15 日
執行個體的基本監控	2010-08-31	新增 EC2 執行個體之基本監控的相關資訊。	2010 年 12 月 12 日
篩選條件和標籤	2010-08-31	新增列出、篩選及建立標籤資源的相關資訊。如需詳細資訊，請參閱 <a href="#">列出與篩選您的資源</a> 和 <a href="#">標記您的 Amazon EC2 資源</a> 。	2010 年 9 月 19 日

功能	API 版本	描述	發行日期
等冪執行個體啟動	2010-08-31	新增在執行執行個體時確保等冪的相關資訊。	2010 年 9 月 19 日
AWS Identity and Access Management 對於 Amazon EC2		Amazon EC2 現在已與 AWS Identity and Access Management (IAM) 整合。如需詳細資訊，請參閱 <a href="#">Amazon EC2 的 Identity and Access Management</a> 。	2010 年 9 月 2 日
Amazon VPC IP 地址目標	2010-06-15	Amazon VPC 使用者現在可以指定 IP 地址，來指派在 VPC 中啟動的執行個體。	2010 年 7 月 12 日
Amazon CloudWatch 監控 Amazon EBS 卷		Amazon CloudWatch 監控現在可以自動用於 Amazon EBS 卷。	2010 年 6 月 14 日
Windows 預留執行個體		Amazon EC2 現在支援 Windows 預留執行個體。	2010 年 2 月 22 日

本文為英文版的機器翻譯版本，如內容有任何歧義或不一致之處，概以英文版為準。