



使用者指南

Amazon Linux 2023



Amazon Linux 2023: 使用者指南

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

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

Table of Contents

Amazon Linux 2023 是什麼？	1
發行節奏	1
主要版本和次要版本	3
使用新版本	3
長期支援政策	4
命名和版本控制	4
效能與操作最佳化	5
與 Fedora 的關係	6
自訂 cloud-init	6
安全更新和功能	8
受管更新	8
雲端內部的安全	8
SELinux 模式	8
合規計劃	9
SSH 伺服器預設	9
OpenSSL 3 的主要功能	9
網路服務	9
核心工具鏈套件 glibc、gcc、binutils	10
套件管理工具	10
預設 SSH 伺服器組態	11
過時的功能	13
compat- 套件	13
在 AL1 中停用的已棄用功能，已在 AL2 中移除	13
32 位元 x86 (i686) 網路晶片	14
aws-apitools-* 取代為 AWS CLI	14
systemd 在 AL2 upstart 中取代	15
在 AL2 中棄用的功能，並在 AL2023 中刪除	15
32 位元 x86 套件	16
aws-apitools-* 取代為 AWS CLI	16
bzip 修訂控制系統	16
C 組第 1 集	17
log4j 熱補丁 () log4j-cve-2021-44228-hotpatch	17
lsb_release 和 system-lsb-core 套件	17
mccrypt	17

OpenJDK 7 () java-1.7.0-openjdk	18
Python 2.7	18
rsyslog-openssl取代 rsyslog-gnutls	18
網路資訊服務 (NIS)/yp	18
在 AL2023 中已棄用	19
32 位元 x86 (i686) 執行階段支援	19
伯克利 DB (libdb)	19
cron	20
IMDSv1	20
pcre 第 1 版	20
System V init (sysvinit)	20
比較 AL2 與 AL2023	21
新增、升級和移除的套件	22
支援各個版本	22
命名和版本控制變更	22
最佳化	22
Python 2.7 已替換為 Python 3	23
安全性更新	23
SELinux	23
OpenSSL 3	23
IMDSV2	24
移除 log4j 熱修補 (log4j-cve-2021-44228-hotpatch)	24
穩定性的確定性升級	25
源自多個上游	25
AMI 根檔案系統和預設的 Amazon EBS 磁碟區類型	25
網路系統服務	25
整合控制群組階層 (cgroup v2)	26
任務排程	26
glibc、gcc 和 binutils 的套件	26
套件管理工具	27
日誌系統	27
curl 和 libcurl 的套件變更	27
GNU Privacy Guard (GNUPG)	27
Amazon Corretto 作為預設 JVM	28
AWS CLI V2	28
UEFI Preferred	28

SSH 伺服器預設組態變更	28
Extra Packages for Enterprise Linux (EPEL)	28
使用 cloud-init	29
圖形桌面支援	29
編譯器三元組	29
32 位元 x86 (i686) 套件	29
lsb_release 和 system-lsb-core 套件	30
AL2023 中的核心變更來自 AL2	30
重視安全的核心設定變更	30
其他核心設定變更	34
核心檔案系統支援	35
Amazon Linux 2 與 AL2023 AMI 的比較	40
Amazon Linux 2 與 AL2023 最小 AMI 的比較	73
Amazon Linux 2 與 AL2023 容器的比較	94
比較 AL1 與 AL2023	102
支援各個版本	102
systemd 取代 upstart 並成為 init 系統	102
Python 2.6 和 2.7 已替換為 Python 3	103
OpenJDK 8 作為最舊版 JDK	103
AL2023 中的內核更改從 AL1	103
Kernel Live Patching	103
核心檔案系統支援	103
重視安全的核心設定變更	105
其他核心設定變更	107
AL1 與 AL2023 AMI 比較	108
AL1 與 AL2023 最小 AMI 比較	141
AL1 和 AL2023 容器的比較	161
系統要求	170
執行 AL2023 的 CPU 需求	170
AL2023 的 ARM CPU 需求	170
AL2023 的 x86-64 CPU 需求	171
執行 AL2023 的記憶體 (記憶體) 需求	171
在上使用 AL2023 AWS	173
開始使用 AWS	173
註冊一個 AWS 帳戶	173
建立具有管理權限的使用者	174

授予程式設計存取權	175
Amazon EC2 上的 AL2023	176
使用亞馬遜主控台啟動 AL2023	177
使用超音波馬達參數啟動 AL2023 AWS CLI	178
使用推出最新的 AL2023 AMI AWS CloudFormation	179
使用特定的 AMI 識別碼啟動 AL2023	180
AL2023 AMI 棄用和生命週期	181
連線至 AL2023 執行個體	181
比較 AL2023 標準 (預設) 與最小 AMI	182
容器中的 AL2023	209
AL2023 基本容器映像	210
AL2023 最小容器圖像	212
構建裸露的 AL2023 容器映像	213
AL2023 容器映像套件清單比較	217
AL2023 最小 AMI 與容器映像的比較	223
Elastic Beanstalk 上的 AL2023	239
AL2023 CloudShell	240
適用於 Amazon ECS 容器主機的 AL2023	240
Amazon ECS 自 AL2 以來的相關變化	241
自訂 Amazon ECS 最佳化 AMI	241
AL2023 上的 Amazon EFS	242
amazon-efs-utils	242
掛載 Amazon EFS 檔案系統	242
AL2023 上的 Amazon EMR	243
基於 AL2023 的 Amazon EMR 發布	243
EKS 上基於 Amazon EMR 的 AL2023	243
AL2023 上 AWS Lambda	243
provided.al2023 Lambda 執行期	243
基於 AL2023 的執行期	243
教學課程	245
在 AL2023 上安裝燈泡	245
步驟 1：準備 LAMP 伺服器	246
步驟 2：測試您的 LAMP 伺服器	250
步驟 3：保護資料庫伺服器	252
步驟 4：(可選) 安裝 phpMyAdmin	253
疑難排解	256

相關主題	256
在 AL2023 上設定傳輸安全傳輸埠	257
必要條件	258
步驟 1：在伺服器上啟用 TLS	259
步驟 2：取得 CA 簽署的憑證	261
步驟 3：測試和強化安全組態	268
疑難排解	271
在 AL2023 上主持一個 WordPress 博客	272
必要條件	273
安裝 WordPress	273
後續步驟	282
說明! 我的公有 DNS 名稱曾經變更且現在部落格無法使用	283
Amazon EC2 之外的 AL2023	285
下載 AL2023 VM 映像	285
支援的組態	285
KVM 需求	286
VMware需求	288
超 V 需求	290
AL2023 VM 組態	291
NoCloud seed.iso 基於組態	292
VMware基於客戶信息的配置	295
AL2023 標準 AMI 和 KVM 映像檔的封裝清單比較	297
標準 AMI 和 VMware OVA 映像的 AL2023 套件清單比較	322
標準 AMI 和超 V 圖像的 AL2023 軟件包列表比較	347
更新 AL2023	373
接收有關新更新的通知	373
管理更新	374
檢查可用的套件更新	374
使用 DNF 和儲存庫版本套用安全更新	376
(安全性) 更新後自動重新啟動服務	378
啟動已啟用最新儲存庫版本的執行個體	379
取得套件支援資訊	380
檢查較新的儲存庫版本	380
新增、啟用或停用新儲存庫	383
使用 cloud-init 新增儲存庫	386
透過 AL2023 上的版本化儲存庫使用確定性升級	387

控制從主要和次要版本收到的更新	387
主要和次要版本升級之間的差異	387
控制 AL2023 儲存庫中可用的套件更新	387
透過使用版本化儲存庫進行確定性升級	388
Kernel Live Patching	394
限制	394
支援的組態和先決條件	395
使用 Kernel Live Patching	395
程式設計語言和執行期	401
C/C++ 和 Fortran	401
Go	402
AL2023 Lambda 函數：Go	402
Java	402
Perl	403
Perl 模組	403
PHP	403
遷移至新 PHP 版本	403
從 PHP 7.x 遷移	404
PHP 模組	404
Python	404
Python 模組	405
Rust	405
AL2023 Lambda 函數：Rust	405
安全與合規	406
安全性公告	406
ALAS 公告	407
ALAS 常見問題集	407
設定 AL2023 的環境管理系統模式	407
AL2023 的預設環境管理系統狀態和模式	408
變更為 enforcing 模式	408
停用 SELinux 的選項	410
在 AL2023 上啟用 FIPS 模式	411
核心強化	412
核心強化選項 (獨立於架構)	413
x86-64 特定核心強化選項	424
aarch64 特定核心強化選項	427

AL2023 上的 UEFI 安全開機	428
在 AL2023 上啟用 UEFI 安全開機	429
註冊現有執行個體	429
從快照註冊映像	430
撤銷更新	430
UEFI 安全開機如何在 AL2023 上運作	431
註冊自有金鑰	431
.....	cdxxxii

Amazon Linux 2023 是什麼？

Amazon Linux 2023 (AL2023) 是 Amazon 下一代亞馬遜 Linux 從 Amazon Web Services (AWS)。使用 AL2023，您可以在安全、穩定且高效能的執行階段環境中開發和執行雲端和企業應用程式。此外，您還可以獲得一個提供長期支持的應用程式環境，並獲得 Linux 的最新創新技術。使用 AL2023 無需負擔額外費用。

AL2023 是 Amazon Linux 2 (AL2) 的繼任者。如需有關 AL2023 和 AL2 之間差異的詳細資訊，請參閱[比較 AL2 與 AL2023](#)和[AL2023 中的 Package 變更](#)。








主題







- [發行節奏](#)
- [命名和版本控制](#)
- [效能與操作最佳化](#)
- [與 Fedora 的關係](#)
- [自訂 cloud-init](#)
- [安全更新和功能](#)
- [網路服務](#)
- [核心工具鏈套件 glibc、gcc、binutils](#)
- [套件管理工具](#)
- [預設 SSH 伺服器組態](#)

發行節奏

Amazon Linux 的新主要版本每兩年發布一次，並包括五年的支援。每個發行版本包括兩個階段的支援。標準支援階段涵蓋前兩年。接下來，維護階段將繼續支援三年。

在標準支援階段，發行版本會每季收到次要版本更新。在維護階段，發行版本只會收到在可用時即發布的安全更新和重要錯誤修正。

年	Amazon Linux 2023	Amazon Linux 2025	Amazon Linux 2027	Amazon Linux 2029
2023	 標準支援			
2024	 標準支援			
2025	Maintenance (維護)	 標準支援		
2026	Maintenance (維護)	 標準支援		
2027	Maintenance (維護)	Maintenance (維護)	 標準支援	
2028	 EOL	Maintenance (維護)	 標準支援	

年	Amazon Linux 2023	Amazon Linux 2025	Amazon Linux 2027	Amazon Linux 2029
2029	 EOL	Maintenance (維護)	Maintenance (維護)	 標準支援
2030	 EOL	 EOL	Maintenance (維護)	 標準支援
2031	 EOL	 EOL	Maintenance (維護)	Maintenance (維護)

主要版本和次要版本

在每個 Amazon Linux 版本 (主要版本、次要版本或安全版本) 中，我們都會發布新的 Linux Amazon Machine Image (AMI)。

- **主要版本發行** — 包括新功能，以及在整個堆疊中的安全性和效能改進。這些改進可能包括對核心、工具鏈、Glib C、OpenSSL，以及任何其他系統程式庫和公用程式的重大變更。Amazon Linux 的主要版本部分基於上游 Fedora Linux 發行版的目前版本。AWS 可能會新增或取代其他非 Fedora 上游的特定套件。
- **次要版本** — 每季更新，包含安全更新、錯誤修正，以及新功能和套件。每個次要版本都是累積的更新清單，除了新功能和套件之外，還包括安全和錯誤修正。這些發行版本可能包含最新的語言執行期，例如 PHP。另外還可能包括其他常見的套件，如 Ansible 和 Docker。

使用新版本

更新將透過新的 Amazon Machine Image (AMI) 發行版和對應的新儲存庫組合進行。預設情況下，新的 AMI 會與其指向的儲存庫耦合。不過，您可以隨著時間經過，將執行中的 Amazon EC2 執行個體指

向較新的儲存庫版本，以便在執行中的執行個體套用更新。您也可以啟動最新 AMI 的新執行個體來進行更新。

長期支援政策

Amazon Linux 會為您的所有套件提供更新，並在主要版本中維持在 Amazon Linux 上建置的應用程式相容性。核心套件 (例如 glibc 程式庫、OpenSSL、OpenSSH 和 DNF 套件管理員) 會獲得 AL2023 主要發行版本壽命週期的支援。不屬於核心套件的套件會根據其特定的上游來源獲得支援。您可以執行下列命令來查看個別套件的特定支援狀態和日期。

```
$ sudo dnf supportinfo --pkg packagename
```

您可以執行下列命令來取得所有目前已安裝套件的資訊。

```
$ sudo dnf supportinfo --show installed
```

核心套件的完整清單會在預覽期間完成。如果您想查看更多核心套件包含的套件，請告訴我們。我們在收集意見回饋時進行評估。您可以透過您指定的 AWS 代表或在 GitHub 的 [amazon-linux-2023 儲存庫](#) 提出問題，來提供有關 AL2023 的意見回饋。

命名和版本控制

AL2023 在兩年的標準支援期間，每三個月提供一次小版本。每個發行版本都會以 0 到 N 之間的增量來識別。0 是指該版序的原始主要發行版本。所有版本都將稱為 Amazon Linux 2023。當 Amazon Linux 2025 發布時，AL2023 將進入擴展支援，並接收有關安全更新和重大錯誤修正的更新。

例如，AL2023 的次要版本具有以下格式：

- 2023.0.20230301
- 2023.1.20230601
- 2023.2.20230901

對應的 AL2023 AMI 格式如下：

- al2023-ami-2023.0.20230301.0-kernel-6.1-x86_64
- al2023-ami-2023.1.20230601.0-kernel-6.1-x86_64
- al2023-ami-2023.2.20230901.0-kernel-6.1-x86_64

在特定的次要版本中，一般 AMI 發行版本會以 AMI 發行日期的時間戳記發行。

- al2023-ami-2023.0.20230301.0-kernel-6.1-x86_64
- al2023-ami-2023.0.20230410.0-kernel-6.1-x86_64
- al2023-ami-2023.0.20230520.0-kernel-6.1-x86_64

識別 AL2 或 AL2023 執行個體的建議方法從讀取通用平台列舉 (CPE) 字串開始。/etc/system-release-cpe 然後，將字串拆為字段。最後，閱讀平台和版本值。

AL2023 還導入用於平台識別的新檔案：

- /etc/amazon-linux-release symlinks 至 /etc/system-release
- /etc/amazon-linux-release-cpe symlinks 至 /etc/system-release-cpe

這兩個檔案指出執行個體是 Amazon Linux。除非您想知道特定的平台和版本值，否則無需讀取檔案或將字串拆為字段。

效能與操作最佳化

Amazon 6.1 核心

- AL2023 使用彈性網路介面卡 (ENA) 和彈性網狀架構介面卡 (EFA) 裝置的最新驅動程式。AL2023 著重於 Amazon EC2 基礎設施中硬體的效能和功能反向移植。
- 核心即時修補程式可用於 x86_64 和 aarch64 執行個體類型。這會減少頻繁重新開機的需求。
- 所有內核構建和運行時配置都包含 AL2 的許多相同的性能和操作優化。

基本工具鏈選項和預設建置旗標

- AL2023 軟件包是使用默認啟用的編譯器優化 (-O2) 構建的
- AL2023 套件的建置 x86-64v2 需要 x86-64 系統 (-march=x86-64-v2)，以及 aarch64 (-march=armv8.2-a+crypto -mtune=neoverse-n1) 的 Graviton 2 或更新版本。
- AL2023 套件是以啟用自動向量化 (-ftree-vectorize) 建置。
- AL2023 套件是在啟用連結時間最佳化 (LTO) 的情況下建置。
- AL2023 使用 Rust、Clang/LLVM 和 Go 的更新版本。

套件選項和版本

- 選擇主要系統元件的反向移植包括在 Amazon EC2 基礎設施上執行的多項效能改進，尤其是 Graviton 執行個體。
- AL2023 與多個 AWS 服務 功能集成在一起。這包括 SSM 代理程式、Amazon Kinesis 代理程式和 AWS CLI CloudFormation
- AL2023 使用 Amazon Corretto 作為 Java 開發套件 (JDK)。
- AL2023 為較新版本提供資料庫引擎和程式設計語言執行期更新，因為它們由上游專案發布。新版本的程式設計語言執行期會在發布時新增。

在雲端環境中部署

- 基礎 AL2023 AMI 和容器映像會經常更新，以支援更換修補執行個體。
- 核心更新包含在 AL2023 AMI 更新中。這代表您不需要使用 `yum update` 和 `reboot` 等的命令來更新核心。
- 除了標準的 AL2023 AMI 之外，還提供最小 AMI 和容器映像。選擇最小 AMI，以使用執行服務所需的最少套件數量來執行環境。
- 預設情況下，AL2023 AMI 和容器會鎖定至特定版本的套件儲存庫。啟動時沒有自動更新。這表示您可以隨時掌控導入任何套件更新的時間。在推出生產前，您可以隨時在 beta/gamma 環境中進行測試。如有問題，您可以使用預先驗證的復原路徑。

與 Fedora 的關係

AL2023 會維護自己的發行版本，並且支援獨立於 Fedora 的生命週期。AL2023 提供開放原始碼軟體的更新版本、更多種類的套件，以及頻繁的發行版本。這樣可以保留熟悉的 RPM 架構作業系統。

AL2023 的正式上市 (GA) 版本無法直接與任何特定的 Fedora 發行版本相比。AL2023 GA 版本包含來自 Fedora 34、35 和 36 的元件。有些元件與 Fedora 中的元件相同，有些元件則經過修改。其他元件更接近 CentOS 9 Streams 中的元件，或者是獨立開發元件。Amazon Linux 核心源自 kernel.org 上的長期支援選項，並從 Fedora 中獨立選出。

自訂 cloud-init

cloud-init 套件是一種開放原始碼應用程式，可在雲端運算環境中引導 Linux 映像。如需詳細資訊，請參閱[雲端初始化文件](#)。

AL2023 包含 cloud-init 的自訂版本。使用 cloud-init 時，您可以指定執行個體在開機時執行的操作。

啟動執行個體時，您可以使用使用者資料欄位將動作傳遞給 cloud-init。這表示您可在許多使用案例下用常用 Amazon Machine Image (AMI)，並在啟動執行個體時進行動態設定。AL2023 也會用 cloud-init 來設定 ec2-user 帳號。

AL2023 會在 /etc/cloud/cloud.cfg.d 和 /etc/cloud/cloud.cfg 中使用 cloud-init 操作。您可以在 /etc/cloud/cloud.cfg.d 目錄中建立自己的 cloud-init 操作檔案。Cloud-init 會以字母順序讀取此目錄中的所有檔案。新檔案會覆寫舊檔案中的值。cloud-init 啟動執行個體時，cloud-init 套件會執行下列設定工作：

- 設定預設的地區設定
- 設定主機名稱
- 剖析和處理使用者資料
- 產生主機私有 SSH 金鑰
- 將使用者的公有 SSH 金鑰加入至 .ssh/authorized_keys，以方便登入和管理
- 準備儲存庫以管理套件
- 處理使用者資料中定義的套件動作
- 執行使用者資料中的使用者指令碼
- 掛載執行個體儲存體磁碟區 (如適用)
 - 根據預設，若 ephemeral0 執行個體儲存體存在，且包含有效的檔案系統，執行個體儲存體會掛載在 /media/ephemeral0；否則將不會掛載。
 - 根據預設，對於 m1.small 和 c1.medium 執行個體類型，將掛載任何與執行個體關聯的置換磁碟區。
- 您可用下列的 cloud-init 指令覆寫預設的執行個體儲存體磁碟區掛載：

```
#cloud-config
mounts:
- [ ephemeral0 ]
```

如需進一步掌控掛載，請參閱 cloud-init 文件中的[掛載](#)。

- 執行個體啟動時，不會格式化支援 TRIM 的執行個體儲存體磁碟區。掛載執行個體儲存體磁碟區前，您必須先分割和格式化執行個體儲存體磁碟區。

如需詳細資訊，請參閱 Amazon EC2 使用者指南中的[執行個體存放區磁碟區 TRIM 支援](#)。

- 您可在啟動執行個體時，使用 disk_setup 模組分割及格式化執行個體儲存體磁碟區。

如需詳細資訊，請參閱 cloud-init 文件中的[磁碟設定](#)。

如需使用 cloud-init 與 SELinux 搭配的相關資訊，請參閱[使用 cloud-init 來啟用 enforcing 模式](#)。

如需有關 cloud-init 使用者資料格式的資訊，請參閱 cloud-init 文件中的[使用者資料格式](#)。

安全更新和功能

AL2023 提供許多安全性更新和解決方案。

主題

- [受管更新](#)
- [雲端內部的安全](#)
- [SELinux 模式](#)
- [合規計劃](#)
- [SSH 伺服器預設](#)
- [OpenSSL 3 的主要功能](#)

受管更新

使用 DNF 和儲存庫版本套用安全性更新。如需詳細資訊，請參閱[在 AL2023 中管理套件和作業系統更新](#)。

雲端內部的安全

安全是 AWS 與您之間共同承擔的責任。[共同的責任模型](#)將其描述為雲端的安全性和雲端中的安全性。如需詳細資訊，請參閱[Amazon 2 中的安全性和合規性](#)。

SELinux 模式

預設情況下，SELinux 處於啟用狀態，並在 AL2023 中設置為寬容模式。在寬容模式中，會記錄權限遭拒的情形，但不會強制執行。

SELinux 政策定義使用者、處理程序、程式、檔案和裝置的權限。使用 SELinux 時，您可以選擇兩個政策之一。這些政策是目標式或多層級安全性 (MLS)。

如需有關 SELinux 模式和政策的詳細資訊，請參閱 [設定 AL2023 的環境管理系統模式](#) 和 [SELinux Project 維基](#)。

合規計劃

獨立稽核員會評估 AL2023 的安全性與合規性，以及許多 AWS 法規遵循方案。

SSH 伺服器預設

AL2023 包括 OpenSSH 8.7。OpenSSH 8.7 預設停用 ssh-rsa 金鑰交換算法。如需詳細資訊，請參閱 [預設 SSH 伺服器組態](#)。

OpenSSL 3 的主要功能

- 憑證管理通訊協定 (CMP, RFC 4210) 同時包含 CRMF (RFC 4211) 和 HTTP 傳輸 (RFC 6712)。
- libcrypto 的 HTTP 或 HTTPS 用戶端支援 GET 和 POST 操作、重新導向、普通和 ASN.1 編碼內容、代理和逾時。
- EVP_KDF 與金鑰衍生函數搭配使用。
- EVP_MAC API 可搭配 MACs 使用。
- Linux 核心 TLS 支援。

如需詳細資訊，請參閱 [《OpenSSL 遷移指南》](#)。

網路服務

該開放原始碼專案 systemd-networkd 在現代 Linux 發行版中廣泛使用。該專案使用類似 systemd 架構其餘部分的聲明式組態語言。其主要組態檔案類型為 .network 和 .link 檔案。

amazon-ec2-net-utils 套件會在 /run/systemd/network 目錄中產生介面特定組態。這些組態會在連接至執行個體時，在介面上啟用 IPv4 和 IPv6 網路。這些組態也會安裝政策路由規則，以協助確保透過對應執行個體的網路介面，將本機來源的流量路由至網路。這些規則可確保正確的流量透過彈性網路介面 (ENI) 從關聯的位址或首碼路由傳送。如需使用 ENI 的詳細資訊，請參閱 Amazon EC2 使用者指南 [中的使用 ENI](#)。

您可以將自訂組態檔案放入 /etc/systemd/network 目錄，以覆寫 /run/systemd/network 中包含的預設組態設定並自訂此網路行為。

[systemd.network](#) 說明文件說明 `systemd-networkd` 服務如何判斷適用於特定介面的組態。它也會為 ENI 支援的介面產生替代名稱 (稱為) `altnames`，以反映各種 AWS 資源的屬性。這些 ENI 支援的介面屬性是 ENI 附件的 ENI ID 和 `DeviceIndex` 欄位。使用各種工具 (例如 `ip` 命令) 時，您可以使用這些介面的屬性來參照這些介面。

AL2023 實例介面名稱是使用 `systemd` 插槽命名配置產生的。如需詳細資訊，請參閱 [systemd.net 命名方案](#)。

此外，AL2023 預設使用 `fq_code1` 主動佇列管理網路傳輸排程演算法。如需詳細資訊，請參閱 [CoDel 概觀](#)。

核心工具鏈套件 glibc、gcc、binutils

Amazon Linux 的套件事子集指定為核心工具鏈套件。作為 AL2023 的主要組成部分，核心軟件包獲得了五年的支持。我們可能會變更套件的版本，但是，長期支援適用於 Amazon Linux 發行版中包含的套件。

這三個核心套件提供一個系統工具鏈，可用在 Amazon Linux 發行版中建置大多數軟體。

套件	定義	用途
glibc 2.34	系統 C 程式庫	獲得大多數提供標準功能的二進位程式採用，以及用於程式與核心間的介面。
gcc 11.2	gcc 編譯器套件	編譯 C、C++、Fortran。
binutils 2.35	組譯工具、連結器以及其他二進位工具	操縱或檢查二進位程式。

建議對任何 `glibc` 程式庫進行更新後皆需重新啟動。對於控制服務的套件更新，可能只需要重新啟動服務便可套用更新。但是，系統重新啟動可確保所有先前的套件和程式庫更新都已完成。

套件管理工具

AL2023 中的預設軟體套件管理工具為 `DNF`。`DNF` 是 AL2 中的軟件包管理工具的后續任務。`YUM`

`DNF` 的用途類似 `YUM`。許多 `DNF` 指令和指令選項與 `YUM` 指令相同。在命令列介面 (CLI) 命令中，在大多數情況下 `dnf` 會取代 `yum`。

例如，對於以下 AL2 yum 命令：

```
$ sudo yum install packagename
$ sudo yum search packagename
$ sudo yum remove packagename
```

在 AL2023 中，它們成為以下命令：

```
$ sudo dnf install packagename
$ sudo dnf search packagename
$ sudo dnf remove packagename
```

在 AL2023 中，yum 命令仍然可用，但作為指向 dnf 命令的指標。因此，當 yum 命令在 Shell 中或指令碼中使用時，所有命令和選項都與 DNF CLI 相同。有關 YUM CLI 和 DNF CLI 之間差異的詳細資訊，請參閱 [DNF CLI 與 YUM 相比的變更](#)。

如需 dnf 命令的命令和選項完整參考，請參閱 man 頁面 `man dnf`。如需詳細資訊，請參閱 [〈DNF 指令參考〉](#)。

預設 SSH 伺服器組態

如果您使用多年前的 SSH 用戶端，連線至執行個體時可能會看到錯誤訊息。如果錯誤告訴您找不到相符的主機金鑰類型，請更新 SSH 主機金鑰以解決此問題。

預設停用 `ssh-rsa` 簽章

AL2023 包含一個預設組態，可停用舊版 `ssh-rsa` 主機金鑰演算法，並產生一組縮減的主機金鑰。用戶端必須支援 `ssh-ed25519` 或 `ecdsa-sha2-nistp256` 主機金鑰演算法。

預設組態接受下列任何金鑰交換演算法：

- `curve25519-sha256`
- `curve25519-sha256@libssh.org`
- `ecdh-sha2-nistp256`
- `ecdh-sha2-nistp384`
- `ecdh-sha2-nistp521`
- `diffie-hellman-group-exchange-sha256`
- `diffie-hellman-group14-sha256`

- `diffie-hellman-group16-sha512`
- `diffie-hellman-group18-sha512`

預設情況下，AL2023 會產生 `ed25519` 和 `ECDSA` 主機金鑰。用戶端支援 `ssh-ed25519` 或 `ecdsa-sha2-nistp256` 主機金鑰演算法。當您透過 SSH 連線至執行個體時，您必須使用支援相容演算法的用戶端，例如 `ssh-ed25519` 或 `ecdsa-sha2-nistp256`。如果您需要使用其他金鑰類型，請使用使用者資料中的 `cloud-config` 片段覆寫生成的金鑰清單。

在下列範例中，`cloud-config` 使用 `ecdsa` 和 `ed25519` 金鑰產生 `rsa` 主機金鑰。

```
#cloud-config
ssh_genkeytypes:
- ed25519
- ecdsa
- rsa
```

如果您使用 RSA 金鑰對進行公開金鑰驗證，您的 SSH 用戶端必須支援 `rsa-sha2-256` 或 `rsa-sha2-512` 簽章。如果您使用不相容的用戶端且無法升級，請重新啟用執行個體的 `ssh-rsa` 支援。要重新啟用 `ssh-rsa` 支持，請使用以下命令激活 LEGACY 系統加密策略。

```
$ sudo dnf install crypto-policies-scripts
$ sudo update-crypto-policies --set LEGACY
```

如需有關管理主機金鑰的詳細資訊，請參閱 [Amazon Linux 主機金鑰](#)。

AL2023 中已淘汰的功能

在 AL2 中棄用且不存在於 AL2023 中的功能在此處記錄。這是功能，例如 AL2 中存在的功能和軟件包，但不在 AL2023 中，並且不會添加到 AL2023 中。如需 AL2 中支援此功能之時間長度的詳細資訊，請參閱 AL2 [中已停用的功能](#)。

AL2023 中還有已棄用的功能，並且將在 future 的版本中刪除。本章說明該功能的含義、何時不再支援，以及何時將從 Amazon Linux 中移除該功能。瞭解已淘汰的功能將有助於您部署 AL2023，並為下一個主要版本的 Amazon Linux 做好準備。

主題

- [compat- 套件](#)
- [在 AL1 中停用的已棄用功能，已在 AL2 中移除](#)
- [在 AL2 中棄用的功能，並在 AL2023 中刪除](#)
- [在 AL2023 中已棄用](#)

compat- 套件

AL2 中任何前置詞為的套件，compat- 都是為了與尚未針對現代套件重建的舊二進位檔案的二進位檔相容性而提供的。Amazon Linux 的每個新主要版本都不會從先前發行版本傳 compat- 送任何套件。

Amazon Linux 發行版本（例如 AL2）中的所有 compat- 軟件包都已被棄用，並且不會出現在後續版本中（例如 AL2023）。我們強烈建議您針對程式庫的更新版本重建軟體。

在 AL1 中停用的已棄用功能，已在 AL2 中移除

本節說明 AL1 中可用且在 AL2 中不再提供的功能。

Note

作為 AL1 維護支援階段的一部分，某些套件的 end-of-life (EOL) 日期早於 AL1 的 EOL。如需詳細資訊，請參閱 [AL1 Package 支援陳述式](#)。

Note

舊版中已停止部分 AL1 功能。如需詳細資訊，請參閱 [AL1 版本說明](#)。

主題

- [32 位元 x86 \(i686\) 網路晶片](#)
- [aws-apitools-* 取代為 AWS CLI](#)
- [systemd 在 AL2 upstart 中取代](#)

32 位元 x86 (i686) 網路晶片

作為 [AL1 2014.09 版本](#) 的一部分，Amazon Linux 宣布這將是生產 32 位 AMI 的最後一個版本。因此，從 [2015.03 版 AL1](#) 開始，Amazon Linux 不再支援以 32 位元模式執行系統。AL2 針對 x86-64 主機上的 32 位元二進位檔案提供有限的執行階段支援，且不提供開發套件來建置新的 32 位元二進位檔案。AL2023 不再包含任何 32 位元使用者空間套件。我們建議使用者先完成轉換為 64 位元程式碼，再移轉至 AL2023。

如果您需要在 AL2023 上運行 32 位二進製文件，則可以在運行在 AL2023 之上的 AL2 容器中使用 AL2 的 32 位用戶空間。

aws-apitools-* 取代為 AWS CLI

AWS CLI 在 2013 年 9 月發行之前，提供 AWS 了一組可用的命令列公用程式，並在中實作 Java，允許使用者進行 Amazon EC2 API 呼叫。這些工具於 2015 年停產，AWS CLI 成為透過命令列與 Amazon EC2 API 互動的首選方式。命令列公用程式集包含下列 aws-apitools-* 套件。

- aws-apitools-as
- aws-apitools-cfn
- aws-apitools-common
- aws-apitools-ec2
- aws-apitools-elb
- aws-apitools-mon

aws-apitools-* 套件的上游支援已於 2017 年 3 月結束。儘管缺乏上游支援，但 Amazon Linux 仍繼續提供其中一些命令列公用程式，例如為 aws-apitools-ec2 使用者提供向後相容性。這 AWS CLI

是一個比aws-apitools-*軟件包更強大和完整的工具，因為它是積極維護的，並提供了使用所有AWS API的方法。

這些aws-apitools-*套件已於2017年3月棄用，而且不會收到進一步的更新。任何這些套件的所有使用者都應該AWS CLI盡快移轉至。這些套件不存在於AL2023中。

AL1還提供了aws-apitools-iam和aws-apitools-rds軟件包，這些軟件包在AL1中已被棄用，並且在AL2以後的Amazon Linux中不存在。

systemd在AL2 upstart 中取代

AL2是第一個使用systemd初始化系統的Amazon Linux版本，upstart在AL1中取代。在從AL1遷移到較新版本的Amazon Linux時，必須變更任何upstart特定組態。無法systemd在AL1上使用，因此只systemd能在轉移upstart到最新主要版本的Amazon Linux（例如AL2或AL2023）的過程中才能完成。

在AL2中棄用的功能，並在AL2023中刪除

本節描述了AL2中可用且不再在AL2023中可用的功能。

主題

- [32 位元 x86 套件](#)
- [aws-apitools-*取代為 AWS CLI](#)
- [bzip2修訂控制系統](#)
- [C 組第 1 集](#)
- [log4j熱補丁 \(\) log4j-cve-2021-44228-hotpatch](#)
- [lsb_release 和 system-lsb-core 套件](#)
- [mccrypt](#)
- [OpenJDK 7 \(\) java-1.7.0-openjdk](#)
- [Python 2.7](#)
- [rsyslog-openssl取代 rsyslog-gnutls](#)
- [網路資訊服務 \(NIS\)/yp](#)

32 位元 x86 套件

作為 [AL1 2014.09 版本](#) 的一部分，我們宣布這將是產生 32 位元 AMI 的最後一個版本。因此，從 [2015.03 版 AL1](#) 開始，Amazon Linux 不再支援以 32 位元模式執行系統。AL2 針對 x86-64 主機上的 32 位元二進位檔案提供有限的執行階段支援，且不提供開發套件來建置新的 32 位元二進位檔案。AL2023 不再包含任何 32 位元使用者空間套件。我們建議客戶完成轉換為 64 位元程式碼。

如果您需要在 AL2023 上運行 32 位二進製文件，則可以在運行在 AL2023 之上的 AL2 容器中使用 AL2 的 32 位用戶空間。

aws-apitools-* 取代為 AWS CLI

AWS CLI 在 2013 年 9 月發行之前，提供 AWS 了一組可用的命令列公用程式，並在中實作 Java，讓客戶可以進行 Amazon EC2 API 呼叫。這些工具在 2015 年已淘汰，AWS CLI 成為透過命令列與 Amazon EC2 API 互動的首選方式。這包括以下 aws-apitools-* 軟件包。

- aws-apitools-as
- aws-apitools-cfn
- aws-apitools-common
- aws-apitools-ec2
- aws-apitools-elb
- aws-apitools-mon

aws-apitools-* 套件的上游支援已於 2017 年 3 月結束。儘管缺乏上游支援，但 Amazon Linux 仍繼續提供其中一些命令列公用程式 (例如 aws-apitools-ec2)，以便為客戶提供向後相容性。這 AWS CLI 是一個比 aws-apitools-* 軟件包更強大和完整的工具，因為它是積極維護的，並提供了使用所有 AWS API 的方法。

這些 aws-apitools-* 套件已於 2017 年 3 月棄用，且不會收到進一步的更新。任何這些套件的所有使用者都應該 AWS CLI 盡快移轉至。這些套件不存在於 AL2023 中。

bzr 修訂控制系統

[GNU Bazaar](#) (bzr) 修訂版控制系統已在 AL2 中停止，並且不再存在於 AL2023 中。

建議使用者將其儲存庫移轉至 git。bzr

C 組第 1 集

AL2023 會移至統一控制群組階層 (cgroup v2)，而 AL2 則使用群組第 1 版。由於 AL2 不支持 cgroup v2，因此此遷移需要在移至 AL2023 的過程中完成。

log4j熱補丁 () **log4j-cve-2021-44228-hotpatch**

Note

該log4j-cve-2021-44228-hotpatch軟件包已在 AL2 中棄用，並在 AL2023 中刪除。

為了響應 [CVE-2021-44228](#)，Amazon Linux 發布了對於 AL1 和 AL2 的阿帕奇 Log4j 的熱補丁的 RPM 打包版本。在[向 Amazon Linux 添加修補程序的公告中](#)，我們注意到「安裝補丁程序不能替代更新到緩解 CVE-2021-44228 或 CVE-2021-45046 的 log4j 版本」。

該熱修補是一種緩解措施，讓您有時間修補 log4j。AL2023 的第一個正式發行版本是在 [CVE-2021-44228](#) 之後的 15 個月，因此 AL2023 不會隨附修補程式 (啟用與否) 一起出貨。

建議在 Amazon Linux 上執行自有 log4j 版本的客戶確認已更新至不受 [CVE-2021-44228](#) 或 [CVE-2021-45046](#) 影響的版本。

lsb_release 和 **system-lsb-core** 套件

在過去，某些軟體會調用 lsb_release 命令 (在 AL2 中由套件 system-lsb-core 提供) 以取得關於執行軟體所用的 Linux 發行版資訊。Linux 標準規範 (LSB) 已導入此命令，且 Linux 發行版已加以採用。Linux 發行版已發展為使用更簡單的標準，以將此資訊保留在 /etc/os-release 和其他相關檔案內。

os-release 標準源自 systemd。如需詳細資訊，請參閱[系統作業系統版本文件](#)。

AL2023 不隨附 lsb_release 命令，也不包含 system-lsb-core 套件。軟體應完成 os-release 標準的轉換，以維持與 Amazon Linux 和其他主要 Linux 發行版的相容性。

mcrypt

程式mcrypt庫和關聯的PHP擴充功能已在 AL2 中棄用，並且不再存在於 AL2023 中。

上游PHP棄用了 [PHP 7.1 中的mcrypt擴充功能](#)，該擴充功能於 2016 年 12 月首次發行，並於 2019 年 10 月發行了最終版本。

上游mccrypt庫最後在 2007 年發布了一個版本，並沒有從 2017 年新提交SourceForge 所需的cvs版本控制遷移，最近的提交（並且僅在 3 年之前）是從 2011 年開始刪除具有維護者的項目的提及。

建議任何剩餘mccrypt的使用者將其程式碼移植至OpenSSL，因為不mccrypt會新增至 AL2023。

OpenJDK 7 () **java-1.7.0-openjdk**

Note

AL2023 提供多種版本的 [Amazon Corretto](#) 來支援Java基於工作負載。OpenJDK 7 套件已在 AL2 中被棄用，而且不再存在於 AL2023 中。AL2023 中可用的最古老的 JDK 由 Corretto 8 提供。

如需 Amazon Linux 上 Java 的詳細資訊，請參閱[AL2023 中的 Java](#)。

Python 2.7

Note

AL2023 已移除 Python 2.7，任何需要 Python 的作業系統組件都編寫為配合 Python 3 運作。若要繼續使用由 Amazon Linux 提供並支援的 Python 版本，請將 Python 2 程式碼轉換為 Python 3。

如需 Amazon Linux 上 Python 的詳細資訊，請參閱[AL2023 中的 Python](#)。

rsyslog-openssl取代 **rsyslog-gnutls**

該rsyslog-gnutls軟件包在 AL2 中已棄用，並且不再存在於 AL2023 中。對於任何rsyslog-openssl包裝的使用，該軟件rsyslog-gnutls包都應該是立即替代品。

網路資訊服務 (NIS)/yp

網路資訊服務 (NIS) 原本稱為「黃頁」，或YP在 AL2 中已棄用，且不再存在於 AL2023 中。這包括下列套件：ypbindypserv、和yp-tools。與整合的其他套件在 AL2023 中NIS已移除此功能。

在 AL2023 中已棄用

本節描述 AL2023 中存在的功能，並且可能在 future 版本的 Amazon Linux 中被刪除的功能。每個部分都會說明功能是什麼，以及預計何時將其從 Amazon Linux 中刪除。

Note

隨著 Linux 生態系統的發展和 future 主要版本的 Amazon Linux 更接近發布，本節將隨著時間的推移而更新。

主題

- [32 位元 x86 \(i686\) 執行階段支援](#)
- [伯克利 DB \(libdb\)](#)
- [cron](#)
- [IMDSv1](#)
- [pcre 第 1 版](#)
- [System V init \(sysvinit\)](#)

32 位元 x86 (i686) 執行階段支援

AL2023 保留執行 32 位元 x86 (i686) 二進位檔案的能力。Amazon Linux 的下一個主要版本可能不再支援執行 32 位元使用者空間二進位檔案。

伯克利 DB (**libdb**)

AL2023 隨附於伯克利數據庫 (libdb) 庫的 5.3.28 版本。這是在許可證更改為 GNU Affero GPLv3 (AGPL) 許可證之前的最後一個版本的伯克利數據庫，來自限制較少的睡貓許可證。

AL2023 中的套件仍然依賴於伯克利 DB (libdb)，並且該程式庫將在下一個主要版本的 Amazon Linux 中移除。

Note

AL2023 中的 dnf 軟件包管理器保留了對伯克利 DB (BDB) 格式 rpm 數據庫的只讀支持。此支援將在下一個主要版本的 Amazon Linux 中移除。

cron

cronie 套件預設安裝在 AL2 AMI 上，為排誠定期任務的傳統 crontab 方式提供支援。在 AL2023 中，cronie 默認情況下不包括在內。因此，依預 crontab 設不再提供支援。

在 AL2023 中，您可以選擇性地安裝 cronie 套件以使用傳統 cron 工作。由於 systemd 提供的新增功能，建議您遷移至 systemd 計時器。

Amazon Linux 的 future 版本 (可能是下一個主要版本) 可能不再包含對傳統任 cron 務的支援，並完成轉換到 systemd 計時器。我們建議您遷移離開使用 cron。

IMDSv1

根據預設，AL2023 AMI 會設定為以 IMDSv2 僅限模式啟動，停用。IMDSv1 仍然可以選擇在啟用 IMDSv1 的情況下使用 AL2023。future 版本的 Amazon Linux 可能 IMDSv2 僅強制執行。

如需 AMI 的 IMDS 組態的詳細資訊，請參閱 Amazon EC2 使用者 [指南中的設定](#) AMI。

pcrc 第 1 版

舊版 pcrc 套件已被取代，並將在下一個主要版本的 Amazon Linux 中移除。pcrc2 套件是後繼版本。雖然 AL2023 的第一個版本隨附的套件數量有限 pcrc，但這些套件仍會移轉至 AL2023 pcrc2 內。已取代的 pcrc 程式庫仍可在 AL2023 中使用。

Note

的已取代版本在 AL2023 的整個生命週期內 pcrc 將不會收到安全性更新。如需有關 pcrc 支援生命週期以及套件將接收安全性更新的時間長度的詳細資訊，請參閱 [套件上的套件支援陳述式](#)。pcrc

System V init (sysvinit)

雖然 AL2023 保留了與 System V service (init) 指令碼的向下相容性，但上游 systemd 專案 (作為其 [v254 發行版本](#) 的一部分) 宣布 [淘汰對 System V 服務指令碼](#) 的支援，並指出在 future 版本的 systemd 如需詳細資訊，請參閱 [systemd](#)。

AL2023 將保留與 System V service (init) 指令碼的向下相容性，但建議使用者遷移至使用原生 systemd 單位檔案，以便在下一個主要版本中移除 System V 服務 (init) 指令碼的支援時做好準備。

比較 AL2 與 AL2023

下列主題說明 AL2 與 AL2023 之間的主要差異。

主題

- [新增、升級和移除的套件](#)
- [支援各個版本](#)
- [命名和版本控制變更](#)
- [最佳化](#)
- [Python 2.7 已替換為 Python 3](#)
- [安全性更新](#)
- [穩定性的確定性升級](#)
- [源自多個上游](#)
- [AMI 根檔案系統和預設的 Amazon EBS 磁碟區類型](#)
- [網路系統服務](#)
- [整合控制群組階層 \(cgroup v2\)](#)
- [任務排程](#)
- [glibc、gcc 和 binutils 的套件](#)
- [套件管理工具](#)
- [日誌系統](#)
- [curl 和 libcurl 的套件變更](#)
- [GNU Privacy Guard \(GNUPG\)](#)
- [Amazon Corretto 作為預設 JVM](#)
- [AWS CLI V2](#)
- [UEFI Preferred](#)
- [SSH 伺服器預設組態變更](#)
- [Extra Packages for Enterprise Linux \(EPEL\)](#)
- [使用 cloud-init](#)
- [圖形桌面支援](#)

- [編譯器三元組](#)
- [32 位元 x86 \(i686\) 套件](#)
- [lsb_release 和 system-lsb-core 套件](#)
- [AL2023 核心變更自 AL2](#)
- [比較 Amazon Linux 2 與 Amazon 2023 AMI 安裝的套件](#)
- [比較 Amazon Linux 2 與 Amazon 2023 最小 AMI 安裝的套件](#)
- [比較 Amazon Linux 2 與 Amazon 2023 基本容器映像安裝的套件](#)

新增、升級和移除的套件

AL2023 包含數千個可供使用的軟體套件。如需 AL2023 中新增、升級或移除的所有套件完整清單 (與舊版 Amazon Linux 相比)，請參閱 [AL2023 中的 Package 件變更](#)。

要請求在 AL2023 中添加或更改軟件包，請在 [亞馬遜-linux-2023 軟件庫](#) 中提出一個問題。GitHub

支援各個版本

對於 AL2023，我們提供五年支援。

如需詳細資訊，請參閱 [發行節奏](#)。

命名和版本控制變更

AL2023 支援 AL2 支援平台識別的相同機制。AL2023 還導入用於平台識別的新檔案。

如需詳細資訊，請參閱 [命名和版本控制](#)。

最佳化

AL2023 可最佳化開機時間，以縮短從執行個體啟動到執行客戶工作負載的時間。這些最佳化涵蓋 Amazon EC2 執行個體核心組態、cloud-init 組態，以及作業系統套件內建的功能，例如 kmod 和 systemd。

如需關於最佳化的詳細資訊，請參閱 [效能與操作最佳化](#)。

Python 2.7 已替換為 Python 3

AL2 為 Python 2.7 提供支援和安全修補直到 2025 年 6 月為止，這包含在我們對 AL2 核心套件的長期支援 (LTS) 承諾中。這種支持超出了 2020 年一月的 Python 2.7 end-of-life 的上游 Python 社區聲明。

AL2 使用的 yum 軟件包管理器，它有一個硬依賴於 Python 2.7。在 AL2023 中，dnf 套件管理器已遷移到 Python 3，不再需要 Python 2.7。AL2023 已經完全轉移到 Python 3。

Note

AL2023 已移除 Python 2.7，任何需要 Python 的作業系統組件都編寫為配合 Python 3 運作。若要繼續使用由 Amazon Linux 提供並支援的 Python 版本，請將 Python 2 程式碼轉換為 Python 3。

如需有關 Amazon Linux 的 Python 詳細資訊，請參閱 [AL2023 中的 Python](#)。

安全性更新

SELinux

預設情況下，適用於 AL2023 的 Security Enhanced Linux (SELinux) 是 enabled 且會設為 permissive 模式。在 permissive 模式中，會記錄權限遭拒的情形，但不會強制執行。

SELinux 是 Amazon Linux 核心的安全功能，在 AL2 是 disabled。SELinux 是核心功能和公用程式的集合，為核心的主要子系統提供強制存取控制 (MAC) 架構。

如需詳細資訊，請參閱 [設定 AL2023 的環境管理系統模式](#)。

如需有關 SELinux 儲存庫、工具和策略的詳細資訊，請參閱 [SELinux Notebook](#)、[SELinux 政策類型](#) 和 [SELinux Project](#)。

OpenSSL 3

AL2023 具有 Open Secure Sockets Layer version 3 (OpenSSL 3) 加密工具組的功能。AL2023 支援 TLS 1.3 和 TLS 1.2 網路通訊協定。

AL2 預設隨附 OpenSSL 1.0.2。您可以針對 OpenSSL 1.1.1 建置應用程式。

如需有關 OpenSSL 的詳細資訊，請參閱 [OpenSSL 遷移指南](#)。

如需詳細資訊，請參閱 [安全更新和功能](#)。

IMDSV2

根據預設，任何使用 AL2023 AMI 啟動的執行個體都IMDSv2只需要，而您的預設躍點限制將設定為 2，以支援容器化工作負載支援。這是透過將 `imds-support` 參數設為 `v2.0` 來完成。如需詳細資訊，請參閱 [Amazon EC2 使用者指南中的設定 AMI](#)。

Note

工作階段記號的有效時間介於 1 秒到 6 小時之間。用來導向 IMDSv2 查詢的 API 要求的位址如下：

- IPv4 : 169.254.169.254
- IPv6 : fd00:ec2::254

您可以手動覆寫這些設定，並IMDSv1使用「執行個體中繼資料」選項啟動屬性 您也可以使用 IAM 控制來強制執行不同的IMDS設定。如需設定和使用執行個體中繼資料服務的詳細資訊，請參閱 [Amazon EC2 使用者指南中的使用IMDSv2](#)、[設定新執行個體的執行個體中繼資料選項](#)和[修改現有執行個體的執行個體中繼資料選項](#)。

移除 log4j 熱修補 (**log4j-cve-2021-44228-hotpatch**)

Note

AL2023 未隨附 `log4j-cve-2021-44228-hotpatch` 套件。

為了響應 [CVE-2021-44228](#)，Amazon Linux 發布了對於 AL1 和 AL2 的阿帕奇 Log4j 的熱補丁的 RPM 打包版本。在[對 Amazon Linux 加入熱修補的公告](#)中，我們發現「安裝熱修補不能替代更新到 log4j 版本，以緩解 CVE-2021-44228 或 CVE-2021-45046。」

該熱修補是一種緩解措施，讓您有時間修補 log4j。AL2023 的第一個正式上市 (GA) 版本是在 [CVE-2021-44228](#) 的 15 個月後，因此 AL2023 不會隨附熱修補 (無論是否啟用)。

在 Amazon Linux 上執行自己的log4j版本的使用者應確保已更新為不受 [CVE-2021-44228](#) 或 [CVE-2021-45046](#) 影響的版本。

AL2023 提供有關 [更新 AL2023](#) 的指引，讓您隨時掌握最新的安全修補。安全建議發佈在 [Amazon Linux 安全中心](#)。

穩定性的確定性升級

通過版本化存儲庫功能進行確定性升級，默認情況下，每個 AL2023 AMI 都會鎖定到特定的存儲庫版本。您可以使用決定性升級，在套件版本和更新之間取得更高的一致性。每個主要或次要版本都包含特定的儲存庫版本。

AL2023 的新功能會預設啟用確定性升級。與 AL2 和其他舊版本使用的手動增量鎖定方法相比，這是一大改進。

如需詳細資訊，請參閱 [透過 AL2023 上的版本化儲存庫使用確定性升級](#)。

源自多個上游

AL2023 以 RPM 為基礎，並包含源自多個版本的 Fedora 及其他發行版本的元件，例如 CentOS 9 Stream。Amazon Linux 核心源自直接從 kernel.org 發行的長期支援 (LTS) 發行版，並從其他發行版中獨立選出。

如需詳細資訊，請參閱 [與 Fedora 的關係](#)。

AMI 根檔案系統和預設的 Amazon EBS 磁碟區類型

AL2023 AMI 和 AL2 的根檔案系統都使用 XFS 檔案系統。對於 AL2023，根裝置檔案系統的 mkfs 選項已針對 Amazon EC2 進一步最佳化。AL2023 也支援許多其他檔案系統，您可以在其他磁碟區使用這些檔案系統，以符合您的特定需求。

AL2023 AMI 預設使用 Amazon EBS gp3 磁碟區，而 AL2 AMI 預設使用 Amazon EBS gp2 磁碟區。您可以在啟動執行個體時變更磁碟區類型。

如需 Amazon EBS 磁碟區類型的詳細資訊，請參閱 [Amazon EBS 一般用途磁碟區](#)。

如需有關啟動 Amazon EC2 執行個體的詳細資訊，請參閱 Amazon EC2 使用者指南中的 [啟動執行個體](#)。

網路系統服務

systemd-networkd 系統服務管理 AL2023 中的網路介面。這是與 AL2 不同的變更，且後者使用 ISC dhclient 或 dhclient。

如需詳細資訊，請參閱 [網路服務](#)。

整合控制群組階層 (cgroup v2)

控制群組 (cgroup) 是 Linux 核心功能，用於階層式整理處理程序並為其分配系統資源。控制組廣泛用於執行容器執行期，並由 systemd 執行。

AL2 支持cgroupv1和 AL2023 支持cgroupv2。如果執行容器化工作負載 (例如 [使用以 AL2023 為基礎的 Amazon ECS AMI 託管容器化工作負載](#) 時)，這就值得注意。

雖然 AL2023 仍然包含可讓系統執行使用的程式碼cgroupv1，但這不是建議或受支援的組態，而且會在 future 的 Amazon Linux 主要版本中完全移除。

[低階 Linux 核心介面](#)和 [systemd cgroup 委託文件](#)現有大量的說明文件。

容器之外的一個常見使用案例是建立對其可使用的系統資源有限制的systemd單位。如需詳細資訊，請參閱[系統資源控制](#)。

任務排程

cronie 套件預設安裝在 AL2 AMI 上，為排誠定期任務的傳統 crontab 方式提供支援。在 AL2023 中，cronie默認情況下不包括在內。因此，依預crontab設不再提供支援。

您可以選擇性安裝 cronie 套件以使用傳統 cron 工作。由於 systemd 提供的新增功能，建議您遷移至 systemd 計時器。

glibc、gcc 和 binutils 的套件

AL2023 包含許多與 AL2 相同的核心套件。

我們已更新以下三個 AL2023 核心工具鏈套件。

套件名稱	AL2	AL2023
glibc	2.26	2.34
gcc	7.3	11.3
binutils	2.29	2.39

如需詳細資訊，請參閱 [核心工具鏈套件 glibc、gcc、binutils](#)。

套件管理工具

AL2023 的預設軟體套件管理工具為 DNF。DNF 是 AL2 套件管理工具 YUM 的後續工具。

如需詳細資訊，請參閱 [套件管理工具](#)。

日誌系統

在 AL2023 中，日誌系統套件已從 AL2 變更。AL2023 預設不會安裝 rsyslog，因此預設無法在 AL2 中使用文字型日誌檔，例如 `/var/log/messages`。AL2023 的預設組態為 `systemd-journal`，並可使用 `journalctl` 檢查。雖然 rsyslog 是 AL2023 的選用套件，但建議使用基於 `systemd` 的新 `journalctl` 介面和相關套件。如需詳細資訊，請參閱 [journalctl](#) 手冊頁面。

curl 和 libcurl 的套件變更

AL2023 將 `curl` 和 `libcurl` 套件的常用通訊協定和功能區分為 `curl-minimal` 和 `libcurl-minimal`。這可讓大多數使用者減少磁碟、記憶體和相依性佔用量，也是 AL2023 AMI 和容器的預設套件。

如果需要 `curl` 的完整功能 (例如 `gopher://` 的支援)，請執行下列命令來安裝 `curl-full` 和 `libcurl-full` 套件。

```
$ dnf swap libcurl-minimal libcurl-full
```

```
$ dnf swap curl-minimal curl-full
```

GNU Privacy Guard (GNUPG)

AL2023 將 `gnupg2` 套件的最小和完整功能區分 `gnupg2-minimal` 和 `gnupg2-full` 套件。預設僅安裝 `gnupg2-minimal` 套件。此套件提供在 `rpm` 套件中驗證數位簽章所需的最小功能。

如需 `gnupg2` 的更多功能，例如從金鑰伺服器下載金鑰，請確定已安裝 `gnupg2-full` 套件。執行下列命令以交換 `gnupg2-minimal` 和 `gnupg2-full`。

```
$ dnf swap gnupg2-minimal gnupg2-full
```

Amazon Corretto 作為預設 JVM

AL2023 隨附 [Amazon Corretto](#) 作為預設 (也是唯一) Java 開發套件 (JDK)。AL2023 中的所有 Java 基礎軟件包都是使用 Amazon Corretto 17。

如果您要從 AL2 遷移，則可以從 AL2 上的等效 OpenJDK 版本平滑過渡到 Amazon Corretto

AWS CLI V2

AL2023 隨附第 2 AWS CLI 版，而 AL2 則隨附的第 1 版。AWS CLI

UEFI Preferred

預設情況下，在支援 UEFI 韌體的執行個體類型上使用 AL2023 AMI 啟動的任何執行個體都會在 UEFI 模式下啟動。這是透過將 Boot Mode AMI 參數設為 `uefi-preferred` 來完成。如需詳細資訊，請參閱 Amazon EC2 使用者指南中的 [開機模式](#)。

SSH 伺服器預設組態變更

對於 AL2023 AMI，我們已變更隨發行版本產生的 `sshd` 主機金鑰類型。我們也已捨棄某些舊金鑰類型，以免在啟動時產生這些金鑰類型。使用者端必須支援 `rsa-sha2-256` 和 `rsa-sha2-512` 通訊協定，或是支援 `ssh-ed25519` 使用 `ed25519` 金鑰。`ssh-rsa` 簽章會預設停用。

此外，預設 `sshd_config` 檔案中的 AL2023 組態設定包含 `UseDNS=no`。這項新設定代表 DNS 減損不太可能防止您與執行個體建立 `ssh` 工作階段。此項取捨是 `authorized_keys` 檔案中的 `from=hostname.domain,hostname.domain` 行項目將無法解析。由於 `sshd` 不再嘗試解析 DNS 名稱，因此必須將每個逗號分隔 `hostname.domain` 值轉換為對應的 IP 地址。

如需詳細資訊，請參閱 [預設 SSH 伺服器組態](#)。

Extra Packages for Enterprise Linux (EPEL)

Extra Packages for Enterprise Linux (EPEL) 是 Fedora 社群的專案，目的是為企業級 Linux 作業系統建立大量套件。該項目主要生產 RHEL 和 CentOS 套件。AL2 與 CentOS 7 高度相容。因此，許多 EPEL7 套件都在 AL2 上運作。但是，AL2023 不支援 EPEL 或 EPEL 形式的儲存庫。

使用 cloud-init

在 AL2023 中，cloud-init 會管理套件儲存庫。在舊版的 Amazon Linux 中，cloud-init 預設安裝安全更新。這不是 AL2023 的預設值。用於在啟動時進行更新 `releasever` 的新確定性升級功能說明 AL2023 在啟動時啟用套件更新的方法。如需詳細資訊，請參閱 [在 AL2023 中管理套件和作業系統更新及穩定性的確定性升級](#)。

使用 AL2023 時，您可以將 cloud-init 與 SELinux 搭配使用。如需詳細資訊，請參閱 [使用 cloud-init 來啟用 enforcing 模式](#)。

Cloud-init 使用 HTTP(S) 以從遠端位置載入 cloud-init 的組態內容。在舊版本中，當遠端資源無法使用時，Amazon Linux 不會提醒您。在 AL2023 中，不可用的遠端資源會產生嚴重錯誤，並且 cloud-init 執行失敗。AL2 的這種行為變更提供更安全的「失敗關閉」預設行為。

如需詳細資訊，請參閱 [自訂 cloud-init](#) 和 [cloud-init 文件](#)。

圖形桌面支援

AL2023 以雲端為中心，並針對使用 Amazon EC2 進行最佳化，目前不包含圖形或桌面環境。若要提供意見反應 GitHub，請參閱 <https://github.com/>。

編譯器三元組

AL2023 針對 GCC 和 LLVM 設定編譯器三元組，以指出 amazon 是供應商。

因此，AL2 `aarch64-redhat-linux-gcc` 在 AL2023 上變成 `aarch64-amazon-linux-gcc`。

對於大多數用戶來說，這應該是完全透明的，並且可能只會影響那些在 AL2023 上構建編譯器的用戶。

32 位元 x86 (i686) 套件

作為 [AL1 的 2014.09 版本](#) 的一部分，它宣布將是生成 32 位 AMI 的最後一個版本。因此，從 [AL1 的 2015.03 版本](#) 開始，Amazon Linux 不再支援以 32 位元模式執行系統。AL2 針對 x86-64 主機的 32 位元二進位檔案提供有限的執行期支援，且未提供開發套件來建置新的 32 位元二進位檔案。AL2023 不再包含任何 32 位元使用者空間套件。我們建議您完成 64 位元程式碼的轉換。

如果您需要在 AL2023 上執行 32 位元二進位檔案，則可在基於 AL2023 的 AL2 容器內部使用 AL2 的 32 位元使用者空間。

lsb_release 和 system-lsb-core 套件

在過去，某些軟體會調用 `lsb_release` 命令 (在 AL2 中由套件 `system-lsb-core` 提供) 以取得關於執行軟體所用的 Linux 發行版資訊。Linux 標準規範 (LSB) 已導入此命令，且 Linux 發行版已加以採用。Linux 發行版已發展為使用更簡單的標準，以將此資訊保留在 `/etc/os-release` 和其他相關檔案內。

`os-release` 標準源自 `systemd`。如需詳細資訊，請參閱[系統作業系統版本文件](#)。

AL2023 不隨附 `lsb_release` 命令，也不包含 `system-lsb-core` 套件。軟體應完成 `os-release` 標準的轉換，以維持與 Amazon Linux 和其他主要 Linux 發行版的相容性。

AL2023 核心變更自 AL2

AL2023 帶來了 6.1 內核，以及許多配置更改，以進一步優化 Amazon Linux 的雲。對於大多數用戶而言，這些更改應該是完全透明的。

重視安全的核心設定變更

CONFIG 選項	AL2/4.14/aarch64	AL2/4.14/x86_64	AL2/5.10/aarch64	AL2/5.10/x86_64	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_BUG_ON_DATA_CORRUPTION	n	y	n	y	y	y
CONFIG_DEBUG_FAULT_MMAP_MIN_ADDR	4096	4096	4096	4096	65536	65536
CONFIG_DEBUG_VMEM	n	y	n	y	n	n
CONFIG_DEBUG_VPORT	n	y	n	y	n	n

CONFIG 選項	AL2/4.14/aarch64	AL2/4.14/x86_64	AL2/5.10/aarch64	AL2/5.10/x86_64	AL2023/6.1/aarch64	AL2023/6.1/x86_64
<u>CONFIG_FORTIFY_SOURCE</u>	n	y	n	y	y	y
<u>CONFIG_HARDENED_USERCOPY_FALLBACK</u>	N/A	N/A	y	y	N/A	N/A
<u>CONFIG_INIT_ON_ALLOC_DEFAULT_ON</u>	N/A	N/A	n	n	n	n
<u>CONFIG_INIT_ON_FREE_DEFAULT_ON</u>	N/A	N/A	n	n	n	n
<u>CONFIG_IOMMU_DEFAULT_DMA_STRICT</u>	N/A	N/A	N/A	N/A	n	n
<u>CONFIG_LDISC_AUTOLOAD</u>	y	y	y	y	n	n
<u>CONFIG_SCHED_CORE</u>	N/A	N/A	N/A	N/A	N/A	y
<u>CONFIG_SCHED_STACK_END_CHECK</u>	n	y	n	y	y	y

CONFIG 選項	AL2/4.14/ aarch64	AL2/4.14/ x86_64	AL2/5.10/ aarch64	AL2/5.10/ x86_64	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64
CONFIG_SECURITY_DMESG_RESTRICT	n	n	n	n	y	y
CONFIG_SECURITY_SELINUX_DISABLE	y	y	y	y	n	n
CONFIG_SHUFFLE_PAGE_ALLOCATOR	N/A	N/A	y	y	y	y
CONFIG_SLAB_FREELIST_HARDENED	n	y	y	y	y	y
CONFIG_SLAB_FREELIST_RANDOM	n	n	y	y	y	y

x86-64 專屬且重視安全的核心設定變更

CONFIG 選項	AL2/4.14/x86_64	AL2/5.10/x86_64	AL2023/6.1/x86_64
CONFIG_AMD_IOMMU	y	y	y
CONFIG_AMD_IOMMU_V2	m	m	y

CONFIG 選項	AL2/4.14/x86_64	AL2/5.10/x86_64	AL2023/6.1/x86_64
CONFIG_RA_NDOMIZE_MEMORY	N/A	y	y

aarch64 (ARM/Graviton) 專屬且重視安全的核心設定變更

CONFIG 選項	AL2/4.14/aarch64	AL2/5.10/aarch64	AL2023/6.1/aarch64
CONFIG_ARM64_PTR_AUTH	N/A	y	y
CONFIG_ARM64_PTR_AUTH_KERNEL	N/A	N/A	y
CONFIG_ARM64_SW_TTBR0_PAN	y	y	y

/dev/mem、/dev/kmem 和 /dev/port

Amazon Linux 2023 禁用 /dev/mem/dev/port (CONFIG_DEVMEM和CONFIG_DEVPORT) 完全基於 AL2 中已經存在的限制的基礎上。

該/dev/kmem代碼已在 5.13 內核中從 Linux 中完全刪除，儘管它在 AL2 中被禁用，但現在不適用於 AL2023。

此選項是[核心自我保護專案建議設定](#)之一。

FORTIFY_SOURCE

AL2023 可CONFIG_FORTIFY_SOURCE在所有受支援的架構上啟用。此功能是安全強化功能。編譯器可以確定和驗證緩衝區大小時，此功能可以偵測常見字符串和記憶體函數中的緩衝區溢出。

此選項是[核心自我保護專案建議設定](#)之一。

線條定律自動載入 () `CONFIG_LDISC_AUTOLOAD`

AL2023 核心不會自動載入線規範，例如透過軟體使用 `TIOCSETDioct1`，除非要求來自具有 `CAP_SYS_MODULE` 權限的程序。

此選項是[核心自我保護專案建議設定](#)之一。

`dmesg` 無權限使用者的存取權 () `CONFIG_SECURITY_DMESG_RESTRICT`

根據預設，AL2023 不允許無權限的使用者存取 `dmesg`。

此選項是[核心自我保護專案建議設定](#)之一。

禁用 `selinuxfs`

AL2023 會停用已停用的 `CONFIG_SECURITY_SELINUX_DISABLE` 核心選項，該選項會在載入政策之前啟用停用 SELinux 的執行階段方法。

此選項是[核心自我保護專案建議設定](#)之一。

其他核心設定變更

CONFIG 選項	AL2/4.14/ aarch64	AL2/4.14/ x86_64	AL2/5.10/ aarch64	AL2/5.10/ x86_64	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64
CONFIG_HZ	100	250	100	250	100	100
CONFIG_NR_CPUS	4096	8192	4096	8192	512	512
CONFIG_PANIC_ON_OOPS	y	n	y	n	y	y
CONFIG_PANIC_ON_OOPS_VALUE	1	0	1	0	1	1
CONFIG_PRINTK	m	m	m	m	n	n

CONFIG 選項	AL2/4.14/aarch64	AL2/4.14/x86_64	AL2/5.10/aarch64	AL2/5.10/x86_64	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_SLIP	m	m	m	m	n	n
CONFIG_XEN_PV	N/A	y	N/A	n	N/A	n

CONFIG_HZ

AL2023 在兩個aarch64平台上都x86-64設置CONFIG_HZ為 100。

CONFIG_NR_CPUS

AL2023 設置CONFIG_NR_CPUS為接近 Amazon EC2 中發現的最大 CPU 內核數的數字。

意外錯誤

AL2023 核心會在發生錯誤時發生恐慌。此功能相當於在核心命令列以 `oops=panic` 啟動。

核心意外是核心偵測到內部錯誤的地方，這可能會影響系統的進一步可靠性。

PPP 和 SLIP 支援

AL2023 不支援 PPP 或滑動通訊協定。

Xen PV 訪客支援

AL2023 不支援以 Xen 光伏客體的身分執行。

核心檔案系統支援

AL2 中的核心將支援掛載的檔案系統中發生了一些變更，以及核心將剖析的磁碟分割配置中的變更。

CONFIG 選項	AL2/4.14/aarch64	AL2/4.14/x86_64	AL2/5.10/aarch64	AL2/5.10/x86_64	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_AFS_FS	n	m	n	m	n	n

CONFIG 選項	AL2/4.14/ aarch64	AL2/4.14/ x86_64	AL2/5.10/ aarch64	AL2/5.10/ x86_64	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64
<u>CONFIG_AF_RRPC</u>	n	m	n	m	n	n
<u>CONFIG_BSD_DISKLABEL</u>	y	y	y	y	n	n
<u>CONFIG_CRAMFS</u>	m	m	m	m	n	n
<u>CONFIG_CRAMFS_BLOCKDEV</u>	N/A	N/A	y	n	N/A	N/A
<u>CONFIG_DM_CLONE</u>	N/A	N/A	n	n	n	n
<u>CONFIG_DM_ERA</u>	m	n	m	n	n	n
<u>CONFIG_DM_INTEGRITY</u>	n	m	n	m	m	m
<u>CONFIG_DM_LOG_WRITES</u>	n	n	m	m	m	m
<u>CONFIG_DM_SWITCH</u>	m	n	m	n	n	n
<u>CONFIG_DM_VERITY</u>	m	n	m	n	n	n
<u>CONFIG_ECRYPT_FS</u>	n	m	n	m	n	n

CONFIG 選項	AL2/4.14/ aarch64	AL2/4.14/ x86_64	AL2/5.10/ aarch64	AL2/5.10/ x86_64	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64
<u>CONFIG_EX_FAT_FS</u>	N/A	N/A	m	m	m	m
<u>CONFIG_EX_T2_FS</u>	n	m	n	m	n	n
<u>CONFIG_EX_T3_FS</u>	n	m	n	m	n	n
<u>CONFIG_GF_S2_FS</u>	m	m	m	m	n	n
<u>CONFIG_HF_SPLUS_FS</u>	n	m	n	m	n	n
<u>CONFIG_HF_S_FS</u>	n	m	n	m	n	n
<u>CONFIG_JF_S_FS</u>	n	m	n	m	n	n
<u>CONFIG_LD_M_PARTITION</u>	n	y	n	y	n	n
<u>CONFIG_MAC_PARTITION</u>	n	y	n	y	n	n
<u>CONFIG_NFS_V2</u>	n	m	n	m	n	n
<u>CONFIG_NTFS_FS</u>	n	m	n	n	n	n

CONFIG 選項	AL2/4.14/ aarch64	AL2/4.14/ x86_64	AL2/5.10/ aarch64	AL2/5.10/ x86_64	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64
CONFIG_ROMFS_FS	n	m	n	m	n	n
CONFIG_S0LARI_X86_PARTITION	n	y	n	y	n	n
CONFIG_SQUASHFS_ZSTD	n	y	n	y	y	y
CONFIG_SU_N_PARTITION	n	y	n	y	n	n

Andrew 檔案系統支援 (AFS)

核心在建置時，不再支援 afs 檔案系統。AL2 並未隨附的使用者空間支援。afs

cramfs 支援

核心在建置時，不再支援 cramfs 檔案系統。AL2023 中的後續任務是squashfs檔案系統。

BSD 磁碟標籤支援

核心在建置時，不再支援 BSD 磁碟標籤。如果需要讀取有 BSD 磁碟標籤的磁碟區，則可啟動各種 BSD。

裝置對應程式變更

AL2023 核心中設定的裝置對應程式目標有幾項變更。

eCryptFs 支持

ecryptfs 檔案系統已在 Amazon Linux 中棄用。的使用者空間元件存ecryptfs在於 AL1 中，在 AL2 中移除，而且 AL2023 不再建置支援的核心。ecryptfs

exFAT

在 AL2 的 5.10 核心中新增了對 exFAT 檔案系統的支援。它在使用 4.14 內核的 AL2 啟動時不存在。AL2023 會繼續支援 exFAT 檔案系統。

ext2、ext3 和 ext4 檔案系統

AL2023 隨附選取 `CONFIG_EXT4_USE_FOR_EXT2` 項，這意味著 ext4 文件系統代碼將用於讀取舊版 ext2 文件系統。

CONFIG_GFS2_FS

此核心不再使用配置 `CONFIG_GFS2_FS` 建置。

Apple Extended HFS 檔案系統支援 (HFS +)

在 AL2 中，只有 x86-64 內核是在 hfsplus 文件系統支持下構建的。AL2 5.15 核心不包含任何架構的 hfsplus 支援。在 AL2023 中，我們完成了 Amazon Linux 中對 hfsplus 支援的棄用。

HFS 檔案系統支援

在 AL2 中，只有 x86-64 內核是在 hfs 文件系統支持下構建的。AL2 5.15 核心不包含任何架構的 hfs 支援。在 AL2023 中，我們完成了 Amazon Linux 中對 hfs 支援的棄用。

JFS 檔案系統支援

在 AL2 中，只有 x86-64 內核是在 jfs 文件系統支持下構建的。AL2 5.15 核心不包含任何架構的 jfs 支援。無論是 AL1 或 AL2 與 JFS 用戶空間一起提供。在 AL2023 中，我們完成了 Amazon Linux 中對 jfs 支援的棄用。

上游 Linux 核心正在 [考慮移除 JFS](#)。因此，如果 JFS 檔案系統上有資料，則應將其移轉至其他檔案系統。

Windows 邏輯磁碟管理員 (動態磁碟) 支援 (CONFIG_LDM_PARTITION)

AL2023 不再支援 Windows 2000、Windows XP、或具有 MS-DOS 樣式分割區的 Windows Vista 動態磁碟。此代碼從未支持隨 Windows Vista 引入的較新基於 GPT 的動態磁碟。

Macintosh 分割對應支援

AL2023 不再支持經典的麥金塔分區圖。現代的 macOS 版本預設建立現代的 GPT 分割表，而不是這種舊類型。

NFSv2 支援

AL2023 已不再支援 NFS2，但仍會繼續支援 NFSv3、NFSv4、NFSv4 和 NFSv4。建議您移轉至 NFSv3 或更新版本。

NTFS (CONFIG_NTFS_FS)

從 AL2 的 5.10 內核開始，替換了 `ntfs` 用於在 Amazon Linux 上訪問 NTFS 文件系統的 `ntfs3` 代碼。AL2023 不再包含 `ntfs` 程式碼，而且完全依賴 `ntfs3` 程式碼來存取 NTFS 檔案系統。

romfs 檔案系統

`squashfs` 檔案系統是 Amazon Linux `romfs` 檔案系統的后續系統，而 AL2023 核心的建置不再支援 `romfs`。

Solaris x86 硬碟分割格式

AL2023 不再支援索拉瑞斯 x86 硬碟分割區格式。

squashfszstd 壓縮

AL2023 在所有支援的架構上新增對 `zstd` 壓縮 `squashfs` 檔案系統的支援。

Sun 分割表支援

AL2023 不再支援 Sun 分割區表格式 (CONFIG_SUN_PARTITION)。

比較 Amazon Linux 2 與 Amazon 2023 AMI 安裝的套件

目前在 Amazon Linux 2 和 AL2023 標準 AMI 上的 RPM 的比較。

套件	AL2 阿 AMI	阿 AMI
<code>acl</code>	2.2.51	2.3.1
<code>acpid</code>	2.0.19	2.0.32
<code>alternatives</code>		1.15
<code>amazon-chrony-config</code>		4.3
amazon-ec2-net-utils		2.4.1

套件	AL2 阿 AMI	阿 AMI
amazon-linux-extras	2.0.3	
amazon-linux-extras-yum-plugin	2.0.3	
amazon-linux-repo-s3		2023.4.20240513
amazon-linux-sb-keys		2023.1
amazon-rpm-config		228
amazon-ssm-agent	3.3.131.0	3.3.380.0
at	3.1.13	3.1.23
attr	2.4.46	2.5.1
audit	2.8.1	3.0.6
audit-libs	2.8.1	3.0.6
authconfig	6.2.8	
aws-cfn-bootstrap	2.0	2.0
awscli	1.18.147	
awscli-2		2.15.30
basesystem	10.0	11
bash	4.2.46	5.2.15
bash-completion	2.1	2.11
bc	1.06.95	1.07.1
bind-export-libs	9.11.4	
bind-libs	9.11.4	9.16.48

套件	AL2 阿 AMI	阿 AMI
bind-libs-lite	9.11.4	
bind-license	9.11.4	9.16.48
bind-utils	9.11.4	9.16.48
binutils	2.29.1	2.39
blktrace	1.0.5	
boost-date-time	1.53.0 (x86_64)	
boost-filesystem		1.75.0
boost-system	1.53.0 (x86_64)	1.75.0
boost-thread	1.53.0 (x86_64)	1.75.0
bridge-utils	1.5	
bzip2	1.0.6	1.0.8
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023.2.64	2023.2.64
c-ares		1.19.0
checkpolicy		3.4
chkconfig	1.7.4	1.15
chrony	4.2	4.3
cloud-init	19.3	22.2.2
cloud-init-cfg-ec2		22.2.2
cloud-utils-growpart	0.31	0.31

套件	AL2 阿 AMI	阿 AMI
coreutils	8.22	8.32
coreutils-common		8.32
cpio	2.12	2.13
cracklib	2.9.0	2.9.6
cracklib-dicts	2.9.0	2.9.6
cronie	1.4.11	
cronie-anacron	1.4.11	
crontabs	1.11	1.11
crypto-policies		20220428
crypto-policies-scripts		20220428
cryptsetup	1.7.4	2.6.1
cryptsetup-libs	1.7.4	2.6.1
curl	8.3.0	
curl-minimal		8.5.0
cyrus-sasl-lib	2.1.26	2.1.27
cyrus-sasl-plain	2.1.26	2.1.27
dbus	1.10.24	1.12.28
dbus-broker		32
dbus-common		1.12.28
dbus-libs	1.10.24	1.12.28

套件	AL2 阿 AMI	阿 AMI
device-mapper	1.02.170	1.02.185
device-mapper-event	1.02.170	
device-mapper-event-libs	1.02.170	
device-mapper-libs	1.02.170	1.02.185
device-mapper-persistent-data	0.7.3	
dhclient	4.2.5	
dhcp-common	4.2.5	
dhcp-libs	4.2.5	
diffutils	3.3	3.8
dmidecode	3.2	
dmraid	1.0.rc16	
dmraid-events	1.0.rc16	
dnf		4.14.0
dnf-data		4.14.0
dnf-plugin-release-notification		1.2
dnf-plugins-core		4.3.0
dnf-plugin-support-info		1.2
dnf-utils		4.3.0

套件	AL2 阿 AMI	阿 AMI
dosfstools	3.0.20	4.2
dracut	033	055
dracut-config-ec2	2.0	3.0
dracut-config-generic	033	055
dwz		0.14
dyninst	9.3.1 (64)	10.2.1
e2fsprogs	1.42.9	1.46.5
e2fsprogs-libs	1.42.9	1.46.5
ec2-hibinit-agent	1.0.8	1.0.8
ec2-instance-connect	1.1	1.1
ec2-instance-connect-selinux	1.1	1.1
ec2-net-utils	1.7.3	
ec2-utils	1.2	2.2.0
ed	1.9	1.14.2
efibootmgr	15 (一個建築 64)	
efi-filesystem		5
efi-srpm-macros		5
efivar		38
efivar-libs	31 (一個建築 64)	38

套件	AL2 阿 AMI	阿 AMI
elfutils-debuginfod-client		0.188
elfutils-default-yama-scope	0.176	0.188
elfutils-libelf	0.176	0.188
elfutils-libs	0.176	0.188
ethtool	4.8	5.15
expat	2.1.0	2.5.0
file	5.11	5.39
file-libs	5.11	5.39
filesystem	3.2	3.14
findutils	4.5.11	4.8.0
fipscheck	1.4.1	
fipscheck-lib	1.4.1	
fonts-srpm-macros		2.0.5
freetype	2.8	
fstrm		0.6.1
fuse-libs	2.9.2	2.9.9
gawk	4.0.2	5.1.0
gdbm	1.13	
gdbm-libs		1.19

套件	AL2 阿 AMI	阿 AMI
gdisk	0.8.10	1.0.8
generic-logos	18.0.0	
GeoIP	1.5.0	
gettext	0.19.8.1	0.21
gettext-libs	0.19.8.1	0.21
ghc-srpm-macros		1.5.0
glib2	2.56.1	2.74.7
glibc	2.26	2.34
glibc-all-langpacks	2.26	2.34
glibc-common	2.26	2.34
glibc-gconv-extra		2.34
glibc-locale-source	2.26	2.34
glibc-minimal-lang pack	2.26	
gmp	6.0.0	6.2.1
gnupg2	2.0.22	
gnupg2-minimal		2.3.7
gnutls		3.8.0
go-srpm-macros		3.2.0
gpgme	1.3.2	1.15.1
gpm-libs	1.20.7	1.20.7

套件	AL2 阿 AMI	阿 AMI
grep	2.20	3.8
groff-base	1.22.2	1.22.4
grub2	2.06	
grub2-common	2.06	2.06
grub2-efi-aa64	2.06 (一個建築 64)	
grub2-efi-aa64-ec2	2.06 (一個建築 64)	2.06 (一個建築 64)
grub2-efi-aa64-modules	2.06 (公爵夫人)	
grub2-efi-x64-ec2	2.06 (十六)	2.06 (十六)
grub2-pc	2.06 (十六)	
grub2-pc-modules	2.06 (公爵夫人)	2.06
grub2-tools	2.06	2.06
grub2-tools-minimal	2.06	2.06
grubby	8.28	8.40
gssproxy	0.7.0	0.8.4
gzip	1.5	1.12
hardlink	1.3	
hibagent	1.1.0	
hostname	3.13	3.23
hunspell	1.3.2	1.7.0
hunspell-en	0.20121024	0.20140811.1

套件	AL2 阿 AMI	阿 AMI
hunspell-en-GB	0.20121024	0.20140811.1
hunspell-en-US	0.20121024	0.20140811.1
hunspell-filesystem		1.7.0
hwdata	0.252	0.353
info	5.1	6.7
inih		49
initscripts	9.49.47	10.09
iproute	5.10.0	5.10.0
iptables	1.8.4	
iptables-libs	1.8.4	
iputils	20180629	20210202
irqbalance	1.7.0	1.9.0
jansson	2.10	2.14
jbigkit-libs	2.0	
jitterentropy		3.4.1
jq		1.7.1
json-c	0.11	0.14
kbd	1.15.5	2.4.0
kbd-legacy	1.15.5	
kbd-misc	1.15.5	2.4.0

套件	AL2 阿 AMI	阿 AMI
kernel	5.10.215	6.1.90
kernel-livepatch-r epo-s3		2023.4.20240513
kernel-srpm-macros		1.0
kernel-tools	5.10.215	6.1.90
keyutils	1.5.8	1.6.3
keyutils-libs	1.5.8	1.6.3
kmod	25	29
kmod-libs	25	29
kpartx	0.4.9	
kpatch-runtime	0.9.4	0.9.7
krb5-libs	1.15.1	1.21
langtable	0.0.31	
langtable-data	0.0.31	
langtable-python	0.0.31	
less	458	608
libacl	2.2.51	2.3.1
libaio	0.3.109	0.3.111
libarchive		3.5.3
libargon2		20171227
libassuan	2.1.0	2.5.5

套件	AL2 阿 AMI	阿 AMI
libattr	2.4.46	2.5.1
libbasicobjects	0.1.1	0.1.1
libblkid	2.30.2	2.37.4
libcap	2.54	2.48
libcap-ng	0.7.5	0.8.2
libcbor		0.7.0
libcollection	0.7.0	0.7.0
libcom_err	1.42.9	1.46.5
libcomps		0.1.20
libconfig	149	1.7.2
libcroco	0.6.12	
libcrypt	2.26	
libcurl	8.3.0	
libcurl-minimal		8.5.0
libdaemon	0.14	
libdb	5.3.21	5.3.28
libdb-utils	5.3.21	
libdhash		0.5.0
libdnf		0.69.0
libdrm	2.4.97	

套件	AL2 阿 AMI	阿 AMI
libdwarf	20130207	
libeconf		0.4.0
libedit	3.0	3.1
libestr	0.1.9	
libev		4.33
libevent	2.0.21	2.1.12
libfastjson	0.99.4	
libfdisk	2.30.2	2.37.4
libffi	3.0.13	3.4.4
libfido2		1.10.0
libgcc	7.3.1	11.4.1
libgcrypt	1.5.3	1.10.2
libgomp	7.3.1	11.4.1
libgpg-error	1.12	1.42
libibverbs		48.0
libicu	50.2	
libidn	1.28	
libidn2	2.3.0	2.3.2
libini_config	1.3.1	1.3.1
libjpeg-turbo	2.0.90	

套件	AL2 阿 AMI	阿 AMI
libkcapi		1.4.0
libkcapi-hmaccalc		1.4.0
libldb		2.6.2
libmaxminddb		1.5.2
libmetalink	0.1.3	0.1.3
libmnl	1.0.3	1.0.4
libmodulemd		2.13.0
libmount	2.30.2	2.37.4
libnetfilter_conntrack	1.0.6	
libnfnetlink	1.0.1	
libnfsidmap	0.25	2.5.4
libnghttp2	1.41.0	1.59.0
libnl3	3.2.28	3.5.0
libnl3-cli	3.2.28	
libpath_utils	0.2.1	0.2.1
libpcap	1.5.3	1.10.1
libpciaccess	0.14 (英文版)	
libpipeline	1.2.3	1.5.3
libpkgconf		1.8.0
libpng	1.5.13	

套件	AL2 阿 AMI	阿 AMI
libpsl	0.21.5	0.21.1
libpwquality	1.2.3	1.4.4
libref_array	0.1.5	0.1.5
librepo		1.14.5
libreport-filessystem		2.15.2
libseccomp	2.5.2	2.5.3
libselinux	2.5	3.4
libselinux-utils	2.5	3.4
libsemanage	2.5	3.4
libsepol	2.5	3.4
libsigsegv		2.13
libsmartcols	2.30.2	2.37.4
libsolv		0.7.22
libss	1.42.9	1.46.5
libssh2	1.4.3	
libsss_certmap		2.9.4
libsss_idmap	1.16.5	2.9.4
libsss_nss_idmap	1.16.5	2.9.4
libsss_sudo		2.9.4
libstdc++	7.3.1	11.4.1

套件	AL2 阿 AMI	阿 AMI
libstoragemgmt	1.6.1	1.9.4
libstoragemgmt-python	1.6.1	
libstoragemgmt-python-clibs	1.6.1	
libsfs	2.1.0	
libtalloc		2.3.4
libtasn1	4.10	4.19.0
libtdb		1.4.7
libteam	1.27	
libtevent		0.13.0
libtextstyle		0.21
libtiff	4.0.3	
libtirpc	0.2.4	1.3.3
libunistring	0.9.3	0.9.10
libuser	0.60	0.63
libutempter	1.1.6	1.2.1
libuuid	2.30.2	2.37.4
libuv		1.47.0
libverto	0.2.5	0.3.2
libverto-libev		0.3.2

套件	AL2 阿 AMI	阿 AMI
libverto-libevent	0.2.5	
libwebp	0.3.0	
libxcrypt		4.4.33
libxml2	2.9.1	2.10.4
libxml2-python	2.9.1	
libyaml	0.1.4	0.2.5
libzstd		1.5.5
lm_sensors-libs	3.4.0	3.6.0
lmdb-libs		0.9.29
logrotate	3.8.6	3.20.1
lsof	4.87	4.94.0
lua	5.1.4	
lua-libs		5.4.4
lua-srpm-macros		1
lvm2	2.02.187	
lvm2-libs	2.02.187	
lz4	1.7.5	
lz4-libs		1.9.4
make	3.82	
man-db	2.6.3	2.9.3

套件	AL2 阿 AMI	阿 AMI
man-pages	3.53	5.10
man-pages-overrides	7.5.2	
mariadb-libs	5.5.68	
mdadm	4.0	
microcode_ctl	2.1 (六十四)	2.1 (六十四)
mlocate	0.26	
mpfr		4.1.0
mtr	0.92	
nano	2.9.8	5.8
ncurses	6.0	6.2
ncurses-base	6.0	6.2
ncurses-libs	6.0	6.2
nettle	2.7.1	3.8
net-tools	2.0	2.0
newt	0.52.15	0.52.21
newt-python	0.52.15	
nfs-utils	1.3.0	2.5.4
npth		1.6
nspr	4.35.0	4.35.0
nss	3.90.0	3.90.0

套件	AL2 阿 AMI	阿 AMI
nss-pem	1.0.3	
nss-softokn	3.90.0	3.90.0
nss-softokn-freebl	3.90.0	3.90.0
nss-sysinit	3.90.0	3.90.0
nss-tools	3.90.0	
nss-util	3.90.0	3.90.0
ntsysv	1.7.4	1.15
numactl-libs	2.0.9	2.0.14
ocaml-srpm-macros		6
oniguruma		6.9.7.1
openblas-srpm-macros		2
openldap	2.4.44	2.4.57
openssh	七分一	8.7 小時
openssh-clients	七分一	8.7 小時
openssh-server	七分一	8.7 小時
openssl	1.0.2K	3.0.8
openssl-libs	1.0.2K	3.0.8
openssl-pkcs11		0.4.12
os-prober	1.58	1.77
p11-kit	0.23.22	0.24.1

套件	AL2 阿 AMI	阿 AMI
p11-kit-trust	0.23.22	0.24.1
package-notes-srpm-macros		0.4
pam	1.1.8	1.5.1
parted	3.1	3.4
passwd	0.79	0.80
pciutils	3.5.1	3.7.0
pciutils-libs	3.5.1	3.7.0
pcre	8.32	
pcre2	10.23	10.40
pcre2-syntax		10.40
perl	5.16.3	
perl-Carp	1.26	1.50
perl-Class-Struct		0.66
perl-constant	1.27	1.33
perl-DynaLoader		1.47
perl-Encode	2.51	3.15
perl-Errno		1.30
perl-Exporter	5.68	5.74
perl-Fcntl		1.13
perl-File-Basename		2.85

套件	AL2 阿 AMI	阿 AMI
perl-File-Path	2.09	2.18
perl-File-stat		1.09
perl-File-Temp	0.23.01	0.231.100
perl-Filter	1.49	
perl-Getopt-Long	2.40	2.52
perl-Getopt-Std		1.12
perl-HTTP-Tiny	0.033	0.078
perl-if		0.60.800
perl-interpreter		5.32.1
perl-I0		1.43
perl-IPC-Open3		1.21
perl-libs	5.16.3	5.32.1
perl-macros	5.16.3	
perl-MIME-Base64		3.16
perl-mro		1.23
perl-overload		1.31
perl-overloading		0.02
perl-parent	0.225	0.238
perl-PathTools	3.40	3.78
perl-Pod-Escapes	1.04	1.07

套件	AL2 阿 AMI	阿 AMI
perl-podlators	2.5.1	4.14
perl-Pod-Perldoc	3.20	3.28.01
perl-Pod-Simple	3.28	3.42
perl-Pod-Usage	1.63	2.01
perl-POSIX		1.94
perl-Scalar-List-U tills	1.27	1.56
perl-SelectSaver		1.02
perl-Socket	2.010	2.032
perl-srpm-macros		1
perl-Storable	2.45	3.21
perl-subst		1.03
perl-Symbol		1.08
perl-Term-ANSIColor		5.01
perl-Term-Cap		1.17
perl-Text-ParseWords	3.29	3.30
perl-Text-Tabs+Wrap		2021.0726
perl-threads	1.87	
perl-threads-shared	1.43	
perl-Time-HiRes	1.9725	
perl-Time-Local	1.2300	1.300

套件	AL2 阿 AMI	阿 AMI
perl-vars		1.05
pinentry	0.8.1	
pkgconf		1.8.0
pkgconfig	0.27.1	
pkgconf-m4		1.8.0
pkgconf-pkg-config		1.8.0
plymouth	0.8.9	
plymouth-core-libs	0.8.9	
plymouth-scripts	0.8.9	
pm-utils	1.4.1	
policycoreutils	2.5	3.4
policycoreutils-python-utils		3.4
popt	1.13	1.18
postfix	2.10.1	
procps-ng	3.3.10	3.3.17
protobuf-c		1.4.1
psacct	6.6.1	6.6.4
psmisc	22.20	23.4
pth	2.0.7	

套件	AL2 阿 AMI	阿 AMI
publicsuffix-list-dafsa	20240208	20240212
pygpgme	0.3	
pyliblzma	0.5.3	
pystache	0.5.3	
python	2.7.18	
python2-botocore	1.18.6	
python2-colorama	0.3.9	
python2-cryptography	1.7.2	
python2-dateutil	2.6.1	
python2-futures	3.0.5	
python2-jmespath	0.9.3	
python2-jsonschema	2.5.1	
python2-oauthlib	2.0.1	
python2-pyasn1	0.1.9	
python2-rpm	4.11.3	
python2-rsa	3.4.1	
python2-s3transfer	0.3.3	
python2-setuptools	41.2.0	
python2-six	1.11.0	
python3	3.7.16	3.9.16

套件	AL2 阿 AMI	阿 AMI
python3-attrs		20.3.0
python3-audit		3.0.6
python3-awscrt		0.19.19
python3-babel		2.9.1
python3-cffi		1.14.5
python3-chardet		4.0.0
python3-colorama		0.4.4
python3-configobj		5.0.6
python3-cryptography		36.0.1
python3-daemon	2.2.3	2.3.0
python3-dateutil		2.8.1
python3-dbus		1.2.18
python3-distro		1.5.0
python3-dnf		4.14.0
python3-dnf-plugins-core		4.3.0
python3-docutils	0.14	0.16
python3-gpg		1.15.1
python3-hawkey		0.69.0
python3-idna		2.10
python3-jinja2		2.11.3

套件	AL2 阿 AMI	阿 AMI
python3-jmespath		0.10.0
python3-jsonpatch		1.21
python3-jsonpointer		2.0
python3-jsonschema		3.2.0
python3-libcomps		0.1.20
python3-libdnf		0.69.0
python3-libs	3.7.16	3.9.16
python3-libselenium		3.4
python3-libsemanage		3.4
python3-libstorage mgmt		1.9.4
python3-lockfile	0.11.0	0.12.2
python3-markupsafe		1.1.1
python3-netifaces		0.10.6
python3-oauthlib		3.0.2
python3-pip	20.2.2	
python3-pip-wheel		21.3.1
python3-ply		3.11
python3-policycore utils		3.4
python3-prettytable		0.7.2

套件	AL2 阿 AMI	阿 AMI
python3-prompt-toolkit		3.0.24
python3-pycparser		2.20
python3-pyrsistent		0.17.3
python3-pyserial		3.4
python3-pysocks		1.7.1
python3-pystache	0.5.4	
python3-pytz		2022.7.1
python3-pyyaml		5.4.1
python3-requests		2.25.1
python3-rpm		4.16.1.3
python3-ruamel-yaml		0.16.6
python3-ruamel-yaml-clib		0.1.2
python3-setools		4.4.1
python3-setuptools	49.1.3	59.6.0
python3-setuptools-wheel		59.6.0
python3-simplejson	3.2.0	
python3-six		1.15.0
python3-systemd		235

套件	AL2 阿 AMI	阿 AMI
python3-urllib3		1.25.10
python3-wcwidth		0.2.5
python-babel	0.9.6	
python-backports	1.0	
python-backports-s sl_match_hostname	3.5.0.1	
python-cffi	1.6.0	
python-chardet	2.2.1	
python-chevron		0.13.1
python-configobj	4.7.2	
python-daemon	1.6	
python-devel	2.7.18	
python-docutils	0.12	
python-enum34	1.0.4	
python-idna	2.4	
python-iniparse	0.4	
python-ipaddress	1.0.16	
python-jinja2	2.7.2	
python-jsonpatch	1.2	
python-jsonpointer	1.9	
python-jwcrypto	0.4.2	

套件	AL2 阿 AMI	阿 AMI
python-kitchen	1.1.1	
python-libs	2.7.18	
python-lockfile	0.9.1	
python-markupsafe	0.11	
python-pillow	2.0.0	
python-ply	3.4	
python-pycparser	2.14	
python-pycurl	7.19.0	
python-repoze-lru	0.4	
python-requests	2.6.0	
python-simplejson	3.2.0	
python-srpm-macros		3.9
python-urlgrabber	3.10	
python-urllib3	1.25.9	
pyxattr	0.5.1	
PyYAML	3.10	
qrencode-libs	3.4.1	
quota	4.01	4.06
quota-nls	4.01	4.06
rdate	1.4	

套件	AL2 阿 AMI	阿 AMI
readline	6.2	8.1
rng-tools	6.8	6.14
rootfiles	8.1	8.1
rpcbind	0.2.0	1.2.6
rpm	4.11.3	4.16.1.3
rpm-build-libs	4.11.3	4.16.1.3
rpm-libs	4.11.3	4.16.1.3
rpm-plugin-selinux		4.16.1.3
rpm-plugin-systemd-inhibit	4.11.3	4.16.1.3
rpm-sign-libs		4.16.1.3
rsync	3.1.2	3.2.6
rsyslog	8.24.0	
rust-srpm-macros		21
sbsigntools		0.9.4
scl-utils	20130529	
screen	4.1.0	4.8.0
sed	4.2.2	4.8
selinux-policy	3.13.1	37.22
selinux-policy-targeted	3.13.1	37.22

套件	AL2 阿 AMI	阿 AMI
setserial	2.17	
setup	2.8.71	2.13.7
setuptools	1.19.11	
sgpio	1.2.0.10	
shadow-utils	4.1.5.1	4.9
shared-mime-info	1.8	
slang	2.2.4	2.3.2
sqlite	3.7.17	
sqlite-libs		3.40.0
sssd-client	1.16.5	2.9.4
sssd-common		2.9.4
sssd-kcm		2.9.4
sssd-nfs-idmap		2.9.4
strace	4.26	6.8
sudo	1.8.23	1.9.15
sysctl-defaults	1.0	1.0
sysstat	10.1.5	12.5.6
systemd	219	252.16
systemd-libs	219	252.16
systemd-networkd		252.16

套件	AL2 阿 AMI	阿 AMI
systemd-pam		252.16
systemd-resolved		252.16
systemd-sysv	219	
systemd-udev		252.16
system-release	2	2023.4.20240513
systemtap-runtime	4.5	4.8
sysvinit-tools	2.88	
tar	1.26	1.34
tbb		2020.3
tcp_wrappers	7.6	
tcp_wrappers-libs	7.6	
tcpdump	4.9.2	4.99.1
tcsh	6.18.01	6.24.07
teamd	1.27	
time	1.7	1.9
traceroute	2.0.22	2.1.3
tzdata	2024a	2024a
unzip	6.0	6.0
update-motd	1.1.2	2.2
usermode	1.111	

套件	AL2 阿 AMI	阿 AMI
userspace-rcu		0.12.1
ustr	1.0.4	
util-linux	2.30.2	2.37.4
util-linux-core		2.37.4
vim-common	9.0.2153	9.0.2153
vim-data	9.0.2153	9.0.2153
vim-enhanced	9.0.2153	9.0.2153
vim-filesystem	9.0.2153	9.0.2153
vim-minimal	9.0.2153	9.0.2153
virt-what	1.18	
wget	1.14	1.21.3
which	2.20	2.21
words	3.0	3.0
xfsdump	3.1.8	3.1.11
xfspgrog	5.0.0	5.18.0
xxd	9.0.2153	9.0.2153
xxhash-libs		0.8.0
xz	5.2.2	5.2.5
xz-libs	5.2.2	5.2.5
yajl	2.0.4	

套件	AL2 阿 AMI	阿 AMI
yum	3.4.3	4.14.0
yum-langpacks	0.4.2	
yum-metadata-parser	1.1.4	
yum-plugin-priorities	1.1.31	
yum-utils	1.1.31	
zip	3.0	3.0
zlib	1.2.7	1.2.11
zram-generator		1.1.2
zram-generator-defaults		1.1.2
zstd		1.5.5

比較 Amazon Linux 2 與 Amazon 2023 最小 AMI 安裝的套件

目前在 Amazon Linux 2 和 AL2023 最小 AMI 上的 RPM 的比較。

套件	AL2 最小	AL2023 最小
acl	2.2.51	
alternatives		1.15
amazon-chrony-config		4.3
amazon-ec2-net-utils		2.4.1
amazon-linux-extras	2.0.3	

套件	AL2 最小	AL2023 最小
amazon-linux-repo-s3		2023.4.20240513
amazon-linux-sb-keys		2023.1
audit	2.8.1	3.0.6
audit-libs	2.8.1	3.0.6
authconfig	6.2.8	
awscli-2		2.15.30
basesystem	10.0	11
bash	4.2.46	5.2.15
bind-export-libs	9.11.4	
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023.2.64	2023.2.64
checkpolicy		3.4
chkconfig	1.7.4	
chrony	4.2	4.3
cloud-init	19.3	22.2.2
cloud-init-cfg-ec2		22.2.2
cloud-utils-growpart	0.31	0.31
coreutils	8.22	8.32
coreutils-common		8.32
cpio	2.12	2.13

套件	AL2 最小	AL2023 最小
cracklib	2.9.0	2.9.6
cracklib-dicts	2.9.0	2.9.6
cronie	1.4.11	
cronie-anacron	1.4.11	
crontabs	1.11	
crypto-policies		20220428
cryptsetup-libs	1.7.4	2.6.1
curl	8.3.0	
curl-minimal		8.5.0
cyrus-sasl-lib	2.1.26	2.1.27
dbus	1.10.24	1.12.28
dbus-broker		32
dbus-common		1.12.28
dbus-libs	1.10.24	1.12.28
device-mapper	1.02.170	1.02.185
device-mapper-libs	1.02.170	1.02.185
dhclient	4.2.5	
dhcp-common	4.2.5	
dhcp-libs	4.2.5	
diffutils	3.3	3.8

套件	AL2 最小	AL2023 最小
dnf		4.14.0
dnf-data		4.14.0
dnf-plugin-release-notification		1.2
dnf-plugins-core		4.3.0
dnf-plugin-support-info		1.2
dracut	033	055
dracut-config-ec2	2.0	3.0
dracut-config-generic	033	055
e2fsprogs	1.42.9	1.46.5
e2fsprogs-libs	1.42.9	1.46.5
ec2-utils	1.2	2.2.0
efibootmgr	15 (一個建築 64)	
efi-filesystem		5
efivar		38
efivar-libs	31 (一個建築 64)	38
elfutils-default-yama-scope	0.176	0.188
elfutils-libelf	0.176	0.188
elfutils-libs	0.176	0.188

套件	AL2 最小	AL2023 最小
expat	2.1.0	2.5.0
file	5.11	5.39
file-libs	5.11	5.39
filesystem	3.2	3.14
findutils	4.5.11	4.8.0
fipscheck	1.4.1	
fipscheck-lib	1.4.1	
freetype	2.8	
fuse-libs	2.9.2	2.9.9
gawk	4.0.2	5.1.0
gdbm	1.13	
gdbm-libs		1.19
gdisk	0.8.10	1.0.8
gettext	0.19.8.1	0.21
gettext-libs	0.19.8.1	0.21
glib2	2.56.1	2.74.7
glibc	2.26	2.34
glibc-all-langpacks	2.26	2.34
glibc-common	2.26	2.34
glibc-locale-source	2.26	2.34

套件	AL2 最小	AL2023 最小
<code>glibc-minimal-langpack</code>	2.26	
<code>gmp</code>	6.0.0	6.2.1
gnupg2	2.0.22	
gnupg2-minimal		2.3.7
<code>gnutls</code>		3.8.0
<code>gpgme</code>	1.3.2	1.15.1
<code>grep</code>	2.20	3.8
<code>groff-base</code>	1.22.2	1.22.4
<code>grub2</code>	2.06	
<code>grub2-common</code>	2.06	2.06
<code>grub2-efi-aa64</code>	2.06 (一個建築 64)	
<code>grub2-efi-aa64-ec2</code>	2.06 (一個建築 64)	2.06 (一個建築 64)
<code>grub2-efi-aa64-modules</code>	2.06 (公爵夫人)	
<code>grub2-efi-x64-ec2</code>	2.06 (十六)	2.06 (十六)
<code>grub2-pc</code>	2.06 (十六)	
<code>grub2-pc-modules</code>	2.06 (公爵夫人)	2.06
<code>grub2-tools</code>	2.06	2.06
<code>grub2-tools-minimal</code>	2.06	2.06
<code>grubby</code>	8.28	8.40

套件	AL2 最小	AL2023 最小
gzip	1.5	1.12
hardlink	1.3	
hostname	3.13	3.23
hwdata		0.353
info	5.1	
inih		49
initscripts	9.49.47	10.09
iproute	5.10.0	5.10.0
iptables	1.8.4	
iptables-libs	1.8.4	
iputils	20180629	20210202
irqbalance	1.7.0	1.9.0
jansson		2.14
jitterentropy		3.4.1
jq		1.7.1
json-c		0.14
kbd		2.4.0
kbd-misc		2.4.0
kernel	4.14.343	6.1.90
kernel-livepatch-r epo-s3		2023.4.20240513

套件	AL2 最小	AL2023 最小
keyutils-libs	1.5.8	1.6.3
kmod	25	29
kmod-libs	25	29
kpartx	0.4.9	
krb5-libs	1.15.1	1.21
less	458	608
libacl	2.2.51	2.3.1
libarchive		3.5.3
libargon2		20171227
libassuan	2.1.0	2.5.5
libattr	2.4.46	2.5.1
libblkid	2.30.2	2.37.4
libcap	2.54	2.48
libcap-ng	0.7.5	0.8.2
libcbor		0.7.0
libcom_err	1.42.9	1.46.5
libcomps		0.1.20
libcroco	0.6.12	
libcrypt	2.26	
libcurl	8.3.0	

套件	AL2 最小	AL2023 最小
libcurl-minimal		8.5.0
libdb	5.3.21	5.3.28
libdb-utils	5.3.21	
libdnf		0.69.0
libeconf		0.4.0
libedit	3.0	3.1
libestr	0.1.9	
libfastjson	0.99.4	
libfdisk	2.30.2	2.37.4
libffi	3.0.13	3.4.4
libfido2		1.10.0
libgcc	7.3.1	11.4.1
libgcrypt	1.5.3	1.10.2
libgomp	7.3.1	11.4.1
libgpg-error	1.12	1.42
libicu	50.2	
libidn	1.28	
libidn2	2.3.0	2.3.2
libkcapi		1.4.0
libkcapi-hmacalc		1.4.0

套件	AL2 最小	AL2023 最小
libmetalink	0.1.3	
libmnl	1.0.3	1.0.4
libmodulemd		2.13.0
libmount	2.30.2	2.37.4
libnetfilter_connt rack	1.0.6	
libnfnetlink	1.0.1	
libnhttp2	1.41.0	1.59.0
libpcap	1.5.3	
libpipeline	1.2.3	1.5.3
libpng	1.5.13	
libpsl	0.21.5	0.21.1
libpwquality	1.2.3	1.4.4
librepo		1.14.5
libreport-filesystem		2.15.2
libseccomp	2.5.2	2.5.3
libselenium	2.5	3.4
libselenium-utils	2.5	3.4
libsemanage	2.5	3.4
libsepol	2.5	3.4
libsigsegv		2.13

套件	AL2 最小	AL2023 最小
libsmartcols	2.30.2	2.37.4
libsolv		0.7.22
libss	1.42.9	1.46.5
libssh2	1.4.3	
libstdc++	7.3.1	11.4.1
libsysfs	2.1.0	
libtasn1	4.10	4.19.0
libtextstyle		0.21
libunistring	0.9.3	0.9.10
libuser	0.60	0.63
libutempter	1.1.6	1.2.1
libuuid	2.30.2	2.37.4
libverto	0.2.5	0.3.2
libxcrypt		4.4.33
libxml2	2.9.1	2.10.4
libyaml	0.1.4	0.2.5
libzstd		1.5.5
logrotate	3.8.6	3.20.1
lua	5.1.4	
lua-libs		5.4.4

套件	AL2 最小	AL2023 最小
lz4	1.7.5	
lz4-libs		1.9.4
make	3.82	
man-db	2.6.3	2.9.3
mariadb-libs	5.5.68	
microcode_ctl	2.1 (六十四)	2.1 (六十四)
mpfr		4.1.0
ncurses	6.0	6.2
ncurses-base	6.0	6.2
ncurses-libs	6.0	6.2
nettle	2.7.1	3.8
net-tools	2.0	2.0
newt	0.52.15	
newt-python	0.52.15	
npth		1.6
nspr	4.35.0	
nss	3.90.0	
nss-pem	1.0.3	
nss-softokn	3.90.0	
nss-softokn-freebl	3.90.0	

套件	AL2 最小	AL2023 最小
nss-sysinit	3.90.0	
nss-tools	3.90.0	
nss-util	3.90.0	
numactl-libs	2.0.9	2.0.14
oniguruma		6.9.7.1
openldap	2.4.44	2.4.57
openssh	七分一	8.7 小時
openssh-clients	七分一	8.7 小時
openssh-server	七分一	8.7 小時
openssl	1.0.2K	3.0.8
openssl-libs	1.0.2K	3.0.8
openssl-pkcs11		0.4.12
os-prober	1.58	1.77
p11-kit	0.23.22	0.24.1
p11-kit-trust	0.23.22	0.24.1
pam	1.1.8	1.5.1
passwd	0.79	0.80
pciutils		3.7.0
pciutils-libs		3.7.0
pcre	8.32	

套件	AL2 最小	AL2023 最小
pcre2	10.23	10.40
pcre2-syntax		10.40
pinentry	0.8.1	
pkgconfig	0.27.1	
policycoreutils	2.5	3.4
popt	1.13	1.18
postfix	2.10.1	
procps-ng	3.3.10	3.3.17
psmisc	22.20	23.4
pth	2.0.7	
publicsuffix-list-dafsa	20240208	20240212
pygpgme	0.3	
pyliblzma	0.5.3	
python	2.7.18	
python2-cryptography	1.7.2	
python2-jjsonschema	2.5.1	
python2-oauthlib	2.0.1	
python2-pyasn1	0.1.9	
python2-rpm	4.11.3	
python2-setuptools	41.2.0	

套件	AL2 最小	AL2023 最小
python2-six	1.11.0	
python3		3.9.16
python3-attrs		20.3.0
python3-audit		3.0.6
python3-awscrt		0.19.19
python3-babel		2.9.1
python3-cffi		1.14.5
python3-chardet		4.0.0
python3-colorama		0.4.4
python3-configobj		5.0.6
python3-cryptography		36.0.1
python3-dateutil		2.8.1
python3-dbus		1.2.18
python3-distro		1.5.0
python3-dnf		4.14.0
python3-dnf-plugins-core		4.3.0
python3-docutils		0.16
python3-gpg		1.15.1
python3-hawkey		0.69.0
python3-idna		2.10

套件	AL2 最小	AL2023 最小
python3-jinja2		2.11.3
python3-jmespath		0.10.0
python3-jsonpatch		1.21
python3-jsonpointer		2.0
python3-jsonschema		3.2.0
python3-libcomps		0.1.20
python3-libdnf		0.69.0
python3-libs		3.9.16
python3-libselenium		3.4
python3-libsemanage		3.4
python3-markupsafe		1.1.1
python3-netifaces		0.10.6
python3-oauthlib		3.0.2
python3-pip-wheel		21.3.1
python3-ply		3.11
python3-policycore utils		3.4
python3-prettytable		0.7.2
python3-prompt-too lkit		3.0.24
python3-pycparser		2.20

套件	AL2 最小	AL2023 最小
python3-pyrsistent		0.17.3
python3-pyserial		3.4
python3-pysocks		1.7.1
python3-pytz		2022.7.1
python3-pyyaml		5.4.1
python3-requests		2.25.1
python3-rpm		4.16.1.3
python3-ruamel-yaml		0.16.6
python3-ruamel-yaml-clib		0.1.2
python3-setools		4.4.1
python3-setuptools		59.6.0
python3-setuptools-wheel		59.6.0
python3-six		1.15.0
python3-systemd		235
python3-urllib3		1.25.10
python3-wcwidth		0.2.5
python-babel	0.9.6	
python-backports	1.0	

套件	AL2 最小	AL2023 最小
python-backports-s sl_match_hostname	3.5.0.1	
python-cffi	1.6.0	
python-chardet	2.2.1	
python-configobj	4.7.2	
python-devel	2.7.18	
python-enum34	1.0.4	
python-idna	2.4	
python-iniparse	0.4	
python-ipaddress	1.0.16	
python-jinja2	2.7.2	
python-jsonpatch	1.2	
python-jsonpointer	1.9	
python-jwcrypto	0.4.2	
python-libs	2.7.18	
python-markupsafe	0.11	
python-ply	3.4	
python-pycparser	2.14	
python-pycurl	7.19.0	
python-repoze-lru	0.4	
python-requests	2.6.0	

套件	AL2 最小	AL2023 最小
python-urlgrabber	3.10	
python-urllib3	1.25.9	
pyxattr	0.5.1	
PyYAML	3.10	
qrencode-libs	3.4.1	
readline	6.2	8.1
rng-tools	6.8	6.14
rootfiles	8.1	8.1
rpm	4.11.3	4.16.1.3
rpm-build-libs	4.11.3	4.16.1.3
rpm-libs	4.11.3	4.16.1.3
rpm-plugin-selinux		4.16.1.3
rpm-plugin-systemd-inhibit	4.11.3	4.16.1.3
rpm-sign-libs		4.16.1.3
rsyslog	8.24.0	
sbsigntools		0.9.4
sed	4.2.2	4.8
selinux-policy	3.13.1	37.22
selinux-policy-targeted	3.13.1	37.22

套件	AL2 最小	AL2023 最小
setup	2.8.71	2.13.7
shadow-utils	4.1.5.1	4.9
shared-mime-info	1.8	
slang	2.2.4	
sqlite	3.7.17	
sqlite-libs		3.40.0
sudo	1.8.23	1.9.15
sysctl-defaults	1.0	1.0
systemd	219	252.16
systemd-libs	219	252.16
systemd-networkd		252.16
systemd-pam		252.16
systemd-resolved		252.16
systemd-sysv	219	
systemd-udev		252.16
system-release	2	2023.4.20240513
sysvinit-tools	2.88	
tar	1.26	1.34
tcp_wrappers-libs	7.6	
tzdata	2024a	2024a

套件	AL2 最小	AL2023 最小
update-motd	1.1.2	2.2
userspace-rcu		0.12.1
ustr	1.0.4	
util-linux	2.30.2	2.37.4
util-linux-core		2.37.4
vim-data	9.0.2153	9.0.2153
vim-minimal	9.0.2153	9.0.2153
which	2.20	2.21
xfspgrog	5.0.0	5.18.0
xz	5.2.2	5.2.5
xz-libs	5.2.2	5.2.5
yum	3.4.3	4.14.0
yum-metadata-parser	1.1.4	
yum-plugin-priorities	1.1.31	
zlib	1.2.7	1.2.11
zram-generator		1.1.2
zram-generator-defaults		1.1.2
zstd		1.5.5

比較 Amazon Linux 2 與 Amazon 2023 基本容器映像安裝的套件

目前在 Amazon Linux 2 和 AL2023 基本容器映像上的 RPM 的比較。

套件	AL2 貨櫃	AL2023 集裝箱
alternatives		1.15
amazon-linux-extras	2.0.3	
amazon-linux-repo-cdn		2023.4.20240513
audit-libs		3.0.6
basesystem	10.0	11
bash	4.2.46	5.2.15
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023.2.64	2023.2.64
chkconfig	1.7.4	
coreutils	8.22	
coreutils-single		8.32
cpio	2.12	
crypto-policies		20220428
curl	8.3.0	
curl-minimal		8.5.0
cyrus-sasl-lib	2.1.26	
diffutils	3.3	

套件	AL2 貨櫃	AL2023 集裝箱
dnf		4.14.0
dnf-data		4.14.0
elfutils-default-yama-scope		0.188
elfutils-libelf	0.176	0.188
elfutils-libs		0.188
expat	2.1.0	2.5.0
file-libs	5.11	5.39
filesystem	3.2	3.14
findutils	4.5.11	
gawk	4.0.2	5.1.0
gdbm	1.13	
gdbm-libs		1.19
glib2	2.56.1	2.74.7
glibc	2.26	2.34
glibc-common	2.26	2.34
glibc-langpack-en	2.26	
glibc-minimal-langpack	2.26	2.34
gmp	6.0.0	6.2.1
gnupg2	2.0.22	

套件	AL2 貨櫃	AL2023 集裝箱
gnupg2-minimal		2.3.7
gpgme	1.3.2	1.15.1
grep	2.20	3.8
info	5.1	
json-c		0.14
keyutils-libs	1.5.8	1.6.3
krb5-libs	1.15.1	1.21
libacl	2.2.51	2.3.1
libarchive		3.5.3
libassuan	2.1.0	2.5.5
libattr	2.4.46	2.5.1
libblkid	2.30.2	2.37.4
libcap	2.54	2.48
libcap-ng		0.8.2
libcom_err	1.42.9	1.46.5
libcomps		0.1.20
libcrypt	2.26	
libcurl	8.3.0	
libcurl-minimal		8.5.0
libdb	5.3.21	

套件	AL2 貨櫃	AL2023 集裝箱
libdb-utils	5.3.21	
libdnf		0.69.0
libffi	3.0.13	3.4.4
libgcc	7.3.1	11.4.1
libgcrypt	1.5.3	1.10.2
libgomp		11.4.1
libgpg-error	1.12	1.42
libidn2	2.3.0	2.3.2
libmetalink	0.1.3	
libmodulemd		2.13.0
libmount	2.30.2	2.37.4
libnghttp2	1.41.0	1.59.0
libpsl	0.21.5	0.21.1
librepo		1.14.5
libreport-filessystem		2.15.2
libselinux	2.5	3.4
libsepol	2.5	3.4
libsigsegv		2.13
libsmartcols		2.37.4
libsolv		0.7.22

套件	AL2 貨櫃	AL2023 集裝箱
libssh2	1.4.3	
libstdc++	7.3.1	11.4.1
libtasn1	4.10	4.19.0
libunistring	0.9.3	0.9.10
libuuid	2.30.2	2.37.4
libverto	0.2.5	0.3.2
libxcrypt		4.4.33
libxml2	2.9.1	2.10.4
libyaml		0.2.5
libzstd		1.5.5
lua	5.1.4	
lua-libs		5.4.4
lz4-libs		1.9.4
mpfr		4.1.0
ncurses	6.0	
ncurses-base	6.0	6.2
ncurses-libs	6.0	6.2
npth		1.6
nspr	4.35.0	
nss	3.90.0	

套件	AL2 貨櫃	AL2023 集裝箱
nss-pem	1.0.3	
nss-softokn	3.90.0	
nss-softokn-freebl	3.90.0	
nss-sysinit	3.90.0	
nss-tools	3.90.0	
nss-util	3.90.0	
openldap	2.4.44	
openssl-libs	1.0.2K	3.0.8
p11-kit	0.23.22	0.24.1
p11-kit-trust	0.23.22	0.24.1
pcre	8.32	
pcre2		10.40
pcre2-syntax		10.40
pinentry	0.8.1	
popt	1.13	1.18
pth	2.0.7	
publicsuffix-list-dafsa	20240208	20240212
pygpgme	0.3	
pyliblzma	0.5.3	
python	2.7.18	

套件	AL2 貨櫃	AL2023 集裝箱
python2-rpm	4.11.3	
python3		3.9.16
python3-dnf		4.14.0
python3-gpg		1.15.1
python3-hawkey		0.69.0
python3-libcomps		0.1.20
python3-libdnf		0.69.0
python3-libs		3.9.16
python3-pip-wheel		21.3.1
python3-rpm		4.16.1.3
python3-setuptools-wheel		59.6.0
python-iniparse	0.4	
python-libs	2.7.18	
python-pycurl	7.19.0	
python-urlgrabber	3.10	
pyxattr	0.5.1	
readline	6.2	8.1
rpm	4.11.3	4.16.1.3
rpm-build-libs	4.11.3	4.16.1.3
rpm-libs	4.11.3	4.16.1.3

套件	AL2 貨櫃	AL2023 集裝箱
rpm-sign-libs		4.16.1.3
sed	4.2.2	4.8
setup	2.8.71	2.13.7
shared-mime-info	1.8	
sqlite	3.7.17	
sqlite-libs		3.40.0
system-release	2	2023.4.20240513
tzdata	2024a	2024a
vim-data	9.0.2153	
vim-minimal	9.0.2153	
xz-libs	5.2.2	5.2.5
yum	3.4.3	4.14.0
yum-metadata-parser	1.1.4	
yum-plugin-ovl	1.1.31	
yum-plugin-priorities	1.1.31	
zlib	1.2.7	1.2.11

比較 AL1 與 AL2023

以下主題描述了 AL1 和 AL2023 之間的主要差異，這些差異尚未涵蓋[與 AL2 的比較](#)。

Note

AL1 於 2023 年 12 月 31 日達成其 end-of-life (EOL)，自 2024 年 1 月 1 日起將不會收到任何安全性更新或錯誤修正。如需 AL1 EOL 和維護支援的詳細資訊，請參閱部落格文章 [Amazon Linux AMI 上的更新](#)。end-of-life 建議您將應用程式升級至 AL2023，其中包括直到 2028 年的長期支援。

主題

- [支援各個版本](#)
- [systemd 取代 upstart 並成為 init 系統](#)
- [Python 2.6 和 2.7 已替換為 Python 3](#)
- [OpenJDK 8 作為最舊版 JDK](#)
- [從 Amazon 1 的 AL2023 內核更改](#)
- [比較 Amazon Linux 1 \(AL1\) 與 Amazon 2023 AMI 安裝的套件](#)
- [比較 Amazon Linux 1 \(AL1\) 與 Amazon Linux 2023 最小 AMI 安裝的套件](#)
- [比較 Amazon Linux 1 \(AL1\) 與 Amazon 2023 基本容器映像安裝的套件](#)

支援各個版本

對於 AL2023，我們從發布之日起提供五年的支持。AL1 已於 2020 年 12 月 31 日結束標準支援，並於 2023 年 12 月 31 日結束維護支援。

如需詳細資訊，請參閱 [發行節奏](#)。

systemd 取代 upstart 並成為 init 系統

在 AL2 中 upstart 被替換 systemd 為 init 系統。AL2023 還使用 systemd 作為其 init 系統，進一步採用的新特性和功能 systemd。

Python 2.6 和 2.7 已替換為 Python 3

雖然 AL1 在 2018.03 版本中將 Python 2.6 標記為 EOL，但是這些套件仍然可以在軟體庫中安裝。AL2 隨 Python 2.7 作為最早支持的 Python 版本一起提供，AL2023 完成了對 Python 3 的過渡。AL2023 儲存庫中不包含任何 Python 2.x 版本。

如需有關 Amazon Linux 的 Python 的詳細資訊，請參閱 [AL2023 中的 Python](#)。

OpenJDK 8 作為最舊版 JDK

AL2023 隨附 [Amazon Corretto](#) 作為預設 (也是唯一) Java 開發套件 (JDK)。AL2023 中的所有 Java 基礎軟件包都是使用 Amazon Corretto 17。

在 AL1 中，OpenJDK 1.6.0 (java-1.6.0-openjdk) 在第一個 2018.03 版本中停止使用，而 OpenJDK 1.7.0 (java-1.7.0-openjdk) 在 2020 年中期停止使用，儘管這兩個版本都在 AL1 儲存庫中提供。AL2023 中可用的最早的 OpenJDK 版本是 OpenJDK 8，由提供。Amazon Corretto 8

從 Amazon 1 的 AL2023 內核更改

Kernel Live Patching

AL2023 和 AL2 都增加了對內核實時補丁功能的支持。這可讓您修補 Linux 核心中重要且重要的安全性弱點，而不需要重新開機或停機。如需詳細資訊，請參閱 [AL2023 上的核心即時修補程式](#)。

核心檔案系統支援

AL1 中的核心將支援掛載的檔案系統發生了幾項變更，以及核心將剖析的磁碟分割配置中的變更。

CONFIG 選項	AL1/4.14/x86_64	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_AFS_FS	m	n	n
CONFIG_AF_RXRPC	m	n	n
CONFIG_BS D_DISKLABEL	y	n	n
CONFIG_CRAMFS	m	n	n

CONFIG 選項	AL1/4.14/x86_64	AL2023/6.1/aarch64	AL2023/6.1/x86_64
<u>CONFIG_CRAMFS_BLOCKDEV</u>	N/A	N/A	N/A
<u>CONFIG_DM_CLONE</u>	N/A	n	n
<u>CONFIG_DM_ERA</u>	n	n	n
<u>CONFIG_DM_INTEGRITY</u>	m	m	m
<u>CONFIG_DM_LOG_WRITES</u>	n	m	m
<u>CONFIG_DM_SWITCH</u>	n	n	n
<u>CONFIG_DM_VERITY</u>	n	n	n
<u>CONFIG_ECRYPT_FS</u>	m	n	n
<u>CONFIG_EXFAT_FS</u>	N/A	m	m
<u>CONFIG_EXT2_FS</u>	m	n	n
<u>CONFIG_EXT3_FS</u>	m	n	n
<u>CONFIG_GFS2_FS</u>	n	n	n
<u>CONFIG_HFSPLUS_FS</u>	m	n	n
<u>CONFIG_HFS_FS</u>	m	n	n
<u>CONFIG_JFS_FS</u>	m	n	n

CONFIG 選項	AL1/4.14/x86_64	AL2023/6.1/aarch64	AL2023/6.1/x86_64
<u>CONFIG_LD M_PARTITION</u>	y	n	n
<u>CONFIG_MA C_PARTITION</u>	y	n	n
<u>CONFIG_NFS_V2</u>	m	n	n
<u>CONFIG_NTFS_FS</u>	m	n	n
<u>CONFIG_ROMFS_FS</u>	m	n	n
<u>CONFIG_S0 LARIS_X86 _PARTITION</u>	y	n	n
<u>CONFIG_SQ UASHFS_ZSTD</u>	y	y	y
<u>CONFIG_SU N_PARTITION</u>	y	n	n

重視安全的核心設定變更

CONFIG 選項	AL1/4.14/x86_64	AL2023/6.1/aarch64	AL2023/6.1/x86_64
<u>CONFIG_BU G_ON_DATA _CORRUPTION</u>	y	y	y
<u>CONFIG_DE FAULT_MMA P_MIN_ADDR</u>	4096	65536	65536
<u>CONFIG_DEVMEM</u>	y	n	n

CONFIG 選項	AL1/4.14/x86_64	AL2023/6.1/aarch64	AL2023/6.1/x86_64
<u>CONFIG_DEVPORT</u>	y	n	n
<u>CONFIG_FORTIFY_SOURCE</u>	y	y	y
<u>CONFIG_HARDENED_USERCOPY_FALLBACK</u>	N/A	N/A	N/A
<u>CONFIG_INIT_ON_ALLOC_DEFAULT_ON</u>	N/A	n	n
<u>CONFIG_INIT_ON_FREE_DEFAULT_ON</u>	N/A	n	n
<u>CONFIG_IOMMU_DEFAULT_STRICT</u>	N/A	n	n
<u>CONFIG_LDISC_AUTOLOAD</u>	y	n	n
<u>CONFIG_SCHED_CORE</u>	N/A	N/A	y
<u>CONFIG_SCHED_STACK_END_CHECK</u>	y	y	y
<u>CONFIG_SECURITY_DMESG_RESTRICT</u>	n	y	y

CONFIG 選項	AL1/4.14/x86_64	AL2023/6.1/aarch64	AL2023/6.1/x86_64
<u>CONFIG_SECURITY_LINUX_DISABLE</u>	y	n	n
<u>CONFIG_SHUFFLE_PAGE_ALLOCATOR</u>	N/A	y	y
<u>CONFIG_SLAB_FREELIST_HARDENED</u>	y	y	y
<u>CONFIG_SLAB_FREELIST_RANDOM</u>	n	y	y

其他核心設定變更

CONFIG 選項	AL1/4.14/x86_64	AL2023/6.1/aarch64	AL2023/6.1/x86_64
<u>CONFIG_HZ</u>	250	100	100
<u>CONFIG_NR_CPUS</u>	8192	512	512
<u>CONFIG_PANIC_ON_OOPS</u>	n	y	y
<u>CONFIG_PANIC_ON_OOPS_VALUE</u>	0	1	1
<u>CONFIG_PPP</u>	m	n	n
<u>CONFIG_SLIP</u>	m	n	n
<u>CONFIG_XEN_PV</u>	y	N/A	n

比較 Amazon Linux 1 (AL1) 與 Amazon 2023 AMI 安裝的套件

AL1 和 AL2023 標準 AMI 上存在的 RPM 的比較。

套件	阿 AMI	阿 AMI
acl	2.2.49	2.3.1
acpid	2.0.19	2.0.32
alsa-lib	1.0.22	
alternatives		1.15
amazon-chroney-config		4.3
amazon-ec2-net-utils		2.4.1
amazon-linux-repo-s3		2023.4.20240513
amazon-linux-sb-keys		2023.1
amazon-rpm-config		228
amazon-ssm-agent	3.2.2222.0	3.3.380.0
at	3.1.10	3.1.23
attr	2.4.46	2.5.1
audit	2.6.5	3.0.6
audit-libs	2.6.5	3.0.6
authconfig	6.2.8	
aws-amitools-ec2	1.5.13	
aws-cfn-bootstrap	1.4	2.0
aws-cli	1.18.107	

套件	阿 AMI	阿 AMI
awscli-2		2.15.30
basesystem	10.0	11
bash	4.2.46	5.2.15
bash-completion		2.11
bc	1.06.95	1.07.1
bind-libs	9.8.2	9.16.48
bind-license		9.16.48
bind-utils	9.8.2	9.16.48
binutils	2.27	2.39
boost-filesystem		1.75.0
boost-system		1.75.0
boost-thread		1.75.0
bzip2	1.0.6	1.0.8
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023.2.62	2023.2.64
c-ares		1.19.0
checkpolicy	2.1.10	3.4
chkconfig	1.3.49.3	1.15
chrony		4.3
cloud-disk-utils	0.27	

套件	阿 AMI	阿 AMI
cloud-init	0.7.6	22.2.2
cloud-init-cfg-ec2		22.2.2
cloud-utils-growpart		0.31
copy-jdk-configs	3.3	
coreutils	8.22	8.32
coreutils-common		8.32
cpio	2.10	2.13
cracklib	2.8.16	2.9.6
cracklib-dicts	2.8.16	2.9.6
cronie	1.4.4	
cronie-anacron	1.4.4	
crontabs	1.10	1.11
crypto-policies		20220428
crypto-policies-scripts		20220428
cryptsetup	1.6.7	2.6.1
cryptsetup-libs	1.6.7	2.6.1
curl	7.61.1	
curl-minimal		8.5.0
cyrus-sasl	2.1.23	
cyrus-sasl-lib	2.1.23	2.1.27

套件	阿 AMI	阿 AMI
cyrus-sasl-plain	2.1.23	2.1.27
dash	0.5.5.1	
db4	4.7.25	
db4-utils	4.7.25	
dbus	1.6.12	1.12.28
dbus-broker		32
dbus-common		1.12.28
dbus-libs	1.6.12	1.12.28
dejavu-fonts-common	2.33	
dejavu-sans-fonts	2.33	
dejavu-serif-fonts	2.33	
device-mapper	1.02.135	1.02.185
device-mapper-event	1.02.135	
device-mapper-event-libs	1.02.135	
device-mapper-libs	1.02.135	1.02.185
device-mapper-persistent-data	0.6.3	
dhclient	4.1.1	
dhcp-common	4.1.1	
diffutils	3.3	3.8

套件	阿 AMI	阿 AMI
dmraid	1.0.rc16	
dmraid-events	1.0.rc16	
dnf		4.14.0
dnf-data		4.14.0
dnf-plugin-release-notification		1.2
dnf-plugins-core		4.3.0
dnf-plugin-support-info		1.2
dnf-utils		4.3.0
dosfstools		4.2
dracut	004	055
dracut-config-ec2		3.0
dracut-config-generic		055
dracut-modules-growroot	0.20	
dump	0.4	
dwz		0.14
dyninst		10.2.1
e2fsprogs	1.43.5	1.46.5
e2fsprogs-libs	1.43.5	1.46.5

套件	阿 AMI	阿 AMI
ec2-hibinit-agent	1.0.0	1.0.8
ec2-instance-connect		1.1
ec2-instance-connect-selinux		1.1
ec2-net-utils	0.7	
ec2-utils	0.7	2.2.0
ed	1.1	1.14.2
efi-filesystem		5
efi-srpm-macros		5
efivar		38
efivar-libs		38
elfutils-debuginfod-client		0.188
elfutils-default-yama-scope		0.188
elfutils-libelf	0.168	0.188
elfutils-libs		0.188
epel-release	6	
ethtool	3.15	5.15
expat	2.1.0	2.5.0
file	5.37	5.39

套件	阿 AMI	阿 AMI
file-libs	5.37	5.39
filesystem	2.4.30	3.14
findutils	4.4.2	4.8.0
fipscheck	1.3.1	
fipscheck-lib	1.3.1	
fontconfig	2.8.0	
fontpackages-files system	1.41	
fonts-srpm-macros		2.0.5
freetype	2.3.11	
fstrm		0.6.1
fuse-libs	2.9.4	2.9.9
gawk	3.1.7	5.1.0
gdbm	1.8.0	
gdbm-libs		1.19
gdisk	0.8.10	1.0.8
generic-logos	17.0.0	
get_reference_source	1.2	
gettext		0.21
gettext-libs		0.21
ghc-srpm-macros		1.5.0

套件	阿 AMI	阿 AMI
giflib	4.1.6	
glib2	2.36.3	2.74.7
glibc	2.17	2.34
glibc-all-langpacks		2.34
glibc-common	2.17	2.34
glibc-gconv-extra		2.34
glibc-locale-source		2.34
gmp	6.0.0	6.2.1
gnupg2	2.0.28	
gnupg2-minimal		2.3.7
gnutls		3.8.0
go-srpm-macros		3.2.0
gpgme	1.4.3	1.15.1
gpm-libs	1.20.6	1.20.7
grep	2.20	3.8
groff	1.22.2	
groff-base	1.22.2	1.22.4
grub	0.97	
grub2-common		2.06
grub2-efi-x64-ec2		2.06

套件	阿 AMI	阿 AMI
grub2-pc-modules		2.06
grub2-tools		2.06
grub2-tools-minimal		2.06
grubby	7.0.15	8.40
gssproxy		0.8.4
gzip	1.5	1.12
hesiod	3.1.0	
hibagent	1.0.0	
hmaccalc	0.9.12	
hostname		3.23
hunspell		1.7.0
hunspell-en		0.20140811.1
hunspell-en-GB		0.20140811.1
hunspell-en-US		0.20140811.1
hunspell-filesystem		1.7.0
hwdata	0.233	0.353
info	5.1	6.7
inih		49
initscripts	9.03.58	10.09
iproute	4.4.0	5.10.0

套件	阿 AMI	阿 AMI
iptables	1.4.21	
iputils	20121221	20210202
irqbalance	1.5.0	1.9.0
jansson		2.14
java-1.7.0-openjdk	1.7.0.321	
javapackages-tools	0.9.1	
jitterentropy		3.4.1
jpackage-utils	1.7.5	
jq		1.7.1
json-c		0.14
kbd	1.15	2.4.0
kbd-misc	1.15	2.4.0
kernel	4.14.336	6.1.90
kernel-livepatch-r epo-s3		2023.4.20240513
kernel-srpm-macros		1.0
kernel-tools	4.14.336	6.1.90
keyutils	1.5.8	1.6.3
keyutils-libs	1.5.8	1.6.3
kmod	14	29
kmod-libs	14	29

套件	阿 AMI	阿 AMI
kpartx	0.4.9	
kpatch-runtime		0.9.7
krb5-libs	1.15.1	1.21
lcms2	2.6	
less	436	608
libacl	2.2.49	2.3.1
libaio	0.3.109	0.3.111
libarchive		3.5.3
libargon2		20171227
libassuan	2.0.3	2.5.5
libattr	2.4.46	2.5.1
libbasicobjects		0.1.1
libblkid	2.23.2	2.37.4
libcap	2.16	2.48
libcap54	2.54	
libcap-ng	0.7.5	0.8.2
libcbor		0.7.0
libcgroup	0.40.rc1	
libcollection		0.7.0
libcom_err	1.43.5	1.46.5

套件	阿 AMI	阿 AMI
libcomps		0.1.20
libconfig		1.7.2
libcurl	7.61.1	
libcurl-minimal		8.5.0
libdb		5.3.28
libdhash		0.5.0
libdnf		0.69.0
libeconf		0.4.0
libedit	2.11	3.1
libev		4.33
libevent	2.0.21	2.1.12
libfdisk		2.37.4
libffi	3.0.13	3.4.4
libfido2		1.10.0
libfontenc	1.0.5	
libgcc		11.4.1
libgcc72	7.2.1	
libgcrypt	1.5.3	1.10.2
libgomp		11.4.1
libgpg-error	1.11	1.42

套件	阿 AMI	阿 AMI
libgssglue	0.1	
libibverbs		48.0
libICE	1.0.6	
libicu	50.2	
libidn	1.18	
libidn2	2.3.0	2.3.2
libini_config		1.3.1
libjpeg-turbo	1.2.90	
libkcap		1.4.0
libkcap-hmacalc		1.4.0
libldb		2.6.2
libmaxinddb		1.5.2
libmetalink		0.1.3
libmnl	1.0.3	1.0.4
libmodulemd		2.13.0
libmount	2.23.2	2.37.4
libnetfilter_conntrack	1.0.4	
libnfnetlink	1.0.1	
libnfsidmap	0.25	2.5.4
libnghttp2	1.33.0	1.59.0

套件	阿 AMI	阿 AMI
libnih	1.0.1	
libnl	1.1.4	
libnl3		3.5.0
libpath_utils		0.2.1
libpcap		1.10.1
libpipeline	1.2.3	1.5.3
libpkgconf		1.8.0
libpng	1.2.49	
libpsl	0.6.2	0.21.1
libpwquality	1.2.3	1.4.4
libref_array		0.1.5
librepo		1.14.5
libreport-filessystem		2.15.2
libseccomp		2.5.3
libselinux	2.1.10	3.4
libselinux-utils	2.1.10	3.4
libsemanage	2.1.6	3.4
libsepol	2.1.7	3.4
libsigsegv		2.13
libSM	1.2.1	

套件	阿 AMI	阿 AMI
libsmartcols	2.23.2	2.37.4
libsolv		0.7.22
libss	1.43.5	1.46.5
libssh2	1.4.2	
libsss_certmap		2.9.4
libsss_idmap		2.9.4
libsss_nss_idmap		2.9.4
libsss_sudo		2.9.4
libstdc++		11.4.1
libstdc++72	7.2.1	
libstoragegmt		1.9.4
libsysfs	2.1.0	
libtalloc		2.3.4
libtasn1	2.3	4.19.0
libtdb		1.4.7
libtevent		0.13.0
libtextstyle		0.21
libtirpc	0.2.4	1.3.3
libudev	173	
libunistring	0.9.3	0.9.10

套件	阿 AMI	阿 AMI
libuser	0.60	0.63
libutempter	1.1.5	1.2.1
libuuid	2.23.2	2.37.4
libuv		1.47.0
libverto	0.2.5	0.3.2
libverto-libev		0.3.2
libX11	1.6.0	
libX11-common	1.6.0	
libXau	1.0.6	
libxcb	1.11	
libXcomposite	0.4.3	
libxcrypt		4.4.33
libXext	1.3.2	
libXfont	1.4.5	
libXi	1.7.2	
libxml2	2.9.1	2.10.4
libxml2-python27	2.9.1	
libXrender	0.9.8	
libxslt	1.1.28	
libXtst	1.2.2	

套件	阿 AMI	阿 AMI
libyam1	0.1.6	0.2.5
libzstd		1.5.5
lm_sensors-libs		3.6.0
lmdb-libs		0.9.29
log4j-cve-2021-44228-hotpatch	1.3	
logrotate	3.7.8	3.20.1
lsof	4.82	4.94.0
lua	5.1.4	
lua-libs		5.4.4
lua-srpm-macros		1
lvm2	2.02.166	
lvm2-libs	2.02.166	
lz4-libs		1.9.4
mailcap	2.1.31	
make	3.82	
man-db	2.6.3	2.9.3
man-pages	4.10	5.10
mdadm	3.2.6	
microcode_ctl	2.1	2.1
mingetty	1.08	

套件	阿 AMI	阿 AMI
mpfr		4.1.0
nano	2.5.3	5.8
nc	1.84	
ncurses	5.7	6.2
ncurses-base	5.7	6.2
ncurses-libs	5.7	6.2
nettle		3.8
net-tools	1.60	2.0
newt	0.52.11	0.52.21
newt-python27	0.52.11	
nfs-utils	1.3.0	2.5.4
npth		1.6
nspr	4.25.0	4.35.0
nss	3.53.1	3.90.0
nss-pem	1.0.3	
nss-softokn	3.53.1	3.90.0
nss-softokn-freebl	3.53.1	3.90.0
nss-sysinit	3.53.1	3.90.0
nss-tools	3.53.1	
nss-util	3.53.1	3.90.0

套件	阿 AMI	阿 AMI
ntp	4.2.8p15	
ntpdate	4.2.8p15	
ntsysv	1.3.49.3	1.15
numactl	2.0.7	
numactl-libs		2.0.14
ocaml-srpm-macros		6
oniguruma		6.9.7.1
openblas-srpm-macros		2
openldap	2.4.40	2.4.57
openssh	七分一	8.7 小時
openssh-clients	七分一	8.7 小時
openssh-server	七分一	8.7 小時
openssl	1.0.2K	3.0.8
openssl-libs		3.0.8
openssl-pkcs11		0.4.12
os-prober		1.77
p11-kit	0.18.5	0.24.1
p11-kit-trust	0.18.5	0.24.1
package-notes-srpm-macros		0.4
pam	1.1.8	1.5.1

套件	阿 AMI	阿 AMI
pam_ccreds	10	
pam_krb5	2.3.11	
pam_passwdqc	1.0.5	
parted	2.1	3.4
passwd	0.79	0.80
pciutils	3.1.10	3.7.0
pciutils-libs	3.1.10	3.7.0
pcre	8.21	
pcre2		10.40
pcre2-syntax		10.40
perl	5.16.3	
perl-Carp	1.26	1.50
perl-Class-Struct		0.66
perl-constant	1.27	1.33
perl-Digest	1.17	
perl-Digest-HMAC	1.03	
perl-Digest-MD5	2.52	
perl-Digest-SHA	5.85	
perl-DynaLoader		1.47
perl-Encode	2.51	3.15

套件	阿 AMI	阿 AMI
perl-Errno		1.30
perl-Exporter	5.68	5.74
perl-Fcntl		1.13
perl-File-Basename		2.85
perl-File-Path	2.09	2.18
perl-File-stat		1.09
perl-File-Temp	0.23.01	0.231.100
perl-Filter	1.49	
perl-Getopt-Long	2.40	2.52
perl-Getopt-Std		1.12
perl-HTTP-Tiny	0.033	0.078
perl-if		0.60.800
perl-interpreter		5.32.1
perl-IO		1.43
perl-IPC-Open3		1.21
perl-libs	5.16.3	5.32.1
perl-macros	5.16.3	
perl-MIME-Base64		3.16
perl-mro		1.23
perl-overload		1.31

套件	阿 AMI	阿 AMI
perl-overloading		0.02
perl-parent	0.225	0.238
perl-PathTools	3.40	3.78
perl-Pod-Escapes	1.04	1.07
perl-podlators	2.5.1	4.14
perl-Pod-Perldoc	3.20	3.28.01
perl-Pod-Simple	3.28	3.42
perl-Pod-Usage	1.63	2.01
perl-POSIX		1.94
perl-Scalar-List-Utils	1.27	1.56
perl-SelectSaver		1.02
perl-Socket	2.010	2.032
perl-srpm-macros		1
perl-Storable	2.45	3.21
perl-subst		1.03
perl-Symbol		1.08
perl-Term-ANSIColor		5.01
perl-Term-Cap		1.17
perl-Text-ParseWords	3.29	3.30
perl-Text-Tabs+Wrap		2021.0726

套件	阿 AMI	阿 AMI
perl-threads	1.87	
perl-threads-shared	1.43	
perl-Time-HiRes	1.9725	
perl-Time-Local	1.2300	1.300
perl-vars		1.05
pinentry	0.7.6	
pkgconf		1.8.0
pkgconfig	0.27.1	
pkgconf-m4		1.8.0
pkgconf-pkg-config		1.8.0
pm-utils	1.4.1	
policycoreutils	2.1.12	3.4
policycoreutils-python-utils		3.4
popt	1.13	1.18
procmail	3.22	
procps	3.2.8	
procps-ng		3.3.17
protobuf-c		1.4.1
psacct	6.3.2	6.6.4
psmisc	22.20	23.4

套件	阿 AMI	阿 AMI
pth	2.0.7	
publicsuffix-list-dafsa		20240212
python27	2.7.18	
python27-babel	0.9.4	
python27-backports	1.0	
python27-backports-ssl_match_hostname	3.4.0.2	
python27-boto	2.48.0	
python27-botocore	1.17.31	
python27-chardet	2.0.1	
python27-colorama	0.4.1	
python27-configobj	4.7.2	
python27-crypto	2.6.1	
python27-daemon	1.5.2	
python27-dateutil	2.1	
python27-devel	2.7.18	
python27-docutils	0.11	
python27-ecdsa	0.11	
python27-futures	3.0.3	
python27-imaging	1.1.6	

套件	阿 AMI	阿 AMI
python27-iniparse	0.3.1	
python27-jinja2	2.7.2	
python27-jmespath	0.9.2	
python27-jsonpatch	1.2	
python27-jsonpointer	1.0	
python27-kitchen	1.1.1	
python27-libs	2.7.18	
python27-lockfile	0.8	
python27-markupsafe	0.11	
python27-paramiko	1.15.1	
python27-pip	9.0.3	
python27-ply	3.4	
python27-pyasn1	0.1.7	
python27-pycurl	7.19.0	
python27-pygpme	0.3	
python27-pyliblzma	0.5.3	
python27-pystache	0.5.3	
python27-pyattr	0.5.0	
python27-PyYAML	3.10	
python27-requests	1.2.3	

套件	阿 AMI	阿 AMI
python27-rsa	3.4.1	
python27-setuptools	36.2.7	
python27-simplejson	3.6.5	
python27-six	1.8.0	
python27-urlgrabber	3.10	
python27-urllib3	1.24.3	
python27-virtualenv	15.1.0	
python3		3.9.16
python3-attrs		20.3.0
python3-audit		3.0.6
python3-awscli		0.19.19
python3-babel		2.9.1
python3-cffi		1.14.5
python3-chardet		4.0.0
python3-colorama		0.4.4
python3-configobj		5.0.6
python3-cryptography		36.0.1
python3-daemon		2.3.0
python3-dateutil		2.8.1
python3-dbus		1.2.18

套件	阿 AMI	阿 AMI
python3-distro		1.5.0
python3-dnf		4.14.0
python3-dnf-plugins-core		4.3.0
python3-docutils		0.16
python3-gpg		1.15.1
python3-hawkey		0.69.0
python3-idna		2.10
python3-jinja2		2.11.3
python3-jmespath		0.10.0
python3-jsonpatch		1.21
python3-jsonpointer		2.0
python3-jsonschema		3.2.0
python3-libcomps		0.1.20
python3-libdnf		0.69.0
python3-libs		3.9.16
python3-libselenium		3.4
python3-libsemanage		3.4
python3-libstorage mgmt		1.9.4
python3-lockfile		0.12.2

套件	阿 AMI	阿 AMI
python3-markupsafe		1.1.1
python3-netifaces		0.10.6
python3-oauthlib		3.0.2
python3-pip-wheel		21.3.1
python3-ply		3.11
python3-policycore utils		3.4
python3-prettytable		0.7.2
python3-prompt-too lkit		3.0.24
python3-pycparser		2.20
python3-pyrsistent		0.17.3
python3-pyserial		3.4
python3-pysocks		1.7.1
python3-pytz		2022.7.1
python3-pyyaml		5.4.1
python3-requests		2.25.1
python3-rpm		4.16.1.3
python3-ruamel-yaml		0.16.6
python3-ruamel-yaml- clib		0.1.2

套件	阿 AMI	阿 AMI
python3-setools		4.4.1
python3-setuptools		59.6.0
python3-setuptools-wheel		59.6.0
python3-six		1.15.0
python3-systemd		235
python3-urllib3		1.25.10
python3-wcwidth		0.2.5
python-chevron		0.13.1
python-srpm-macros		3.9
quota	4.00	4.06
quota-nls	4.00	4.06
readline	6.2	8.1
rmt	0.4	
rng-tools	5	6.14
rootfiles	8.1	8.1
rpcbind	0.2.0	1.2.6
rpm	4.11.3	4.16.1.3
rpm-build-libs	4.11.3	4.16.1.3
rpm-libs	4.11.3	4.16.1.3
rpm-plugin-selinux		4.16.1.3

套件	阿 AMI	阿 AMI
rpm-plugin-systemd-inhibit		4.16.1.3
rpm-python27	4.11.3	
rpm-sign-libs		4.16.1.3
rsync	3.0.6	3.2.6
rsyslog	5.8.10	
ruby	2.0	
ruby20	2.0.0.648	
ruby20-irb	2.0.0.648	
ruby20-libs	2.0.0.648	
rubygem20-bigdecimal	1.2.0	
rubygem20-json	1.8.3	
rubygem20-psych	2.0.0	
rubygem20-rdoc	4.2.2	
rubygems20	2.0.14.1	
rust-srpm-macros		21
sbsigntools		0.9.4
screen	4.0.3	4.8.0
sed	4.2.1	4.8
selinux-policy		37.22

套件	阿 AMI	阿 AMI
selinux-policy-targeted		37.22
sendmail	8.14.4	
setserial	2.17	
setup	2.8.14	2.13.7
sgpio	1.2.0.10	
shadow-utils	4.1.4.2	4.9
shared-mime-info	1.1	
slang	2.2.1	2.3.2
sqlite	3.7.17	
sqlite-libs		3.40.0
sssd-client		2.9.4
sssd-common		2.9.4
sssd-kcm		2.9.4
sssd-nfs-idmap		2.9.4
strace		6.8
sudo	1.8.23	1.9.15
sysctl-defaults	1.0	1.0
sysfsutils	2.1.0	
sysstat		12.5.6
systemd		252.16

套件	阿 AMI	阿 AMI
systemd-libs		252.16
systemd-networkd		252.16
systemd-pam		252.16
systemd-resolved		252.16
systemd-udev		252.16
system-release	2018.03	2023.4.20240513
systemtap-runtime		4.8
sysvinit	2.87	
tar	1.26	1.34
tbb		2020.3
tcp_wrappers	7.6	
tcp_wrappers-libs	7.6	
tcpdump		4.99.1
tcsh		6.24.07
time	1.7	1.9
tmpwatch	2.9.16	
traceroute	2.0.14	2.1.3
ttmkfdir	3.0.9	
tzdata	2023c	2024a
tzdata-java	2023c	

套件	阿 AMI	阿 AMI
udev	173	
unzip	6.0	6.0
update-motd	1.0.1	2.2
upstart	0.6.5	
userspace-rcu		0.12.1
ustr	1.0.4	
util-linux	2.23.2	2.37.4
util-linux-core		2.37.4
vim-common	9.0.2120	9.0.2153
vim-data	9.0.2120	9.0.2153
vim-enhanced	9.0.2120	9.0.2153
vim-filesystem	9.0.2120	9.0.2153
vim-minimal	9.0.2120	9.0.2153
wget	1.18	1.21.3
which	2.19	2.21
words	3.0	3.0
xfsdump		3.1.11
xfspgrog		5.18.0
xorg-x11-fonts-Type1	7.2	
xorg-x11-font-utils	7.2	

套件	阿 AMI	阿 AMI
xxd	9.0.2120	9.0.2153
xxhash-libs		0.8.0
xz	5.2.2	5.2.5
xz-libs	5.2.2	5.2.5
yum	3.4.3	4.14.0
yum-metadata-parser	1.1.4	
yum-plugin-priorities	1.1.31	
yum-plugin-upgrade-helper	1.1.31	
yum-utils	1.1.31	
zip	3.0	3.0
zlib	1.2.8	1.2.11
zram-generator		1.1.2
zram-generator-defaults		1.1.2
zstd		1.5.5

比較 Amazon Linux 1 (AL1) 與 Amazon Linux 2023 最小 AMI 安裝的套件

目前在 AL1 和 AL2023 最小 AMI 上的 RPM 的比較。

套件	AL1 最小	AL2023 最小
acpid	2.0.19	
alternatives		1.15
amazon-chrony-config		4.3
amazon-ec2-net-utils		2.4.1
amazon-linux-repo-s3		2023.4.20240513
amazon-linux-sb-keys		2023.1
audit	2.6.5	3.0.6
audit-libs	2.6.5	3.0.6
authconfig	6.2.8	
awscli-2		2.15.30
basesystem	10.0	11
bash	4.2.46	5.2.15
binutils	2.27	
bzip2	1.0.6	
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023.2.62	2023.2.64
checkpolicy	2.1.10	3.4
chkconfig	1.3.49.3	
chrony		4.3
cloud-disk-utils	0.27	

套件	AL1 最小	AL2023 最小
cloud-init	0.7.6	22.2.2
cloud-init-cfg-ec2		22.2.2
cloud-utils-growpart		0.31
coreutils	8.22	8.32
coreutils-common		8.32
cpio	2.10	2.13
cracklib	2.8.16	2.9.6
cracklib-dicts	2.8.16	2.9.6
cronie	1.4.4	
cronie-anacron	1.4.4	
crontabs	1.10	
crypto-policies		20220428
cryptsetup-libs		2.6.1
curl	7.61.1	
curl-minimal		8.5.0
cyrus-sasl	2.1.23	
cyrus-sasl-lib	2.1.23	2.1.27
dash	0.5.5.1	
db4	4.7.25	
db4-utils	4.7.25	

套件	AL1 最小	AL2023 最小
dbus		1.12.28
dbus-broker		32
dbus-common		1.12.28
dbus-libs	1.6.12	1.12.28
device-mapper		1.02.185
device-mapper-libs		1.02.185
dhclient	4.1.1	
dhcp-common	4.1.1	
diffutils	3.3	3.8
dnf		4.14.0
dnf-data		4.14.0
dnf-plugin-release-notification		1.2
dnf-plugins-core		4.3.0
dnf-plugin-support-info		1.2
dracut	004	055
dracut-config-ec2		3.0
dracut-config-generic		055
dracut-modules-growroot	0.20	

套件	AL1 最小	AL2023 最小
e2fsprogs	1.43.5	1.46.5
e2fsprogs-libs	1.43.5	1.46.5
ec2-utils	0.7	2.2.0
ed	1.1	
efi-filesystem		5
efivar		38
efivar-libs		38
elfutils-default-yama-scope		0.188
elfutils-libelf	0.168	0.188
elfutils-libs		0.188
ethtool	3.15	
expat	2.1.0	2.5.0
file	5.37	5.39
file-libs	5.37	5.39
filesystem	2.4.30	3.14
findutils	4.4.2	4.8.0
fipscheck	1.3.1	
fipscheck-lib	1.3.1	
fuse-libs	2.9.4	2.9.9
gawk	3.1.7	5.1.0

套件	AL1 最小	AL2023 最小
gdbm	1.8.0	
gdbm-libs		1.19
gdisk	0.8.10	1.0.8
generic-logos	17.0.0	
get_reference_source	1.2	
gettext		0.21
gettext-libs		0.21
glib2	2.36.3	2.74.7
glibc	2.17	2.34
glibc-all-langpacks		2.34
glibc-common	2.17	2.34
glibc-locale-source		2.34
gmp	6.0.0	6.2.1
gnupg2	2.0.28	
gnupg2-minimal		2.3.7
gnutls		3.8.0
gpgme	1.4.3	1.15.1
grep	2.20	3.8
groff	1.22.2	
groff-base	1.22.2	1.22.4

套件	AL1 最小	AL2023 最小
grub	0.97	
grub2-common		2.06
grub2-efi-x64-ec2		2.06
grub2-pc-modules		2.06
grub2-tools		2.06
grub2-tools-minimal		2.06
grubby	7.0.15	8.40
gzip	1.5	1.12
hesiod	3.1.0	
hmacalc	0.9.12	
hostname		3.23
hwdata	0.233	0.353
info	5.1	
inih		49
initscripts	9.03.58	10.09
iproute	4.4.0	5.10.0
iptables	1.4.21	
iputils	20121221	20210202
irqbalance		1.9.0
jansson		2.14

套件	AL1 最小	AL2023 最小
jitterentropy		3.4.1
jq		1.7.1
json-c		0.14
kbd	1.15	2.4.0
kbd-misc	1.15	2.4.0
kernel	4.14.336	6.1.90
kernel-livepatch-r epo-s3		2023.4.20240513
keyutils-libs	1.5.8	1.6.3
kmod	14	29
kmod-libs	14	29
krb5-libs	1.15.1	1.21
less	436	608
libacl	2.2.49	2.3.1
libarchive		3.5.3
libargon2		20171227
libassuan	2.0.3	2.5.5
libattr	2.4.46	2.5.1
libblkid	2.23.2	2.37.4
libcap	2.16	2.48
libcap54	2.54	

套件	AL1 最小	AL2023 最小
libcap-ng	0.7.5	0.8.2
libcbor		0.7.0
libcgroup	0.40.rc1	
libcom_err	1.43.5	1.46.5
libcomps		0.1.20
libcurl	7.61.1	
libcurl-minimal		8.5.0
libdb		5.3.28
libdnf		0.69.0
libeconf		0.4.0
libedit	2.11	3.1
libfdisk		2.37.4
libffi	3.0.13	3.4.4
libfido2		1.10.0
libgcc		11.4.1
libgcc72	7.2.1	
libgcrypt	1.5.3	1.10.2
libgomp		11.4.1
libpgp-error	1.11	1.42
libicu	50.2	

套件	AL1 最小	AL2023 最小
libidn	1.18	
libidn2	2.3.0	2.3.2
libkcapi		1.4.0
libkcapi-hmaccalc		1.4.0
libmnl	1.0.3	1.0.4
libmodulemd		2.13.0
libmount	2.23.2	2.37.4
libnetfilter_contrack	1.0.4	
libnfnetlink	1.0.1	
libnghttp2	1.33.0	1.59.0
libnih	1.0.1	
libpipeline		1.5.3
libpsl	0.6.2	0.21.1
libpwquality	1.2.3	1.4.4
librepo		1.14.5
libreport-filessystem		2.15.2
libseccomp		2.5.3
libselinux	2.1.10	3.4
libselinux-utils	2.1.10	3.4
libsemanage	2.1.6	3.4

套件	AL1 最小	AL2023 最小
libsepol	2.1.7	3.4
libsigsegv		2.13
libsmartcols	2.23.2	2.37.4
libsolv		0.7.22
libss	1.43.5	1.46.5
libssh2	1.4.2	
libstdc++		11.4.1
libstdc++72	7.2.1	
libsysfs	2.1.0	
libtasn1	2.3	4.19.0
libtextstyle		0.21
libudev	173	
libunistring	0.9.3	0.9.10
libuser	0.60	0.63
libutempter	1.1.5	1.2.1
libuuid	2.23.2	2.37.4
libverto	0.2.5	0.3.2
libxcrypt		4.4.33
libxml2	2.9.1	2.10.4
libyam1	0.1.6	0.2.5

套件	AL1 最小	AL2023 最小
libzstd		1.5.5
logrotate	3.7.8	3.20.1
lua	5.1.4	
lua-libs		5.4.4
lz4-libs		1.9.4
make	3.82	
man-db		2.9.3
microcode_ctl	2.1	2.1
mingetty	1.08	
mpfr		4.1.0
ncurses	5.7	6.2
ncurses-base	5.7	6.2
ncurses-libs	5.7	6.2
nettle		3.8
net-tools	1.60	2.0
newt	0.52.11	
newt-python27	0.52.11	
npth		1.6
nspr	4.25.0	
nss	3.53.1	

套件	AL1 最小	AL2023 最小
nss-pem	1.0.3	
nss-softokn	3.53.1	
nss-softokn-freebl	3.53.1	
nss-sysinit	3.53.1	
nss-tools	3.53.1	
nss-util	3.53.1	
ntp	4.2.8p15	
ntpdate	4.2.8p15	
numactl-libs		2.0.14
oniguruma		6.9.7.1
openldap	2.4.40	2.4.57
openssh	七分一	8.7 小時
openssh-clients		8.7 小時
openssh-server	七分一	8.7 小時
openssl	1.0.2K	3.0.8
openssl-libs		3.0.8
openssl-pkcs11		0.4.12
os-prober		1.77
p11-kit	0.18.5	0.24.1
p11-kit-trust	0.18.5	0.24.1

套件	AL1 最小	AL2023 最小
pam	1.1.8	1.5.1
passwd	0.79	0.80
pciutils	3.1.10	3.7.0
pciutils-libs	3.1.10	3.7.0
pcre	8.21	
pcre2		10.40
pcre2-syntax		10.40
pinentry	0.7.6	
pkgconfig	0.27.1	
policycoreutils	2.1.12	3.4
popt	1.13	1.18
procmail	3.22	
procps	3.2.8	
procps-ng		3.3.17
psmisc	22.20	23.4
pth	2.0.7	
publicsuffix-list-dafsa		20240212
python27	2.7.18	
python27-babel	0.9.4	
python27-backports	1.0	

套件	AL1 最小	AL2023 最小
python27-backports-ssl_match_hostname	3.4.0.2	
python27-chardet	2.0.1	
python27-configobj	4.7.2	
python27-iniparse	0.3.1	
python27-jinja2	2.7.2	
python27-jsonpatch	1.2	
python27-jsonpointer	1.0	
python27-libs	2.7.18	
python27-markupsafe	0.11	
python27-pycurl	7.19.0	
python27-pygpme	0.3	
python27-pyliblzma	0.5.3	
python27-pyattr	0.5.0	
python27-PyYAML	3.10	
python27-requests	1.2.3	
python27-setuptools	36.2.7	
python27-six	1.8.0	
python27-urlgrabber	3.10	
python27-urllib3	1.24.3	
python3		3.9.16

套件	AL1 最小	AL2023 最小
python3-attrs		20.3.0
python3-audit		3.0.6
python3-awscrt		0.19.19
python3-babel		2.9.1
python3-cffi		1.14.5
python3-chardet		4.0.0
python3-colorama		0.4.4
python3-configobj		5.0.6
python3-cryptography		36.0.1
python3-dateutil		2.8.1
python3-dbus		1.2.18
python3-distro		1.5.0
python3-dnf		4.14.0
python3-dnf-plugins-core		4.3.0
python3-docutils		0.16
python3-gpg		1.15.1
python3-hawkey		0.69.0
python3-idna		2.10
python3-jinja2		2.11.3
python3-jmespath		0.10.0

套件	AL1 最小	AL2023 最小
python3-jsonpatch		1.21
python3-jsonpointer		2.0
python3-jjsonschema		3.2.0
python3-libcomps		0.1.20
python3-libdnf		0.69.0
python3-libs		3.9.16
python3-libselenium		3.4
python3-libsemanage		3.4
python3-markupsafe		1.1.1
python3-netifaces		0.10.6
python3-oauthlib		3.0.2
python3-pip-wheel		21.3.1
python3-ply		3.11
python3-policycore utils		3.4
python3-prettytable		0.7.2
python3-prompt-too lkit		3.0.24
python3-pycparser		2.20
python3-pyrsistent		0.17.3
python3-pyserial		3.4

套件	AL1 最小	AL2023 最小
python3-pysocks		1.7.1
python3-pytz		2022.7.1
python3-pyyaml		5.4.1
python3-requests		2.25.1
python3-rpm		4.16.1.3
python3-ruamel-yaml		0.16.6
python3-ruamel-yaml-clib		0.1.2
python3-setools		4.4.1
python3-setuptools		59.6.0
python3-setuptools-wheel		59.6.0
python3-six		1.15.0
python3-systemd		235
python3-urllib3		1.25.10
python3-wcwidth		0.2.5
readline	6.2	8.1
rng-tools		6.14
rootfiles	8.1	8.1
rpm	4.11.3	4.16.1.3
rpm-build-libs	4.11.3	4.16.1.3

套件	AL1 最小	AL2023 最小
rpm-libs	4.11.3	4.16.1.3
rpm-plugin-selinux		4.16.1.3
rpm-plugin-systemd-inhibit		4.16.1.3
rpm-python27	4.11.3	
rpm-sign-libs		4.16.1.3
rsyslog	5.8.10	
sbsigntools		0.9.4
sed	4.2.1	4.8
selinux-policy		37.22
selinux-policy-targeted		37.22
sendmail	8.14.4	
setserial	2.17	
setup	2.8.14	2.13.7
shadow-utils	4.1.4.2	4.9
shared-mime-info	1.1	
slang	2.2.1	
sqlite	3.7.17	
sqlite-libs		3.40.0
sudo	1.8.23	1.9.15

套件	AL1 最小	AL2023 最小
sysctl-defaults	1.0	1.0
sysfsutils	2.1.0	
systemd		252.16
systemd-libs		252.16
systemd-networkd		252.16
systemd-pam		252.16
systemd-resolved		252.16
systemd-udev		252.16
system-release	2018.03	2023.4.20240513
sysvinit	2.87	
tar	1.26	1.34
tcp_wrappers-libs	7.6	
tzdata	2023c	2024a
udev	173	
update-motd	1.0.1	2.2
upstart	0.6.5	
userspace-rcu		0.12.1
ustr	1.0.4	
util-linux	2.23.2	2.37.4
util-linux-core		2.37.4

套件	AL1 最小	AL2023 最小
vim-data	9.0.2120	9.0.2153
vim-minimal	9.0.2120	9.0.2153
which	2.19	2.21
xfspgrog		5.18.0
xz	5.2.2	5.2.5
xz-libs	5.2.2	5.2.5
yum	3.4.3	4.14.0
yum-metadata-parser	1.1.4	
yum-plugin-priorities	1.1.31	
yum-plugin-upgrade-helper	1.1.31	
zlib	1.2.8	1.2.11
zram-generator		1.1.2
zram-generator-defaults		1.1.2
zstd		1.5.5

比較 Amazon Linux 1 (AL1) 與 Amazon 2023 基本容器映像安裝的套件

AL1 和 AL2023 基本容器映像上存在的 RPM 的比較。

套件	AL1 貨櫃	AL2023 集裝箱
alternatives		1.15
amazon-linux-repo-cdn		2023.4.20240513
audit-libs		3.0.6
basesystem	10.0	11
bash	4.2.46	5.2.15
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023.2.62	2023.2.64
chkconfig	1.3.49.3	
coreutils	8.22	
coreutils-single		8.32
crypto-policies		20220428
curl	7.61.1	
curl-minimal		8.5.0
cyrus-sasl-lib	2.1.23	
db4	4.7.25	
db4-utils	4.7.25	
dnf		4.14.0
dnf-data		4.14.0
elfutils-default-yama-scope		0.188

套件	AL1 貨櫃	AL2023 集裝箱
elfutils-libelf	0.168	0.188
elfutils-libs		0.188
expat	2.1.0	2.5.0
file-libs	5.37	5.39
filesystem	2.4.30	3.14
gawk	3.1.7	5.1.0
gdbm	1.8.0	
gdbm-libs		1.19
glib2	2.36.3	2.74.7
glibc	2.17	2.34
glibc-common	2.17	2.34
glibc-minimal-lang pack		2.34
gmp	6.0.0	6.2.1
gnupg2	2.0.28	
gnupg2-minimal		2.3.7
gpgme	1.4.3	1.15.1
grep	2.20	3.8
gzip	1.5	
info	5.1	
json-c		0.14

套件	AL1 貨櫃	AL2023 集裝箱
keyutils-libs	1.5.8	1.6.3
krb5-libs	1.15.1	1.21
libacl	2.2.49	2.3.1
libarchive		3.5.3
libassuan	2.0.3	2.5.5
libattr	2.4.46	2.5.1
libblkid		2.37.4
libcap	2.16	2.48
libcap-ng		0.8.2
libcom_err	1.43.5	1.46.5
libcomps		0.1.20
libcurl	7.61.1	
libcurl-minimal		8.5.0
libdnf		0.69.0
libffi	3.0.13	3.4.4
libgcc		11.4.1
libgcc72	7.2.1	
libgcrypt	1.5.3	1.10.2
libgomp		11.4.1
libgpg-error	1.11	1.42

套件	AL1 貨櫃	AL2023 集裝箱
libc	50.2	
libidn2	2.3.0	2.3.2
libmodulemd		2.13.0
libmount		2.37.4
libnghttp2	1.33.0	1.59.0
libpsl	0.6.2	0.21.1
librepo		1.14.5
libreport-filessystem		2.15.2
libselinux	2.1.10	3.4
libsepol	2.1.7	3.4
libsigsegv		2.13
libsmartcols		2.37.4
libsolv		0.7.22
libssh2	1.4.2	
libstdc++		11.4.1
libstdc++72	7.2.1	
libtasn1	2.3	4.19.0
libunistring	0.9.3	0.9.10
libuuid		2.37.4
libverto	0.2.5	0.3.2

套件	AL1 貨櫃	AL2023 集裝箱
libxcrypt		4.4.33
libxml2	2.9.1	2.10.4
libxml2-python27	2.9.1	
libyaml		0.2.5
libzstd		1.5.5
lua	5.1.4	
lua-libs		5.4.4
lz4-libs		1.9.4
make	3.82	
mpfr		4.1.0
ncurses	5.7	
ncurses-base	5.7	6.2
ncurses-libs	5.7	6.2
npth		1.6
nspr	4.25.0	
nss	3.53.1	
nss-pem	1.0.3	
nss-softokn	3.53.1	
nss-softokn-freebl	3.53.1	
nss-sysinit	3.53.1	

套件	AL1 貨櫃	AL2023 集裝箱
nss-tools	3.53.1	
nss-util	3.53.1	
openldap	2.4.40	
openssl	1.0.2K	
openssl-libs		3.0.8
p11-kit	0.18.5	0.24.1
p11-kit-trust	0.18.5	0.24.1
pcre	8.21	
pcre2		10.40
pcre2-syntax		10.40
pinentry	0.7.6	
pkgconfig	0.27.1	
popt	1.13	1.18
pth	2.0.7	
publicsuffix-list-dafsa		20240212
python27	2.7.18	
python27-chardet	2.0.1	
python27-iniparse	0.3.1	
python27-kitchen	1.1.1	
python27-libs	2.7.18	

套件	AL1 貨櫃	AL2023 集裝箱
python27-pycurl	7.19.0	
python27-pygpme	0.3	
python27-pyliblzma	0.5.3	
python27-pyattr	0.5.0	
python27-urlgrabber	3.10	
python3		3.9.16
python3-dnf		4.14.0
python3-gpg		1.15.1
python3-hawkey		0.69.0
python3-libcomps		0.1.20
python3-libdnf		0.69.0
python3-libs		3.9.16
python3-pip-wheel		21.3.1
python3-rpm		4.16.1.3
python3-setuptools-wheel		59.6.0
readline	6.2	8.1
rpm	4.11.3	4.16.1.3
rpm-build-libs	4.11.3	4.16.1.3
rpm-libs	4.11.3	4.16.1.3
rpm-python27	4.11.3	

套件	AL1 貨櫃	AL2023 集裝箱
rpm-sign-libs		4.16.1.3
sed	4.2.1	4.8
setup	2.8.14	2.13.7
shared-mime-info	1.1	
sqlite	3.7.17	
sqlite-libs		3.40.0
sysctl-defaults	1.0	
system-release	2018.03	2023.4.20240513
tar	1.26	
tzdata	2023c	2024a
xz-libs	5.2.2	5.2.5
yum	3.4.3	4.14.0
yum-metadata-parser	1.1.4	
yum-plugin-ovl	1.1.31	
yum-plugin-priorities	1.1.31	
yum-utils	1.1.31	
zlib	1.2.8	1.2.11

AL2023 系統需求

本節說明使用 AL2023 的系統需求。

主題

- [執行 AL2023 的 CPU 需求](#)
- [執行 AL2023 的記憶體 \(記憶體\) 需求](#)

執行 AL2023 的 CPU 需求

若要執行任何 AL2023 程式碼，使用的處理器必須符合特定的最低需求。嘗試在不符合這些需求的 CPU 上執行 AL2023，可能會在程式碼執行的早期導致非法指令錯誤。

最低需求適用於[Amazon EC2 上的 AL2023 容器中的 AL2023](#)、和[Amazon EC2 之外的 AL2023](#)。

AL2023 的 ARM CPU 需求

所有 AL2023aarch64 (ARM) 二進製文件都是為 64 位構建的。沒有可用的 32 位元 ARM 二進位檔案，因此需要 64 位元 ARM CPU。

Note

對於 ARM 型執行個體，AL2023 僅支援使用 Graviton2 或更新版本處理器的執行個體類型。AL2023 不支援 A1 執行個體。

AL2023 需要符合 ARMv8.2 標準且具有密碼編譯延伸功能的處理器 (ARMv8.2+crypto)。的所有 AL2023 套件aarch64都是使用-march=armv8.2-a+crypto編譯器旗標建置的。雖然我們嘗試在較舊的ARM處理器上執行 AL2023 程式碼時，嘗試列印正確的錯誤訊息，但第一個錯誤訊息可能是非法的指令錯誤。

Note

由於 AL2023 的aarch64基本 CPU 需求，因此之前的所有Raspberry Pi系統都Raspberry Pi 5 不符合最低 CPU 需求。

AL2023 的 x86-64 CPU 需求

所有 AL2023 x86-64 二進製文件都是通過傳遞給編譯器 `-march=x86-64-v2` 來為體 x86-64 系結構的 x86-64v2 修訂而構建的。

該架構的 x86-64v2 修訂版在基準架 x86-64 構之上添加了以下 CPU 功能：

- CMPXCHG16B
- LAHF-SAHF
- POPCNT
- SSE3
- SSE4_1
- SSE4_2
- SSSE3

這大致映射到 2009 年或更高版本發布的 x86-64 處理器。範例包括 Intel Nehalem、AMD Jaguar、Atom Silvermont、以及 VIA Nano 及 Eden C 微架構。

在 Amazon EC2 中，所有 x86-64 執行個體類型都支援 x86-64v2，包括 M1、C1 和 M2 執行個體系列。

沒有建置 32 位元 x86 (i686) AL2023 二進位檔案。雖然 AL2023 保留對執行 32 位元使用者空間二進位檔的支援，但此功能已被取代，並且可能會在 future 的主要版本的 Amazon Linux 中移除。如需詳細資訊，請參閱 [32 位元 x86 \(i686\) 套件](#)。

執行 AL2023 的記憶體 (記憶體) 需求

Amazon EC2 執行個體類型 .nano 系列 (t2.nano、t3.nanot3a.nano、和 t4g.nano) 具有 512 MB 記憶體，這是 AL2023 的最低需求。

Note

雖然 512 MB 是最小的需求，但這些執行個體類型受到記憶體限制，而且功能和效能可能受到限制。

AL2023 映像檔尚未在記憶體少於 512 MB 的系統上進行測試。以少於 512 MB 的記憶體執行以 AL2023 為基礎的容器映像，將視容器化工作負載而定。

某些工作負載 (例如某些 AL2023 版本 `dnf update` 之間) 可能需要超過 512 MB 的記憶體。基於這個原因，[AL2023.3](#) 版本預設會針對記憶體 `zram` 體少於 800 MB 的執行個體啟用。對於容器化工作負載，這意味著某些工作負載在具有此數量記憶體的 AL2023 執行個體上可能會正常執行，但在受限於此記憶體使用量的容器中執行時會失敗。

對於 RAM 小於 800 MB 的執行個體類型，AL2023 (截至 [AL2023.3](#) 或更新版本) 預設啟用基於 `zram` 的交換。記憶體少於 800 MB 的 Amazon EC2 執行個體類型範例包括 `t4g`、`nanot3a`、`nanot3`、`nanot2`、`nano`、和 `t1.micro`。這表示這些執行個體類型的記憶體不足案例較少，因為 AL2023 會隨需壓縮和解壓縮記憶體分頁。如此就能使用之前需要具有更多記憶體的執行個體類型，且會犧牲執行壓縮所需 CPU 使用量的工作負載。

在上使用 AL2023 AWS

您可以將 AL2023 設定為與其他配合使用 AWS 服務。例如，您可以在啟動 [Amazon 彈性運算雲端 \(亞馬 Amazon EC2\)](#) 執行個體時選擇 AL2023 AMI。

對於這些設定程序，您可以使用 AWS Identity and Access Management (IAM) 服務。如需 IAM 的完整資訊，請參閱下列參考資料：

- [AWS Identity and Access Management \(IAM\)](#)
- [IAM 使用者指南](#)

主題

- [開始使用 AWS](#)
- [Amazon EC2 上的 AL2023](#)
- [在容器中使用 AL2023](#)
- [AL2023 開啟 AWS Elastic Beanstalk](#)
- [在中使用 AL2023 AWS CloudShell](#)
- [使用以 AL2023 為基礎的 Amazon ECS AMI 託管容器化工作負載](#)
- [在 AL2023 上使用 Amazon Elastic File System](#)
- [使用建立在 AL2023 上的 Amazon EMR](#)
- [在中使用 AL2023 AWS Lambda](#)

開始使用 AWS

註冊一個 AWS 帳戶

如果您沒有 AWS 帳戶，請完成以下步驟來建立一個。

若要註冊成為 AWS 帳戶

1. 開啟 <https://portal.aws.amazon.com/billing/signup>。
2. 請遵循線上指示進行。

部分註冊程序需接收來電，並在電話鍵盤輸入驗證碼。

當您註冊一個時 AWS 帳戶，將創建AWS 帳戶根使用者一個。根使用者有權存取該帳戶中的所有 AWS 服務 和資源。安全性最佳做法是將管理存取權指派給使用者，並僅使用 root 使用者來執行需要 root 使用者存取權的工作。

AWS 註冊過程完成後，會向您發送確認電子郵件。您可以隨時登錄 <https://aws.amazon.com/> 並選擇我的帳戶，以檢視您目前的帳戶活動並管理帳戶。

建立具有管理權限的使用者

註冊後，請保護 AWS 帳戶 AWS 帳戶根使用者、啟用和建立系統管理使用者 AWS IAM Identity Center，這樣您就不會將 root 使用者用於日常工作。

保護您的 AWS 帳戶根使用者

1. 選擇 Root 使用者並輸入您的 AWS 帳戶 電子郵件地址，以帳戶擁有者身分登入。[AWS Management Console](#)在下一頁中，輸入您的密碼。

如需使用根使用者登入的說明，請參閱 AWS 登入 使用者指南中的[以根使用者身分登入](#)。

2. 若要在您的根使用者帳戶上啟用多重要素驗證 (MFA)。

如需指示，請參閱《IAM 使用者指南》中的[為 AWS 帳戶 根使用者啟用虛擬 MFA 裝置 \(主控台\)](#)。

建立具有管理權限的使用者

1. 啟用 IAM Identity Center。

如需指示，請參閱 AWS IAM Identity Center 使用者指南中的[啟用 AWS IAM Identity Center](#)。

2. 在 IAM 身分中心中，將管理存取權授予使用者。

[若要取得有關使用 IAM Identity Center 目錄 做為身分識別來源的自學課程，請參閱《使用指南》IAM Identity Center 目錄中的「以預設值設定使用AWS IAM Identity Center 者存取」。](#)

以具有管理權限的使用者身分登入

- 若要使用您的 IAM Identity Center 使用者簽署，請使用建立 IAM Identity Center 使用者時傳送至您電子郵件地址的簽署 URL。

如需使用 IAM 身分中心使用者 [登入的說明](#)，請參閱 [使用AWS 登入者指南中的登入 AWS 存取入口網站](#)。

指派存取權給其他使用者

1. 在 IAM 身分中心中，建立遵循套用最低權限許可的最佳做法的權限集。

如需指示，請參閱《AWS IAM Identity Center 使用指南》中的「[建立權限集](#)」。

2. 將使用者指派給群組，然後將單一登入存取權指派給群組。

如需指示，請參閱《AWS IAM Identity Center 使用指南》中的「[新增群組](#)」。

授予程式設計存取權

如果使用者想要與 AWS 之外的 AWS Management Console 授與程式設計存取權的方式取決於正在存取的使用者類型。

若要授與使用者程式設計存取權，請選擇下列其中一個選項。

哪個使用者需要程式設計存取權？	到	By
人力身分 (IAM Identity Center 中管理的使用者)	使用臨時登入資料來簽署對 AWS CLI、AWS SDK 或 AWS API 的程式設計要求。	<p>請依照您要使用的介面所提供的指示操作。</p> <ul style="list-style-type: none"> • 如需詳細資訊 AWS CLI，請參閱 《使 AWS CLI 用AWS Command Line Interface者指南》 AWS IAM Identity Center 中的〈配置使用〉。 • 如需 AWS SDK、工具和 AWS API，請參閱 AWS SDK 和工具參考指南中的 IAM 身分中心身分驗證。

哪個使用者需要程式設計存取權？	到	By
IAM	使用臨時登入資料來簽署對 AWS CLI、AWS SDK 或 AWS API 的程式設計要求。	遵循《IAM 使用者指南 》中的 〈將臨時登入資料搭配 AWS 資源使用〉 中的指示
IAM	(不建議使用) 使用長期認證簽署對 AWS CLI、AWS SDK 或 AWS API 的程式設計要求。	請依照您要使用的介面所提供的指示操作。 <ul style="list-style-type: none"> • 如需相關資訊 AWS CLI，請參閱使用指南中的使用 IAM 使用者登入資料進行驗證。AWS Command Line Interface • 對於 AWS SDK 和工具，請參閱 AWS SDK 和工具參考指南中的使用長期憑據進行身份驗證。 • 如需 AWS API，請參閱 IAM 使用者指南中的管理 IAM 使用者的存取金鑰。

Amazon EC2 上的 AL2023

使用下列其中一個程序來啟動具有 AL2023 AMI 的 Amazon EC2 執行個體。您可以選擇標準 AMI 或最小 AMI。如需標準 AMI 和最小 AMI 之間差異的詳細資訊，請參閱[比較 AL2023 標準 \(預設\) 與最小 AMI](#)。

主題

- [使用亞馬遜主控台啟動 AL2023](#)
- [使用超音波馬達參數啟動 AL2023 AWS CLI](#)
- [使用推出最新的 AL2023 AMI AWS CloudFormation](#)
- [使用特定的 AMI 識別碼啟動 AL2023](#)
- [AL2023 AMI 棄用和生命週期](#)

- [連線至 AL2023 執行個體](#)
- [比較 AL2023 標準和最小的 AMI](#)

使用亞馬遜主控台啟動 AL2023

使用 Amazon EC2 主控台來啟動 AL2023 AMI。

Note

對於 ARM 型執行個體，AL2023 僅支援使用 Graviton2 或更新版本處理器的執行個體類型。AL2023 不支援 A1 執行個體。

請遵循下列步驟，以從 Amazon EC2 主控台並使用 AL2023 AMI 啟動 Amazon EC2 執行個體。

若要使用 AL2023 AMI 啟動 EC2 執行個體

1. 在 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選擇 AMIs (AMI)。
3. 從下拉式功能表中選擇 Public images (公有映像)。
4. 在搜尋欄位中輸入 **al2023-ami**。

Note

確保 Amazon 出現在 Owner alias (所有者別名) 欄中。

5. 從清單選取映像。在 Source (來源) 下，您可以判斷 AMI 是標準或最小。AL2023 AMI 名稱可使用以下格式來解釋：

```
'al2023-[ami || ami-minimal]-2023.0.[release build date].[build number]-kernel-[version number]-[arm64 || x86_64]'
```

6. 下圖顯示 AL2023 AMI 的部分清單。

<input type="checkbox"/>	Name	AMI ID	AMI name	Source	Owner	Owner alias
<input type="checkbox"/>	-	ami-000a4d9c6067d5d0d	al2023-ami-2023.0.20230222.1...	amazon/al2023-ami-2023.0.20230222.1-kernel-6.1-arm64	137112412989	amazon
<input type="checkbox"/>	-	ami-0a409f3927bd2662f	al2023-ami-2023.0.20230222.1...	amazon/al2023-ami-2023.0.20230222.1-kernel-6.1-x86_64	137112412989	amazon
<input type="checkbox"/>	-	ami-043e11d11db3d437e	al2023-ami-minimal-2023.0.20...	amazon/al2023-ami-minimal-2023.0.20230222.1-kernel-6.1-ar...	137112412989	amazon
<input type="checkbox"/>	-	ami-0d19aa82c9a61ef2c	al2023-ami-minimal-2023.0.20...	amazon/al2023-ami-minimal-2023.0.20230222.1-kernel-6.1-x8...	137112412989	amazon

如需有關啟動 Amazon EC2 執行個體的詳細資訊，請參閱 [Amazon EC2 使用者指南中的開始使用 Amazon EC2 Linux 執行個體](#)。

使用超音波馬達參數啟動 AL2023 AWS CLI

在中 AWS CLI，您可以使用 AMI 的 SSM 參數值來啟動新的 AL2023 執行個體。更具體來說，請使用下列清單的其中一個動態 SSM 參數值，並在 SSM 參數值/ 前面加 `/aws/service/ami-amazon-linux-latest/`。您可以在 AWS CLI 中使用此方式來啟動執行個體。

- `al2023-ami-kernel-default-arm64` 適用於 arm64 架構
- `al2023-ami-minimal-kernel-default-arm64` 適用於 arm64 架構 (最小 AMI)
- `al2023-ami-kernel-default-x86_64` 適用於 x86_64 架構
- `al2023-ami-minimal-kernel-default-x86_64` 適用於 x86_64 架構 (最小 AMI)

Note

每個 `##` 項目皆為範例參數。請以您自己的資訊取代。

```
$ aws ec2 run-instances \  
  --image-id \  
    resolve:ssm:/aws/service/ami-amazon-linux-latest/al2023-ami-kernel-default-x86_64 \  
  --instance-type m5.xlarge \  
  --region us-east-1 \  
  --key-name aws-key-us-east-1 \  
  --security-group-ids sg-004a7650
```

`--image-id` 旗標指定 SSM 參數值。

`--instance-type` 旗標指定執行個體的類型和大小。此旗標必須與您選取的 AMI 類型相容。

此 `--region` 旗標會指定 AWS 區域 您建立執行個體的位置。

此 `--key-name` 旗標會 AWS 區域指定用來連線至執行個體的金鑰。如果您沒有提供位於建立執行個體所在區域的金鑰，就無法使用 SSH 連線至執行個體。

`--security-group-ids` 旗標指定確定輸入和輸出網路流量存取權限的安全群組。

⚠ Important

您 AWS CLI 必須指定現有的安全群組，以允許透過連接埠從遠端機器存取執行個體TCP:22。如果沒有指定的安全群組，新執行個體會放置在預設安全群組中。在預設安全群組中，您的執行個體只能與 VPC 中的其他執行個體連線。

如需詳細資訊，請參閱《AWS Command Line Interface 使用者指南》中的[啟動、列出及終止 Amazon EC2 執行個體](#)。

使用推出最新的 AL2023 AMI AWS CloudFormation

若要使用啟動 AL2023 AMI AWS CloudFormation，請使用下列其中一個範本。

📌 Note

x86_64 和 Arm64 AMI 各自需要不同的執行個體類型。如需詳細資訊，請參閱 [Amazon EC2 執行個體類型](#)

JSON 範本：

```
{
  "Parameters": {
    "LatestAmiId": {
      "Type": "AWS::SSM::Parameter::Value<AWS::EC2::Image::Id>",
      "Default": "/aws/service/ami-amazon-linux-latest/al2023-ami-minimal-kernel-
default-x86_64"
    }
  },
  "Resources": {
    "MyEC2Instance": {
      "Type": "AWS::EC2::Instance",
      "Properties": {
        "InstanceType": "t2.large",
        "ImageId": {
          "Ref": "LatestAmiId"
        }
      }
    }
  }
}
```

```
}
```

YAML 範本：

```
Parameters:
  LatestAmiId:
    Type: 'AWS::SSM::Parameter::Value<AWS::EC2::Image::Id>'
    Default: '/aws/service/ami-amazon-linux-latest/al2023-ami-minimal-kernel-default-x86_64'

Resources:
  Instance:
    Type: 'AWS::EC2::Instance'
    Properties:
      InstanceType: 't2.large'
      ImageId: !Ref LatestAmiId
```

如有需要，請務必替換「預設」部分末尾的 AMI 參數。以下參數值可用於：

- al2023-ami-kernel-6.1-arm64 適用於 arm64 架構
- al2023-ami-minimal-kernel-6.1-arm64 適用於 arm64 架構 (最小 AMI)
- al2023-ami-kernel-6.1-x86_64 適用於 x86_64 架構
- al2023-ami-minimal-kernel-6.1-x86_64 適用於 x86_64 架構 (最小 AMI)

以下是動態核心規格。預設核心版本會隨著每次重大核心版本更新而自動變更。

- al2023-ami-kernel-default-arm64 適用於 arm64 架構
- al2023-ami-minimal-kernel-default-arm64 適用於 arm64 架構 (最小 AMI)
- al2023-ami-kernel-default-x86_64 適用於 x86_64 架構
- al2023-ami-minimal-kernel-default-x86_64 適用於 x86_64 架構 (最小 AMI)

使用特定的 AMI 識別碼啟動 AL2023

您可以使用 AMI ID 啟動特定的 AL2023 AMI。您可以透過查看 Amazon EC2 控制台中的 AMI 列表，來確定需要哪個 AL2023 AMI ID。或者，您可以使用 AWS Systems Manager。如果您使用系統管理員，請務必從上一節列出的 AMI 別名中選取 AMI 別名。如需詳細資訊，請參閱[使用 AWS Systems Manager 參數存放區查詢最新的 Amazon Linux AMI ID](#)。

AL2023 AMI 棄用和生命週期

每個新的 AL2023 發布都含有新的 AMI。當 AMI 註冊時，會標記棄用日期。每個 AL2023 AMI 的棄用日期都是自發行日起 90 天，以符合針對每個核心版本提供的 [AL2023 上的核心即時修補程式](#) 時間段。

Note

90 天棄用日期是指個別 AMI，並不是指 AL2023 [發行節奏](#) 或產品支援期限。

如需 AMI 淘汰的詳細資訊，請參閱 Amazon EC2 使用者指南中的 [淘汰 AMI](#)。

定期使用更新的 AMI 啟動執行個體，以確保執行個體使用最新的安全性更新 (包括更新的核心) 啟動。如果您啟動舊版 AMI 並套用更新，執行個體會有一段時間沒有最新的安全性更新。為確保您使用最新的 AMI，建議使用 SSM 參數。

如需使用 SSM 參數啟動執行個體的詳細資訊，請參閱：

- [使用超音波馬達參數啟動 AL2023 AWS CLI](#)
- [使用推出最新的 AL2023 AMI AWS CloudFormation](#)

連線至 AL2023 執行個體

使用安全殼層或連線 AWS Systems Manager 至您的 AL2023 執行個體。

使用 SSH 連線至您的執行個體

如需如何使用 SSH Connect 至執行個體的指示，請參閱 Amazon EC2 使用者指南中的 [使用安全殼層連線到 Linux 執行個體](#)。

使用 Connect 至您的執行個體 AWS Systems Manager

如需如何使用 Connect AWS Systems Manager 至 AL2023 執行個體的指示，請參閱 Amazon EC2 使用者指南中的 [使用工作階段管理員連線到 Linux 執行個體](#)。

使用 Amazon EC2 實例 Connect

AL2023 AMI (不包括最小 AMI) 會隨附預設安裝 EC2 執行個體 Connect 代理程式。若要將 EC2 執行個體 Connect 與從最小 AMI 啟動的 AL2023 執行個體搭配使用，您必須安裝 `ec2-instance-connect` 套件。如需使用 EC2 執行個體 Connect 的 [相關說明](#)，請參閱 [Amazon EC2 使用者指南中的使用 EC2 執行個體 Connect 到 Linux 執行個體](#)。

比較 AL2023 標準和最小的 AMI

您可以使用標準 (預設) 或最小 AL2023 AMI 來啟動 Amazon EC2 執行個體。如需如何使用標準或最小 AMI 類型啟動 Amazon EC2 執行個體的指示，請參閱[Amazon EC2 上的 AL2023](#)。

標準的 AL2023 AMI 隨附了所有最常用的應用程序和安裝工具。如果您想快速開始並且對自訂 AMI 不感興趣，建議使用標準 AMI。

最小的 AL2023 AMI 是簡化的基本版本，其中僅包含執行作業系統 (OS) 所需的最基本工具和公用程式。如果您希望將作業系統的使用空間降到最低，建議您使用最小 AMI。最小 AMI 可稍微降低磁碟空間使用率，並提高長期成本效益。如果您想要更小的作業系統，且不介意手動安裝工具和應用程序，則適合使用最小 AMI。

容器映像更接近套件組中的 AL2023 最小 AMI。

比較 Amazon Linux 2023 映像安裝的套件

AL2023 AMI，最小 AMI 和容器映像上存在的 RPM 的比較。

套件	AMI	最小 AMI	容器
acl	2.3.1		
acpid	2.0.32		
alternatives	1.15	1.15	1.15
amazon-chrony-config	4.3	4.3	
amazon-ec2-net-utils	2.4.1	2.4.1	
amazon-linux-repo-cdn			2023.4.20240513
amazon-linux-repo-s3	2023.4.20240513	2023.4.20240513	
amazon-linux-sb-keys	2023.1	2023.1	

套件	AMI	最小 AMI	容器
amazon-rpm-config	228		
amazon-ssm-agent	3.3.380.0		
at	3.1.23		
attr	2.5.1		
audit	3.0.6	3.0.6	
audit-libs	3.0.6	3.0.6	3.0.6
aws-cfn-bootstrap	2.0		
awscli-2	2.15.30	2.15.30	
basesystem	11	11	11
bash	5.2.15	5.2.15	5.2.15
bash-completion	2.11		
bc	1.07.1		
bind-libs	9.16.48		
bind-license	9.16.48		
bind-utils	9.16.48		
binutils	2.39		
boost-filesystem	1.75.0		
boost-system	1.75.0		

套件	AMI	最小 AMI	容器
boost-thread	1.75.0		
bzip2	1.0.8		
bzip2-libs	1.0.8	1.0.8	1.0.8
ca-certificates	2023.2.64	2023.2.64	2023.2.64
c-ares	1.19.0		
checkpolicy	3.4	3.4	
chkconfig	1.15		
chrony	4.3	4.3	
cloud-init	22.2.2	22.2.2	
cloud-init-cfg-ec2	22.2.2	22.2.2	
cloud-utils-growpart	0.31	0.31	
coreutils	8.32	8.32	
coreutils-common	8.32	8.32	
coreutils-single			8.32
cpio	2.13	2.13	
cracklib	2.9.6	2.9.6	
cracklib-dicts	2.9.6	2.9.6	
crontabs	1.11		

套件	AMI	最小 AMI	容器
crypto-policies	20220428	20220428	20220428
crypto-policies-scripts	20220428		
cryptsetup	2.6.1		
cryptsetup-libs	2.6.1	2.6.1	
curl-minimal	8.5.0	8.5.0	8.5.0
cyrus-sasl-lib	2.1.27	2.1.27	
cyrus-sasl-plain	2.1.27		
dbus	1.12.28	1.12.28	
dbus-broker	32	32	
dbus-common	1.12.28	1.12.28	
dbus-libs	1.12.28	1.12.28	
device-mapper	1.02.185	1.02.185	
device-mapper-libs	1.02.185	1.02.185	
diffutils	3.8	3.8	
dnf	4.14.0	4.14.0	4.14.0
dnf-data	4.14.0	4.14.0	4.14.0
dnf-plugin-release-notification	1.2	1.2	

套件	AMI	最小 AMI	容器
dnf-plugins-core	4.3.0	4.3.0	
dnf-plugin-support-info	1.2	1.2	
dnf-utils	4.3.0		
dosfstools	4.2		
dracut	055	055	
dracut-config-ec2	3.0	3.0	
dracut-config-generic	055	055	
dwz	0.14		
dyninst	10.2.1		
e2fsprogs	1.46.5	1.46.5	
e2fsprogs-libs	1.46.5	1.46.5	
ec2-hibinit-agent	1.0.8		
ec2-instance-connect	1.1		
ec2-instance-connect-selinux	1.1		
ec2-utils	2.2.0	2.2.0	

套件	AMI	最小 AMI	容器
ed	1.14.2		
efi-filesystem	5	5	
efi-srpm-macros	5		
efivar	38	38	
efivar-libs	38	38	
elfutils- debuginfod- client	0.188		
elfutils- default-yama- scope	0.188	0.188	0.188
elfutils-libelf	0.188	0.188	0.188
elfutils-libs	0.188	0.188	0.188
ethtool	5.15		
expat	2.5.0	2.5.0	2.5.0
file	5.39	5.39	
file-libs	5.39	5.39	5.39
filesystem	3.14	3.14	3.14
findutils	4.8.0	4.8.0	
fonts-srpm- macros	2.0.5		
fstrm	0.6.1		

套件	AMI	最小 AMI	容器
fuse-libs	2.9.9	2.9.9	
gawk	5.1.0	5.1.0	5.1.0
gdbm-libs	1.19	1.19	1.19
gdisk	1.0.8	1.0.8	
gettext	0.21	0.21	
gettext-libs	0.21	0.21	
ghc-srpm-macros	1.5.0		
glib2	2.74.7	2.74.7	2.74.7
glibc	2.34	2.34	2.34
glibc-all-langpacks	2.34	2.34	
glibc-common	2.34	2.34	2.34
glibc-gconv-extra	2.34		
glibc-locale-source	2.34	2.34	
glibc-minimal-langpack			2.34
gmp	6.2.1	6.2.1	6.2.1
gnupg2-minimal	2.3.7	2.3.7	2.3.7
gnutls	3.8.0	3.8.0	
go-srpm-macros	3.2.0		

套件	AMI	最小 AMI	容器
gpgme	1.15.1	1.15.1	1.15.1
gpm-libs	1.20.7		
grep	3.8	3.8	3.8
groff-base	1.22.4	1.22.4	
grub2-common	2.06	2.06	
grub2-efi-aa64-ec2	2.06 (一個建築 64)	2.06 (一個建築 64)	
grub2-efi-x64-ec2	2.06 (十六)	2.06 (十六)	
grub2-pc-modules	2.06	2.06	
grub2-tools	2.06	2.06	
grub2-tools-minimal	2.06	2.06	
grubby	8.40	8.40	
gssproxy	0.8.4		
gzip	1.12	1.12	
hostname	3.23	3.23	
hunspell	1.7.0		
hunspell-en	0.20140811.1		
hunspell-en-GB	0.20140811.1		
hunspell-en-US	0.20140811.1		

套件	AMI	最小 AMI	容器
hunspell-filesystem	1.7.0		
hwdata	0.353	0.353	
info	6.7		
inih	49	49	
initscripts	10.09	10.09	
iproute	5.10.0	5.10.0	
iputils	20210202	20210202	
irqbalance	1.9.0	1.9.0	
jansson	2.14	2.14	
jitterentropy	3.4.1	3.4.1	
jq	1.7.1	1.7.1	
json-c	0.14	0.14	0.14
kbd	2.4.0	2.4.0	
kbd-misc	2.4.0	2.4.0	
kernel	6.1.90	6.1.90	
kernel- li vepatch-repo- s3	2023.4.20240513	2023.4.20240513	
kernel-srpm- macros	1.0		
kernel-tools	6.1.90		

套件	AMI	最小 AMI	容器
keyutils	1.6.3		
keyutils-libs	1.6.3	1.6.3	1.6.3
kmod	29	29	
kmod-libs	29	29	
kpatch-runtime	0.9.7		
krb5-libs	1.21	1.21	1.21
less	608	608	
libacl	2.3.1	2.3.1	2.3.1
libaio	0.3.111		
libarchive	3.5.3	3.5.3	3.5.3
libargon2	20171227	20171227	
libassuan	2.5.5	2.5.5	2.5.5
libattr	2.5.1	2.5.1	2.5.1
libbasicobjects	0.1.1		
libblkid	2.37.4	2.37.4	2.37.4
libcap	2.48	2.48	2.48
libcap-ng	0.8.2	0.8.2	0.8.2
libcbor	0.7.0	0.7.0	
libcollection	0.7.0		
libcom_err	1.46.5	1.46.5	1.46.5

套件	AMI	最小 AMI	容器
libcomps	0.1.20	0.1.20	0.1.20
libconfig	1.7.2		
libcurl-minimal	8.5.0	8.5.0	8.5.0
libdb	5.3.28	5.3.28	
libdhash	0.5.0		
libdnf	0.69.0	0.69.0	0.69.0
libeconf	0.4.0	0.4.0	
libedit	3.1	3.1	
libev	4.33		
libevent	2.1.12		
libfdisk	2.37.4	2.37.4	
libffi	3.4.4	3.4.4	3.4.4
libfido2	1.10.0	1.10.0	
libgcc	11.4.1	11.4.1	11.4.1
libgcrypt	1.10.2	1.10.2	1.10.2
libgomp	11.4.1	11.4.1	11.4.1
libgpg-error	1.42	1.42	1.42
libibverbs	48.0		
libidn2	2.3.2	2.3.2	2.3.2
libini_config	1.3.1		

套件	AMI	最小 AMI	容器
libkcapi	1.4.0	1.4.0	
libkcapi-hmaccalc	1.4.0	1.4.0	
libldb	2.6.2		
libmaxinddb	1.5.2		
libmetalink	0.1.3		
libmnl	1.0.4	1.0.4	
libmodulemd	2.13.0	2.13.0	2.13.0
libmount	2.37.4	2.37.4	2.37.4
libnfsidmap	2.5.4		
libnghttp2	1.59.0	1.59.0	1.59.0
libnl3	3.5.0		
libpath_utils	0.2.1		
libpcap	1.10.1		
libpipeline	1.5.3	1.5.3	
libpkgconf	1.8.0		
libpsl	0.21.1	0.21.1	0.21.1
libpwquality	1.4.4	1.4.4	
libref_array	0.1.5		
librepo	1.14.5	1.14.5	1.14.5

套件	AMI	最小 AMI	容器
libreport-filesystem	2.15.2	2.15.2	2.15.2
libseccomp	2.5.3	2.5.3	
libselinux	3.4	3.4	3.4
libselinux-utils	3.4	3.4	
libsemanage	3.4	3.4	
libsepol	3.4	3.4	3.4
libsigsegv	2.13	2.13	2.13
libsmartcols	2.37.4	2.37.4	2.37.4
libsolv	0.7.22	0.7.22	0.7.22
libss	1.46.5	1.46.5	
libsss_certmap	2.9.4		
libsss_idmap	2.9.4		
libsss_nss_idmap	2.9.4		
libsss_sudo	2.9.4		
libstdc++	11.4.1	11.4.1	11.4.1
libstoragemgmt	1.9.4		
libtalloc	2.3.4		
libtasn1	4.19.0	4.19.0	4.19.0

套件	AMI	最小 AMI	容器
libtdb	1.4.7		
libtevent	0.13.0		
libtextstyle	0.21	0.21	
libtirpc	1.3.3		
libunistring	0.9.10	0.9.10	0.9.10
libuser	0.63	0.63	
libutempter	1.2.1	1.2.1	
libuuid	2.37.4	2.37.4	2.37.4
libuv	1.47.0		
libverto	0.3.2	0.3.2	0.3.2
libverto-libev	0.3.2		
libxcrypt	4.4.33	4.4.33	4.4.33
libxml2	2.10.4	2.10.4	2.10.4
libyaml	0.2.5	0.2.5	0.2.5
libzstd	1.5.5	1.5.5	1.5.5
lm_sensors-libs	3.6.0		
lmdb-libs	0.9.29		
logrotate	3.20.1	3.20.1	
lsof	4.94.0		
lua-libs	5.4.4	5.4.4	5.4.4

套件	AMI	最小 AMI	容器
lua-srpm-macros	1		
lz4-libs	1.9.4	1.9.4	1.9.4
man-db	2.9.3	2.9.3	
man-pages	5.10		
microcode_ctl	2.1 (六十四)	2.1 (六十四)	
mpfr	4.1.0	4.1.0	4.1.0
nano	5.8		
ncurses	6.2	6.2	
ncurses-base	6.2	6.2	6.2
ncurses-libs	6.2	6.2	6.2
nettle	3.8	3.8	
net-tools	2.0	2.0	
newt	0.52.21		
nfs-utils	2.5.4		
npth	1.6	1.6	1.6
nspr	4.35.0		
nss	3.90.0		
nss-softokn	3.90.0		
nss-softokn-freebl	3.90.0		
nss-sysinit	3.90.0		

套件	AMI	最小 AMI	容器
nss-util	3.90.0		
ntsysv	1.15		
numactl-libs	2.0.14	2.0.14	
ocaml-srpm-macros	6		
oniguruma	6.9.7.1	6.9.7.1	
openblas-srpm-macros	2		
openldap	2.4.57	2.4.57	
openssh	8.7 小時	8.7 小時	
openssh-clients	8.7 小時	8.7 小時	
openssh-server	8.7 小時	8.7 小時	
openssl	3.0.8	3.0.8	
openssl-libs	3.0.8	3.0.8	3.0.8
openssl-pkcs11	0.4.12	0.4.12	
os-prober	1.77	1.77	
p11-kit	0.24.1	0.24.1	0.24.1
p11-kit-trust	0.24.1	0.24.1	0.24.1
package-notes-srpm-macros	0.4		
pam	1.5.1	1.5.1	

套件	AMI	最小 AMI	容器
parted	3.4		
passwd	0.80	0.80	
pciutils	3.7.0	3.7.0	
pciutils-libs	3.7.0	3.7.0	
pcre2	10.40	10.40	10.40
pcre2-syntax	10.40	10.40	10.40
perl-Carp	1.50		
perl-Class-Struct	0.66		
perl-constant	1.33		
perl-DynaLoader	1.47		
perl-Encode	3.15		
perl-Errno	1.30		
perl-Exporter	5.74		
perl-Fcntl	1.13		
perl-File-Basename	2.85		
perl-File-Path	2.18		
perl-File-stat	1.09		
perl-File-Temp	0.231.100		

套件	AMI	最小 AMI	容器
perl-Getopt-Long	2.52		
perl-Getopt-Std	1.12		
perl-HTTP-Tiny	0.078		
perl-if	0.60.800		
perl-inte rpreter	5.32.1		
perl-IO	1.43		
perl-IPC-Open3	1.21		
perl-libs	5.32.1		
perl-MIME-Base64	3.16		
perl-mro	1.23		
perl-overload	1.31		
perl-overloading	0.02		
perl-parent	0.238		
perl-PathTools	3.78		
perl-Pod-Escapes	1.07		
perl-podlators	4.14		
perl-Pod-Perldoc	3.28.01		

套件	AMI	最小 AMI	容器
perl-Pod-Simple	3.42		
perl-Pod-Usage	2.01		
perl-POSIX	1.94		
perl-Scalar-List-Utils	1.56		
perl-SelectSaver	1.02		
perl-Socket	2.032		
perl-srpm-macros	1		
perl-Storable	3.21		
perl-subst	1.03		
perl-Symbol	1.08		
perl-Term-ANSIColor	5.01		
perl-Term-Cap	1.17		
perl-Text-ParseWords	3.30		
perl-Text-Tabs+Wrap	2021.0726		
perl-Time-Local	1.300		
perl-vars	1.05		
pkgconf	1.8.0		

套件	AMI	最小 AMI	容器
pkgconf-m4	1.8.0		
pkgconf-pkg-config	1.8.0		
policycoreutils	3.4	3.4	
policycoreutils-python-utils	3.4		
popt	1.18	1.18	1.18
procps-ng	3.3.17	3.3.17	
protobuf-c	1.4.1		
psacct	6.6.4		
psmisc	23.4	23.4	
publicsuffix-list-dafsa	20240212	20240212	20240212
python3	3.9.16	3.9.16	3.9.16
python3-attrs	20.3.0	20.3.0	
python3-audit	3.0.6	3.0.6	
python3-awscli	0.19.19	0.19.19	
python3-babel	2.9.1	2.9.1	
python3-cffi	1.14.5	1.14.5	
python3-chardet	4.0.0	4.0.0	

套件	AMI	最小 AMI	容器
python3-colorama	0.4.4	0.4.4	
python3-onfigobj	5.0.6	5.0.6	
python3-cryptography	36.0.1	36.0.1	
python3-daemon	2.3.0		
python3-dateutil	2.8.1	2.8.1	
python3-dbus	1.2.18	1.2.18	
python3-distro	1.5.0	1.5.0	
python3-dnf	4.14.0	4.14.0	4.14.0
python3-dnf-plugins-core	4.3.0	4.3.0	
python3-docutils	0.16	0.16	
python3-gpg	1.15.1	1.15.1	1.15.1
python3-hawkey	0.69.0	0.69.0	0.69.0
python3-idna	2.10	2.10	
python3-jinja2	2.11.3	2.11.3	
python3-jmespath	0.10.0	0.10.0	

套件	AMI	最小 AMI	容器
python3-j sonpatch	1.21	1.21	
python3-j sonpointer	2.0	2.0	
python3-j sonschema	3.2.0	3.2.0	
python3-l ibcomps	0.1.20	0.1.20	0.1.20
python3-libdnf	0.69.0	0.69.0	0.69.0
python3-libs	3.9.16	3.9.16	3.9.16
python3-l ibselinux	3.4	3.4	
python3-l ibsemanage	3.4	3.4	
python3-l ibstoragemgmt	1.9.4		
python3-l ockfile	0.12.2		
python3-m arkupsafe	1.1.1	1.1.1	
python3-n etifaces	0.10.6	0.10.6	
python3-o authlib	3.0.2	3.0.2	

套件	AMI	最小 AMI	容器
python3-pip-wheel	21.3.1	21.3.1	21.3.1
python3-ply	3.11	3.11	
python3-policycoreutils	3.4	3.4	
python3-prettytable	0.7.2	0.7.2	
python3-prompt-toolkit	3.0.24	3.0.24	
python3-pyparser	2.20	2.20	
python3-pyrsistent	0.17.3	0.17.3	
python3-pyserial	3.4	3.4	
python3-pysocks	1.7.1	1.7.1	
python3-pytz	2022.7.1	2022.7.1	
python3-pyyaml	5.4.1	5.4.1	
python3-requests	2.25.1	2.25.1	
python3-rpm	4.16.1.3	4.16.1.3	4.16.1.3
python3-ruamel-yaml	0.16.6	0.16.6	

套件	AMI	最小 AMI	容器
python3-ruamel-yaml- clib	0.1.2	0.1.2	
python3-setools	4.4.1	4.4.1	
python3-s etuptools	59.6.0	59.6.0	
python3-s etuptools- wheel	59.6.0	59.6.0	59.6.0
python3-six	1.15.0	1.15.0	
python3-systemd	235	235	
python3-urllib3	1.25.10	1.25.10	
python3-wcwidth	0.2.5	0.2.5	
python-chevron	0.13.1		
python-srpm- macros	3.9		
quota	4.06		
quota-nls	4.06		
readline	8.1	8.1	8.1
rng-tools	6.14	6.14	
rootfiles	8.1	8.1	
rpcbind	1.2.6		
rpm	4.16.1.3	4.16.1.3	4.16.1.3

套件	AMI	最小 AMI	容器
rpm-build-libs	4.16.1.3	4.16.1.3	4.16.1.3
rpm-libs	4.16.1.3	4.16.1.3	4.16.1.3
rpm-plugin-selinux	4.16.1.3	4.16.1.3	
rpm-plugin-systemd-inhibit	4.16.1.3	4.16.1.3	
rpm-sign-libs	4.16.1.3	4.16.1.3	4.16.1.3
rsync	3.2.6		
rust-srpm-macros	21		
sbsigntools	0.9.4	0.9.4	
screen	4.8.0		
sed	4.8	4.8	4.8
selinux-policy	37.22	37.22	
selinux-policy-targeted	37.22	37.22	
setup	2.13.7	2.13.7	2.13.7
shadow-utils	4.9	4.9	
slang	2.3.2		
sqlite-libs	3.40.0	3.40.0	3.40.0
sssd-client	2.9.4		
sssd-common	2.9.4		

套件	AMI	最小 AMI	容器
sssd-kcm	2.9.4		
sssd-nfs-idmap	2.9.4		
strace	6.8		
sudo	1.9.15	1.9.15	
sysctl-defaults	1.0	1.0	
sysstat	12.5.6		
systemd	252.16	252.16	
systemd-libs	252.16	252.16	
systemd-networkd	252.16	252.16	
systemd-pam	252.16	252.16	
systemd-resolved	252.16	252.16	
systemd-udev	252.16	252.16	
system-release	2023.4.20240513	2023.4.20240513	2023.4.20240513
systemtap-runtime	4.8		
tar	1.34	1.34	
tbb	2020.3		
tcpdump	4.99.1		
tcsh	6.24.07		

套件	AMI	最小 AMI	容器
time	1.9		
traceroute	2.1.3		
tzdata	2024a	2024a	2024a
unzip	6.0		
update-motd	2.2	2.2	
userspace-rcu	0.12.1	0.12.1	
util-linux	2.37.4	2.37.4	
util-linux-core	2.37.4	2.37.4	
vim-common	9.0.2153		
vim-data	9.0.2153	9.0.2153	
vim-enhanced	9.0.2153		
vim-filesystem	9.0.2153		
vim-minimal	9.0.2153	9.0.2153	
wget	1.21.3		
which	2.21	2.21	
words	3.0		
xfsdump	3.1.11		
xfspgms	5.18.0	5.18.0	
xxd	9.0.2153		
xxhash-libs	0.8.0		

套件	AMI	最小 AMI	容器
xz	5.2.5	5.2.5	
xz-libs	5.2.5	5.2.5	5.2.5
yum	4.14.0	4.14.0	4.14.0
zip	3.0		
zlib	1.2.11	1.2.11	1.2.11
zram-generator	1.1.2	1.1.2	
zram-generator-defaults	1.1.2	1.1.2	
zstd	1.5.5	1.5.5	

在容器中使用 AL2023

Note

如需如何使用 AL2023 在 Amazon ECS 上託管容器化工作負載的詳細資訊，請參閱。[適用於 Amazon ECS 容器主機的 AL2023](#)

根據使用案例，AL2023 可以在容器內使用多種方法。最類似於 Amazon Linux 2 容器映像和 AL2023 最小的 AMI。[AL2023 基本容器映像](#)

[針對進階使用者，我們提供 AL2023.2 版本中引入的最小容器映像檔，以及說明如何建置裸機容器的文件。](#)

AL2023 也可用於託管容器化工作負載，無論是以 AL2023 為基礎的容器映像，或是以其他 Linux 發行版為基礎的容器。您可以直接使用 [適用於 Amazon ECS 容器主機的 AL2023](#) 或使用提供的容器執行期套件。docker、containerd 和 nerdctl 套件可以在 AL2023 上安裝和使用。

主題

- [使用 AL2023 基本容器映像檔](#)

- [AL2023 最小容器圖像](#)
- [構建裸露的 AL2023 容器映像](#)
- [比較 Amazon Linux 2023 容器映像安裝的套件](#)
- [比較 Amazon Linux 2023 最小 AMI 與容器映像安裝的套件](#)

使用 AL2023 基本容器映像檔

AL2023 容器映像檔是由 AL2023 AMI 所包含的相同軟體元件所建立。該映像可用於任何環境中並作為 Docker 工作負載的基礎映像。如果在 [Amazon Elastic Compute Cloud \(Amazon EC2\)](#) 中的應用程式中使用 Amazon Linux AMI，則可以使用 Amazon Linux 容器映像對應用程式進行容器化。

在您的本機開發環境中使用 Amazon Linux 容器映像檔，然後將您的應用程式推送至 AWS 使用 [Amazon 彈性容器服務 \(Amazon ECS\)](#)。如需詳細資訊，請參閱《Amazon Elastic Container Registry 使用者指南》中的[搭配使用 Amazon ECR 映像與 Amazon ECS](#)。

Amazon Linux 容器映像可在 Amazon ECR Public 使用。您可以通過指定的 AWS 代表提供有關 AL2023 的反饋，也可以通過在[亞馬遜-linu](#) x-2023 軟件庫中提交問題來提交問題。 GitHub

從 Amazon ECR Public 中提取 Amazon Linux 容器映像

1. 向 Amazon Linux Public 登錄檔驗證您的 Docker 用戶端。驗證字符有效時間為 12 小時。如需詳細資訊，請參閱《Amazon Elastic Container Registry 使用者指南》中的[私有登錄檔身分驗證](#)。

Note

使用版本 2 的 AWS CLI 最新版本支援此 `get-login-password` 指令。如需詳細資訊，請參閱《AWS Command Line Interface 使用者指南》中的[安裝 AWS Command Line Interface](#)。

```
$ aws ecr-public get-login-password --region us-east-1 | docker login --username  
AWS --password-stdin public.ecr.aws
```

輸出如下。

```
Login succeeded
```

2. 使用 `docker pull` 命令提取 Amazon Linux 容器映像。若要在 Amazon ECR 公共映像庫上查看 Amazon Linux 容器映像，請參閱 [Amazon ECR 公共映像庫 - amazonlinux](#)。

Note

當您提取 AL2023 Docker 容器映像時，您可以使用下列其中一種格式的標籤：

- 若要取得 AL2023 容器映像的最新版本，請使用 `:2023` 標籤。
- 若要取得 AL2023 的特定版本，您可以使用下列格式：
 - `:2023.[0-7 release quarter].[release date].[build number]`

以下範例使用標籤 `:2023` 並提取 AL2023 最新可用的容器映像。

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023
```

3. (選用) 在本機執行容器。

```
$ docker run -it --security-opt seccomp=unconfined public.ecr.aws/amazonlinux/amazonlinux:2023 /bin/bash
```

從 Docker Hub 提取 AL2023 容器映像

1. 使用 `docker pull` 命令提取 AL2023 容器映像。

```
$ docker pull amazonlinux:2023
```

2. (選用) 在本機執行容器。

```
$ docker run -it amazonlinux:2023 /bin/bash
```

Note

AL2023 的容器映像僅使用 `dnf` 套件管理員來安裝軟體套件。這表示沒有 `amazon-linux-extras` 或同等的命令可用於其他軟體。

AL2023 最小容器圖像

Note

標準的 AL2023 容器映像適用於大多數使用案例，並且適應最小容器映像可能比適應 AL2023 基本容器映像更多的工作。

AL2023.2 中引入的 AL2023 最小容器映像檔與基礎容器映像檔不同，因為它只包含安裝其他套件所需的最低限度套件。最小容器映像檔被設計為最小的套件集，而不是一組方便的套件。

AL2023 最小容器映像是由 AL2023 中已有的軟體元件建置而成。在最小容器映像的關鍵區別是用 `microdnf` 來提供 `dnf` 包管理器，而不是 Python 基於功能齊全 `dnf`。如此可讓最小容器映像縮小，因為沒有包含在 AL2023 AMI 和基本容器映像中的 `dnf` 套件管理員的完整功能集。

AL2023 最小容器映像檔構成 `provided.al2023` AWS Lambda 執行階段環境的基礎。

如需最小容器映像檔中包含的套件的詳細清單，請參閱 [比較 Amazon Linux 2023 容器映像安裝的套件](#)。

最小容器映像的大小

由於 AL2023 最小容器映像所包含的封裝數量少於 AL2023 基礎容器映像，因此也明顯縮小。下表比較目前和過去版本的 Amazon Linux 容器映像檔選項。

Note

映像大小如 [Amazon ECR 公共映像庫的 Amazon Linux](#) 所示。

映像	版本	映像大小	注意
Amazon Linux 1 (AL1)	2018.03.0.20230918 .0	62.3MB	僅限 x86-64
Amazon Linux 2	2.0.20230926.0	64.2MB	aarch64 比 x86-64 大 1.6 MB

映像	版本	映像大小	注意
Amazon Linux 2023 基本容器映像	2023.2.20231002.0	52.4MB	
Amazon Linux 2023 最小容器映像	2023.2.20231002.0- minimal	35.2MB	

使用 AL2023 最小容器映像

AL2023 最小容器映像檔可在上使用，而 ECR 且 `2023-minimal` 標籤將始終指向最新的 AL2023 最小容器映像，而 `minimal` 標籤可能會更新為比 AL2023 更新的 Amazon Linux 版本。

您可以使用 `docker` 以下示例提取這些標籤：

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:minimal
```

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023-minimal
```

下面的示例顯示 Dockerfile 了一個採用最小容器映像並在其上安裝 GCC：

```
FROM public.ecr.aws/amazonlinux/amazonlinux:2023-minimal
RUN dnf install -y gcc && dnf clean all
```

構建裸露的 AL2023 容器映像

AL2023 容器映像檔是由 AL2023 AMI 所包含的相同軟體元件所建立。它包含一個軟體，可讓基礎容器層的行為與在 Amazon EC2 執行個體 (例如套件管理員) 上執行的行為類似 `dnf`。本節說明如何從頭開始建構容器，該容器僅包含應用程式所需的最低限度相依性。

Note

標準 AL2023 容器映像檔適用於大多數使用案例。使用標準容器映像可讓您輕鬆地以映像為基礎進行建置。裸露的容器映像使得在圖像之上構建變得更加困難。

為應用程式建立具有最小相依性的容器

1. 決定執行期相依性。這將根據您的應用程式而有所不同。
2. 構建一個可建置 FROM scratch 的 Dockerfile / Containerfile。下列的 Dockerfile 範例可用來建置只包含 bash shell 及其相依性的容器。

```
FROM public.ecr.aws/amazonlinux/amazonlinux:2023 as build
RUN mkdir /sysroot
RUN dnf --releasever=$(rpm -q system-release --qf '%{VERSION}') \
  --installroot /sysroot \
  -y \
  --setopt=install_weak_deps=False \
  install bash

FROM scratch
COPY --from=build /sysroot /
WORKDIR /
ENTRYPOINT ["/bin/bash"]
```

- 此 Dockerfile 的運作方式：
 1. 啟動名為 build 的 AL2023 容器。此容器將用於引導準系統容器，此容器本身不會部署，而是生成要部署的容器。
 2. 建立 /sysroot 目錄。此目錄將會是 build 容器安裝準系統容器所需相依性的目錄。在後續步驟中，/sysroot 路徑會封裝成準系統映像的根目錄。

使用 `--installroot` 選項並以此方式 `dnf`，是建立其他 AL2023 映像的方式。這是 `dnf` 用來執行安裝程式和映像建立工具的功能。

3. 調用 `dnf` 以將套件安裝到 /sysroot。

`rpm -q system-release --qf '%{VERSION}'` 命令查詢 (-q) `system-release` 套件、設定查詢格式 (`--qf`) 以印出受查詢的套件版本 (`%{VERSION}` 變數是 RPM 版本的 `rpm` 變數)。

將 `dnf` 的 `--releasever` 引數設為 build 容器中的 `system-release` 版本，每次發行 Amazon Linux 的更新版容器基本映像時，Dockerfile 就能用來重建準系統容器。

可以將其設置 `--releasever` 為任何 Amazon 版本，例如 這樣做意味著 `build` 容器將作為最新的 AL2023 版本運行，但無論當前的 AL2023 版本如何，都可以從 2023.4.20240513 構建準系統容器。

`--setopt=install_weak_deps=False` 組態選項會告訴 `dnf` 只安裝必要的相依性，而非推薦或建議的相依性。

- 將已安裝的系統複製到空白 (FROM scratch) 容器的根目錄中。
- 在此情況 `/bin/bash` 下，將 `ENTRYPOINT` 設為所需的二進位。
- 建立目錄，並將步驟 2 範例的內容新增至名為 `Dockerfile` 的檔案。

```
$ mkdir al2023-barebones-bash-example
$ cd al2023-barebones-bash-example
$ cat > Dockerfile <<EOF
FROM public.ecr.aws/amazonlinux/amazonlinux:2023 as build
RUN mkdir /sysroot
RUN dnf --releasever=$(rpm -q system-release --qf '%{VERSION}') \
  --installroot /sysroot \
  -y \
  --setopt=install_weak_deps=False \
  install bash && dnf --installroot /sysroot clean all

FROM scratch
COPY --from=build /sysroot /
WORKDIR /
ENTRYPOINT ["/bin/bash"]
EOF
```

- 執行下列命令來建置容器。

```
$ docker build -t al2023-barebones-bash-example
```

- 使用下列命令執行容器，以查看僅 `bash` 容器的最小程度。

```
$ docker run -it --rm al2023-barebones-bash-example
bash-5.2# rpm
bash: rpm: command not found
bash-5.2# du -sh /usr/
bash: du: command not found
bash-5.2# ls
```



```
bash: ls: command not found
bash-5.2# echo /bin/*
/bin/alias /bin/bash /bin/bashbug /bin/bashbug-64 /bin/bg /bin/catchsegv /bin/cd /
bin/command /bin/fc /bin/fg /bin/gencat /bin/getconf /bin/getent /bin/getopts /
bin/hash /bin/iconv /bin/jobs /bin/ld.so /bin/ldd /bin/locale /bin/localedef /
bin/pldd /bin/read /bin/sh /bin/sotruss /bin/sprof /bin/type /bin/tzselect /bin/
ulimit /bin/umask /bin/unalias /bin/wait /bin/zdump
```

如需更實際的範例，下列程序會為顯示 Hello World! 的 C 應用程式建置容器。

1. 建立空白目錄，然後新增 C 原始碼和 Dockerfile。

```
$ mkdir al2023-barebones-c-hello-world-example
$ cd al2023-barebones-c-hello-world-example
$ cat > hello-world.c <<EOF
#include <stdio.h>
int main(void)
{
    printf("Hello World!\n");
    return 0;
}
EOF

$ cat > Dockerfile <<EOF
FROM public.ecr.aws/amazonlinux/amazonlinux:2023 as build
COPY hello-world.c /
RUN dnf -y install gcc
RUN gcc -o hello-world hello-world.c
RUN mkdir /sysroot
RUN mv hello-world /sysroot/
RUN dnf --releasever=$(rpm -q system-release --qf '%{VERSION}') \
    --installroot /sysroot \
    -y \
    --setopt=install_weak_deps=False \
    install glibc && dnf --installroot /sysroot clean all

FROM scratch
COPY --from=build /sysroot /
WORKDIR /
ENTRYPOINT ["/hello-world"]
EOF
```

2. 使用下列命令來建置容器。

```
$ docker build -t al2023-barebones-c-hello-world-example .
```

3. 使用下列命令來執行容器。

```
$ docker run -it --rm al2023-barebones-c-hello-world-example
Hello World!
```

比較 Amazon Linux 2023 容器映像安裝的套件

AL2023 基本容器映像上存在的 RPM 與 AL2023 最小容器映像上存在的 RPM 進行比較的比較。

套件	容器	最小容器
alternatives	1.15	1.15
amazon-linux-repo-cdn	2023.4.20240513	2023.4.20240513
audit-libs	3.0.6	3.0.6
basesystem	11	11
bash	5.2.15	5.2.15
bzip2-libs	1.0.8	1.0.8
ca-certificates	2023.2.64	2023.2.64
coreutils-single	8.32	8.32
crypto-policies	20220428	20220428
curl-minimal	8.5.0	8.5.0
dnf	4.14.0	
dnf-data	4.14.0	4.14.0

套件	容器	最小容器
elfutils-default-yama-scope	0.188	
elfutils-libelf	0.188	
elfutils-libs	0.188	
expat	2.5.0	
file-libs	5.39	5.39
filesystem	3.14	3.14
gawk	5.1.0	5.1.0
gdbm-libs	1.19	
glib2	2.74.7	2.74.7
glibc	2.34	2.34
glibc-common	2.34	2.34
glibc-minimal-langpack	2.34	2.34
gmp	6.2.1	6.2.1
gnupg2-minimal	2.3.7	2.3.7
gobject-introspection		1.73.0
gpgme	1.15.1	1.15.1
grep	3.8	3.8
json-c	0.14	0.14

套件	容器	最小容器
keyutils-libs	1.6.3	1.6.3
krb5-libs	1.21	1.21
libacl	2.3.1	2.3.1
libarchive	3.5.3	3.5.3
libassuan	2.5.5	2.5.5
libattr	2.5.1	2.5.1
libblkid	2.37.4	2.37.4
libcap	2.48	2.48
libcap-ng	0.8.2	0.8.2
libcom_err	1.46.5	1.46.5
libcomps	0.1.20	
libcurl-minimal	8.5.0	8.5.0
libdnf	0.69.0	0.69.0
libffi	3.4.4	3.4.4
libgcc	11.4.1	11.4.1
libgcrypt	1.10.2	1.10.2
libgomp	11.4.1	
libgpg-error	1.42	1.42
libidn2	2.3.2	2.3.2
libmodulemd	2.13.0	2.13.0

套件	容器	最小容器
libmount	2.37.4	2.37.4
libnghttp2	1.59.0	1.59.0
libpeas		1.32.0
libpsl	0.21.1	0.21.1
librepo	1.14.5	1.14.5
libreport-filessystem	2.15.2	2.15.2
libselinux	3.4	3.4
libsepol	3.4	3.4
libsigsegv	2.13	2.13
libsmartcols	2.37.4	2.37.4
libsolv	0.7.22	0.7.22
libstdc++	11.4.1	11.4.1
libtasn1	4.19.0	4.19.0
libunistring	0.9.10	0.9.10
libuuid	2.37.4	2.37.4
libverto	0.3.2	0.3.2
libxcrypt	4.4.33	
libxml2	2.10.4	2.10.4
libyaml	0.2.5	0.2.5
libzstd	1.5.5	1.5.5

套件	容器	最小容器
lua-libs	5.4.4	5.4.4
lz4-libs	1.9.4	1.9.4
microdnf		3.8.1
microdnf-dnf		3.8.1
mpfr	4.1.0	4.1.0
ncurses-base	6.2	6.2
ncurses-libs	6.2	6.2
npth	1.6	1.6
openssl-libs	3.0.8	3.0.8
p11-kit	0.24.1	0.24.1
p11-kit-trust	0.24.1	0.24.1
pcre2	10.40	10.40
pcre2-syntax	10.40	10.40
popt	1.18	1.18
publicsuffix-list-dafsa	20240212	20240212
python3	3.9.16	
python3-dnf	4.14.0	
python3-gpg	1.15.1	
python3-hawkey	0.69.0	
python3-libcomps	0.1.20	

套件	容器	最小容器
python3-libdnf	0.69.0	
python3-libs	3.9.16	
python3-pip-wheel	21.3.1	
python3-rpm	4.16.1.3	
python3-setuptools-wheel	59.6.0	
readline	8.1	8.1
rpm	4.16.1.3	4.16.1.3
rpm-build-libs	4.16.1.3	
rpm-libs	4.16.1.3	4.16.1.3
rpm-sign-libs	4.16.1.3	
sed	4.8	4.8
setup	2.13.7	2.13.7
sqlite-libs	3.40.0	3.40.0
system-release	2023.4.20240513	2023.4.20240513
tzdata	2024a	
xz-libs	5.2.5	5.2.5
yum	4.14.0	
zlib	1.2.11	1.2.11

比較 Amazon Linux 2023 最小 AMI 與容器映像安裝的套件

AL2023 微型 AMI 上存在的 RPM 與 AL2023 基礎和最小容器映像上存在的 RPM 的比較。

套件	最小 AMI	容器	最小容器
alternatives	1.15	1.15	1.15
amazon-chrony-config	4.3		
amazon-ec2-net-utils	2.4.1		
amazon-linux-repo-cdn		2023.4.20240513	2023.4.20240513
amazon-linux-repo-s3	2023.4.20240513		
amazon-linux-sb-keys	2023.1		
audit	3.0.6		
audit-libs	3.0.6	3.0.6	3.0.6
awscli-2	2.15.30		
basesystem	11	11	11
bash	5.2.15	5.2.15	5.2.15
bzip2-libs	1.0.8	1.0.8	1.0.8
ca-certificates	2023.2.64	2023.2.64	2023.2.64
checkpolicy	3.4		
chrony	4.3		

套件	最小 AMI	容器	最小容器
cloud-init	22.2.2		
cloud-init-cfg-ec2	22.2.2		
cloud-utils-growpart	0.31		
coreutils	8.32		
coreutils-common	8.32		
coreutils-single		8.32	8.32
cpio	2.13		
cracklib	2.9.6		
cracklib-dicts	2.9.6		
crypto-policies	20220428	20220428	20220428
cryptsetup-libs	2.6.1		
curl-minimal	8.5.0	8.5.0	8.5.0
cyrus-sasl-lib	2.1.27		
dbus	1.12.28		
dbus-broker	32		
dbus-common	1.12.28		
dbus-libs	1.12.28		
device-mapper	1.02.185		

套件	最小 AMI	容器	最小容器
device-mapper-libs	1.02.185		
diffutils	3.8		
dnf	4.14.0	4.14.0	
dnf-data	4.14.0	4.14.0	4.14.0
dnf-plugin-release-notification	1.2		
dnf-plugins-core	4.3.0		
dnf-plugin-support-info	1.2		
dracut	055		
dracut-config-ec2	3.0		
dracut-config-generic	055		
e2fsprogs	1.46.5		
e2fsprogs-libs	1.46.5		
ec2-utils	2.2.0		
efi-filesystem	5		
efivar	38		
efivar-libs	38		

套件	最小 AMI	容器	最小容器
elfutils- default-yama- scope	0.188	0.188	
elfutils-libelf	0.188	0.188	
elfutils-libs	0.188	0.188	
expat	2.5.0	2.5.0	
file	5.39		
file-libs	5.39	5.39	5.39
filesystem	3.14	3.14	3.14
findutils	4.8.0		
fuse-libs	2.9.9		
gawk	5.1.0	5.1.0	5.1.0
gdbm-libs	1.19	1.19	
gdisk	1.0.8		
gettext	0.21		
gettext-libs	0.21		
glib2	2.74.7	2.74.7	2.74.7
glibc	2.34	2.34	2.34
glibc-all- langpacks	2.34		
glibc-common	2.34	2.34	2.34

套件	最小 AMI	容器	最小容器
glibc-locale-source	2.34		
glibc-minimal-langpack		2.34	2.34
gmp	6.2.1	6.2.1	6.2.1
gnupg2-minimal	2.3.7	2.3.7	2.3.7
gnutls	3.8.0		
gobject-introspection			1.73.0
gpgme	1.15.1	1.15.1	1.15.1
grep	3.8	3.8	3.8
groff-base	1.22.4		
grub2-common	2.06		
grub2-efi-aa64-ec2	2.06 (一個建築 64)		
grub2-efi-x64-ec2	2.06 (十六)		
grub2-pc-modules	2.06		
grub2-tools	2.06		
grub2-tools-minimal	2.06		
grubby	8.40		

套件	最小 AMI	容器	最小容器
gzip	1.12		
hostname	3.23		
hwdata	0.353		
inih	49		
initscripts	10.09		
iproute	5.10.0		
iputils	20210202		
irqbalance	1.9.0		
jansson	2.14		
jitterentropy	3.4.1		
jq	1.7.1		
json-c	0.14	0.14	0.14
kbd	2.4.0		
kbd-misc	2.4.0		
kernel	6.1.90		
kernel- li vepatch-repo- s3	2023.4.20240513		
keyutils-libs	1.6.3	1.6.3	1.6.3
kmod	29		
kmod-libs	29		

套件	最小 AMI	容器	最小容器
krb5-libs	1.21	1.21	1.21
less	608		
libacl	2.3.1	2.3.1	2.3.1
libarchive	3.5.3	3.5.3	3.5.3
libargon2	20171227		
libassuan	2.5.5	2.5.5	2.5.5
libattr	2.5.1	2.5.1	2.5.1
libblkid	2.37.4	2.37.4	2.37.4
libcap	2.48	2.48	2.48
libcap-ng	0.8.2	0.8.2	0.8.2
libcbor	0.7.0		
libcom_err	1.46.5	1.46.5	1.46.5
libcomps	0.1.20	0.1.20	
libcurl-minimal	8.5.0	8.5.0	8.5.0
libdb	5.3.28		
libdnf	0.69.0	0.69.0	0.69.0
libeconf	0.4.0		
libedit	3.1		
libfdisk	2.37.4		
libffi	3.4.4	3.4.4	3.4.4

套件	最小 AMI	容器	最小容器
libfido2	1.10.0		
libgcc	11.4.1	11.4.1	11.4.1
libgcrypt	1.10.2	1.10.2	1.10.2
libgomp	11.4.1	11.4.1	
libgpg-error	1.42	1.42	1.42
libidn2	2.3.2	2.3.2	2.3.2
libkcapi	1.4.0		
libkcapi-hmaccalc	1.4.0		
libmnl	1.0.4		
libmodulemd	2.13.0	2.13.0	2.13.0
libmount	2.37.4	2.37.4	2.37.4
libnghttp2	1.59.0	1.59.0	1.59.0
libpeas			1.32.0
libpipeline	1.5.3		
libpsl	0.21.1	0.21.1	0.21.1
libpwquality	1.4.4		
librepo	1.14.5	1.14.5	1.14.5
libreport-filessystem	2.15.2	2.15.2	2.15.2
libseccomp	2.5.3		

套件	最小 AMI	容器	最小容器
libselinux	3.4	3.4	3.4
libselinux- utils	3.4		
libsemanage	3.4		
libsepol	3.4	3.4	3.4
libsigsegv	2.13	2.13	2.13
libsmartcols	2.37.4	2.37.4	2.37.4
libsolv	0.7.22	0.7.22	0.7.22
libss	1.46.5		
libstdc++	11.4.1	11.4.1	11.4.1
libtasn1	4.19.0	4.19.0	4.19.0
libtextstyle	0.21		
libunistring	0.9.10	0.9.10	0.9.10
libuser	0.63		
libutempter	1.2.1		
libuuid	2.37.4	2.37.4	2.37.4
libverto	0.3.2	0.3.2	0.3.2
libxcrypt	4.4.33	4.4.33	
libxml2	2.10.4	2.10.4	2.10.4
libyaml	0.2.5	0.2.5	0.2.5
libzstd	1.5.5	1.5.5	1.5.5

套件	最小 AMI	容器	最小容器
logrotate	3.20.1		
lua-libs	5.4.4	5.4.4	5.4.4
lz4-libs	1.9.4	1.9.4	1.9.4
man-db	2.9.3		
microcode_ctl	2.1 (六十四)		
microdnf			3.8.1
microdnf-dnf			3.8.1
mpfr	4.1.0	4.1.0	4.1.0
ncurses	6.2		
ncurses-base	6.2	6.2	6.2
ncurses-libs	6.2	6.2	6.2
nettle	3.8		
net-tools	2.0		
npth	1.6	1.6	1.6
numactl-libs	2.0.14		
oniguruma	6.9.7.1		
openldap	2.4.57		
openssh	8.7 小時		
openssh-clients	8.7 小時		
openssh-server	8.7 小時		

套件	最小 AMI	容器	最小容器
openssl	3.0.8		
openssl-libs	3.0.8	3.0.8	3.0.8
openssl-pkcs11	0.4.12		
os-prober	1.77		
p11-kit	0.24.1	0.24.1	0.24.1
p11-kit-trust	0.24.1	0.24.1	0.24.1
pam	1.5.1		
passwd	0.80		
pciutils	3.7.0		
pciutils-libs	3.7.0		
pcre2	10.40	10.40	10.40
pcre2-syntax	10.40	10.40	10.40
policycoreutils	3.4		
popt	1.18	1.18	1.18
procps-ng	3.3.17		
psmisc	23.4		
publicsuffix-list-dafsa	20240212	20240212	20240212
python3	3.9.16	3.9.16	
python3-attrs	20.3.0		
python3-audit	3.0.6		

套件	最小 AMI	容器	最小容器
python3-awscrt	0.19.19		
python3-babel	2.9.1		
python3-cffi	1.14.5		
python3-chardet	4.0.0		
python3-colorama	0.4.4		
python3-configobj	5.0.6		
python3-cryptography	36.0.1		
python3-dateutil	2.8.1		
python3-dbus	1.2.18		
python3-distro	1.5.0		
python3-dnf	4.14.0	4.14.0	
python3-dnf-plugins-core	4.3.0		
python3-docutils	0.16		
python3-gpg	1.15.1	1.15.1	
python3-hawkey	0.69.0	0.69.0	
python3-idna	2.10		
python3-jinja2	2.11.3		

套件	最小 AMI	容器	最小容器
python3-j mespath	0.10.0		
python3-j sonpatch	1.21		
python3-j sonpointer	2.0		
python3-j sonschema	3.2.0		
python3-l ibcomps	0.1.20	0.1.20	
python3-libdnf	0.69.0	0.69.0	
python3-libs	3.9.16	3.9.16	
python3-l ibselinux	3.4		
python3-l ibsemanage	3.4		
python3-m arkupsafe	1.1.1		
python3-n etifaces	0.10.6		
python3-o authlib	3.0.2		
python3-pip- wheel	21.3.1	21.3.1	
python3-ply	3.11		

套件	最小 AMI	容器	最小容器
python3-p olicycoreutils	3.4		
python3-p rettytable	0.7.2		
python3-prompt- toolkit	3.0.24		
python3-p ycparser	2.20		
python3-p yrsistent	0.17.3		
python3-p yserial	3.4		
python3-pysocks	1.7.1		
python3-pytz	2022.7.1		
python3-pyyaml	5.4.1		
python3-r equests	2.25.1		
python3-rpm	4.16.1.3	4.16.1.3	
python3-ruamel- yaml	0.16.6		
python3-ruamel- yaml-clib	0.1.2		
python3-setools	4.4.1		

套件	最小 AMI	容器	最小容器
python3-s etuptools	59.6.0		
python3-s etuptools- wheel	59.6.0	59.6.0	
python3-six	1.15.0		
python3-systemd	235		
python3-urllib3	1.25.10		
python3-wcwidth	0.2.5		
readline	8.1	8.1	8.1
rng-tools	6.14		
rootfiles	8.1		
rpm	4.16.1.3	4.16.1.3	4.16.1.3
rpm-build-libs	4.16.1.3	4.16.1.3	
rpm-libs	4.16.1.3	4.16.1.3	4.16.1.3
rpm-plugin- selinux	4.16.1.3		
rpm-plugin- systemd-inhibit	4.16.1.3		
rpm-sign-libs	4.16.1.3	4.16.1.3	
sbsigntools	0.9.4		
sed	4.8	4.8	4.8

套件	最小 AMI	容器	最小容器
selinux-policy	37.22		
selinux-policy-targeted	37.22		
setup	2.13.7	2.13.7	2.13.7
shadow-utils	4.9		
sqlite-libs	3.40.0	3.40.0	3.40.0
sudo	1.9.15		
sysctl-defaults	1.0		
systemd	252.16		
systemd-libs	252.16		
systemd-networkd	252.16		
systemd-pam	252.16		
systemd-resolved	252.16		
systemd-udev	252.16		
system-release	2023.4.20240513	2023.4.20240513	2023.4.20240513
tar	1.34		
tzdata	2024a	2024a	
update-motd	2.2		
userspace-rcu	0.12.1		

套件	最小 AMI	容器	最小容器
util-linux	2.37.4		
util-linux-core	2.37.4		
vim-data	9.0.2153		
vim-minimal	9.0.2153		
which	2.21		
xfspg	5.18.0		
xz	5.2.5		
xz-libs	5.2.5	5.2.5	5.2.5
yum	4.14.0	4.14.0	
zlib	1.2.11	1.2.11	1.2.11
zram-generator	1.1.2		
zram-generator-defaults	1.1.2		
zstd	1.5.5		

AL2023 開啟 AWS Elastic Beanstalk

AWS Elastic Beanstalk 是一種用於部署和擴展 Web 應用程式和服務的服務。您只要上傳程式碼，Elastic Beanstalk 即會自動處理部署作業；無論是容量佈建、負載平衡、自動擴展或應用程式運作狀態監控，該服務都能滿足所需。如需詳細資訊，請參閱 [AWS Elastic Beanstalk](#)。

若要使用 Elastic Beanstalk，您須建立應用程式、將應用程式版本 (採用諸如 Java .war 檔案之應用程式原始碼套件的形式) 上傳至 Elastic Beanstalk，然後提供應用程式的部分資訊。Elastic Beanstalk 會自動啟動環境，並建立並設定執程式碼所需的 AWS 資源。如需詳細資訊，請參閱 [《AWS Elastic Beanstalk 開發人員指南》](#)。

Elastic Beanstalk Linux 平台使用 Amazon EC2 執行個體，而這些執行個體執行 Amazon Linux。自 2023 年 8 月 4 日起，Elastic Beanstalk 提供以下基於 Amazon Linux 2023 的平台分支：Docker、Tomcat、Java SE、Node.js、PHP 和 Python。Elastic Beanstalk 正在努力在更多 Elastic Beanstalk 平台上發佈對 AL2023 的支援。

您可以在《[Elastic Beanstalk 開發人員指南](#)》的「[Elastic Beanstalk Linux 平台](#)」一節中查看 Elastic Beanstalk 平台支援和目前建置在 AL2023 上的完整平台清單。

您可以在 [Elastic Beanstalk 版本備註](#) 中查看新 Elastic Beanstalk 平台的版本備註以及現有平台的版本

在中使用 AL2023 AWS CloudShell

AWS CloudShell 是一個以瀏覽器為基礎的預先驗證殼層，您可以直接從 AWS Management Console 您可以 CloudShell 從幾種不同 AWS Management Console 的方式導覽至。如需詳細資訊，請參閱[如何開始使用 AWS CloudShell?](#)

AWS CloudShell, 這是目前基於 Amazon Linux 2, 將遷移到 AL2023. AL2023 的遷移將從 2023 年 12 月 4 日開 AWS 區域 始全部開始推出。有關 CloudShell 遷移到 AL2023 的詳細資訊，請參閱[從 Amazon 2 AWS CloudShell 遷移到 Amazon](#)

使用以 AL2023 為基礎的 Amazon ECS AMI 託管容器化工作負載

Note

如需如何在容器內使用 AL2023 的詳細資訊，請參閱[容器中的 AL2023](#)。

Amazon Elastic Container Service (Amazon ECS) 為全受管容器協同運作服務，可讓您輕鬆部署、管理和擴展容器化應用程式。Amazon ECS 是全受管服務，內建 AWS 組態和操作最佳實務。它與 AWS 和第三方工具集成，例如 Amazon Elastic Container Registry (Amazon ECR) 和碼頭工具。這種整合可讓團隊能夠更輕鬆地專注於建置應用程式，而無需為環境分心。您可以在雲端跨 AWS 區域執行和擴展容器工作負載，而無需進行控制平面管理等複雜任務。

您可以使用以 AL2023 為基礎的 Amazon ECS 最佳化 AMI，在 AL2023 上託管容器化工作負載。如需詳細資訊，請參閱 [Amazon ECS 最佳化 AMI](#)

與 AL2 相比，Amazon ECS 的 AL2023 變化

與 AL2 一樣，AL2023 提供了作為 Amazon ECS Linux 執行個體執行個體執行所需的基本套件。在 AL2 中 containerd、docker、和 ecs-init 套件可透過取得 amazon-linux-extras，而 AL2023 則在核心儲存庫中包含這些套件。

通過版本化儲存庫功能進行確定性升級，默認情況下，每個 AL2023 AMI 都會鎖定到特定的儲存庫版本。對於 AL2023 Amazon ECS 優化 AMI 也是如此。在部署之前，您可以仔細管理和測試環境的所有更新，並提供在發生問題時還原為先前 AMI 內容的簡便方法。如需此 AL2023 功能的詳細資訊，請參閱 [透過 AL2023 上的版本化儲存庫使用確定性升級](#)。

AL2023 可透過 AL2 支援的群組第 1 個介面切換至群組 V2。如需詳細資訊，請參閱 [整合控制群組階層 \(cgroup v2\)](#)。

Note

[2023.2.20230920 之前的 AL2023 版本 \(第一個 AL2023 3.2 發行版本\)](#) 包含一個在群組中處理記憶體不足 (OOM) 的 systemd 錯誤。cgroup 中的所有進程總是被殺死，而不是一次選擇一個進程的 OOM 殺手，這是預期的行為。

與 AL2 行為相比，這是一個回歸，並且在 AL2023 的 2023.2.20230920 版本中被固定。

[此專案提供建置 Amazon ECS 最佳化 AMI 的程式碼。](#) [amazon-ecs-ami GitHub 發行說明](#) 描述了哪個 AL2023 版本映射到哪個 Amazon ECS AMI 版本。

自訂基於 AL2023 的 Amazon ECS 最佳化 AMI

Important

我們建議您使用 Amazon ECS 優化 AL2023 AMI。如需詳細資訊，請參閱 [Amazon 彈性容器服務開發人員指南中的 Amazon ECS 優化 AMI](#)。

您可以使用與 Amazon ECS 用來建立自訂 AMI 相同的建置指令碼。如需詳細資訊，請參閱 [Amazon ECS 最佳化 Linux AMI 建置](#) 指令碼。

在 AL2023 上使用 Amazon Elastic File System

Amazon Elastic File System (Amazon EFS) 提供無伺服器、完全彈性的檔案儲存功能，讓您無需佈建或管理儲存容量和效能，即可分享檔案資料。Amazon EFS 可隨需擴展至 PB 級，而不會中斷應用程式，並可隨著您新增和移除檔案而自動擴展及縮減。因為 Amazon EFS 採用簡單的 Web 服務介面，您可以快速輕鬆地建立和設定檔案系統。此服務會為您管理所有檔案儲存基礎設施，這表示您可以避免部署、修補和維護複雜檔案系統組態的複雜性。

Amazon EFS 支援網路檔案系統第 4 版 (NFSv4.1 和 NFSv4.0) 協定，因此您目前使用的應用程式和工具都可以與 Amazon EFS 無縫配合使用。多個運算執行個體 (包括 Amazon EC2、Amazon ECS 和 AWS Lambda) 可以同時存取 Amazon EFS 檔案系統。因此，EFS 檔案系統可針對在一個以上運算執行個體或伺服器執行的工作負載和應用程式，提供共同的資料來源。

在 AL2023 上安裝 `amazon-efs-utils`

該 `amazon-efs-utils` 套件可在 AL2023 儲存庫中使用，以便安裝並用於存取 Amazon EFS 檔案系統。

在 AL2023 上安裝 `amazon-efs-utils` 套件

- `amazon-efs-utils` 使用以下命令進行安裝。

```
$ dnf -y install amazon-efs-utils
```

在 AL2023 上掛載 Amazon EFS 檔案系統

安裝 `amazon-efs-utils` 完成之後，您可以在 AL2023 執行個體上掛載 Amazon EFS 檔案系統。

在 AL2023 上掛載 Amazon EFS 檔案系統

- 若要使用檔案系統 ID 進行裝載，請使用下列指令。

```
sudo mount -t efs file-system-id efs-mount-point/
```

您也可以掛載檔案系統，以便使用 TLS 加密傳輸中的資料，或使用 DNS 名稱或掛載目標 IP，而不是檔案系統 ID。如需詳細資訊，請參閱 [使用 EFS 掛載協助程式在 Amazon Linux 執行個體上掛載](#)。

使用建立在 AL2023 上的 Amazon EMR

Amazon EMR 是一種 Web 服務，可讓您輕鬆地使用 Apache Hadoop 和 AWS 提供的服務，以高效率地處理大量資料。

基於 AL2023 的 Amazon EMR 發布

Amazon EMR 版本 7.0.0 是基於 AL2023 構建的第一個版本。在此版本中，AL2023 是 Amazon EMR 的基礎操作系統，將 AL2023 的所有優勢帶入 Amazon EMR。如需詳細資訊，請參閱 [Amazon EMR 7.0.0 版本說明](#)。

EKS 上基於 Amazon EMR 的 AL2023

EKS 6.13 上的 Amazon EMR 是導入 AL2023 作為選項的第一個版本。在此版本中，您可以將 Spark 與 AL2023 同時啟動作為作業系統，並搭配 Java 17 執行期。如需詳細資訊，請參閱 [EKS 6.13 版本說明上的 Amazon EMR](#)，以及 [EKS 上的所有 Amazon EMR 發行說明](#)。

在中使用 AL2023 AWS Lambda

使用 AWS Lambda，您無需佈建或管理伺服器即可執行程式碼。您只要按實際使用的運算時間付費即可，未執行程式碼時不必支付任何費用。可以針對幾乎任何類型的應用程式或後端服務執行程式碼，且通常都不需要管理。您只需上傳程式碼，Lambda 就會處理執行程式碼及擴展所需的各項工作，藉此維持高可用性。

AL2023 **provided.al2023** 託管運行時和容器映像

`provided.al2023` 基本執行階段以 [AL2023 最小容器映像](#) 為基礎，並提供以 AL2023 為基礎的 Lambda 受管執行階段和 [容器基礎映像](#)。由於 `provided.al2023` 執行階段是以 AL2023 最小容器映像檔為基礎，因此在 40 MB 以下的 `provided.al2` 執行階段大幅小於 109 MB 的執行階段。

如需詳細資訊，請參閱 [Lambda 執行階段](#) 和 [使用 Lambda 容器映像](#)。

基於 AL2023 的 Lambda 執行階段

受管理語言執行階段的未來發行版本 (例如 Node.js 20、Python 3.12、Java 21 和 .NET 8) 是以 AL2023 為基礎，並將用 `provided.al2023` 作基礎映像，如 [AL2023 執行階段公告](#) 中所述。

基於 AL2023 的 Lambda 函數

- [AL2023 Lambda 函數寫入 Go](#)

- [AL2023 Lambda 函數寫入 Rust](#)

如需詳細資訊，請參閱AWS Lambda 開發人員指南中的[Lambda 執行階段](#)。

教學課程

以下教學說明如何使用執行 Amazon Linux 2023 (AL2023) 的亞馬遜 EC2 執行個體執行一般任務。如需視訊教學課程，請參閱[AWS 教學影片和實驗室](#)。

如需 AL2 指示，請參閱 [Amazon EC2 使用者指南中針對執行 Linux 的 Amazon EC2 執行個體教學課程](#)。

教學課程

- [教學課程：在 AL2023 上安裝燈泡伺服器](#)
- [教學課程：在 AL2023 上設定 SSL/TLS](#)
- [教學課程：在 AL2023 上主持一個 WordPress 部落格](#)

教學課程：在 AL2023 上安裝燈泡伺服器

下列程序可協助您在 AL2023 執行個體 (有時稱為 LAMP 網頁伺服器或 LAMP 堆疊) 上安裝具有 PHP 和 [MariaDB](#) (社群開發的 MySQL 分支) 支援的 Apache 網頁伺服器。您可以使用此伺服器託管一個靜態網站或部署一個將資訊讀取和寫入資料庫的動態 PHP 應用程式。

Important

這些程序適用於搭配 AL2023 使用。如果您要在不同的發行版 (例如 Ubuntu 或 Red Hat Enterprise Linux) 上設定 LAMP Web 伺服器，本教學不適用於您。如需 Ubuntu，請參閱下列社群文件：[ApacheMySQL PHP](#)。若是其他發行版，請參閱其特定文件。

任務

- [步驟 1：準備 LAMP 伺服器](#)
- [步驟 2：測試您的 LAMP 伺服器](#)
- [步驟 3：保護資料庫伺服器](#)
- [步驟 4：\(可選\) 安裝 phpMyAdmin](#)
- [疑難排解](#)
- [相關主題](#)

步驟 1：準備 LAMP 伺服器

必要條件

- 本教學課程假設您已使用 AL2023 啟動新執行個體，並具有可從網際網路存取的公用 DNS 名稱。如需詳細資訊，請參閱 [Amazon EC2 上的 AL2023](#)。您也必須先設定您的安全群組，允許 SSH (連接埠 22)、HTTP (連接埠 80) 和 HTTPS (連接埠 443) 連線。如需這些先決條件的詳細資訊，請參閱 Amazon EC2 使用者指南中的 [授權 Linux 執行個體的入站流量](#)。
- 下列程序會安裝 AL2023 (目前為 8.1) 上可用的最新 PHP 版本。若您預計不使用此教學所提及的 PHP 應用程式，改使用其他 PHP 應用程式，您應檢查其與 PHP 8.1 的相容性。

準備 LAMP 伺服器

1. 連線到您的執行個體。如需詳細資訊，請參閱 [連線至 AL2023 執行個體](#)。
2. 為確保所有軟體套件皆為最新版本，請對您的執行個體執行快速軟體更新。本程序可能需費時幾分鐘，但確定您擁有最新的安全更新和錯誤修正至關重要。

-y 選項不要求確認就會安裝更新。如果您要先檢查更新再安裝，則可以略過此選項。

```
[ec2-user ~]$ sudo dnf update -y
```

3. 安裝最新版本的阿帕奇網頁伺服器和適用於 AL2023 的 PHP 套件。

```
[ec2-user ~]$ sudo dnf install -y httpd wget php-fpm php-mysql php-json php php-devel
```

4. 安裝 MariaDB 軟體套件。使用 dnf install 命令可以同時安裝多個軟體套件和所有相關的依存項目。

```
[ec2-user ~]$ sudo dnf install mariadb105-server
```

您可以使用下列命令來檢視這些套件的目前版本：

```
[ec2-user ~]$ sudo dnf info package_name
```

範例：

```
[root@ip-172-31-25-170 ec2-user]# dnf info mariadb105
```

```
Last metadata expiration check: 0:00:16 ago on Tue Feb 14 21:35:13 2023.
Installed Packages
Name       : mariadb105
Epoch     : 3
Version   : 10.5.16
Release   : 1.amzn2023.0.6
Architecture : x86_64
Size      : 18 M
Source    : mariadb105-10.5.16-1.amzn2023.0.6.src.rpm
Repository : @System
From repo  : amazonlinux
Summary   : A very fast and robust SQL database server
URL       : http://mariadb.org
License   : GPLv2 and LGPLv2
Description : MariaDB is a community developed fork from MySQL - a multi-user,
multi-threaded
           : SQL database server. It is a client/server implementation consisting
of
           : a server daemon (mariadb) and many different client programs and
libraries.
           : The base package contains the standard MariaDB/MySQL client programs
and
           : utilities.
```

5. 啟動 Apache Web 伺服器。

```
[ec2-user ~]$ sudo systemctl start httpd
```

6. 使用 systemctl 命令來設定 Apache Web 伺服器在每次系統開機時啟動。

```
[ec2-user ~]$ sudo systemctl enable httpd
```

要確認 httpd 已啟用，您可以執行以下命令：


```
[ec2-user ~]$ sudo systemctl is-enabled httpd
```

7. 如果您尚未這樣做，請新增安全規則以允許對內 HTTP (連接埠 80) 連線到您的執行個體。根據預設，launch-wizard-*N* 安全群組在啟動期間已針對您的執行個體建立完畢。如果您未新增安全群組規則，該群組只會包含允許 SSH 連線的單一規則。

- a. 前往 <https://console.aws.amazon.com/ec2/> 開啟 Amazon EC2 主控台。
- b. 在左側的導覽窗格中，選取 Instances (執行個體)，然後選取您的執行個體。

- c. 在 Security (安全性) 標籤上，檢視對內規則。您應該會看到下列規則：

Port range	Protocol	Source
22	tcp	0.0.0.0/0

 Warning

使用 0.0.0.0/0 可讓所有 IPv4 地址使用 SSH 存取您的執行個體。通常在測試環境中短暫進行此作業是沒有問題的，但用在生產環境則不安全。在生產環境中，您應只授權特定 IP 地址或特定範圍的地址存取您的執行個體。

- d. 如果允許 HTTP (連接埠 80) 連線的傳入規則不存在，您必須現在新增規則。選擇安全群組的連結。使用請參閱[授權 Linux 執行個體的入站流量中的](#)程序，新增具有下列值的新輸入安全性規則：

- 類型：HTTP
- Protocol (通訊協定)：TCP
- Port Range (連接埠範圍)：80
- Source (來源)：自訂

8. 測試您的 Web 伺服器。在 Web 瀏覽器中，輸入執行個體的公有 DNS 地址 (或公有 IP 地址)。如果 /var/www/html 中沒有內容，則您應該會看到 Apache 測試頁，其會顯示 "It works!" (成功了!) 訊息。

您可以使用 Amazon EC2 主控台取得執行個體的公有 DNS (請查看 Public IPv4 DNS (公有 IPv4 DNS) 欄；如果此欄為隱藏，請選擇 Preferences (偏好設定) (齒輪狀圖示) 並將 Public IPv4 DNS (公有 IPv4 DNS) 選項切換為開啟)。

確認執行個體的安全性群組包含允許連接埠 80 上的 HTTP 流量的規則。如需詳細資訊，請參閱[規則新增至安全性群組](#)。

 Important

如果您未使用 Amazon Linux，則可能也需要在您的執行個體上設定防火牆以允許這些連線。如需如何設定防火牆的詳細資訊，請參閱針對您特定散發的文件。

Apache httpd 提供保存在稱為 Apache 文件根目錄中的檔案。Amazon Linux Apache 文件根目錄是 `/var/www/html`，預設情況下由根擁有。

若要允許 `ec2-user` 帳戶操作此目錄中的檔案，您必須修改目錄的所有權和許可。有多種方法可以達成這件任務。在本教學中，您會將 `ec2-user` 新增至 `apache` 群組以向 `apache` 群組授予 `/var/www` 目錄的所有權，並指派寫入許可。

設定檔案許可

1. 將您的使用者 (在此案例中為 `ec2-user`) 新增至 `apache` 群組。

```
[ec2-user ~]$ sudo usermod -a -G apache ec2-user
```

2. 登出並重新登入，以取得新的群組並驗證您的成員資格。

- a. 登出 (使用 `exit` 命令或關閉終端機視窗)：

```
[ec2-user ~]$ exit
```

- b. 若要在 `apache` 群組中驗證您的會員資格，請重新連線至您的執行個體，然後執行下列命令：

```
[ec2-user ~]$ groups  
ec2-user adm wheel apache systemd-journal
```

3. 將 `/var/www` 的群組所有權及其內容變更為 `apache` 群組。

```
[ec2-user ~]$ sudo chown -R ec2-user:apache /var/www
```

4. 若要新增群組寫入許可並在將來的子目錄上設定群組 ID，請變更 `/var/www` 及其子目錄的目錄許可。

```
[ec2-user ~]$ sudo chmod 2775 /var/www && find /var/www -type d -exec sudo chmod  
2775 {} \;
```

5. 若要新增群組寫入許可，請以遞迴方式變更 `/var/www` 及其子目錄的檔案許可：

```
[ec2-user ~]$ find /var/www -type f -exec sudo chmod 0664 {} \;
```

現在，`ec2-user` (以及 `apache` 群組未來的任何成員) 可以新增、刪除和編輯 Apache 文件根目錄中的檔案，所以您可以新增內容 (例如靜態網站或 PHP 應用程式)。

保護您的 Web 伺服器 (選擇性)

執行 HTTP 通訊協定的 Web 伺服器不會為其傳送或接收的資料提供傳輸安全性。當您使用 Web 瀏覽器連線到 HTTP 伺服器時，您前往的 URL、您收到的網頁內容以及您提交的任何 HTML 表單內容 (包括密碼)，網路路徑的任何一處的竊聽者都可以看到。保護您的 Web 伺服器的最佳實務是安裝對 HTTPS (HTTP Secure) 的支援，HTTPS 使用 SSL/TLS 加密保護您的資料。

如需在伺服器上啟用 HTTPS 的相關資訊，請參閱 [教學課程：在 AL2023 上設定 SSL/TLS](#)。

步驟 2：測試您的 LAMP 伺服器

如果伺服器已安裝且正在執行，且檔案許可設定正確，則您的 `ec2-user` 帳戶應該能夠在可透過網際網路使用的 `/var/www/html` 目錄中建立 PHP 檔案。

測試您的 LAMP 伺服器

1. 在 Apache 文件根資料夾中建立 PHP 檔案。



```
[ec2-user ~]$ echo "<?php phpinfo(); ?>" > /var/www/html/phpinfo.php
```

如果您嘗試執行此命令時出現拒絕許可錯誤，請嘗試登出並重新登入，以取得您在 [設定檔案許可](#) 設定的適當群組許可。

2. 在 Web 瀏覽器中，輸入您剛才建立的檔案 URL。此 URL 為您執行個體的公有 DNS 地址，其後跟隨斜線和檔案名稱。例如：

```
http://my.public.dns.amazonaws.com/phpinfo.php
```

您現在應該會看見 PHP 資訊頁面：

PHP Version 8.1.7		
System	Linux ip-172-31-16-77.ec2.internal 5.15.57-28.127.amzn2022.aarch64 #1 SMP Thu Aug 4 17:06:57 UTC 2022 aarch64	
Build Date	Jun 7 2022 18:21:38	
Build System	Linux	
Build Provider	Amazon Linux	
Compiler	gcc (GCC) 11.3.1 20220421 (Red Hat 11.3.1-2)	
Architecture	aarch64	
Server API	FPM/FastCGI	
Virtual Directory Support	disabled	
Configuration File (php.ini) Path	/etc	
Loaded Configuration File	/etc/php.ini	
Scan this dir for additional .ini files	/etc/php.d	
Additional .ini files parsed	/etc/php.d/10-opcache.ini, /etc/php.d/20-bz2.ini, /etc/php.d/20-calendar.ini, /etc/php.d/20-ctype.ini, /etc/php.d/20-curl.ini, /etc/php.d/20-dom.ini, /etc/php.d/20-exif.ini, /etc/php.d/20-fileinfo.ini, /etc/php.d/20-ftp.ini, /etc/php.d/20-gd.ini, /etc/php.d/20-gettext.ini, /etc/php.d/20-iconv.ini, /etc/php.d/20-mbstring.ini, /etc/php.d/20-mysqld.ini, /etc/php.d/20-pdo.ini, /etc/php.d/20-phar.ini, /etc/php.d/20-simplexml.ini, /etc/php.d/20-sockets.ini, /etc/php.d/20-sqlite3.ini, /etc/php.d/20-tokenizer.ini, /etc/php.d/20-xml.ini, /etc/php.d/20-xmlwriter.ini, /etc/php.d/20-xsl.ini, /etc/php.d/30-mysqli.ini, /etc/php.d/30-pdo_mysql.ini, /etc/php.d/30-pdo_sqlite.ini, /etc/php.d/30-xmlreader.ini	
PHP API	20210902	
PHP Extension	20210902	
Zend Extension	420210902	
Zend Extension Build	API420210902,NTS	
PHP Extension Build	API20210902,NTS	
Debug Build	no	
Thread Safety	disabled	
Zend Signal Handling	enabled	
Zend Memory Manager	enabled	
Zend Multibyte Support	provided by mbstring	
IPv6 Support	enabled	
DTrace Support	available, disabled	
Registered PHP Streams	https, ftps, compress.zlib, php, file, glob, data, http, ftp, compress.bzip2, phar	
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2, tlsv1.3	
Registered Stream Filters	zlib.*, string.rot13, string.toupper, string.tolower, convert.*, consumed, dechunk, bzip2.*, convert.iconv.*	
This program makes use of the Zend Scripting Language Engine: Zend Engine v4.1.7, Copyright (c) Zend Technologies with Zend OPcache v8.1.7, Copyright (c), by Zend Technologies		

如果你未看見此頁面，請確認 `/var/www/html/phpinfo.php` 檔案在前述步驟中正確建立。您也可以使用下列命令來確認已安裝所有必要的套件。

```
[ec2-user ~]$ sudo dnf list installed httpd mariadb-server php-mysqld
```

如果您的輸出未列出所需之任何套件，請使用 `sudo yum install package` 命令來安裝。

3. 刪除 `phpinfo.php` 檔案。雖然這可能是有用的資訊，但基於安全因素，您不應將其廣播至網際網路。

```
[ec2-user ~]$ rm /var/www/html/phpinfo.php
```

現在您應擁有功能齊全的 LAMP Web 伺服器。如果您將內容新增至 `/var/www/html` 的文件根目錄，則應該可以在執行個體的公有 DNS 地址檢視該內容。

步驟 3：保護資料庫伺服器

MariaDB 伺服器的預設安裝有幾項非常適合測試和開發的功能，但針對生產伺服器應將其停用或移除。`mysql_secure_installation` 命令將引導您完成設定根目錄密碼並從安裝中移除不安全功能的程序。即使您不打算使用 MariaDB 伺服器，我們也建議您執行此程序。

保護 MariaDB 伺服器

1. 啟動 MariaDB 伺服器。

```
[ec2-user ~]$ sudo systemctl start mariadb
```

2. 執行 `mysql_secure_installation`。

```
[ec2-user ~]$ sudo mysql_secure_installation
```

- a. 請在系統提示時，輸入根帳戶的密碼。
 - i. 輸入目前的根密碼。根據預設，根帳戶並未設定密碼。按 Enter。
 - ii. 輸入 **Y** 來設定密碼，然後輸入兩次安全密碼。如需建立安全密碼的詳細資訊，請參閱 <https://identitysafe.norton.com/password-generator/>。請確保此密碼存放於安全處。

為 MariaDB 設定根密碼僅是確保資料庫安全的最基本措施。在建置或安裝資料庫驅動的應用程式時，通常會為該應用程式建立資料庫服務使用者，並避免使用根帳戶進行資料庫管理以外的任何作業。

- b. 輸入 **Y** 來移除匿名使用者帳戶。
 - c. 輸入 **Y** 來停用遠端根登入。
 - d. 輸入 **Y** 來移除測試資料庫。
 - e. 輸入 **Y** 來載入使用者權限資料表並儲存您的變更。
3. (選用) 如果您不打算立即使用 MariaDB 伺服器，請將其停止。再次需要時，您可以將其重啟。

```
[ec2-user ~]$ sudo systemctl stop mariadb
```

4. (選用) 如果您希望 MariaDB 伺服器在每次系統開機時啟動，請輸入下列命令。

```
[ec2-user ~]$ sudo systemctl enable mariadb
```

步驟 4：(可選) 安裝 phpMyAdmin

[phpMyAdmin](#) 是一種基於 Web 的數據庫管理工具，可用於查看和編輯 EC2 實例上的 MySQL 數據庫。請遵循下列步驟在您的 Amazon Linux 執行個體上安裝並設定 phpMyAdmin。

Important

我們不建議使用 phpMyAdmin 來存取 LAMP 伺服器，除非您在 Apache 中啟用了 SSL/TLS；否則，您的資料庫管理員密碼和其他資料將透過網際網路不安全傳輸。如需開發人員提供的安全性建議，請參閱[保護您的 phpMyAdmin 安裝](#)。如需有關在 EC2 執行個體上保護 Web 伺服器的一般資訊，請參閱[教學課程：在 AL2023 上設定 SSL/TLS](#)。

若要安裝 phpMyAdmin

1. 安裝所需的依存項目。

```
[ec2-user ~]$ sudo dnf install php-mbstring php-xml -y
```

2. 重新啟動 Apache。

```
[ec2-user ~]$ sudo systemctl restart httpd
```

3. 重新啟動 php-fpm。

```
[ec2-user ~]$ sudo systemctl restart php-fpm
```

4. 在 `/var/www/html` 中導覽至 Apache 文件根。

```
[ec2-user ~]$ cd /var/www/html
```

- 請從 <https://www.phpmyadmin.net/downloads> 中選取最新 phpMyAdmin 發行版本的原始碼套件。若要直接將檔案下載到您的執行個體，請複製連結並將其貼入 wget 命令，如下所示：

```
[ec2-user html]$ wget https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz
```

- 以下列命令建立 phpMyAdmin 資料夾，並將套件擷取至該資料夾中。

```
[ec2-user html]$ mkdir phpMyAdmin && tar -xvzf phpMyAdmin-latest-all-languages.tar.gz -C phpMyAdmin --strip-components 1
```

- 刪除 `phpMyAdmin-latest-all-languages.tar.gz` ####

```
[ec2-user html]$ rm phpMyAdmin-latest-all-languages.tar.gz
```

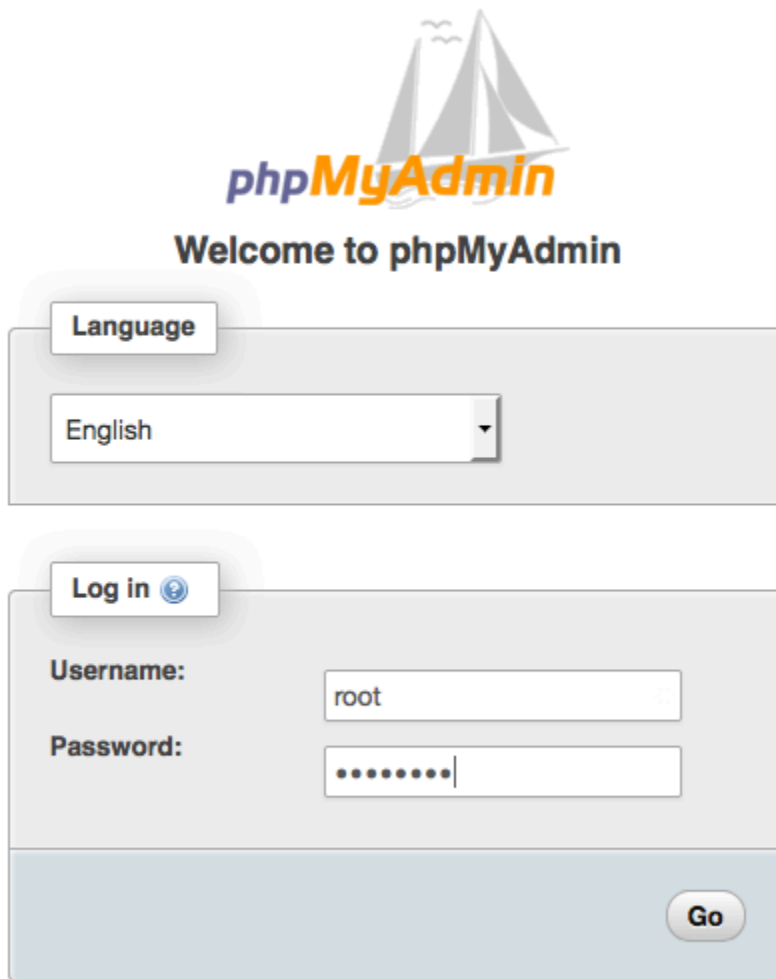
- (選用) 如果 MySQL 伺服器未執行，請現在將其啟動。

```
[ec2-user ~]$ sudo systemctl start mariadb
```

- 在網頁瀏覽器中，輸入 phpMyAdmin 安裝的 URL。此 URL 為您執行個體的公有 DNS 地址 (或公有 IP 地址)，其後跟隨斜線和您安裝目錄的檔案名稱。例如：

```
http://my.public.dns.amazonaws.com/phpMyAdmin
```

您應該會看到 phpMyAdmin 登入頁面：



The image shows the phpMyAdmin login interface. At the top is the phpMyAdmin logo and the text "Welcome to phpMyAdmin". Below this is a "Language" section with a dropdown menu set to "English". Underneath is a "Log in" section with a "Log in" button and a help icon. It contains two input fields: "Username:" with "root" entered, and "Password:" with masked characters. A "Go" button is located at the bottom right of the login section.

10. 使用您先前建立的root使用者名稱和 MySQL 根密碼登入您的 phpMyAdmin 安裝。

必須先設定您的安裝，才能將其投入使用。我們建議您從手動建立組態檔案開始，如下所示：

- a. 若要從最小的組態檔開始，請使用您最愛的文字編輯器建立新檔案，然後將 `config.sample.inc.php` 的內容複製到其中。
- b. 將檔案另存為 `config.inc.php` 在包含的 phpMyAdmin 目錄中 `index.php`。
- c. 如需任何其他設定，請參閱安裝指示的「[使用 phpMyAdmin 安裝程式指令碼](#)」一節中的檔案後建立指示。

若要取得有關使用的資訊 phpMyAdmin，請參閱[phpMyAdmin 使用指南](#)。

疑難排解

本節提供解決設定新 LAMP 伺服器時可能遇到之常見問題的建議。

我無法使用 Web 瀏覽器連線至我的伺服器

請執行下列檢查以查看您的 Apache Web 伺服器是否正在執行且可存取。

- Web 伺服器是否正在執行？

要確認 httpd 已啟用，您可以執行以下命令：

```
[ec2-user ~]$ sudo systemctl is-enabled httpd
```

如果 httpd 程序未執行，請重複 [準備 LAMP 伺服器](#) 所述的步驟。

- 防火牆是否設定正確？

確認執行個體的安全性群組包含允許連接埠 80 上的 HTTP 流量的規則。如需詳細資訊，請參閱[將規則新增至安全性群組](#)。

我無法使用 HTTPS 連線至我的伺服器

請執行下列檢查以查看您的 Apache Web 伺服器是否設為支援 HTTPS。

- Web 伺服器是否正確設定？

安裝 Apache 後，伺服器即針對 HTTP 流量進行設定。若要支援 HTTPS，請在伺服器上啟用 TLS 並安裝 SSL 憑證。如需相關資訊，請參閱[教學課程：在 AL2023 上設定 SSL/TLS](#)。

- 防火牆是否設定正確？

確認執行個體的安全性群組包含允許連接埠 443 上的 HTTPS 流量的規則。如需詳細資訊，請參閱[授權 Linux 執行個體的入站流量](#)。

相關主題

如需有關將檔案傳輸至執行個體或在 Web 伺服器上安裝 WordPress 部落格的詳細資訊，請參閱下列文件：

- 使用 Amazon EC2 使用者[指南中的 WinSCP 將檔案傳輸到您的 Linux 執行個體](#)。

- 使用 Amazon EC2 使用者指南中的 [SCP 用戶端將檔案傳輸到 Linux 執行個體](#)。
- [教學課程：在 AL2023 上主持一個 WordPress 部落格](#)

如需本教學所使用的命令和軟體之詳細資訊，請參閱下列網頁：

- Apache Web 伺服器：<http://httpd.apache.org/>
- MariaDB 資料庫伺服器：<https://mariadb.org/>
- PHP 程式設計語言：<http://php.net/>

如需為您的 Web 伺服器註冊網域名稱或將現有網域名稱轉移至此主機的詳細資訊，請參閱 Amazon Route 53 開發人員指南中的 [建立和遷移網域和子網域至 Amazon Route 53](#)。

教學課程：在 AL2023 上設定 SSL/TLS

Secure Sockets Layer/Transport Layer Security (SSL/TLS) 會在 Web 伺服器與 Web 用戶端之間建立加密通路，保護傳輸中的資料以防遭到竊聽。本教學課程說明如何在具有 AL2023 和 Apache 網頁伺服器的 EC2 執行個體上手動新增對 SSL/TLS 的支援。本教學假設您未使用負載平衡器。如果您正在使用 Elastic Load Balancing，您可以選擇使用來自 [AWS Certificate Manager](#) 的憑證設定負載平衡器上的 SSL 卸載。

基於歷史因素，Web 加密通常僅簡單以 SSL 指稱。雖然 Web 瀏覽器仍然支援 SSL，但其後來的通訊協定 TLS 較不易受攻擊。AL2023 預設會停用伺服器端對所有版本的 SSL 支援。[安全標準機構](#) 認為 TLS 1.0 不安全。TLS 1.0 和 TLS 1.1 已在 2021 年 3 月正式棄用。本教學課程包含完全基於啟用 TLS 1.2 的指導。TLS 1.3 已於 2018 年完成，只要支援並啟用基礎 TLS 程式庫 (本教學課程中的 OpenSSL)，就可以在 AL2 中使用。[用戶端必須在 2023 年 6 月 28 日之前支援 TLS 1.2 或更新版本](#)。如需更新之加密標準的詳細資訊，請參閱 [RFC 7568](#) 和 [RFC 8446](#)。

本教學課程會將現代 Web 加密簡稱為 TLS。

Important

這些程序適用於搭配 AL2023 使用。如果您嘗試設定的 EC2 執行個體執行不同的發行版或執行舊版 Amazon Linux 的執行個體，則本教學中的一些程序可能不適用。若為 Ubuntu，請參閱以下 Ubuntu 社群文件：[Open SSL on Ubuntu](#)。對於 Red Hat Enterprise Linux，請參閱以下：[設定 Apache HTTP Web 伺服器](#)。若是其他發行版，請參閱其特定文件。

Note

或者，您可以將 AWS Certificate Manager (ACM) 用於 AWS Nitro Enclave，這是一個隔離區應用程式，可讓您將公有和私有 SSL/TLS 憑證與 Web 應用程式和在具有 Nitro Enclaves 的 Amazon EC2 執行個體上執行的伺服器搭配使用。AWS Nitro Enclaves 是 Amazon EC2 功能，可建立隔離運算環境，保護並安全處理高度敏感資料，例如 SSL/TLS 憑證和私有金鑰。ACM for Nitro Enclaves 可搭配在您 Amazon EC2 Linux 執行個體上執行的 nginx，建立私有金鑰、分配憑證和私有金鑰，以及管理憑證續約。

若要使用 ACM for Nitro Enclaves，您必須使用啟用飛地的 Linux 執行個體。

如需詳細資訊，請參閱[什麼是 AWS 硝基飛地？](#) 以及[硝基飛地 AWS Certificate Manager 用戶指南](#)中的 AWS 硝基飛地。

目錄

- [必要條件](#)
- [步驟 1：在伺服器上啟用 TLS](#)
- [步驟 2：取得 CA 簽署的憑證](#)
- [步驟 3：測試和強化安全組態](#)
- [疑難排解](#)

必要條件

開始本教學之前，請先完成下列步驟：

- 啟動 EBS 支援的 AL2023 執行個體。如需詳細資訊，請參閱 [Amazon EC2 上的 AL2023](#)。
- 設定安全群組允許執行個體接受下列 TCP 連接埠上的連線：
 - SSH (連接埠 22)
 - HTTP (連接埠 80)
 - HTTPS (連接埠 443)

如需詳細資訊，請參閱 Amazon EC2 使用者指南中的[授權 Linux 執行個體的入站流量](#)。

- 安裝 Apache Web 伺服器。如需 step-by-step 指示，請參閱[教學課程：在 AL2023 上安裝燈泡伺服器](#)。只需要 httpd 套件和其相依性，因此您可以忽略包含 PHP 和 MariaDB 的說明。

- 若要識別和驗證網站，TLS 公有金鑰基礎設施 (PKI) 依賴網域名稱系統 (DNS)。若要使用 EC2 執行個體來託管公有網站，您需要註冊 Web 伺服器的網域名稱，或將現有網域名稱傳輸至 Amazon EC2 主機。現有多個第三方網域註冊和 DNS 託管服務可用，或者您可以使用 [Amazon Route 53](#)。

步驟 1：在伺服器上啟用 TLS

此程序會引導您完成使用自我簽署數位憑證在 AL2023 上設定 TLS 的程序。

Note

自簽憑證可用於測試環境，而非生產環境。如果您在網際網路公開自簽憑證，來您網站光顧的訪客將會收到安全警告。

在伺服器上啟用 TLS

- 連線至執行個體，並確認 Apache 正在執行。如需詳細資訊，請參閱 [連線至 AL2023 執行個體](#)。

```
[ec2-user ~]$ sudo systemctl is-enabled httpd
```

如果傳回的值不是 "enabled"，請啟動 Apache，並設定為每次系統開機時都啟動。

```
[ec2-user ~]$ sudo systemctl start httpd && sudo systemctl enable httpd
```

- 為確保所有軟體套件皆為最新版本，請對您的執行個體執行快速軟體更新。本程序可能需費時幾分鐘，但確定您擁有最新的安全更新和錯誤修正至關重要。

Note

-y 選項不要求確認就會安裝更新。如果您要先檢查更新再安裝，則可以略過此選項。

```
[ec2-user ~]$ sudo dnf install openssl mod_ssl
```

- 輸入以下命令後，您將看到一條提示，從中可以輸入有關網站的資訊。

```
[ec2-user ~]$ sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/pki/tls/private/apache-selfsigned.key -out /etc/pki/tls/certs/apache-selfsigned.crt
```

這會在 `apache-selfsigned.crt` 目錄中產生新檔案 `/etc/pki/tls/certs/`。指定的檔案名稱符合 `SSLCertificateFile` 中的 `/etc/httpd/conf.d/ssl.conf` 指示詞所指派的預設值。

您的執行個體現在有下列檔案，供您用來設定安全伺服器和建立用於測試的憑證：

- `/etc/httpd/conf.d/ssl.conf`

`mod_ssl` 的組態檔案。檔案中包含的指示詞會告知 Apache 可在何處找到加密金鑰和憑證、要允許的 TLS 通訊協定版本，以及要接受的加解密碼。這將是您的本機憑證檔案：

- `/etc/pki/tls/certs/apache-selfsigned.crt`

此檔案包含自我簽署憑證和憑證的私有金鑰。Apache 規定憑證和金鑰必須是 PEM 格式，此格式由 "BEGIN" 和 "END" 行所框住的 Base64 編碼 ASCII 字元構成，如下方的憑證縮寫範例所示。

```
-----BEGIN PRIVATE KEY-----
MIIIEvgIBADANBgkqhkiG9w0BAQEFAASCBKgwggSkAgEAAoIBAQD2KKx/8Zk94m1q
3gQMZF9ZN66Ls19+3tHAgQ5Fpo9KJDhzLj00CI8u1PTcGmAah5kEitCEc0wzmNeo
BC10wYR6G0rGaKtK9Dn7CuIjvubtUysVyQoMVPQ971deakHWeRMiEJFXg6kZZ0vr
GvwnKoMh3D1K44D9dX7IDua2P1Yx5+eroA+1Lqf32ZSaA00bBIMIYTHigwbHMZoT
...
56tE7THvH7v0Ef4/iU0sIrEzaMaJ0mqkmY1A70qQGQKBgBF3H1qNRNHuyMcPODFs
27hDzPDinrquSEvoZlIggkDMlh2irTiipJ/GhkvTpoQ1v0fK/VXw8vSgeaBuhwJvS
LXU9HvYq0U604FgD3nAyB9hI0BE13r1HjUvbjT7moH+RhnNz6eeqdsccCS09VtRAo
4QQvAq0a8UheYeoXLdWcHaLP
-----END PRIVATE KEY-----

-----BEGIN CERTIFICATE-----
MIIIEazCCA10gAwIBAgICWxQwDQYJKoZIhvcNAQELBQAwbGExCzAJBgNVBAYTAi0t
MRIwEAYDVQQIDAlTb211U3RhdGUxETAPBgNVBACMFNvbWVkaXR5MRkwFwYDVQQK
DBBtb211T3JnYW5pemF0aW9uMR8wHQYDVQQLDBZTb211T3JnYW5pemF0aW9uYWxv
bml0MRkwFwYDVQDDBBpcC0xNzItMzEtMjMjMjMSQwIgwYJKoZIhvcNAQkBFhVy
...
z5rRUE/XzxRLBZ0oWZpNWTXJkQ3uFYH6s/sBwtHpKKZMz0vDedREjNKAvk4ws6F0
CuIjvubtUysVyQoMVPQ971deakHWeRMiEJFXg6kZZ0vrGvwnKoMh3D1K44D9d1U3
WanXWehT6FiSZvB4sTEXXJN2jdw8g+sHGnZ8zC0sc1knYhHrCVD2vnBlZJKSZvak
3ZazhBxtQSukFM0nWPP2a0DMMFGYUH0d0BQE8sBJxg==
```

```
-----END CERTIFICATE-----
```

檔案名稱和副檔名僅為使用上的方便，不會影響功能。例如，只要 `cert.crt` 檔案中的相關指示詞使用相同的名稱，您就可以呼叫憑證 `cert.pem`、`ssl.conf` 或其他檔案名稱。

Note

當您將預設 TLS 檔案取代為自己的自訂檔案時，請確認自訂檔案為 PEM 格式。

4. 重新啟動 Apache。

```
[ec2-user ~]$ sudo systemctl restart httpd
```

Note

請確定可以在 EC2 執行個體上存取 TCP 連接埠 443，如先前所述。

5. Apache Web 伺服器現在應該支援透過連接埠 443 的 HTTPS (安全 HTTP)。測試方式是將 EC2 執行個體的 IP 地址或完整網域名稱加上字首 `https://` 後，一起輸入到瀏覽器 URL 列。

因為您要使用自簽的不受信任主機憑證連線至網站，所以瀏覽器可能會顯示一系列的安全警告。請覆寫警告，並繼續前往網站。

如果預設 Apache 測試頁面開啟，即表示您已於伺服器順利設定 TLS。現在所有在瀏覽器與伺服器之間傳遞的資料皆會加密。

Note

為了避免網站訪客碰到警告畫面，您必須取得 CA 簽署的受信任憑證，其不僅會加密也可將您公開驗證為網站擁有者。

步驟 2：取得 CA 簽署的憑證

您可以使用下列程序取得 CA 簽署的憑證：

- 從私有金鑰產生憑證簽署請求 (CSR)
- 將 CSR 提交至憑證授權機構 (CA)

- 取得簽署的主機憑證
- 設定 Apache 來使用憑證

在密碼編譯方面，自簽的 TLS X.509 主機憑證與 CA 簽署的憑證完全相同。兩者的差異在於往來的形式，無關數學性質。CA 允諾會至少先驗證網域的所有權，再將憑證發給申請人。每個 Web 瀏覽器皆含有受瀏覽器廠商信任能執行這項操作的 CA 名單。X.509 憑證主要包含對應至私有伺服器金鑰的公有金鑰，以及以密碼編譯方式繫結至公有金鑰的 CA 簽章。瀏覽器透過 HTTPS 連接至 Web 伺服器時，伺服器會呈現憑證給瀏覽器，讓瀏覽器檢查其信任的 CA 名單。如果簽署者位於名單上，或可透過包含其他受信任簽署者的「信任鏈」存取，瀏覽器會與伺服器協議一快速加密資料通路，並載入頁面。

憑證通常因包含驗證請求的勞力而需耗費成本，因此請貨比三家。一些 CA 免費提供基本層級憑證。這些 CA 當中最值得注意的是 [Let's Encrypt](#) 專案，此專案也支援自動化憑證建立和續約程序。如需使用 Let's Encrypt 憑證的詳細資訊，請參閱[取得 Certbot](#)。

如果您打算提供商業級服務，[AWS Certificate Manager](#) 會是不錯的選擇。

主機憑證的基礎就是金鑰。自 2019 年起，[政府](#)和[產業](#)團體建議採用最小金鑰 (模數) 大小為 2048 位元的 RSA 金鑰，用以保護文件至 2030 年。在 AL2023 中 OpenSSL 產生的預設模數大小為 2048 位元，適合用於 CA 簽署的憑證。對於需要自訂金鑰的人員，例如具有較大模數或使用不同加密演算法的金鑰，以下程序提供選用的步驟。

Important

除非您擁有已註冊和託管的 DNS 網域，否則有關取得 CA 簽署的主機憑證的這些指示將不適用。

取得 CA 簽署的憑證

1. 連線至執行個體，並導覽至 `/etc/pki/tls/private/`。這是存放伺服器的 TLS 私有金鑰的目錄。如果您偏好使用現有主機金鑰來產生 CSR，請跳到步驟 3。如需連線至執行個體的詳細資訊，請參閱 [連線至 AL2023 執行個體](#)
2. (選用) 產生新私有金鑰。以下是金鑰組態的一些範例。任何產生的金鑰皆可用於您的 Web 伺服器，但金鑰所實作的安全程度和類型會不同。
 - 範例 1：建立預設 RSA 主機金鑰。產生的檔案 `custom.key` 是 2048 位元 RSA 私有金鑰。

```
[ec2-user ~]$ sudo openssl genrsa -out custom.key
```


- 範例 2：建立具有較大模數的較嚴格 RSA 金鑰。產生的檔案 **custom.key** 是 4096 位元 RSA 私有金鑰。

```
[ec2-user ~]$ sudo openssl genrsa -out custom.key 4096
```

- 範例 3：建立具有密碼保護的 4096 位元加密 RSA 金鑰。產生的檔案 **custom.key** 是以 AES-128 密碼加密的 4096 位元 RSA 私有金鑰。

Important

加密金鑰可提供更好的安全，但因為加密的金鑰需要密碼，所以無法自動啟動與其相依的服務。每次使用此金鑰時，您都必須透過 SSH 連線提供密碼 (在前述範例中為 "abcde12345")。

```
[ec2-user ~]$ sudo openssl genrsa -aes128 -passout pass:abcde12345 -out  
custom.key 4096
```

- 範例 4：使用非 RSA 密碼來加密金鑰。RSA 密碼編譯因為其公有金鑰的大小 (取決於兩個大質數的乘積)，可能相當慢。不過，為 TLS 建立使用非 RSA 密碼的金鑰是有可能的。傳送對等安全層級時，以橢圓曲線數學原理為基礎的金鑰會較小，且運算速度較快。

```
[ec2-user ~]$ sudo openssl ecparam -name prime256v1 -out custom.key -genkey
```

結果是使用 prime256v1 (OpenSSL 支援的一種「具名曲線」) 的 256 位元橢圓曲線私有金鑰。[根據 NIST](#)，其密碼編譯強度略大於 2048 位元 RSA 金鑰。

Note

並非所有 CA 都提供與 RSA 金鑰相同的 elliptic-curve-based 金鑰支援層級。

請確定新的私有金鑰具有高限制的所有權和許可 (擁有者=root、群組=root、僅限擁有者的讀寫權)。命令如下範例所示。

```
[ec2-user ~]$ sudo chown root:root custom.key  
[ec2-user ~]$ sudo chmod 600 custom.key  
[ec2-user ~]$ ls -al custom.key
```


上述命令會產生下列結果。

```
-rw----- root root custom.key
```

在您建立和設定滿意的金鑰之後，即可建立 CSR。

3. 使用偏好的金鑰建立 CSR。以下範例使用 **custom.key**。

```
[ec2-user ~]$ sudo openssl req -new -key custom.key -out csr.pem
```

OpenSSL 會開啟對話方塊，並提示您輸入下表中顯示的資訊。對於基本的已驗證網域之主機憑證，Common Name (通用名稱) 以外的所有欄位皆為選用欄位。

名稱	描述	範例
Country Name (國家/地區名稱)	兩個字母的 ISO 縮寫，用來代表您的國家/地區。	US (=美國)
State or Province Name (州或省名稱)	您組織位在的州名或省名。此名稱不得使用縮寫。	華盛頓州
Locality Name (地區名稱)	您組織的位置 (例如城市)。	西雅圖
Organization Name (組織名稱)	您組織的完整法定名稱。請不要使用您組織名稱的縮寫。	範例公司
Organizational Unit Name (組織單位名稱)	額外組織資訊 (如果有的話)。	範例部門

名稱	描述	範例
Common Name (通用名稱)	此值必須完全符合您預期使用者會在瀏覽器輸入的 web 地址。這通常表示字首為主機名稱或別名的網域名稱，格式為 www.example.com 。在使用自簽憑證且沒有 DNS 解析的測試中，通用名稱可能只包含主機名稱。CA 也提供費用較高的憑證，其可接受萬用字元名稱 (例如 *.example.com)。	www.example.com
電子郵件地址	伺服器管理員的電子郵件地址。	someone@example.com

最後，OpenSSL 會提示您輸入選用的挑戰密碼。此密碼只會套用至 CSR 以及您與 CA 之間的交易，因此請遵循 CA 對於這個和另一個選用欄位 (選用公司名稱) 的建議。CSR 挑戰密碼不會影響伺服器操作。

產生的檔案 **csr.pem** 會包含您的公有金鑰、您公有金鑰的數位簽章，以及您輸入的中繼資料。

- 將 CSR 提交給 CA。這通常包含在文字編輯器開啟 CSR 檔案，以及將內容複製至 Web 表單。此時，系統可能會要求您提供要放在憑證上的一或多個主體別名 (SAN)。如果 **www.example.com** 是通用名稱，則 **example.com** 會是不錯的 SAN，反之亦然。輸入其中任一名稱的網站訪客會看到無錯誤連線。如果您的 CA Web 表單允許這項操作，請在 SAN 清單中包含通用名稱。部分 CA 會自動予以包含。

在核准您的請求之後，您會收到 CA 所簽署的新主機憑證。系統也可能會指示您下載「中繼憑證」檔案，其中包含完成 CA 信任鏈所需的其他憑證。

Note

您的 CA 可能會傳送多種格式的檔案給您，以供不同用途所需。在本教學中，您只應該使用 PEM 格式的憑證檔案，而憑證檔案通常 (但不一定) 會標上 **.pem** 或 **.crt** 副檔名。如果您不確定要使用的檔案，請使用文字編輯器開啟檔案，並尋找包含一或多個以下列這一行開頭之區塊的檔案。

```
- - - - -BEGIN CERTIFICATE - - - - -
```

此檔案的結尾也應該是下列這一行。

```
- - - -END CERTIFICATE - - - - -
```

您也可以在此命令列測試檔案，如下所示。

```
[ec2-user certs]$ openssl x509 -in certificate.crt -text
```

確認檔案中出現這幾行。請不要使用結尾為 .p7b、.p7c 或類似副檔名的檔案。

5. 將新的 CA 簽署憑證和任何中繼憑證放在 /etc/pki/tls/certs 目錄中。

Note

有數種方式可以將新的憑證上傳至 EC2 執行個體，但最直接且有益的方式是在本機電腦和執行個體上開啟文字編輯器 (例如，vi、nano 或記事本)，然後在其間複製和貼上檔案內容。對 EC2 執行個體執行這些操作時，您需要有 root [sudo] 許可。因此，您可以立即看到是否有任何許可或路徑問題。不過，請注意不要在複製內容時新增其他行，或以任何方式變更它們。

從/etc/pki/tls/certs目錄內，檢查檔案擁有權、群組和權限設定是否符合高度限制的AL2023 預設值 (擁有者 = root、群組 = root、僅限擁有者讀取/寫入)。以下範例顯示要使用的命令。

```
[ec2-user certs]$ sudo chown root:root custom.crt  
[ec2-user certs]$ sudo chmod 600 custom.crt  
[ec2-user certs]$ ls -al custom.crt
```

這些命令應該會產生下列結果。

```
-rw----- root root custom.crt
```

中繼憑證檔案的許可較不嚴格 (擁有者=root、群組=root、擁有者可以寫入、群組可以讀取、世界可以讀取)。以下範例顯示要使用的命令。

```
[ec2-user certs]$ sudo chown root:root intermediate.crt  
[ec2-user certs]$ sudo chmod 644 intermediate.crt  
[ec2-user certs]$ ls -al intermediate.crt
```

這些命令應該會產生下列結果。

```
-rw-r--r-- root root intermediate.crt
```

- 將您用來建立 CSR 的私有金鑰放在 `/etc/pki/tls/private/` 目錄中。

Note

有數種方式可以將自訂金鑰上傳至 EC2 執行個體，但最直接且有益的方式是在本機電腦和執行個體上開啟文字編輯器 (例如，vi、nano 或記事本)，然後在其間複製和貼上檔案內容。對 EC2 執行個體執行這些操作時，您需要有 root [sudo] 許可。因此，您可以立即看到是否有任何許可或路徑問題。不過，請注意不要在複製內容時新增其他行，或以任何方式變更它們。

從 `/etc/pki/tls/private` 目錄內部，使用下列指令來驗證檔案擁有權、群組和權限設定是否符合嚴格限制性的 AL2023 預設值 (擁有者 = root、群組 = root、僅供擁有者讀取/寫入)。

```
[ec2-user private]$ sudo chown root:root custom.key
[ec2-user private]$ sudo chmod 600 custom.key
[ec2-user private]$ ls -al custom.key
```

這些命令應該會產生下列結果。

```
-rw----- root root custom.key
```

- 編輯 `/etc/httpd/conf.d/ssl.conf` 以反映新的憑證和金鑰檔案。
 - 在 Apache 的 `SSLCertificateFile` 指示詞中，提供 CA 簽署主機憑證的路徑和檔案名稱：

```
SSLCertificateFile /etc/pki/tls/certs/custom.crt
```

- 如果您收到中繼憑證檔案 (在此範例中為 `intermediate.crt`)，請使用 Apache 的 `SSLCACertificateFile` 指示詞提供其路徑和檔案名稱：

```
SSLCACertificateFile /etc/pki/tls/certs/intermediate.crt
```

Note

部分 CA 會將主機憑證和中繼憑證結合至單一檔案，讓 `SSLCACertificateFile` 指示詞變得非必要。請參閱 CA 所提供的說明。

- c. 在 Apache 的 `custom.key` 指示詞中，提供私有金鑰 (在此範例中為 `SSLCertificateKeyFile`) 的路徑和檔案名稱：

```
SSLCertificateKeyFile /etc/pki/tls/private/custom.key
```

8. 儲存 `/etc/httpd/conf.d/ssl.conf`，並重新啟動 Apache。

```
[ec2-user ~]$ sudo systemctl restart httpd
```

9. 在瀏覽器 URL 列輸入網域名稱 (字首為 `https://`)，以測試伺服器。瀏覽器應該會透過 HTTPS 載入測試頁面，而不會產生錯誤。

步驟 3：測試和強化安全組態

在您的 TLS 運作並向大眾公開之後，您應測試其實際安全程度。這項操作能夠利用線上服務輕鬆完成，例如 [Qualys SSL Labs](#) 可免費為您的安全設定執行透徹的分析。根據結果，您可以決定透過控制所接受的通訊協定、偏好的密碼，以及排除的項目，來強化預設安全組態。如需詳細資訊，請參閱 [how Qualys formulates its scores](#)。

⚠ Important

實際測試對於伺服器安全而言十分重要。微小的組態錯誤可能會導致嚴重安全漏洞和資料遺失。由於建議的安全實務為因應研究與浮現的威脅而持續地變動，所以定期安全稽核是良好伺服器管理的必要項目。

在 [Qualys SSL Labs](#) 網站上，輸入伺服器的完整網域名稱，格式為 `www.example.com`。約兩分鐘之後，您會收到網站的評等 (從 A 到 F)，以及發現之項目的詳細分析。下表摘要說明網域的報告，其設定與 AL2023 上的預設 Apache 組態相同，且具有預設 Certbot 憑證。

整體評分

B

憑證	100%
通訊協定支援	95%
金鑰交換	70%
密碼強度	90%

雖然概觀顯示組態大致上很正確，但詳細報告指出幾個潛在問題，這裡依嚴重程度列出：

x 支援某些較舊的瀏覽器使用 RC4 加密。加密是加密演算法的數學核心。用來加密 TLS 資料串流的快速加密 RC4 已知有數個[嚴重缺點](#)。除非您有絕佳理由來支援舊版瀏覽器，否則應該停用此功能。

x 支援舊版 TLS。此組態支援 TLS 1.0 (已廢除) 和 TLS 1.1 (即將廢除)。自 2018 年起，只建議使用 TLS 1.2。

x 不完全支援前向保密。[前向保密](#)是使用衍生自私有金鑰之暫時 (短暫) 工作階段金鑰來加密的演算法的一項功能。這表示攻擊者實際上無法解密 HTTPS 資料，即使他們擁有 Web 伺服器的長期私有金鑰也是一樣。

更正和打造前瞻性的 TLS 組態

1. 在文字編輯器中開啟組態檔案 `/etc/httpd/conf.d/ssl.conf`，在下一行的開頭輸入 "#"，以變更為註解。

```
#SSLProtocol all -SSLv3
```

2. 新增下列指示詞：

```
#SSLProtocol all -SSLv3  
SSLProtocol -SSLv2 -SSLv3 -TLSv1 -TLSv1.1 +TLSv1.2
```

此指示詞明確停用 SSL 版本 2 和 3，以及 TLS 版本 1.0 和 1.1。對於完全只使用 TLS 1.2 的用戶端，伺服器現在拒絕接受其加密連線。指示詞中的冗長言詞更清楚向讀者表達伺服器的設定用途。

Note

以此種方式停用 TLS 1.0 和 1.1 版，可封鎖小部分的過期 Web 瀏覽器存取您的網站。

修改允許的加密清單

1. 在組態檔案 `/etc/httpd/conf.d/ssl.conf` 中，找出含有 **SSLCipherSuite** 指示詞的區段，在現有一行的開頭輸入 `"#"`，以變更為註解。

```
#SSLCipherSuite HIGH:MEDIUM:!aNULL:!MD5
```

2. 指定明確的加密套件，並指定加密順序將前向保密列為優先，避免不安全的加密。這裡使用的 **SSLCipherSuite** 指示詞是根據 [Mozilla SSL Configuration Generator](#) 的輸出，此工具可針對您伺服器上執行的特定軟體來量身打造 TLS 組態。(如需詳細資訊，請參閱 Mozilla 的實用資源 [Security/Server Side TLS](#)。) 首先，根據下列命令的輸出來確定您的 Apache 和 OpenSSL 版本。

```
[ec2-user ~]$ yum list installed | grep httpd
```

```
[ec2-user ~]$ yum list installed | grep openssl
```

例如，如果傳回的資訊是 Apache 2.4.34 和 OpenSSL 1.0.2，我們會在產生器中輸入此資訊。如果您選擇 "modern" 相容性模型，則會建立 **SSLCipherSuite** 指示詞來強制實施安全性，但仍適用於大多數瀏覽器。如果您的軟體不支援新式組態，您可以更新軟體或改為選擇 "intermediate" 組態。

```
SSLCipherSuite ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-  
ECDSA-CHACHA20-POLY1305:  
ECDHE-RSA-CHACHA20-POLY1305:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-  
SHA256:  
ECDHE-ECDSA-AES256-SHA384:ECDHE-RSA-AES256-SHA384:ECDHE-ECDSA-AES128-SHA256:ECDHE-  
RSA-AES128-SHA256
```

挑選的加密在名稱中包含 ECDHE (Elliptic Curve Diffie-Hellman Ephemeral 的縮寫)。ephemeral 這個字表示前向保密。附帶效果是這些加密不支援 RC4。

建議您使用明確密碼清單，而不是依賴預設值或是內容不可見的簡潔指示詞。

將產生的指示詞複製到 `/etc/httpd/conf.d/ssl.conf` 中。

Note

指示詞複製到 `/etc/httpd/conf.d/ssl.conf` 時必須是單行，而且加密名稱之間只有冒號 (沒有空格)，這裡顯示成多行是為了方便閱讀。

3. 最後，移除下列這一行開頭的 "#"，以取消註解。

```
#SSLHonorCipherOrder on
```

此指示詞會強制伺服器優先選擇排名較高的加密，包括 (在此案例中) 支援前向保密的加密。開啟此指示詞時，伺服器會先嘗試建立嚴密的安全連線，再備援至具有較低安全的允許密碼。

完成這兩道程序後，請儲存 `/etc/httpd/conf.d/ssl.conf` 的變更，並重新啟動 Apache。

如果您在 [Qualys SSL Labs](#) 上重新測試網域，應該會發現 RC4 漏洞和其他警告已消失，而摘要如下所示。

整體評分	A
憑證	100%
通訊協定支援	100%
金鑰交換	90%
密碼強度	90%

每個 OpenSSL 更新都會產生新密碼，並移除舊密碼的支援。保留您的 EC2 AL2023 執行個體 up-to-date、留意 [OpenSSL](#) 的安全性公告，並在技術媒體上留意新安全漏洞的報告。

疑難排解

- 除非我輸入密碼，否則我的 Apache Web 伺服器不會啟動。

如果您已安裝一個加密、密碼受保護的私有伺服器金鑰，這會是預期行為。

您可以移除金鑰的加密和密碼需求。假設您在預設目錄中有稱為 `custom.key` 的私有加密 RSA 金鑰，且其上的密碼為 `abcde12345`，請於 EC2 執行個體上執行下列命令，以產生此金鑰的未加密版本。

```
[ec2-user ~]$ cd /etc/pki/tls/private/  
[ec2-user private]$ sudo cp custom.key custom.key.bak  
[ec2-user private]$ sudo openssl rsa -in custom.key -passin pass:abcde12345 -out  
custom.key.nocrypt
```



```
[ec2-user private]$ sudo mv custom.key.nocrypt custom.key
[ec2-user private]$ sudo chown root:root custom.key
[ec2-user private]$ sudo chmod 600 custom.key
[ec2-user private]$ sudo systemctl restart httpd
```

Apache 現在應該會啟動，而且系統不會提示您輸入密碼。

- 我在執行 `sudo dnf install -y mod_ssl` 時收到錯誤。

在您安裝 SSL 的必要套件時，可能會看到與下列類似的錯誤。

```
Error: httpd24-tools conflicts with httpd-tools-2.2.34-1.16.amzn1.x86_64
Error: httpd24 conflicts with httpd-2.2.34-1.16.amzn1.x86_64
```

這通常表示您的 EC2 執行個體並未執行 AL2023。本教程僅支持從官方 AL2023 AMI 新創建的實例。

教學課程：在 AL2023 上主持一個 WordPress 部落格

下列程序可協助您在 AL2023 執行個體上安裝、設定和保護 WordPress 部落格。本教程是使用 Amazon EC2 的一個很好的介紹，因為您可以完全控制託管 WordPress 博客的 Web 服務器，這在傳統託管服務中並不典型。

您需負責為伺服器更新軟體套件及維護安全性修補程式。對於不需要直接與 Web 伺服器組態互動的更自動化 WordPress 安裝，此 AWS CloudFormation 服務會提供一個 WordPress 範本，讓您快速開始使用。如需詳細資訊，請參閱 AWS CloudFormation 使用者指南中的 [入門](#)。如果您想要在 Windows 執行個體上託管部 WordPress 部落格，請參閱 Amazon EC2 使用者指南中的 [在 Amazon EC2 Windows 執行個體上部署部 WordPress 部落格](#)。如果您需要具有解耦資料庫的高可用性解決方案，請參閱開發人員指 AWS Elastic Beanstalk 南中的 [部署高可用性 WordPress 網站](#)。

Important

這些程序適用於搭配 AL2023 使用。如需其他分發的詳細資訊，請參閱其特定文件。本教學課程中的許多步驟不適用於 Ubuntu 執行個體。如需在 Ubuntu 執行個體 WordPress 上安裝的說明，請參閱 Ubuntu 文件 [WordPress](#) 中的。您也可以使 [CodeDeploy](#) 用在 Amazon Linux，macOS 或 Unix 系統上完成此任務。

主題

- [必要條件](#)
- [安裝 WordPress](#)
- [後續步驟](#)
- [說明! 我的公有 DNS 名稱曾經變更且現在部落格無法使用](#)

必要條件

強烈建議您將彈性 IP 位址 (EIP) 與用來託管 WordPress 部落格的執行個體建立關聯。如此可避免您執行個體的公有 DNS 地址變更及損害安裝。如果您擁有網域名稱，而且您想將該網域名稱用於部落格，您可更新網域名稱的 DNS 記錄，使其指向您的 EIP 地址 (如需這類的協助，請聯絡您的網域名稱註冊商)。您可免費將一個 EIP 地址與執行中的執行個體建立關聯。如需詳細資訊，請參閱《Amazon EC2 使用者指南》中的[彈性 IP 地址](#)。[教學課程：在 AL2023 上安裝燈泡伺服器](#) 教學課程亦提供設定安全群組以允許 HTTP 和 HTTPS 流量的步驟，還有多個步驟可確保您的 web 伺服器已設定正確的檔案許可權限。如需將規則新增至安全性群組的相關資訊，請參閱將[規則新增至安全性群組](#)。

如果您沒有網域名稱可用於部落格，您可向 Route 53 註冊網域名稱，並將執行個體的 EIP 地址與網域名稱建立關聯。如需詳細資訊，請參閱 Amazon Route 53 開發人員指南中的[使用 Amazon Route 53 註冊網域名稱](#)。

安裝 WordPress

Connect 至您的執行個體，然後下載 WordPress 安裝套件。如需連線至執行個體的詳細資訊，請參閱[連線至 AL2023 執行個體](#)。

1. 使用下列命令，下載並安裝這些套件。

```
dnf install wget php-mysqlnd httpd php-fpm php-mysqlcli mariadb105-server php-json
php php-devel -y
```

2. 您可能會注意到輸出中顯示帶有類似措辭的警告 (版本可能會隨時間而有所變化)：

```
WARNING:
  A newer release of "Amazon Linux" is available.

  Available Versions:

dnf update --releasever=2023.0.20230202

  Release notes:
```

```
https://aws.amazon.com
```

```
Version 2023.0.20230204:
```

```
Run the following command to update to 2023.0.20230204:
```

```
dnf update --releasever=2023.0.20230204 ... etc
```

最佳作法是我們建議 up-to-date 盡可能保留作業系統，但是您可能想要逐一查看每個版本，以確保您的環境中沒有衝突。若步驟 1 中記下的先前套件安裝失敗，您可能需要更新至所列出較新版本的其中一個，然後再試一次。

3. 使用 `wget` 指令下載最新的 WordPress 安裝套件。以下命令將一律下載最新版本。

```
[ec2-user ~]$ wget https://wordpress.org/latest.tar.gz
```

4. 解壓縮並解除封存安裝套件。安裝資料夾將解壓縮到名為 `wordpress` 的資料夾。

```
[ec2-user ~]$ tar -xzf latest.tar.gz
```

若要建立 WordPress 安裝的資料庫使用者和資料庫

您的 WordPress 安裝需要將資訊 (例如部落格貼文和使用者註解) 儲存在資料庫中。此程序協助您建立部落格的資料庫，以及有權在該資料庫中讀取和儲存資訊的使用者。

1. 啟動資料庫和 Web 伺服器。

```
[ec2-user ~]$ sudo systemctl start mariadb httpd
```

2. 以 `root` 使用者的身分登入資料庫伺服器。出現提示時，輸入您資料庫的 `root` 密碼；此密碼可能不同於您的 `root` 系統密碼，假如您尚未設定資料庫伺服器的密碼，此密碼可能為空白。

如果您尚未建立資料庫伺服器的保護機制，請務必這麼做。如需詳細資訊，請參閱 [步驟 3：保護資料庫伺服器](#) (AL2023)。

```
[ec2-user ~]$ mysql -u root -p
```

3. 為您的 MySQL 資料庫建立使用者和密碼。您的 WordPress 安裝會使用這些值與 MySQL 資料庫進行通訊。請輸入下列命令，並換成唯一的使用者名稱與密碼。

```
CREATE USER 'wordpress-user'@'localhost' IDENTIFIED BY 'your_strong_password';
```

務必為使用者建立高強度密碼。請勿在密碼中使用單引號字元 (')，因為這會使上述命令中斷。請勿重複使用現有密碼，並將此密碼存放在安全的地方。

4. 建立資料庫。為資料庫提供一個描述性有意義的名稱，例如 `wordpress-db`。

Note

以下命令中資料庫名稱前後的標點符號稱為反引號。標準鍵盤上的反引號 (`) 鍵通常位在 Tab 鍵上方。反引號不一定為必要，但反引號可讓您在資料庫名稱中使用其他的非法字元，例如連字號。

```
CREATE DATABASE `wordpress-db`;
```

5. 將資料庫的完整權限授與您之前建立的 WordPress 使用者。

```
GRANT ALL PRIVILEGES ON `wordpress-db`.* TO "wordpress-user"@"localhost";
```

6. 排清資料庫權限，以套用所有變更。

```
FLUSH PRIVILEGES;
```

7. 離開 `mysql` 用戶端。

```
exit
```

建立及編輯 `wp-config.php` 檔案

WordPress 安裝資料夾包含一個名為的範例規劃檔 `wp-config-sample.php`。在此程序中您將複製該檔案，並依照您的特定組態編輯檔案。

1. 將 `wp-config-sample.php` 檔案複製到稱為 `wp-config.php` 的檔案。此動作將建立一個新的組態檔案，並保留原本的範例檔案做為備份。

```
[ec2-user ~]$ cp wordpress/wp-config-sample.php wordpress/wp-config.php
```

2. 用您喜愛的文字編輯器 (例如 `wp-config.php` 或 `nano`) 編輯 `vim` 檔案，並輸入您的安裝值。如果您沒有喜愛的文字編輯器，`nano` 很適合入門者。

```
[ec2-user ~]$ nano wordpress/wp-config.php
```

- a. 找出定義 DB_NAME 的行，並將 database_name_here 變更為您在 [Step 4](#) 的 [若要建立 WordPress 安裝的資料庫使用者和資料庫](#) 中建立的資料庫名稱。

```
define('DB_NAME', 'wordpress-db');
```

- b. 找出定義 DB_USER 的行，並將 username_here 變更為您在 [Step 3](#) 的 [若要建立 WordPress 安裝的資料庫使用者和資料庫](#) 中建立的資料庫使用者。

```
define('DB_USER', 'wordpress-user');
```

- c. 找出定義 DB_PASSWORD 的行，並將 password_here 變更為您在 [Step 3](#) 的 [若要建立 WordPress 安裝的資料庫使用者和資料庫](#) 中建立的高強度密碼。

```
define('DB_PASSWORD', 'your_strong_password');
```

- d. 找出稱為 Authentication Unique Keys and Salts 的區段。這些 SALT 值 KEY 和值為 WordPress 使用者儲存在其本機電腦上的瀏覽器 Cookie 提供了一層加密。基本上，於此新增的隨機值將使您的網站更安全。請前往 <https://api.wordpress.org/secret-key/1.1/salt/> 隨機產生一組金鑰值，您可將其複製並貼入到 wp-config.php 檔案內。若要將文字貼入到 PuTTY 終端，請將游標移至想要貼入文字的位置，然後在 PuTTY 終端內按一下滑鼠右鍵。

如需有關安全金鑰的詳細資訊，請前往 <https://wordpress.org/support/article/editing-wp-config-php/#security-金鑰>。

Note

下列值僅供範例使用；請勿使用這些值進行安裝。

```
define('AUTH_KEY',          ' #U$$+[RXN8:b^-L 0(WU_+ c+WFkI~c]o)-bHw+)/
Aj[wTwSiZ<Qb[mghEXcRh-');
define('SECURE_AUTH_KEY',  'Zsz._P=l/|y.Lq)XjlkwS1y5NJ76E6EJ.AV0pCKZZB,*~*r ?
60P$eJT@;+(ndLg');
define('LOGGED_IN_KEY',    'ju}qwre3V*+8f_z0Wf?{LlGsQ]Ye@2Jh^,8x>)Y |;(^[Iw]Pi
+LG#A4R?7N`YB3');
```

```
define('NONCE_KEY',      'P(g62HeZxEes|LnI^i=H,[Xwk9I&[2s|:?0N}VJM%?;v2v]v+;
+^9eXUahg@: :Cj');
define('AUTH_SALT',      'C$DpB4Hj[JK:#{qL`sRVa: {:7yShy(9A@5wg+`JJVb1fk%-
Bx*M4(qc[Qg%JT!h)');
define('SECURE_AUTH_SALT', 'd!uRu#}+q#{f$Z?Z9uFPG.$ {+S{n~1M&%@~gL>U>NV<zpD-@2-
Es7Q10-bp28EKv');
define('LOGGED_IN_SALT',  ';j{00P*owZf)kVD+FVLn-~ >.|Y%Ug4#I^*LVd9QeZ^&XmK|
e(76miC+&W&+^0P/');
define('NONCE_SALT',      '-97r*V/cgxLmp?Zy4zUU4r99QQ_rGs2LTd%P;|
_e1tS)8_B/, .6[=UK<J_y9?JWG');
```

- e. 儲存檔案並結束您的文字編輯器。

若要在 Apache 文件根目錄下安裝您的 WordPress 檔案

- 現在您已經解壓縮了安裝資料夾、建立 MySQL 資料庫和使用者，以及自訂 WordPress 組態檔案，您已準備好將安裝檔案複製到 Web 伺服器文件根目錄，以便執行完成安裝的安裝指令碼。這些文件的位置取決於您是否希望您的 WordPress 博客在 Web 服務器的實際根目錄（例如，*my.public.dns.amazonaws.com*）或根目錄下的子目錄或文件夾中（例如，*my.public.dns.amazonaws.com/blog*）。
- 如果 WordPress 要在文檔根目錄下運行，請複製 wordpress 安裝目錄的內容（但不是目錄本身），如下所示：

```
[ec2-user ~]$ cp -r wordpress/* /var/www/html/
```

- 如果您想 WordPress 要在文件根目錄下的替代目錄中執行，請先建立該目錄，然後將檔案複製到該目錄中。在這個例子中，WordPress 將從目錄中運行 blog：

```
[ec2-user ~]$ mkdir /var/www/html/blog
[ec2-user ~]$ cp -r wordpress/* /var/www/html/blog/
```

Important

基於安全起見，如果您不是要立即移至下一個步驟，現在請先停止 Apache web 伺服器 (httpd)。在您將安裝移到 Apache 文件根目錄下之後，WordPress 安裝指令碼不受保護，如果 Apache Web 伺服器正在執行，攻擊者可能會取得您部落格的存取權。若要停止 Apache Web 伺服器，輸入命令 `sudo service httpd stop`。如果您要移至下一個步驟，則不需要停止 Apache web 伺服器。

允許使 WordPress 用永久鏈接

WordPress 永久連結必須使用 Apache `.htaccess` 檔案才能正常運作，但在 Amazon Linux 上預設不會啟用此功能。請使用此程序允許 Apache 文件根下的所有覆寫。

1. 使用您喜愛的文字編輯器 (例如 `httpd.conf` 或 `nano`) 開啟 `vim` 檔案。如果您沒有喜愛的文字編輯器，`nano` 很適合入門者。

```
[ec2-user ~]$ sudo vim /etc/httpd/conf/httpd.conf
```

2. 找出開頭為 `<Directory "/var/www/html">` 的區段。

```
<Directory "/var/www/html">
#
# Possible values for the Options directive are "None", "All",
# or any combination of:
#   Indexes Includes FollowSymLinks SymLinksifOwnerMatch ExecCGI MultiViews
#
# Note that "MultiViews" must be named *explicitly* --- "Options All"
# doesn't give it to you.
#
# The Options directive is both complicated and important. Please see
# http://httpd.apache.org/docs/2.4/mod/core.html#options
# for more information.
#
Options Indexes FollowSymLinks

#
# AllowOverride controls what directives may be placed in .htaccess files.
# It can be "All", "None", or any combination of the keywords:
#   Options FileInfo AuthConfig Limit
#
AllowOverride None

#
# Controls who can get stuff from this server.
#
Require all granted
</Directory>
```

3. 變更上述區段中的 `AllowOverride None` 行，以讀取 `AllowOverride All`。

Note

此檔案中有多個 AllowOverride 行；請確定您變更的是 <Directory "/var/www/html"> 區段中的行。

```
AllowOverride All
```

4. 儲存檔案並結束您的文字編輯器。

若要在 AL2023 上安裝 PHP 圖形繪圖程式庫

適用於 PHP 的 GD 程式庫可讓您修改影像。如果您需要裁剪部落格的標題映像，請安裝此程式庫。您安裝 phpMyAdmin 的版本可能需要此程式庫的特定最低版本 (例如 8.1 版)。

使用以下命令在 AL2023 上安裝 PHP 圖形繪圖庫。例如，如果您從來源安裝 php8.1 作為安裝 LAMP 堆疊的一部分，則此命令會安裝 PHP 圖形繪製庫的 8.1 版。

```
[ec2-user ~]$ sudo dnf install php-gd
```

若要確認已安裝的版本，請使用下列命令：

```
[ec2-user ~]$ sudo dnf list installed | grep php-gd
```

下列為範例輸出：

```
php-gd.x86_64                8.1.30-1.amzn2                @amazonlinux
```

在 Amazon Linux AMI 安裝 PHP 圖形繪製庫

適用於 PHP 的 GD 程式庫可讓您修改影像。如果您需要裁剪部落格的標題映像，請安裝此程式庫。您安裝 phpMyAdmin 的版本可能需要此程式庫的特定最低版本 (例如 8.1 版)。

若要確認哪些版本可用，請使用下列命令：

```
[ec2-user ~]$ dnf list | grep php
```


以下是 PHP 圖形繪製庫 (8.1 版) 輸出的範例行：

```
php8.1.aarch64                                8.1.7-1.amzn2023.0.1
                                                @amazonlinux
php8.1-cli.aarch64                            8.1.7-1.amzn2023.0.1
                                                @amazonlinux
php8.1-common.aarch64                        8.1.7-1.amzn2023.0.1
                                                @amazonlinux
php8.1-devel.aarch64                          8.1.7-1.amzn2023.0.1
                                                @amazonlinux
php8.1-fpm.aarch64                            8.1.7-1.amzn2023.0.1
                                                @amazonlinux
php8.1-gd.aarch64                             8.1.7-1.amzn2023.0.1
                                                @amazonlinux
```

使用下列命令，在 Amazon Linux AMI 上安裝特定版本的 PHP 圖形繪製庫 (例如，php8.1 版)：

```
[ec2-user ~]$ sudo dnf install -y php8.1-gd
```

修正 Apache web 伺服器的檔案權限

中的某些可用功能 WordPress 需要對 Apache 文件根目錄的寫入權限 (例如透過「管理」畫面上傳媒體)。如果您尚未這樣做，請先套用下列的群組成員資格與許可 (詳細資訊詳述於 [LAMP web 伺服器教學課程](#))。

1. 將 `/var/www` 及其內容的檔案所有權授予 apache 使用者。

```
[ec2-user ~]$ sudo chown -R apache /var/www
```

2. 將 `/var/www` 及其內容的群組所有權授予 apache 群組。

```
[ec2-user ~]$ sudo chgrp -R apache /var/www
```

3. 變更 `/var/www` 及其子目錄的目錄許可，以新增群組寫入許可並設定日後子目錄的群組 ID。

```
[ec2-user ~]$ sudo chmod 2775 /var/www
[ec2-user ~]$ find /var/www -type d -exec sudo chmod 2775 {} \;
```

4. 請以遞迴方式變更 `/var/www` 及其子目錄的檔案許可。

```
[ec2-user ~]$ find /var/www -type f -exec sudo chmod 0644 {} \;
```

Note

如果您打算還將其用 WordPress 作 FTP 服務器，則需要在此處進行更多寬鬆的「組」設置。請檢閱[中的建議步驟和安全性設定](#)，WordPress 以完成此操作。

5. 重新啟動 Apache web 伺服器，以套用新的群組與許可。

```
[ec2-user ~]$ sudo systemctl restart httpd
```

若要使用 AL2023 執行 WordPress 安裝指令碼

您已準備好安裝 WordPress。使用的指令取決於作業系統。此程序中的指令可與 AL2023 搭配使用。使用 AL2023 AMI 後面的程序。

1. 使用 systemctl 命令，確定每次系統開機時 httpd 和資料庫服務都會啟動。

```
[ec2-user ~]$ sudo systemctl enable httpd && sudo systemctl enable mariadb
```

2. 確認資料庫伺服器正在執行。

```
[ec2-user ~]$ sudo systemctl status mariadb
```

如果資料庫服務未執行，請啟動服務。

```
[ec2-user ~]$ sudo systemctl start mariadb
```

3. 確認 Apache web 伺服器 (httpd) 正在執行。

```
[ec2-user ~]$ sudo systemctl status httpd
```

如果 httpd 服務未執行，請啟動服務。

```
[ec2-user ~]$ sudo systemctl start httpd
```

4. 在網頁瀏覽器中，輸入 WordPress 部落格的 URL (執行個體的公用 DNS 位址，或該位址後面接著 `blog料夾`)。您應該會看到 WordPress 安裝指令碼。提供 WordPress 安裝所需的資訊。選擇 `[安裝] WordPress` 以完成安裝。如需詳細資訊，請參閱[步驟 5：在 WordPress 網站上執行安裝指令碼](#)。

若要使用 AL2023 AMI 執行 WordPress 安裝指令碼

1. 使用 `chkconfig` 命令，確定每次系統開機時 `httpd` 和資料庫服務都會啟動。

```
[ec2-user ~]$ sudo chkconfig httpd on && sudo chkconfig mariadb on
```

2. 確認資料庫伺服器正在執行。

```
[ec2-user ~]$ sudo service mariadb status
```

如果資料庫服務未執行，請啟動服務。

```
[ec2-user ~]$ sudo service mariadb start
```

3. 確認 Apache web 伺服器 (`httpd`) 正在執行。

```
[ec2-user ~]$ sudo service httpd status
```

如果 `httpd` 服務未執行，請啟動服務。

```
[ec2-user ~]$ sudo service httpd start
```

4. 在網頁瀏覽器中，輸入 WordPress 部落格的 URL (執行個體的公用 DNS 位址，或該位址後面接著 `blog` 資料夾)。您應該會看到 WordPress 安裝指令碼。提供 WordPress 安裝所需的資訊。選擇 [安裝] WordPress 以完成安裝。如需詳細資訊，請參閱 [步驟 5：在 WordPress 網站上執行安裝指令碼](#)。

後續步驟

測試完 WordPress 博客後，請考慮更新其配置。

使用自訂的網域名稱

如果您有網域名稱與 EC2 執行個體的 EIP 地址相關聯，您可設定部落格使用該名稱，而不是使用 EC2 公有 DNS 地址。如需詳細資訊，請參閱 [變更網站上的 WordPress 網站 URL](#)。

設定部落格

您可設定部落格使用不同的 [主題](#) 和 [外掛程式](#)，為讀者提供更為個人化的使用體驗。但安裝程序有時會出錯，使您失去整個部落格。因此我們強烈建議您為執行個體建立備份的 Amazon Machine Image

(AMI)，然後再嘗試安裝任何主題或外掛程式，如此安裝期間發生任何錯誤時便能還原部落格。如需詳細資訊，請參閱 Amazon EC2 使用者指南中的[建立您自己的 AMI](#)。

增加容量

如果您的 WordPress 部落格越來越受歡迎，而您需要更多的運算能力或儲存空間，請考慮下列步驟：

- 擴展執行個體的儲存空間。如需詳細資訊，請參閱 [Amazon EBS 彈性磁碟區](#)。
- 將 MySQL 資料庫移往 [Amazon RDS](#)，善加運用該服務可輕鬆擴展的能力。

改善網際網路流量的網路效能

如果您希望您的部落格能夠提高來自世界各地使用者的流量，請考慮 [AWS Global Accelerator](#)。Global Accelerator 可改善使用者用戶端裝置與執行 WordPress 應用程式之間的網際網路流量效能，協助您降低延遲 AWS。Global Accelerator 會使用 [AWS 全域網路](#)，將流量導向至距離用戶端最近的 AWS 區域中運作良好的應用程式端點。

進一步了解 WordPress

以下鏈接包含有關的更多信息 WordPress。

- [有關詳情 WordPress](#)，請參閱 [食品 WordPress 法典委員會的幫助文檔](#)。
- 如需有關疑難排解安裝的詳細資訊，請移至 [常見的安裝程式問題](#)。
- 如需讓 WordPress 部落格更安全的相關資訊，請前往 [強化 WordPress](#)。
- 如需保留 WordPress 部落格的相關資訊 up-to-date，請前往 [更新 WordPress](#)。

說明! 我的公有 DNS 名稱曾經變更且現在部落格無法使用

您的 WordPress 安裝會使用 EC2 執行個體的公有 DNS 地址自動設定。如果您停止並重新啟動執行個體，公有 DNS 地址便會變更 (除非與彈性 IP 地址建立關聯)，而且部落格將無法再使用，因為其參考資源的地址已不存在 (或已指派至其他的 EC2 執行個體)。更詳細的問題描述以及幾種可能的解決方案概述在 <https://wordpress.org/support/article/changing-the-site-url/>。

如果您的 WordPress 安裝發生了這種情況，您可以使用以下程序恢復博客，該程序使用 wp-cli 命令行界面 WordPress。

若要變更您的 WordPress 網站 URL wp-cli

1. 使用 SSH 連結至您的 EC2 執行個體。

2. 記下執行個體的舊網站 URL 和新網站 URL。安裝時，舊網站 URL 可能是 EC2 執行個體的公有 DNS 名稱 WordPress。新網站 URL 為 EC2 執行個體目前的公有 DNS 名稱。如果您不確定舊網站的 URL，請用 curl 使用下列命令來尋找 URL。

```
[ec2-user ~]$ curl localhost | grep wp-content
```

輸出應該會顯示舊公有 DNS 名稱的參考，其看起來類似 (舊網站的 URL 為紅色)：

```
<script type='text/javascript' src='http://ec2-52-8-139-223.us-west-1.compute.amazonaws.com/wp-content/themes/twentyfifteen/js/functions.js?ver=20150330'></script>
```

3. 使用下列命令下載 wp-cli。

```
[ec2-user ~]$ curl -O https://raw.githubusercontent.com/wp-cli/builds/gh-pages/phar/wp-cli.phar
```

4. 使用下列命令搜尋並取代 WordPress 安裝中的舊網站 URL。將舊網站和新網站 URL 替換為 EC2 執行個體，以及 WordPress 安裝路徑 (通常為 /var/www/html 或 /var/www/html/blog)。

```
[ec2-user ~]$ php wp-cli.phar search-replace 'old_site_url' 'new_site_url' --path=/path/to/wordpress/installation --skip-columns=guid
```

5. 在網頁瀏覽器中，輸入 WordPress 部落格的新網站 URL，以確認網站再次正常運作。如果不是，請參閱 [變更網站 URL](#) 和 [常見安裝問題](#) 以取得詳細資訊。

在 Amazon EC2 之外使用 Amazon Linux 2023

Amazon Linux 2023 容器映像可以在相容的容器執行期環境中執行。如需如何在容器內使用 Amazon Linux 2023 的詳細資訊，請參閱 [容器中的 AL2023](#)。

Amazon Linux 2023 (AL2023) 也可以作為虛擬化客體執行，而不是直接在 Amazon EC2 上執行。目前有 KVM (qcow2)、VMware (OVA) 和超 V (vhdx) 映像可供使用。

Note

Amazon Linux 2023 映像的組態與 Amazon Linux 2 不同。
如果您是從將 [Amazon Linux 2 作為內部部署的虛擬機器執行](#) 的環境轉換，則需要調整您的組態以與 AL2023 相容。

下載 Amazon 2023 映像檔以搭配 KVM、VMware 和超 V 使用

[Amazon 2023 磁碟映像檔可與 KVM、VMware 和超 V 搭配使用，可從網站下載。](#)

支援的 Amazon Linux 2023 組態，可用於非 Amazon EC2 虛擬化環境

本節涵蓋在非 Amazon EC2 虛擬化環境 (例如在 KVM、VMware 或超 V) 中執行亞馬遜 Linux 2023 的需求。

此基礎 [AL2023 系統需求](#) 適用於所有非 Amazon EC2 虛擬化環境。下列主題詳細說明每個 Hypervisor 環境的支援裝置型號清單。

KVM、VMware 和 Hyper-V 提供許多組態選項，並且需要小心設定這些選項以符合您的安全性、效能和可靠性需求。如需詳細資訊，請查看 Hypervisor 提供的說明文件。

主題

- [在 KVM 上執行 AL2023 的需求](#)
- [在上執行 AL2023 的需求 VMware](#)
- [在超 V 上運行 Amazon Linux 2023 的要求](#)

在 KVM 上執行 AL2023 的需求

本節說明在 KVM 上執行 AL2023 的需求。AL2023 的 KVM 映像可同時用於 aarch64 和 x86-64 架構。這些需求是 KVM 映像檔[AL2023 系統需求](#)的基礎之外。

主題

- [在 KVM 上執行 AL2023 的 KVM 主機需求](#)
- [在 KVM 上支援 AL2023 的裝置](#)
- [在 KVM 上支援 AL2023 的開機模式 \(UEFI和BIOS\)](#)
- [在 KVM 上執行 AL2023 的限制](#)

在 KVM 上執行 AL2023 的 KVM 主機需求

KVM 映像檔目前在執行 Ubuntu 22.04.3 LTS 版本 (由此 Ubuntu qemu 版本6.2+dfsg-2ubuntu6.15提供) 的主機上符合資格，該主機使用的q35機器類型x86-64和virt機器類型。aarch64

在 KVM 上支援 AL2023 的裝置

經測試可與 AL2023 KVM 映像 (**aarch64** 和 **x86-64** 皆可) 搭配使用的 **qemu** 裝置型號：

- virtio-blk (virtio 區塊型儲存裝置)
- virtio-scsi (有磁碟裝置的 virtio SCSI 控制器)
- virtio-net (virtio 網路裝置)
- ahci (與虛擬光碟機搭配使用)
- usb-storage (透過 xhci)

在 AL2023 KVM 影像認證中啟用的其他**qemu**裝置型號，但沒有大量行使如下：

- VGA (qemu VGA)，只在 x86-64 上
- virtio-rng (虛擬隨機數產生器)
- 舊式 AT 鍵盤和 PS/2 滑鼠裝置
- 舊版序列裝置

在 KVM 上支援 AL2023 的開機模式 (UEFI和BIOS)

x86-64 映像會同時使用舊版 BIOS 和 UEFI 開機模式進行測試。aarch64 映像會以 UEFI 開機模式進行測試。

Note

根據預設，使用UEFI開機模式時，某些虛擬機器管理員會以啟用安全開機的 Microsoft 安全開機金鑰佈建虛擬機器。此組態將無法啟動 AL2023。

由於 AL2023 開機載入程式並非由 Microsoft 簽署，因此必須在沒有 UEFI 金鑰的情況下佈建虛擬機器，或使用 AL2023 金鑰進行安全開機佈建。

Important

KVM映像檔的安全開機支援尚未驗證。

在 KVM 上執行 AL2023 的限制

在 KVM 上執行 AL2023 時，有一些已知的限制。

Note

實作某些列出的不受支援功能的程式碼可能存在於 AL2023 中，而且運作正常。存在不受支援的功能清單，因此您可以針對目前的工作依賴項目做出明智的決策，以及 Amazon Linux 團隊在 future 更新中將有資格作為哪些工作資格。

在 KVM 上執行 AL2023 的已知限制

- KVM 客體代理程式目前未封裝或支援。
- 不支援熱插拔 CPU、記憶體或任何其他裝置類型。
- 不支援虛擬機器休眠。
- 不支援虛擬機器移轉。
- 不支援任何裝置 (例如透過 PCI 直通或 USB 直通) 的傳遞。

在上執行 AL2023 的需求 VMware

本節說明在上執行 AL2023 的需求VMware。AL2023 的VMware映像檔僅適用於x86-64架構。VMware的影像aarch64不可用或不受支援。這些要求是除了VMware影像[AL2023 系統需求](#)的基礎之外。

主題

- [VMware在上執行 AL2023 的主機需求 VMware](#)
- [開啟 AL2023 的裝置支援 VMware](#)
- [開啟 AL2023 的開機模式 \(UEFI和BIOS\) 支援 VMware](#)
- [在上執行 AL2023 的限制 VMware](#)

VMware在上執行 AL2023 的主機需求 VMware

AL2023 VMware OVA 影像目前符合下列條件：

- VMware工作站 17.5.0 在使用英特爾 (R) 至強 (R) 白金 8124M 處理器的主機上運行
- VMware使用英特爾 (R) 至強 (R) 白金 8275CL 處理器的 vSphere 8.0

AL2023 O VMware VA 影像會指定機器硬體版本為 13。

VMware機器硬體版本 13 受以下支援：

- ESXi 6.5 或更新版本
- VMware工作站 14 或更新版本

開啟 AL2023 的裝置支援 VMware

下列VMware裝置型號已測試可與 AL2023 VMware OVA 影像搭配使用 (**x86-64**僅限)：

- vmw_pvscsi(VMware半虛擬化SCSI控制器)
- vmxnet3(VMware半虛擬化網路裝置)
- ata_piix (僅限與虛擬光碟機搭配的舊式 IDE)

在 AL2023 VMware 影像認證中啟用的其他VMware裝置型號，但沒有大量行使：

- vmw_vmci和相關vsock介面 (VMware客體代理程式的虛擬通訊端傳輸)
- vmw_balloon 記憶氣球裝置
- VMwareSVGA控制器
- 舊式 AT 鍵盤和 PS/2 滑鼠裝置

根據預設，VMware客體代理程式套件 (open-vm-tools) 可用並安裝在 AL2023 VMware OVA 映像中。

開啟 AL2023 的開機模式 (UEFI和BIOS) 支援 VMware

從 2023.3.20231211 發行版本開始，AL2023 VMware OVA 映像檔已在舊版和開機模式下進行驗證。BIOS UEFIOVA 預設組態仍然是舊版，BIOS但使用者可以變更。

Important

需要安全開機支援UEFI，但尚未在上執行的 AL2023 驗證VMware。

在上執行 AL2023 的限制 VMware

在上執行 AL2023 時存在一些已知的限制VMware。

Note

實作某些列出的非支援功能的程式碼可能存在於 AL2023 中，而且運作正常。存在不支援的功能清單，因此客戶可以根據目前的工作需求做出明智的決策，以及 Amazon Linux 團隊會在未來更新中認證哪些工作符合資格。

在上執行 AL2023 的已知限制 VMware

- UEFI安全開機目前未在啟用 AL2023 的情況下進行驗證VMware。
- 不支援熱插拔 CPU、記憶體或任何其他裝置類型。
- 不支援虛擬機器休眠。
- 不支援虛擬機器移轉。
- 不支援任何裝置 (例如透過 PCI 直通或 USB 直通) 的傳遞。

在超 V 上運行 Amazon Linux 2023 的要求

本節介紹了在超 V 上運行 Amazon Linux 2023 的要求。AL2023 的超 V 映像檔僅適用於 x86-64 架構。的 Hyper-V 映像 aarch64 目前無法使用或不受支援。

本節涵蓋 Hyper-V 映像基礎 [AL2023 系統需求](#) 之上的其他需求。

主題

- [在超 V 上執行 Amazon Linux 2023 的超 V 主機需求](#)
- [超 V 上 Amazon Linux 2023 的設備支持](#)
- [在超 V 上運行 Amazon 2023 的限制](#)

在超 V 上執行 Amazon Linux 2023 的超 V 主機需求

超 V 上的 Amazon Linux 2023 的主要資格發生在 EC2 實例 c5.metal 上運行的視窗服務器 2022 上。

超 V 上 Amazon Linux 2023 的設備支持

Amazon Linux 2023 已在第 1 代和第 2 代 Hyper-V 虛擬機器上使用以下虛擬化硬體組進行測試：

- 第 1 代 (舊版 BIOS 開機) 虛擬機器
- 第 2 代 (UEFI 開機-無安全開機) 虛擬機器
- 下列裝置型號已經過測試，可與 AL2023 超 V 影像搭配使用：
 - hv_storvsc 適用於根磁碟和第 2 代虛擬機器上模擬光碟機的 Hyper-V 虛擬儲存裝置
 - 第 1 代虛擬機器上虛擬光碟機的模擬 PIIX IDE ata_piix
 - 超 V 虛擬乙太網路 hv_netvsc
- 以下設備型號已啟用，但經過輕微測試：
 - 第 1 代虛擬機器上的舊版 VGA 文字模式
 - 第 2 代虛擬機器 simpledrmfb 上以 UEFI 韌體為基礎的幀緩衝區
 - 超 V 氣球 hv_balloon
 - 超 V 氣球 hv_balloon
 - 超 V 隱藏/滑鼠 hid_hyperv
- AL2023 目前尚未啟用下列裝置模式：
 - 超 V 型 PCI 直通
 - 超 V DRM 顯示卡

Important

對於第 2 代虛擬機器，不支援安全開機，且必須在啟動虛擬機器之前停用，才能成功啟動 Amazon Linux 2023。Hyper-V 目前僅支援使用由微軟自己金鑰簽署的軟體元件的安全開機，而 Amazon Linux 開機載入程式則由 Amazon 私密金鑰簽署。Hyper-V 目前不支援匯入第三方金鑰。

在超 V 上運行 Amazon 2023 的限制

以下是在超 V 上執行 Amazon 2023 的一些已知限制：

Note

實作某些列出的非支援功能的程式碼可能存在於 AL2023 中，而且運作正常。存在不支援的功能清單，因此客戶可以根據目前的工作需求做出明智的決策，以及 Amazon Linux 團隊會在未來更新中認證哪些工作符合資格。

在超 V 上執行 AL2023 的已知限制

- 目前不支援 UEFI 安全開機模式，也不適用於超 V 上的 AL2023
- 不支援熱插拔 CPU、記憶體或任何其他裝置類型。
- 不支援虛擬機器 (VM) 休眠。
- 不支援虛擬機器 (VM) 遷移。
- 不支援任何裝置 (例如透過 PCI 直通或 USB 直通) 的傳遞。

Amazon Linux 2023 在 Amazon EC2 外部使用時的設定和 **cloud-init** 組態

本節說明如何在不直接在 Amazon EC2 上執行時 (例如在 KVM、VMware 或 Hyper-V 上執行) 時設定和設定亞馬遜 Linux 2023 虛擬機器。

預設情況下，Amazon Linux 2023 虛擬機器映像不會隨任何使用者密碼或 SSH 金鑰佈建，而是在第一個探索到的網路界面上透過 DHCP 取得網路組態。這表示在預設情況下，如果沒有其他設定，就無法連線到產生的虛擬機器。

因此，需要將某種形式的組態提供給虛擬機器。針對 Amazon Linux 執行此操作的標準機制是透過 `cloud-init` 資料來源進行。

Amazon Linux 2023 已通過以下資料來源的資格：

NoCloud

這是透過虛擬光碟機設定內部部署映像的傳統方法，其中包含具有 `cloud-init` 組態檔案的種子 ISO9660 映像。

VMware

此外，Amazon Linux 2023 還支援透過 VMware `guestinfo.userdata` 和 `guestinfo.metadata` 的 VMware 特定資料來源設定在 vSphere 上執行的 VMware 映像。

Note

資料來源的組態可能與 Amazon Linux 2 不同。更具體來說，Amazon Linux 2023 會將 `systemd-networkd` 作為組態，且需要使用 [cloud-init 網路組態說明文件](#) 所述的「Networking Config Version 2」。

如需 Amazon Linux 2023 中 `cloud-init` 封裝版本 `cloud-init` 組態機制的完整文件，請參閱[上游 cloud-init 文件](#)。

NoCloud (**seed.iso**) KVM 和 VMware 上 Amazon 2023 的 **cloud-init** 組態

本節介紹如何建立和使用 `seed.iso` 映像檔來設定在 KVM 或 VMware 上執行的 Amazon Linux 2023。由 KVM 於 VMware 環境中沒有 [Amazon EC2 執行個體中繼資料服務 \(IMDS\)](#)，因此需要另一種設定 Amazon Linux 2023 的方法，而提供 `seed.iso` 映像檔就是其中一種方法。

`seed.iso` 開機映像包括虛擬機器開機和設定所需的初始組態資訊，例如網路組態、主機名稱和使用者資料。

Note

`seed.iso` 開機映像僅包括 VM 開機所需的組態資訊。不包括 Amazon Linux 2023 作業系統檔案。

若要產生 `seed.iso` 映像，您至少需要兩個組態檔案，有時需要三個：

meta-data

此檔案通常會包含虛擬機器的主機名稱。

user-data

此檔案通常會設定使用者帳戶，並指定其密碼、ssh 金鑰對和/或存取機制。預設情況下，Amazon Linux 2023 KVM 映像會建立 `ec2-user` 使用者帳戶。您可以使用 `user-data` 組態檔案來設定此預設使用者帳戶的密碼和/或 SSH 金鑰。

network-config (選用)

此檔案通常會為虛擬機器提供網路組態，該設定將覆寫預設的組態。預設組態是在第一個可用的網路介面上使用 DHCP。

建立 `seed.iso` 磁碟映像

1. 在 Linux 或 macOS 電腦上，建立名為 `seedconfig` 的新資料夾並導覽到該資料夾。

Note

Windows 或其他作業系統也可以用來完成這些步驟，但您必須找到與 `mkisofs` 等效的工具，才能完成建立 `seed.iso` 映像。

2. 建立 `meta-data` 組態檔案。
 - a. 建立名為 `meta-data` 的新檔案。
 - b. 使用您偏好的編輯器開啟 `meta-data` 檔案、新增下列命令，然後將 `vm-hostname` 替換成虛擬機器的主機名稱：

```
local-hostname: vm-hostname
```

- c. 儲存並關閉 `meta-data` 組態檔案。
3. 建立 `user-data` 組態檔案。
 - a. 建立名為 `user-data` 的新檔案。
 - b. 使用您偏好的編輯器開啟 `user-data` 檔案，然後新增下列項目，以及視需要進行取代：

```
#cloud-config
#vim:syntax=yaml
users:
# A user by the name 'ec2-user' is created in the image by default.
- default
- name: ec2-user
ssh_authorized_keys:
- ssh-rsa ssh-key
# In the above line, replace ssh key with the content of your ssh public key.
```

- c. 您可以選擇性地將更多使用者帳戶新增至user-data組態檔案。

您也可以指定其他使用者帳戶、其存取機制、密碼和金鑰對。如需支援指令的詳細資訊，請參閱[上游 cloud-init 文件](#)。

- d. 儲存並關閉 user-data 組態檔案。

4. (選擇性) 建立 network-config 組態檔

- a. 建立名為 network-config 的新檔案。

- b. 使用您偏好的編輯器開啟 network-config 檔案、新增下列命令，然後將各種 IP 地址替換成適合您設定的 IP 地址。

```
version: 2
ethernets:
  enp1s0:
    addresses:
      - 192.168.122.161/24
    gateway4: 192.168.122.1
    nameservers:
      addresses: 192.168.122.1
```

Note

cloud-init 網路組態提供與介面 MAC 位址相符的機制，而不是指定可根據 VM 組態變更的介面名稱。如需此 (及其他) cloud-init 網路設定功能的詳細資訊，請參閱[上游 cloud-init Network Config Version 2 說明文件](#)。

- c. 儲存並關閉 network-config 組態檔案。

5. 使用在先前步驟中建立的 meta-data、user-data 和選用的 network-config 組態檔，以建立 seed.iso 磁碟映像。

根據您建立 seed.iso 磁碟映像的作業系統，執行下列其中一個動作。

- 在 Linux 系統上，使用 **mkisofs** 或 **genisoimage** 等的工具來建立完成的 seed.iso 檔案。導覽至 seedconfig 資料夾並執行下列命令：

```
$ mkisofs -output seed.iso -volid cidata -joliet -rock user-data meta-data
```

- 如果您使用 network-config，請將其包含在 **mkisofs** 的調用中：

```
$ mkisofs -output seed.iso -volid cidata -joliet -rock user-data meta-data  
network-config
```

- 在 macOS 系統上，您可以使用 **hdiutil** 等的工具來產生完成的 seed.iso 檔案。由於 **hdiutil** 採用路徑名稱而非檔案清單，因此無論是否建立 network-config 組態檔案，都可以使用相同的調用。

```
$ hdiutil makehybrid -o seed.iso -hfs -joliet -iso -default-volume-name cidata  
seedconfig/
```

6. 產生的 seed.iso 檔案現在可以透過虛擬光碟機附加到新的 Amazon Linux 2023 虛擬機器，讓 cloud-init 能在第一次開機時找到，並將組態套用至系統。

VMware對於 AL2023 的訪客資訊cloud-init組態 VMware

VMware環境沒有 [Amazon EC2 執行個體中繼資料服務 \(IMDS\)](#)，因此需要另一種設定 AL2023 的方法。本節說明如何對 VMware vSphere 中可用的 seed.iso 虛擬光碟機使用替代組態機制。

這種配置方法使用 VMwareextraconfig 機制將配置數據提供給 cloud-init。對於以下每個鍵，必須提供對應的 **keyname.encoding** 屬性。

可以向 VMwareextraconfig 機制提供以下鍵。

guestinfo.metadata

JSON 或包含 cloud-init 中繼資料的 YAML

guestinfo.userdata

包含 cloud-config 格式 cloud-init 使用者資料的 YAML 文件。

guestinfo.vendordata (選用)

YAML 包含 cloud-init 供應商資料

對應的編碼屬性 (guestinfo.metadata.encoding、guestinfo.userdata.encoding 和 guestinfo.vendordata.encoding) 可以包含：

base64

屬性的內容已 base64 編碼。

gzip+base64

base64 編碼後，會以 gzip 壓縮屬性的內容。

Note

該 seed.iso 方法支持單獨的 (可選) network-config 配置文件。VMware guestinfo 不同於提供網路組態的方式。下一節將提供其他資訊。

如果需要明確的網路組態，應該以兩個 YAML 或 JSON 屬性的形式嵌入 metadata：

network

包含 JSON 或 YAML 格式的編碼網路設定。

network.encoding

包含上述網路組態資料的編碼。cloud-init 支援的編碼與 guestinfo 資料的編碼相同：base64 和 gzip+base64。

Example 使用 VMware vSphere **govc** CLI 工具將組態傳遞給 **guestinfo**

1. 如中所述 meta-data/user-data，準備、和選用的 network-config 組態檔案 [NoCloud \(seed.iso\)](#) KVM 和 VMware 上 Amazon 2023 的 cloud-init 組態。
2. 將配置文件轉換為可用的格式 VMware guestinfo。

```
# 'meta-data', `user-data` and `network-config` are the configuration
# files in the same format that would be used by a NoCloud (seed.iso)
# data source, read-them and convert them to VMware guestinfo
```

```
#
# The VM_NAME variable is assumed to be set to the name of the VM
# It is assumed that the necessary govc environment (credentials etc...) are
  already set

metadata=$(cat "meta-data")
userdata=$(cat "user-data")
if [ -e "network-config" ] ; then
  # We need to embed the network config inside the meta-data
  netconf=$(base64 -w0 "network-config")
  metadata=$(printf "%s\nnetwork: %s\nnetwork.encoding: base64" "$metadata"
"$netconf")
fi
metadata=$(base64 -w0 <<< "$metadata")
govc vm.change -vm "$VM_NAME" \
  -e guestinfo.metadata="$metadata" \
  -e guestinfo.metadata.encoding="base64"
userdata=$(base64 -w0 <<< "$userdata")
govc vm.change -vm "$VM_NAME" \
  -e guestinfo.userdata="$userdata" \
  -e guestinfo.userdata.encoding="base64"
```

比較安裝在 Amazon 2023 標準 AMI 上的套件與 AL2023 KVM 映像檔

AL2023 標準 AMI 上存在的 RPM 與 AL2023 KVM 映像上存在的 RPM 進行比較。

套件	AMI	KVM
acl	2.3.1	2.3.1
acpid	2.0.32	
alternatives	1.15	1.15
amazon-chrony-config	4.3	
amazon-ec2-net-utils	2.4.1	
amazon-linux-onprem		1.2

套件	AMI	KVM
amazon-linux-repo-cdn		2023.4.20240513
amazon-linux-repo-s3	2023.4.20240513	
amazon-linux-sb-keys	2023.1	2023.1
amazon-onprem-network		1.2
amazon-rpm-config	228	228
amazon-ssm-agent	3.3.380.0	3.3.380.0
at	3.1.23	3.1.23
attr	2.5.1	2.5.1
audit	3.0.6	3.0.6
audit-libs	3.0.6	3.0.6
aws-cfn-bootstrap	2.0	
awscli-2	2.15.30	2.15.30
basesystem	11	11
bash	5.2.15	5.2.15
bash-completion	2.11	2.11
bc	1.07.1	1.07.1
bind-libs	9.16.48	9.16.48
bind-license	9.16.48	9.16.48
bind-utils	9.16.48	9.16.48

套件	AMI	KVM
binutils	2.39	2.39
boost-filesystem	1.75.0	1.75.0
boost-system	1.75.0	1.75.0
boost-thread	1.75.0	1.75.0
bzip2	1.0.8	1.0.8
bzip2-libs	1.0.8	1.0.8
ca-certificates	2023.2.64	2023.2.64
c-ares	1.19.0	
checkpolicy	3.4	3.4
chkconfig	1.15	1.15
chrony	4.3	4.3
cloud-init	22.2.2	22.2.2
cloud-init-cfg-ec2	22.2.2	
cloud-init-cfg-onpre m		22.2.2
cloud-utils-growpart	0.31	0.31
coreutils	8.32	8.32
coreutils-common	8.32	8.32
cpio	2.13	2.13
cracklib	2.9.6	2.9.6
cracklib-dicts	2.9.6	2.9.6

套件	AMI	KVM
crontabs	1.11	1.11
crypto-policies	20220428	20220428
crypto-policies-scripts	20220428	20220428
cryptsetup	2.6.1	2.6.1
cryptsetup-libs	2.6.1	2.6.1
curl-minimal	8.5.0	8.5.0
cyrus-sasl-lib	2.1.27	2.1.27
cyrus-sasl-plain	2.1.27	2.1.27
dbus	1.12.28	1.12.28
dbus-broker	32	32
dbus-common	1.12.28	1.12.28
dbus-libs	1.12.28	1.12.28
device-mapper	1.02.185	1.02.185
device-mapper-libs	1.02.185	1.02.185
diffutils	3.8	3.8
dnf	4.14.0	4.14.0
dnf-data	4.14.0	4.14.0
dnf-plugin-release-notification	1.2	1.2
dnf-plugins-core	4.3.0	4.3.0

套件	AMI	KVM
dnf-plugin-support-info	1.2	1.2
dnf-utils	4.3.0	4.3.0
dosfstools	4.2	4.2
dracut	055	055
dracut-config-ec2	3.0	
dracut-config-generic	055	055
dwz	0.14	0.14
dyninst	10.2.1	10.2.1
e2fsprogs	1.46.5	1.46.5
e2fsprogs-libs	1.46.5	1.46.5
ec2-hibinit-agent	1.0.8	
ec2-instance-connect	1.1	
ec2-instance-connect-selinux	1.1	
ec2-utils	2.2.0	
ed	1.14.2	1.14.2
efi-filesystem	5	5
efi-srpm-macros	5	5
efivar	38	38

套件	AMI	KVM
efivar-libs	38	38
elfutils-debuginfod-client	0.188	0.188
elfutils-default-yama-scope	0.188	0.188
elfutils-libelf	0.188	0.188
elfutils-libs	0.188	0.188
ethtool	5.15	5.15
expat	2.5.0	2.5.0
file	5.39	5.39
file-libs	5.39	5.39
filesystem	3.14	3.14
findutils	4.8.0	4.8.0
fonts-srpm-macros	2.0.5	2.0.5
fstrm	0.6.1	0.6.1
fuse-libs	2.9.9	2.9.9
gawk	5.1.0	5.1.0
gdbm-libs	1.19	1.19
gdisk	1.0.8	1.0.8
gettext	0.21	0.21
gettext-libs	0.21	0.21

套件	AMI	KVM
ghc-srpm-macros	1.5.0	1.5.0
glib2	2.74.7	2.74.7
glibc	2.34	2.34
glibc-all-langpacks	2.34	2.34
glibc-common	2.34	2.34
glibc-gconv-extra	2.34	2.34
glibc-locale-source	2.34	2.34
gmp	6.2.1	6.2.1
gnupg2-minimal	2.3.7	2.3.7
gnutls	3.8.0	3.8.0
go-srpm-macros	3.2.0	3.2.0
gpgme	1.15.1	1.15.1
gpm-libs	1.20.7	1.20.7
grep	3.8	3.8
groff-base	1.22.4	1.22.4
grub2-common	2.06	2.06
grub2-efi-aa64-ec2	2.06 (阿建築 64)	2.06 (阿建築 64)
grub2-efi-x64-ec2	2.06 (英文版)	2.06 (英文版)
grub2-pc		2.06 (英文版)
grub2-pc-modules	2.06	2.06 (公爵夫人)

套件	AMI	KVM
grub2-tools	2.06	2.06
grub2-tools-minimal	2.06	2.06
grubby	8.40	8.40
gssproxy	0.8.4	0.8.4
gzip	1.12	1.12
hostname	3.23	3.23
hunspell	1.7.0	1.7.0
hunspell-en	0.20140811.1	0.20140811.1
hunspell-en-GB	0.20140811.1	0.20140811.1
hunspell-en-US	0.20140811.1	0.20140811.1
hunspell-filesystem	1.7.0	1.7.0
hwdata	0.353	0.353
info	6.7	6.7
inih	49	49
initscripts	10.09	10.09
iproute	5.10.0	5.10.0
iputils	20210202	20210202
irqbalance	1.9.0	1.9.0
jansson	2.14	2.14
jitterentropy	3.4.1	3.4.1

套件	AMI	KVM
jq	1.7.1	
json-c	0.14	0.14
kbd	2.4.0	2.4.0
kbd-misc	2.4.0	2.4.0
kernel	6.1.90	6.1.90
kernel-livepatch-r epo-cdn		2023.4.20240513
kernel-livepatch-r epo-s3	2023.4.20240513	
kernel-modules-extra		6.1.90
kernel-modules-ext ra-common		6.1.90
kernel-srpm-macros	1.0	1.0
kernel-tools	6.1.90	6.1.90
keyutils	1.6.3	1.6.3
keyutils-libs	1.6.3	1.6.3
kmod	29	29
kmod-libs	29	29
kpatch-runtime	0.9.7	0.9.7
krb5-libs	1.21	1.21
less	608	608

套件	AMI	KVM
libacl	2.3.1	2.3.1
libaio	0.3.111	0.3.111
libarchive	3.5.3	3.5.3
libargon2	20171227	20171227
libassuan	2.5.5	2.5.5
libattr	2.5.1	2.5.1
libbasicobjects	0.1.1	0.1.1
libblkid	2.37.4	2.37.4
libcap	2.48	2.48
libcap-ng	0.8.2	0.8.2
libcbor	0.7.0	0.7.0
libcollection	0.7.0	0.7.0
libcom_err	1.46.5	1.46.5
libcomps	0.1.20	0.1.20
libconfig	1.7.2	1.7.2
libcurl-minimal	8.5.0	8.5.0
libdb	5.3.28	5.3.28
libdhash	0.5.0	
libdnf	0.69.0	0.69.0
libeconf	0.4.0	0.4.0

套件	AMI	KVM
libedit	3.1	3.1
libev	4.33	4.33
libevent	2.1.12	2.1.12
libfdisk	2.37.4	2.37.4
libffi	3.4.4	3.4.4
libfido2	1.10.0	1.10.0
libgcc	11.4.1	11.4.1
libgcrypt	1.10.2	1.10.2
libgomp	11.4.1	11.4.1
libgpg-error	1.42	1.42
libibverbs	48.0	48.0
libidn2	2.3.2	2.3.2
libini_config	1.3.1	1.3.1
libkcapi	1.4.0	1.4.0
libkcapi-hmacalc	1.4.0	1.4.0
libldb	2.6.2	
libmaxminddb	1.5.2	1.5.2
libmetalink	0.1.3	0.1.3
libmnl	1.0.4	1.0.4
libmodulemd	2.13.0	2.13.0

套件	AMI	KVM
libmount	2.37.4	2.37.4
libnfsidmap	2.5.4	2.5.4
libnghttp2	1.59.0	1.59.0
libnl3	3.5.0	3.5.0
libpath_utils	0.2.1	0.2.1
libpcap	1.10.1	1.10.1
libpipeline	1.5.3	1.5.3
libpkgconf	1.8.0	1.8.0
libpsl	0.21.1	0.21.1
libpwquality	1.4.4	1.4.4
libref_array	0.1.5	0.1.5
librepo	1.14.5	1.14.5
libreport-filessystem	2.15.2	2.15.2
libseccomp	2.5.3	2.5.3
libselenium	3.4	3.4
libselenium-utils	3.4	3.4
libsemanage	3.4	3.4
libsepol	3.4	3.4
libsigsegv	2.13	2.13
libsmartcols	2.37.4	2.37.4

套件	AMI	KVM
libsolv	0.7.22	0.7.22
libss	1.46.5	1.46.5
libsss_certmap	2.9.4	
libsss_idmap	2.9.4	2.9.4
libsss_nss_idmap	2.9.4	2.9.4
libsss_sudo	2.9.4	
libstdc++	11.4.1	11.4.1
libstoragemgmt	1.9.4	1.9.4
libtalloc	2.3.4	
libtasn1	4.19.0	4.19.0
libtdb	1.4.7	
libtevent	0.13.0	
libtextstyle	0.21	0.21
libtirpc	1.3.3	1.3.3
libunistring	0.9.10	0.9.10
libuser	0.63	0.63
libutempter	1.2.1	1.2.1
libuuid	2.37.4	2.37.4
libuv	1.47.0	1.47.0
libverto	0.3.2	0.3.2

套件	AMI	KVM
libverto-libev	0.3.2	0.3.2
libxcrypt	4.4.33	4.4.33
libxml2	2.10.4	2.10.4
libyaml	0.2.5	0.2.5
libzstd	1.5.5	1.5.5
lm_sensors-libs	3.6.0	3.6.0
lmdb-libs	0.9.29	0.9.29
logrotate	3.20.1	3.20.1
lsof	4.94.0	4.94.0
lua-libs	5.4.4	5.4.4
lua-srpm-macros	1	1
lz4-libs	1.9.4	1.9.4
man-db	2.9.3	2.9.3
man-pages	5.10	5.10
microcode_ctl	2.1 (英文版)	2.1 (英文版)
mpfr	4.1.0	4.1.0
nano	5.8	5.8
ncurses	6.2	6.2
ncurses-base	6.2	6.2
ncurses-libs	6.2	6.2

套件	AMI	KVM
nettle	3.8	3.8
net-tools	2.0	2.0
newt	0.52.21	0.52.21
nfs-utils	2.5.4	2.5.4
npth	1.6	1.6
nspr	4.35.0	4.35.0
nss	3.90.0	3.90.0
nss-softokn	3.90.0	3.90.0
nss-softokn-freebl	3.90.0	3.90.0
nss-sysinit	3.90.0	3.90.0
nss-util	3.90.0	3.90.0
ntsysv	1.15	1.15
numactl-libs	2.0.14	2.0.14
ocaml-srpm-macros	6	6
oniguruma	6.9.7.1	
openblas-srpm-macros	2	2
openldap	2.4.57	2.4.57
openssh	8.7 小時	8.7 小時
openssh-clients	8.7 小時	8.7 小時
openssh-server	8.7 小時	8.7 小時

套件	AMI	KVM
openssl	3.0.8	3.0.8
openssl-libs	3.0.8	3.0.8
openssl-pkcs11	0.4.12	0.4.12
os-prober	1.77	1.77
p11-kit	0.24.1	0.24.1
p11-kit-trust	0.24.1	0.24.1
package-notes-srpm-macros	0.4	0.4
pam	1.5.1	1.5.1
parted	3.4	3.4
passwd	0.80	0.80
pciutils	3.7.0	3.7.0
pciutils-libs	3.7.0	3.7.0
pcre2	10.40	10.40
pcre2-syntax	10.40	10.40
perl-Carp	1.50	1.50
perl-Class-Struct	0.66	0.66
perl-constant	1.33	1.33
perl-DynaLoader	1.47	1.47
perl-Encode	3.15	3.15
perl-Errno	1.30	1.30

套件	AMI	KVM
perl-Exporter	5.74	5.74
perl-Fcntl	1.13	1.13
perl-File-Basename	2.85	2.85
perl-File-Path	2.18	2.18
perl-File-stat	1.09	1.09
perl-File-Temp	0.231.100	0.231.100
perl-Getopt-Long	2.52	2.52
perl-Getopt-Std	1.12	1.12
perl-HTTP-Tiny	0.078	0.078
perl-if	0.60.800	0.60.800
perl-interpreter	5.32.1	5.32.1
perl-I0	1.43	1.43
perl-IPC-Open3	1.21	1.21
perl-libs	5.32.1	5.32.1
perl-MIME-Base64	3.16	3.16
perl-mro	1.23	1.23
perl-overload	1.31	1.31
perl-overloading	0.02	0.02
perl-parent	0.238	0.238
perl-PathTools	3.78	3.78

套件	AMI	KVM
perl-Pod-Escapes	1.07	1.07
perl-podlators	4.14	4.14
perl-Pod-Perldoc	3.28.01	3.28.01
perl-Pod-Simple	3.42	3.42
perl-Pod-Usage	2.01	2.01
perl-POSIX	1.94	1.94
perl-Scalar-List-Utils	1.56	1.56
perl-SelectSaver	1.02	1.02
perl-Socket	2.032	2.032
perl-srpm-macros	1	1
perl-Storable	3.21	3.21
perl-subst	1.03	1.03
perl-Symbol	1.08	1.08
perl-Term-ANSIColor	5.01	5.01
perl-Term-Cap	1.17	1.17
perl-Text-ParseWords	3.30	3.30
perl-Text-Tabs+Wrap	2021.0726	2021.0726
perl-Time-Local	1.300	1.300
perl-vars	1.05	1.05
pkgconf	1.8.0	1.8.0

套件	AMI	KVM
pkgconf-m4	1.8.0	1.8.0
pkgconf-pkg-config	1.8.0	1.8.0
policycoreutils	3.4	3.4
policycoreutils-python-utils	3.4	
popt	1.18	1.18
procps-ng	3.3.17	3.3.17
protobuf-c	1.4.1	1.4.1
psacct	6.6.4	6.6.4
psmisc	23.4	23.4
publicsuffix-list-dafsa	20240212	20240212
python3	3.9.16	3.9.16
python3-attrs	20.3.0	20.3.0
python3-audit	3.0.6	3.0.6
python3-awscrt	0.19.19	0.19.19
python3-babel	2.9.1	2.9.1
python3-cffi	1.14.5	1.14.5
python3-chardet	4.0.0	4.0.0
python3-colorama	0.4.4	0.4.4
python3-configobj	5.0.6	5.0.6

套件	AMI	KVM
python3-cryptography	36.0.1	36.0.1
python3-daemon	2.3.0	
python3-dateutil	2.8.1	2.8.1
python3-dbus	1.2.18	1.2.18
python3-distro	1.5.0	1.5.0
python3-dnf	4.14.0	4.14.0
python3-dnf-plugins-core	4.3.0	4.3.0
python3-docutils	0.16	0.16
python3-gpg	1.15.1	1.15.1
python3-hawkey	0.69.0	0.69.0
python3-idna	2.10	2.10
python3-jinja2	2.11.3	2.11.3
python3-jmespath	0.10.0	0.10.0
python3-jsonpatch	1.21	1.21
python3-jsonpointer	2.0	2.0
python3-jjsonschema	3.2.0	3.2.0
python3-libcomps	0.1.20	0.1.20
python3-libdnf	0.69.0	0.69.0
python3-libs	3.9.16	3.9.16
python3-libselenium	3.4	3.4

套件	AMI	KVM
python3-libsemanage	3.4	3.4
python3-libstorage mgmt	1.9.4	1.9.4
python3-lockfile	0.12.2	
python3-markupsafe	1.1.1	1.1.1
python3-netifaces	0.10.6	0.10.6
python3-oauthlib	3.0.2	3.0.2
python3-pip-wheel	21.3.1	21.3.1
python3-ply	3.11	3.11
python3-policycore utils	3.4	3.4
python3-prettytable	0.7.2	0.7.2
python3-prompt-too lkit	3.0.24	3.0.24
python3-pycparser	2.20	2.20
python3-pyrsistent	0.17.3	0.17.3
python3-pyserial	3.4	3.4
python3-pysocks	1.7.1	1.7.1
python3-pytz	2022.7.1	2022.7.1
python3-pyyaml	5.4.1	5.4.1
python3-requests	2.25.1	2.25.1

套件	AMI	KVM
python3-rpm	4.16.1.3	4.16.1.3
python3-ruamel-yaml	0.16.6	0.16.6
python3-ruamel-yaml-clib	0.1.2	0.1.2
python3-setools	4.4.1	4.4.1
python3-setuptools	59.6.0	59.6.0
python3-setuptools-wheel	59.6.0	59.6.0
python3-six	1.15.0	1.15.0
python3-systemd	235	235
python3-urllib3	1.25.10	1.25.10
python3-wcwidth	0.2.5	0.2.5
python-chevront	0.13.1	
python-srpm-macros	3.9	3.9
quota	4.06	4.06
quota-nls	4.06	4.06
readline	8.1	8.1
rng-tools	6.14	6.14
rootfiles	8.1	8.1
rpcbind	1.2.6	1.2.6
rpm	4.16.1.3	4.16.1.3

套件	AMI	KVM
rpm-build-libs	4.16.1.3	4.16.1.3
rpm-libs	4.16.1.3	4.16.1.3
rpm-plugin-selinux	4.16.1.3	4.16.1.3
rpm-plugin-systemd-inhibit	4.16.1.3	4.16.1.3
rpm-sign-libs	4.16.1.3	4.16.1.3
rsync	3.2.6	3.2.6
rust-srpm-macros	21	21
sbsigntools	0.9.4	0.9.4
screen	4.8.0	4.8.0
sed	4.8	4.8
selinux-policy	37.22	37.22
selinux-policy-targeted	37.22	37.22
setup	2.13.7	2.13.7
shadow-utils	4.9	4.9
slang	2.3.2	2.3.2
sqlite-libs	3.40.0	3.40.0
sssd-client	2.9.4	2.9.4
sssd-common	2.9.4	
sssd-kcm	2.9.4	

套件	AMI	KVM
sssd-nfs-idmap	2.9.4	
strace	6.8	6.8
sudo	1.9.15	1.9.15
sysctl-defaults	1.0	1.0
sysstat	12.5.6	12.5.6
systemd	252.16	252.16
systemd-libs	252.16	252.16
systemd-networkd	252.16	252.16
systemd-pam	252.16	252.16
systemd-resolved	252.16	252.16
systemd-udev	252.16	252.16
system-release	2023.4.20240513	2023.4.20240513
systemtap-runtime	4.8	4.8
tar	1.34	1.34
tbb	2020.3	2020.3
tcpdump	4.99.1	4.99.1
tcsh	6.24.07	6.24.07
time	1.9	1.9
traceroute	2.1.3	2.1.3
tzdata	2024a	2024a

套件	AMI	KVM
unzip	6.0	6.0
update-motd	2.2	2.2
userspace-rcu	0.12.1	0.12.1
util-linux	2.37.4	2.37.4
util-linux-core	2.37.4	2.37.4
vim-common	9.0.2153	9.0.2153
vim-data	9.0.2153	9.0.2153
vim-enhanced	9.0.2153	9.0.2153
vim-filesystem	9.0.2153	9.0.2153
vim-minimal	9.0.2153	9.0.2153
wget	1.21.3	1.21.3
which	2.21	2.21
words	3.0	3.0
xfsdump	3.1.11	3.1.11
xfsplogs	5.18.0	5.18.0
xxd	9.0.2153	9.0.2153
xxhash-libs	0.8.0	0.8.0
xz	5.2.5	5.2.5
xz-libs	5.2.5	5.2.5
yum	4.14.0	4.14.0

套件	AMI	KVM
zip	3.0	3.0
zlib	1.2.11	1.2.11
zram-generator	1.1.2	
zram-generator-def aults	1.1.2	
zstd	1.5.5	1.5.5

比較安裝在 Amazon 2023 標準 AMI 上的軟件包與 AL2023 VMware OVA 映像

AL2023 標準 AMI 上存在的 RPM 與 AL2023 VMware OVA 映像上存在的 RPM 進行比較的比較。

套件	AMI	VMware OVA
acl	2.3.1	2.3.1
acpid	2.0.32	
alternatives	1.15	1.15
amazon-chrony-config	4.3	
amazon-ec2-net-utils	2.4.1	
amazon-linux-onprem		1.2
amazon-linux-repo- cdn		2023.4.20240513
amazon-linux-repo-s3	2023.4.20240513	
amazon-linux-sb-keys	2023.1	2023.1

套件	AMI	VMware OVA
amazon-onprem-network		1.2
amazon-rpm-config	228	228
amazon-ssm-agent	3.3.380.0	3.3.380.0
at	3.1.23	3.1.23
attr	2.5.1	2.5.1
audit	3.0.6	3.0.6
audit-libs	3.0.6	3.0.6
aws-cfn-bootstrap	2.0	
awscli-2	2.15.30	2.15.30
basesystem	11	11
bash	5.2.15	5.2.15
bash-completion	2.11	2.11
bc	1.07.1	1.07.1
bind-libs	9.16.48	9.16.48
bind-license	9.16.48	9.16.48
bind-utils	9.16.48	9.16.48
binutils	2.39	2.39
boost-filesystem	1.75.0	1.75.0
boost-system	1.75.0	1.75.0
boost-thread	1.75.0	1.75.0

套件	AMI	VMware OVA
bzip2	1.0.8	1.0.8
bzip2-libs	1.0.8	1.0.8
ca-certificates	2023.2.64	2023.2.64
c-ares	1.19.0	
checkpolicy	3.4	3.4
chkconfig	1.15	1.15
chrony	4.3	4.3
cloud-init	22.2.2	22.2.2
cloud-init-cfg-ec2	22.2.2	
cloud-init-cfg-onpre		22.2.2
cloud-utils-growpart	0.31	0.31
coreutils	8.32	8.32
coreutils-common	8.32	8.32
cpio	2.13	2.13
cracklib	2.9.6	2.9.6
cracklib-dicts	2.9.6	2.9.6
crontabs	1.11	1.11
crypto-policies	20220428	20220428
crypto-policies-scripts	20220428	20220428

套件	AMI	VMware OVA
cryptsetup	2.6.1	2.6.1
cryptsetup-libs	2.6.1	2.6.1
curl-minimal	8.5.0	8.5.0
cyrus-sasl-lib	2.1.27	2.1.27
cyrus-sasl-plain	2.1.27	2.1.27
dbus	1.12.28	1.12.28
dbus-broker	32	32
dbus-common	1.12.28	1.12.28
dbus-libs	1.12.28	1.12.28
device-mapper	1.02.185	1.02.185
device-mapper-libs	1.02.185	1.02.185
diffutils	3.8	3.8
dnf	4.14.0	4.14.0
dnf-data	4.14.0	4.14.0
dnf-plugin-release-notification	1.2	1.2
dnf-plugins-core	4.3.0	4.3.0
dnf-plugin-support-info	1.2	1.2
dnf-utils	4.3.0	4.3.0
dosfstools	4.2	4.2

套件	AMI	VMware OVA
dracut	055	055
dracut-config-ec2	3.0	
dracut-config-generic	055	055
dwz	0.14	0.14
dyninst	10.2.1	10.2.1
e2fsprogs	1.46.5	1.46.5
e2fsprogs-libs	1.46.5	1.46.5
ec2-hibinit-agent	1.0.8	
ec2-instance-connect	1.1	
ec2-instance-connect-selinux	1.1	
ec2-utils	2.2.0	
ed	1.14.2	1.14.2
efi-filesystem	5	5
efi-srpm-macros	5	5
efivar	38	38
efivar-libs	38	38
elfutils-debuginfod-client	0.188	0.188
elfutils-default-yama-scope	0.188	0.188

套件	AMI	VMware OVA
elfutils-libelf	0.188	0.188
elfutils-libs	0.188	0.188
ethtool	5.15	5.15
expat	2.5.0	2.5.0
file	5.39	5.39
file-libs	5.39	5.39
filesystem	3.14	3.14
findutils	4.8.0	4.8.0
fonts-srpm-macros	2.0.5	2.0.5
fstrm	0.6.1	0.6.1
fuse3		3.10.4
fuse3-libs		3.10.4
fuse-common		3.10.4
fuse-libs	2.9.9	2.9.9
gawk	5.1.0	5.1.0
gdbm-libs	1.19	1.19
gdisk	1.0.8	1.0.8
gettext	0.21	0.21
gettext-libs	0.21	0.21
ghc-srpm-macros	1.5.0	1.5.0

套件	AMI	VMware OVA
glib2	2.74.7	2.74.7
glibc	2.34	2.34
glibc-all-langpacks	2.34	2.34
glibc-common	2.34	2.34
glibc-gconv-extra	2.34	2.34
glibc-locale-source	2.34	2.34
gmp	6.2.1	6.2.1
gnupg2-minimal	2.3.7	2.3.7
gnutls	3.8.0	3.8.0
go-srpm-macros	3.2.0	3.2.0
gpgme	1.15.1	1.15.1
gpm-libs	1.20.7	1.20.7
grep	3.8	3.8
groff-base	1.22.4	1.22.4
grub2-common	2.06	2.06
grub2-efi-x64-ec2	2.06	2.06
grub2-pc		2.06
grub2-pc-modules	2.06	2.06
grub2-tools	2.06	2.06
grub2-tools-minimal	2.06	2.06

套件	AMI	VMware OVA
grubby	8.40	8.40
gssproxy	0.8.4	0.8.4
gzip	1.12	1.12
hostname	3.23	3.23
hunspell	1.7.0	1.7.0
hunspell-en	0.20140811.1	0.20140811.1
hunspell-en-GB	0.20140811.1	0.20140811.1
hunspell-en-US	0.20140811.1	0.20140811.1
hunspell-filesystem	1.7.0	1.7.0
hwdata	0.353	0.353
info	6.7	6.7
inih	49	49
initscripts	10.09	10.09
iproute	5.10.0	5.10.0
iputils	20210202	20210202
irqbalance	1.9.0	1.9.0
jansson	2.14	2.14
jitterentropy	3.4.1	3.4.1
jq	1.7.1	
json-c	0.14	0.14

套件	AMI	VMware OVA
kbd	2.4.0	2.4.0
kbd-misc	2.4.0	2.4.0
kernel	6.1.90	6.1.90
kernel-livepatch-r epo-cdn		2023.4.20240513
kernel-livepatch-r epo-s3	2023.4.20240513	
kernel-modules-extra		6.1.90
kernel-modules-ext ra-common		6.1.90
kernel-srpm-macros	1.0	1.0
kernel-tools	6.1.90	6.1.90
keyutils	1.6.3	1.6.3
keyutils-libs	1.6.3	1.6.3
kmod	29	29
kmod-libs	29	29
kpatch-runtime	0.9.7	0.9.7
krb5-libs	1.21	1.21
less	608	608
libacl	2.3.1	2.3.1
libaio	0.3.111	0.3.111

套件	AMI	VMware OVA
libarchive	3.5.3	3.5.3
libargon2	20171227	20171227
libassuan	2.5.5	2.5.5
libattr	2.5.1	2.5.1
libbasicobjects	0.1.1	0.1.1
libblkid	2.37.4	2.37.4
libcap	2.48	2.48
libcap-ng	0.8.2	0.8.2
libcbor	0.7.0	0.7.0
libcollection	0.7.0	0.7.0
libcom_err	1.46.5	1.46.5
libcomps	0.1.20	0.1.20
libconfig	1.7.2	1.7.2
libcurl-minimal	8.5.0	8.5.0
libdb	5.3.28	5.3.28
libdhash	0.5.0	
libdnf	0.69.0	0.69.0
libeconf	0.4.0	0.4.0
libedit	3.1	3.1
libev	4.33	4.33

套件	AMI	VMware OVA
libevent	2.1.12	2.1.12
libfdisk	2.37.4	2.37.4
libffi	3.4.4	3.4.4
libfido2	1.10.0	1.10.0
libgcc	11.4.1	11.4.1
libgcrypt	1.10.2	1.10.2
libgomp	11.4.1	11.4.1
libgpg-error	1.42	1.42
libibverbs	48.0	48.0
libidn2	2.3.2	2.3.2
libini_config	1.3.1	1.3.1
libkcapi	1.4.0	1.4.0
libkcapi-hmacalc	1.4.0	1.4.0
libldb	2.6.2	
libmaxminddb	1.5.2	1.5.2
libmetalink	0.1.3	0.1.3
libmnl	1.0.4	1.0.4
libmodulemd	2.13.0	2.13.0
libmount	2.37.4	2.37.4
libmspack		0.10.1

套件	AMI	VMware OVA
libnfsidmap	2.5.4	2.5.4
libnghttp2	1.59.0	1.59.0
libnl3	3.5.0	3.5.0
libpath_utils	0.2.1	0.2.1
libpcap	1.10.1	1.10.1
libpipeline	1.5.3	1.5.3
libpkgconf	1.8.0	1.8.0
libpsl	0.21.1	0.21.1
libpwquality	1.4.4	1.4.4
libref_array	0.1.5	0.1.5
librepo	1.14.5	1.14.5
libreport-filessystem	2.15.2	2.15.2
libseccomp	2.5.3	2.5.3
libselenium	3.4	3.4
libselenium-utils	3.4	3.4
libsemanage	3.4	3.4
libsepol	3.4	3.4
libsigsegv	2.13	2.13
libsmartcols	2.37.4	2.37.4
libsolv	0.7.22	0.7.22

套件	AMI	VMware OVA
libss	1.46.5	1.46.5
libsss_certmap	2.9.4	
libsss_idmap	2.9.4	2.9.4
libsss_nss_idmap	2.9.4	2.9.4
libsss_sudo	2.9.4	
libstdc++	11.4.1	11.4.1
libstoragegmt	1.9.4	1.9.4
libtalloc	2.3.4	
libtasn1	4.19.0	4.19.0
libtdb	1.4.7	
libtevent	0.13.0	
libtextstyle	0.21	0.21
libtirpc	1.3.3	1.3.3
libtool-ltdl		2.4.7
libunistring	0.9.10	0.9.10
libuser	0.63	0.63
libutempter	1.2.1	1.2.1
libuuid	2.37.4	2.37.4
libuv	1.47.0	1.47.0
libverto	0.3.2	0.3.2

套件	AMI	VMware OVA
libverto-libev	0.3.2	0.3.2
libxcrypt	4.4.33	4.4.33
libxml2	2.10.4	2.10.4
libxslt		1.1.34
libyaml	0.2.5	0.2.5
libzstd	1.5.5	1.5.5
lm_sensors-libs	3.6.0	3.6.0
lmdb-libs	0.9.29	0.9.29
logrotate	3.20.1	3.20.1
lsuf	4.94.0	4.94.0
lua-libs	5.4.4	5.4.4
lua-srpm-macros	1	1
lz4-libs	1.9.4	1.9.4
man-db	2.9.3	2.9.3
man-pages	5.10	5.10
microcode_ctl	2.1	2.1
mpfr	4.1.0	4.1.0
nano	5.8	5.8
ncurses	6.2	6.2
ncurses-base	6.2	6.2

套件	AMI	VMware OVA
ncurses-libs	6.2	6.2
nettle	3.8	3.8
net-tools	2.0	2.0
newt	0.52.21	0.52.21
nfs-utils	2.5.4	2.5.4
npth	1.6	1.6
nspr	4.35.0	4.35.0
nss	3.90.0	3.90.0
nss-softokn	3.90.0	3.90.0
nss-softokn-freebl	3.90.0	3.90.0
nss-sysinit	3.90.0	3.90.0
nss-util	3.90.0	3.90.0
ntsysv	1.15	1.15
numactl-libs	2.0.14	2.0.14
ocaml-srpm-macros	6	6
oniguruma	6.9.7.1	
openblas-srpm-macros	2	2
openldap	2.4.57	2.4.57
openssh	8.7 小時	8.7 小時
openssh-clients	8.7 小時	8.7 小時

套件	AMI	VMware OVA
openssh-server	8.7 小時	8.7 小時
openssl	3.0.8	3.0.8
openssl-libs	3.0.8	3.0.8
openssl-pkcs11	0.4.12	0.4.12
open-vm-tools		12.3.0
os-prober	1.77	1.77
p11-kit	0.24.1	0.24.1
p11-kit-trust	0.24.1	0.24.1
package-notes-srpm-macros	0.4	0.4
pam	1.5.1	1.5.1
parted	3.4	3.4
passwd	0.80	0.80
pciutils	3.7.0	3.7.0
pciutils-libs	3.7.0	3.7.0
pcre2	10.40	10.40
pcre2-syntax	10.40	10.40
perl-Carp	1.50	1.50
perl-Class-Struct	0.66	0.66
perl-constant	1.33	1.33
perl-DynaLoader	1.47	1.47

套件	AMI	VMware OVA
perl-Encode	3.15	3.15
perl-Errno	1.30	1.30
perl-Exporter	5.74	5.74
perl-Fcntl	1.13	1.13
perl-File-Basename	2.85	2.85
perl-File-Path	2.18	2.18
perl-File-stat	1.09	1.09
perl-File-Temp	0.231.100	0.231.100
perl-Getopt-Long	2.52	2.52
perl-Getopt-Std	1.12	1.12
perl-HTTP-Tiny	0.078	0.078
perl-if	0.60.800	0.60.800
perl-interpreter	5.32.1	5.32.1
perl-IO	1.43	1.43
perl-IPC-Open3	1.21	1.21
perl-libs	5.32.1	5.32.1
perl-MIME-Base64	3.16	3.16
perl-mro	1.23	1.23
perl-overload	1.31	1.31
perl-overloading	0.02	0.02

套件	AMI	VMware OVA
perl-parent	0.238	0.238
perl-PathTools	3.78	3.78
perl-Pod-Escapes	1.07	1.07
perl-podlators	4.14	4.14
perl-Pod-Perldoc	3.28.01	3.28.01
perl-Pod-Simple	3.42	3.42
perl-Pod-Usage	2.01	2.01
perl-POSIX	1.94	1.94
perl-Scalar-List-Utils	1.56	1.56
perl-SelectSaver	1.02	1.02
perl-Socket	2.032	2.032
perl-srpm-macros	1	1
perl-Storable	3.21	3.21
perl-subst	1.03	1.03
perl-Symbol	1.08	1.08
perl-Term-ANSIColor	5.01	5.01
perl-Term-Cap	1.17	1.17
perl-Text-ParseWords	3.30	3.30
perl-Text-Tabs+Wrap	2021.0726	2021.0726
perl-Time-Local	1.300	1.300

套件	AMI	VMware OVA
perl-vars	1.05	1.05
pkgconf	1.8.0	1.8.0
pkgconf-m4	1.8.0	1.8.0
pkgconf-pkg-config	1.8.0	1.8.0
policycoreutils	3.4	3.4
policycoreutils-python-utils	3.4	
popt	1.18	1.18
procps-ng	3.3.17	3.3.17
protobuf-c	1.4.1	1.4.1
psacct	6.6.4	6.6.4
psmisc	23.4	23.4
publicsuffix-list-dafsa	20240212	20240212
python3	3.9.16	3.9.16
python3-attrs	20.3.0	20.3.0
python3-audit	3.0.6	3.0.6
python3-awscli	0.19.19	0.19.19
python3-babel	2.9.1	2.9.1
python3-cffi	1.14.5	1.14.5
python3-chardet	4.0.0	4.0.0

套件	AMI	VMware OVA
python3-colorama	0.4.4	0.4.4
python3-configobj	5.0.6	5.0.6
python3-cryptography	36.0.1	36.0.1
python3-daemon	2.3.0	
python3-dateutil	2.8.1	2.8.1
python3-dbus	1.2.18	1.2.18
python3-distro	1.5.0	1.5.0
python3-dnf	4.14.0	4.14.0
python3-dnf-plugins-core	4.3.0	4.3.0
python3-docutils	0.16	0.16
python3-gpg	1.15.1	1.15.1
python3-hawkey	0.69.0	0.69.0
python3-idna	2.10	2.10
python3-jinja2	2.11.3	2.11.3
python3-jmespath	0.10.0	0.10.0
python3-jsonpatch	1.21	1.21
python3-jsonpointer	2.0	2.0
python3-jsonschemata	3.2.0	3.2.0
python3-libcomps	0.1.20	0.1.20
python3-libdnf	0.69.0	0.69.0

套件	AMI	VMware OVA
python3-libs	3.9.16	3.9.16
python3-libselinux	3.4	3.4
python3-libsemanage	3.4	3.4
python3-libstorage mgmt	1.9.4	1.9.4
python3-lockfile	0.12.2	
python3-markupsafe	1.1.1	1.1.1
python3-netifaces	0.10.6	0.10.6
python3-oauthlib	3.0.2	3.0.2
python3-pip-wheel	21.3.1	21.3.1
python3-ply	3.11	3.11
python3-policycore utils	3.4	3.4
python3-prettytable	0.7.2	0.7.2
python3-prompt-too lkit	3.0.24	3.0.24
python3-pycparser	2.20	2.20
python3-pyrsistent	0.17.3	0.17.3
python3-pyserial	3.4	3.4
python3-pysocks	1.7.1	1.7.1
python3-pytz	2022.7.1	2022.7.1

套件	AMI	VMware OVA
python3-pyyaml	5.4.1	5.4.1
python3-requests	2.25.1	2.25.1
python3-rpm	4.16.1.3	4.16.1.3
python3-ruamel-yaml	0.16.6	0.16.6
python3-ruamel-yaml-clib	0.1.2	0.1.2
python3-setools	4.4.1	4.4.1
python3-setuptools	59.6.0	59.6.0
python3-setuptools-wheel	59.6.0	59.6.0
python3-six	1.15.0	1.15.0
python3-systemd	235	235
python3-urllib3	1.25.10	1.25.10
python3-wcwidth	0.2.5	0.2.5
python-chevront	0.13.1	
python-srpm-macros	3.9	3.9
quota	4.06	4.06
quota-nls	4.06	4.06
readline	8.1	8.1
rng-tools	6.14	6.14
rootfiles	8.1	8.1

套件	AMI	VMware OVA
rpcbind	1.2.6	1.2.6
rpm	4.16.1.3	4.16.1.3
rpm-build-libs	4.16.1.3	4.16.1.3
rpm-libs	4.16.1.3	4.16.1.3
rpm-plugin-selinux	4.16.1.3	4.16.1.3
rpm-plugin-systemd-inhibit	4.16.1.3	4.16.1.3
rpm-sign-libs	4.16.1.3	4.16.1.3
rsync	3.2.6	3.2.6
rust-srpm-macros	21	21
sbsigntools	0.9.4	0.9.4
screen	4.8.0	4.8.0
sed	4.8	4.8
selinux-policy	37.22	37.22
selinux-policy-targeted	37.22	37.22
setup	2.13.7	2.13.7
shadow-utils	4.9	4.9
slang	2.3.2	2.3.2
sqlite-libs	3.40.0	3.40.0
sssd-client	2.9.4	2.9.4

套件	AMI	VMware OVA
sssd-common	2.9.4	
sssd-kcm	2.9.4	
sssd-nfs-idmap	2.9.4	
strace	6.8	6.8
sudo	1.9.15	1.9.15
sysctl-defaults	1.0	1.0
sysstat	12.5.6	12.5.6
systemd	252.16	252.16
systemd-libs	252.16	252.16
systemd-networkd	252.16	252.16
systemd-pam	252.16	252.16
systemd-resolved	252.16	252.16
systemd-udev	252.16	252.16
system-release	2023.4.20240513	2023.4.20240513
systemtap-runtime	4.8	4.8
tar	1.34	1.34
tbb	2020.3	2020.3
tcpdump	4.99.1	4.99.1
tcsh	6.24.07	6.24.07
time	1.9	1.9

套件	AMI	VMware OVA
traceroute	2.1.3	2.1.3
tzdata	2024a	2024a
unzip	6.0	6.0
update-motd	2.2	2.2
userspace-rcu	0.12.1	0.12.1
util-linux	2.37.4	2.37.4
util-linux-core	2.37.4	2.37.4
vim-common	9.0.2153	9.0.2153
vim-data	9.0.2153	9.0.2153
vim-enhanced	9.0.2153	9.0.2153
vim-filesystem	9.0.2153	9.0.2153
vim-minimal	9.0.2153	9.0.2153
wget	1.21.3	1.21.3
which	2.21	2.21
words	3.0	3.0
xfsdump	3.1.11	3.1.11
xfspgrog	5.18.0	5.18.0
xmlsec1		1.2.33
xmlsec1-openssl		1.2.33
xxd	9.0.2153	9.0.2153

套件	AMI	VMware OVA
xxhash-libs	0.8.0	0.8.0
xz	5.2.5	5.2.5
xz-libs	5.2.5	5.2.5
yum	4.14.0	4.14.0
zip	3.0	3.0
zlib	1.2.11	1.2.11
zram-generator	1.1.2	
zram-generator-def aults	1.1.2	
zstd	1.5.5	1.5.5

將 Amazon 2023 標準 AMI 上安裝的軟件包與 AL2023 超 V 映像進行比較

AL2023 標準 AMI 上存在的 RPM 與 AL2023 超 V 映像上存在的 RPM 的比較的比較。

套件	AMI	超 V 型 VHDX
acl	2.3.1	2.3.1
acpid	2.0.32	
alternatives	1.15	1.15
amazon-chrony-config	4.3	
amazon-ec2-net-utils	2.4.1	
amazon-linux-onprem		1.2

套件	AMI	超 V 型 VHDX
amazon-linux-repo-cdn		2023.4.20240319
amazon-linux-repo-s3	2023.4.20240319	
amazon-linux-sb-keys	2023.1	2023.1
amazon-onprem-network		1.2
amazon-rpm-config	228	228
amazon-ssm-agent	3.2.2303.0	3.2.2303.0
at	3.1.23	3.1.23
attr	2.5.1	2.5.1
audit	3.0.6	3.0.6
audit-libs	3.0.6	3.0.6
aws-cfn-bootstrap	2.0	
awscli-2	2.14.5	2.14.5
basesystem	11	11
bash	5.2.15	5.2.15
bash-completion	2.11	2.11
bc	1.07.1	1.07.1
bind-libs	9.16.48	9.16.48
bind-license	9.16.48	9.16.48
bind-utils	9.16.48	9.16.48

套件	AMI	超 V 型 VHDX
binutils	2.39	2.39
boost-filesystem	1.75.0	1.75.0
boost-system	1.75.0	1.75.0
boost-thread	1.75.0	1.75.0
bzip2	1.0.8	1.0.8
bzip2-libs	1.0.8	1.0.8
ca-certificates	2023.2.64	2023.2.64
c-ares	1.19.0	
checkpolicy	3.4	3.4
chkconfig	1.15	1.15
chrony	4.3	4.3
cloud-init	22.2.2	22.2.2
cloud-init-cfg-ec2	22.2.2	
cloud-init-cfg-onpre m		22.2.2
cloud-utils-growpart	0.31	0.31
coreutils	8.32	8.32
coreutils-common	8.32	8.32
cpio	2.13	2.13
cracklib	2.9.6	2.9.6
cracklib-dicts	2.9.6	2.9.6

套件	AMI	超 V 型 VHDX
crontabs	1.11	1.11
crypto-policies	20220428	20220428
crypto-policies-scripts	20220428	20220428
cryptsetup	2.6.1	2.6.1
cryptsetup-libs	2.6.1	2.6.1
curl-minimal	8.5.0	8.5.0
cyrus-sasl-lib	2.1.27	2.1.27
cyrus-sasl-plain	2.1.27	2.1.27
dbus	1.12.28	1.12.28
dbus-broker	32	32
dbus-common	1.12.28	1.12.28
dbus-libs	1.12.28	1.12.28
device-mapper	1.02.185	1.02.185
device-mapper-libs	1.02.185	1.02.185
diffutils	3.8	3.8
dnf	4.14.0	4.14.0
dnf-data	4.14.0	4.14.0
dnf-plugin-release-notification	1.2	1.2
dnf-plugins-core	4.3.0	4.3.0

套件	AMI	超 V 型 VHDX
dnf-plugin-support-info	1.2	1.2
dnf-utils	4.3.0	4.3.0
dosfstools	4.2	4.2
dracut	055	055
dracut-config-ec2	3.0	
dracut-config-generic	055	055
dwz	0.14	0.14
dyninst	10.2.1	10.2.1
e2fsprogs	1.46.5	1.46.5
e2fsprogs-libs	1.46.5	1.46.5
ec2-hibinit-agent	1.0.8	
ec2-instance-connect	1.1	
ec2-instance-connect-selinux	1.1	
ec2-utils	2.2.0	
ed	1.14.2	1.14.2
efi-filesystem	5	5
efi-srpm-macros	5	5
efivar	38	38

套件	AMI	超 V 型 VHDX
efivar-libs	38	38
elfutils-debuginfod-client	0.188	0.188
elfutils-default-yama-scope	0.188	0.188
elfutils-libelf	0.188	0.188
elfutils-libs	0.188	0.188
ethtool	5.15	5.15
expat	2.5.0	2.5.0
file	5.39	5.39
file-libs	5.39	5.39
filesystem	3.14	3.14
findutils	4.8.0	4.8.0
fonts-srpm-macros	2.0.5	2.0.5
fstrm	0.6.1	0.6.1
fuse-libs	2.9.9	2.9.9
gawk	5.1.0	5.1.0
gdbm-libs	1.19	1.19
gdisk	1.0.8	1.0.8
gettext	0.21	0.21
gettext-libs	0.21	0.21

套件	AMI	超 V 型 VHDX
ghc-srpm-macros	1.5.0	1.5.0
glib2	2.74.7	2.74.7
glibc	2.34	2.34
glibc-all-langpacks	2.34	2.34
glibc-common	2.34	2.34
glibc-gconv-extra	2.34	2.34
glibc-locale-source	2.34	2.34
gmp	6.2.1	6.2.1
gnupg2-minimal	2.3.7	2.3.7
gnutls	3.8.0	3.8.0
go-srpm-macros	3.2.0	3.2.0
gpgme	1.15.1	1.15.1
gpm-libs	1.20.7	1.20.7
grep	3.8	3.8
groff-base	1.22.4	1.22.4
grub2-common	2.06	2.06
grub2-efi-x64-ec2	2.06	2.06
grub2-pc		2.06
grub2-pc-modules	2.06	2.06
grub2-tools	2.06	2.06

套件	AMI	超 V 型 VHDX
grub2-tools-minimal	2.06	2.06
grubby	8.40	8.40
gssproxy	0.8.4	0.8.4
gzip	1.12	1.12
hostname	3.23	3.23
hunspell	1.7.0	1.7.0
hunspell-en	0.20140811.1	0.20140811.1
hunspell-en-GB	0.20140811.1	0.20140811.1
hunspell-en-US	0.20140811.1	0.20140811.1
hunspell-filesystem	1.7.0	1.7.0
hwdata	0.353	0.353
hyperv-daemons		0
hyperv-daemons-lic ense		0
hypervfcopyd		0
hypervkvpd		0
hyperv-tools		0
hypervvssd		0
info	6.7	6.7
inih	49	49
initscripts	10.09	10.09

套件	AMI	超 V 型 VHDX
iproute	5.10.0	5.10.0
iputils	20210202	20210202
irqbalance	1.9.0	1.9.0
jansson	2.14	2.14
jitterentropy	3.4.1	3.4.1
jq	1.7.1	1.7.1
json-c	0.14	0.14
kbd	2.4.0	2.4.0
kbd-misc	2.4.0	2.4.0
kernel	6.1.79	6.1.79
kernel-livepatch-r epo-cdn		2023.4.20240319
kernel-livepatch-r epo-s3	2023.4.20240319	
kernel-modules-extra		6.1.79
kernel-modules-ext ra-common		6.1.79
kernel-srpm-macros	1.0	1.0
kernel-tools	6.1.79	6.1.79
keyutils	1.6.3	1.6.3
keyutils-libs	1.6.3	1.6.3

套件	AMI	超 V 型 VHDX
kmod	29	29
kmod-libs	29	29
kpatch-runtime	0.9.7	0.9.7
krb5-libs	1.21	1.21
less	608	608
libacl	2.3.1	2.3.1
libaio	0.3.111	0.3.111
libarchive	3.5.3	3.5.3
libargon2	20171227	20171227
libassuan	2.5.5	2.5.5
libattr	2.5.1	2.5.1
libbasicobjects	0.1.1	0.1.1
libblkid	2.37.4	2.37.4
libcap	2.48	2.48
libcap-ng	0.8.2	0.8.2
libcbor	0.7.0	0.7.0
libcollection	0.7.0	0.7.0
libcom_err	1.46.5	1.46.5
libcomps	0.1.20	0.1.20
libconfig	1.7.2	1.7.2

套件	AMI	超 V 型 VHDX
libcurl-minimal	8.5.0	8.5.0
libdb	5.3.28	5.3.28
libdhash	0.5.0	
libdnf	0.69.0	0.69.0
libeconf	0.4.0	0.4.0
libedit	3.1	3.1
libev	4.33	4.33
libevent	2.1.12	2.1.12
libfdisk	2.37.4	2.37.4
libffi	3.4.4	3.4.4
libfido2	1.10.0	1.10.0
libgcc	11.4.1	11.4.1
libgcrypt	1.10.2	1.10.2
libgomp	11.4.1	11.4.1
libgpg-error	1.42	1.42
libibverbs	48.0	48.0
libidn2	2.3.2	2.3.2
libini_config	1.3.1	1.3.1
libkcapi	1.4.0	1.4.0
libkcapi-hmacalc	1.4.0	1.4.0

套件	AMI	超 V 型 VHDX
libldb	2.6.2	
libmaxminddb	1.5.2	1.5.2
libmetalink	0.1.3	0.1.3
libmnl	1.0.4	1.0.4
libmodulemd	2.13.0	2.13.0
libmount	2.37.4	2.37.4
libnfsidmap	2.5.4	2.5.4
libnghttp2	1.57.0	1.57.0
libnl3	3.5.0	3.5.0
libpath_utils	0.2.1	0.2.1
libpcap	1.10.1	1.10.1
libpipeline	1.5.3	1.5.3
libpkgconf	1.8.0	1.8.0
libpsl	0.21.1	0.21.1
libpwquality	1.4.4	1.4.4
libref_array	0.1.5	0.1.5
librepo	1.14.5	1.14.5
libreport-filesystem	2.15.2	2.15.2
libseccomp	2.5.3	2.5.3
libselinux	3.4	3.4

套件	AMI	超 V 型 VHDX
libselinux-utils	3.4	3.4
libsemanage	3.4	3.4
libsepol	3.4	3.4
libsigsegv	2.13	2.13
libsmartcols	2.37.4	2.37.4
libsolv	0.7.22	0.7.22
libss	1.46.5	1.46.5
libsss_certmap	2.9.4	
libsss_idmap	2.9.4	2.9.4
libsss_nss_idmap	2.9.4	2.9.4
libsss_sudo	2.9.4	
libstdc++	11.4.1	11.4.1
libstoragegmt	1.9.4	1.9.4
libtalloc	2.3.4	
libtasn1	4.19.0	4.19.0
libtdb	1.4.7	
libtevent	0.13.0	
libtextstyle	0.21	0.21
libtirpc	1.3.3	1.3.3
libunistring	0.9.10	0.9.10

套件	AMI	超 V 型 VHDX
libuser	0.63	0.63
libutempter	1.2.1	1.2.1
libuuid	2.37.4	2.37.4
libuv	1.47.0	1.47.0
libverto	0.3.2	0.3.2
libverto-libev	0.3.2	0.3.2
libxcrypt	4.4.33	4.4.33
libxml2	2.10.4	2.10.4
libyaml	0.2.5	0.2.5
libzstd	1.5.5	1.5.5
lm_sensors-libs	3.6.0	3.6.0
lmdb-libs	0.9.29	0.9.29
logrotate	3.20.1	3.20.1
lsof	4.94.0	4.94.0
lua-libs	5.4.4	5.4.4
lua-srpm-macros	1	1
lz4-libs	1.9.4	1.9.4
man-db	2.9.3	2.9.3
man-pages	5.10	5.10
microcode_ctl	2.1	2.1

套件	AMI	超 V 型 VHDX
mpfr	4.1.0	4.1.0
nano	5.8	5.8
ncurses	6.2	6.2
ncurses-base	6.2	6.2
ncurses-libs	6.2	6.2
nettle	3.8	3.8
net-tools	2.0	2.0
newt	0.52.21	0.52.21
nfs-utils	2.5.4	2.5.4
npth	1.6	1.6
nspr	4.35.0	4.35.0
nss	3.90.0	3.90.0
nss-softokn	3.90.0	3.90.0
nss-softokn-freebl	3.90.0	3.90.0
nss-sysinit	3.90.0	3.90.0
nss-util	3.90.0	3.90.0
ntsysv	1.15	1.15
numactl-libs	2.0.14	2.0.14
ocaml-srpm-macros	6	6
oniguruma	6.9.7.1	6.9.7.1

套件	AMI	超 V 型 VHDX
openblas-srpm-macros	2	2
openldap	2.4.57	2.4.57
openssh	8.7 小時	8.7 小時
openssh-clients	8.7 小時	8.7 小時
openssh-server	8.7 小時	8.7 小時
openssl	3.0.8	3.0.8
openssl-lib	3.0.8	3.0.8
openssl-pkcs11	0.4.12	0.4.12
os-prober	1.77	1.77
p11-kit	0.24.1	0.24.1
p11-kit-trust	0.24.1	0.24.1
package-notes-srpm-macros	0.4	0.4
pam	1.5.1	1.5.1
parted	3.4	3.4
passwd	0.80	0.80
pciutils	3.7.0	3.7.0
pciutils-lib	3.7.0	3.7.0
pcre2	10.40	10.40
pcre2-syntax	10.40	10.40
perl-Carp	1.50	1.50

套件	AMI	超 V 型 VHDX
perl-Class-Struct	0.66	0.66
perl-constant	1.33	1.33
perl-DynaLoader	1.47	1.47
perl-Encode	3.15	3.15
perl-Errno	1.30	1.30
perl-Exporter	5.74	5.74
perl-Fcntl	1.13	1.13
perl-File-Basename	2.85	2.85
perl-File-Path	2.18	2.18
perl-File-stat	1.09	1.09
perl-File-Temp	0.231.100	0.231.100
perl-Getopt-Long	2.52	2.52
perl-Getopt-Std	1.12	1.12
perl-HTTP-Tiny	0.078	0.078
perl-if	0.60.800	0.60.800
perl-interpreter	5.32.1	5.32.1
perl-IO	1.43	1.43
perl-IPC-Open3	1.21	1.21
perl-libs	5.32.1	5.32.1
perl-MIME-Base64	3.16	3.16

套件	AMI	超 V 型 VHDX
perl-mro	1.23	1.23
perl-overload	1.31	1.31
perl-overloading	0.02	0.02
perl-parent	0.238	0.238
perl-PathTools	3.78	3.78
perl-Pod-Escapes	1.07	1.07
perl-podlators	4.14	4.14
perl-Pod-Perldoc	3.28.01	3.28.01
perl-Pod-Simple	3.42	3.42
perl-Pod-Usage	2.01	2.01
perl-POSIX	1.94	1.94
perl-Scalar-List-Utils	1.56	1.56
perl-SelectSaver	1.02	1.02
perl-Socket	2.032	2.032
perl-srpm-macros	1	1
perl-Storable	3.21	3.21
perl-subst	1.03	1.03
perl-Symbol	1.08	1.08
perl-Term-ANSIColor	5.01	5.01
perl-Term-Cap	1.17	1.17

套件	AMI	超 V 型 VHDX
perl-Text-ParseWords	3.30	3.30
perl-Text-Tabs+Wrap	2021.0726	2021.0726
perl-Time-Local	1.300	1.300
perl-vars	1.05	1.05
pkgconf	1.8.0	1.8.0
pkgconf-m4	1.8.0	1.8.0
pkgconf-pkg-config	1.8.0	1.8.0
policycoreutils	3.4	3.4
policycoreutils-python-utils	3.4	
popt	1.18	1.18
procps-ng	3.3.17	3.3.17
protobuf-c	1.4.1	1.4.1
psacct	6.6.4	6.6.4
psmisc	23.4	23.4
publicsuffix-list-dafsa	20240212	20240212
python3	3.9.16	3.9.16
python3-attrs	20.3.0	20.3.0
python3-audit	3.0.6	3.0.6
python3-awscrt	0.19.19	0.19.19

套件	AMI	超 V 型 VHDX
python3-babel	2.9.1	2.9.1
python3-cffi	1.14.5	1.14.5
python3-chardet	4.0.0	4.0.0
python3-colorama	0.4.4	0.4.4
python3-configobj	5.0.6	5.0.6
python3-cryptography	36.0.1	36.0.1
python3-daemon	2.3.0	
python3-dateutil	2.8.1	2.8.1
python3-dbus	1.2.18	1.2.18
python3-distro	1.5.0	1.5.0
python3-dnf	4.14.0	4.14.0
python3-dnf-plugins-core	4.3.0	4.3.0
python3-docutils	0.16	0.16
python3-gpg	1.15.1	1.15.1
python3-hawkey	0.69.0	0.69.0
python3-idna	2.10	2.10
python3-jinja2	2.11.3	2.11.3
python3-jmespath	0.10.0	0.10.0
python3-jsonpatch	1.21	1.21
python3-jsonpointer	2.0	2.0

套件	AMI	超 V 型 VHDX
python3-jsonschema	3.2.0	3.2.0
python3-libcomps	0.1.20	0.1.20
python3-libdnf	0.69.0	0.69.0
python3-libs	3.9.16	3.9.16
python3-libselenium	3.4	3.4
python3-libsemanage	3.4	3.4
python3-libstorage mgmt	1.9.4	1.9.4
python3-lockfile	0.12.2	
python3-markupsafe	1.1.1	1.1.1
python3-netifaces	0.10.6	0.10.6
python3-oauthlib	3.0.2	3.0.2
python3-pip-wheel	21.3.1	21.3.1
python3-ply	3.11	3.11
python3-policycore utils	3.4	3.4
python3-prettytable	0.7.2	0.7.2
python3-prompt-too lkit	3.0.24	3.0.24
python3-pycparser	2.20	2.20
python3-pyrsistent	0.17.3	0.17.3

套件	AMI	超 V 型 VHDX
python3-pyserial	3.4	3.4
python3-pysocks	1.7.1	1.7.1
python3-pytz	2022.7.1	2022.7.1
python3-pyyaml	5.4.1	5.4.1
python3-requests	2.25.1	2.25.1
python3-rpm	4.16.1.3	4.16.1.3
python3-ruamel-yaml	0.16.6	0.16.6
python3-ruamel-yaml-clib	0.1.2	0.1.2
python3-setools	4.4.1	4.4.1
python3-setuptools	59.6.0	59.6.0
python3-setuptools-wheel	59.6.0	59.6.0
python3-six	1.15.0	1.15.0
python3-systemd	235	235
python3-urllib3	1.25.10	1.25.10
python3-wcwidth	0.2.5	0.2.5
python-chevron	0.13.1	
python-srpm-macros	3.9	3.9
quota	4.06	4.06
quota-nls	4.06	4.06

套件	AMI	超 V 型 VHDX
readline	8.1	8.1
rng-tools	6.14	6.14
rootfiles	8.1	8.1
rpcbind	1.2.6	1.2.6
rpm	4.16.1.3	4.16.1.3
rpm-build-libs	4.16.1.3	4.16.1.3
rpm-libs	4.16.1.3	4.16.1.3
rpm-plugin-selinux	4.16.1.3	4.16.1.3
rpm-plugin-systemd-inhibit	4.16.1.3	4.16.1.3
rpm-sign-libs	4.16.1.3	4.16.1.3
rsync	3.2.6	3.2.6
rust-srpm-macros	21	21
sbsigntools	0.9.4	0.9.4
screen	4.8.0	4.8.0
sed	4.8	4.8
selinux-policy	37.22	37.22
selinux-policy-targeted	37.22	37.22
setup	2.13.7	2.13.7
shadow-utils	4.9	4.9

套件	AMI	超 V 型 VHDX
slang	2.3.2	2.3.2
sqlite-libs	3.40.0	3.40.0
sssd-client	2.9.4	2.9.4
sssd-common	2.9.4	
sssd-kcm	2.9.4	
sssd-nfs-idmap	2.9.4	
strace	5.16	5.16
sudo	1.9.14	1.9.14
sysctl-defaults	1.0	1.0
sysstat	12.5.6	12.5.6
systemd	252.16	252.16
systemd-libs	252.16	252.16
systemd-networkd	252.16	252.16
systemd-pam	252.16	252.16
systemd-resolved	252.16	252.16
systemd-udev	252.16	252.16
system-release	2023.4.20240319	2023.4.20240319
systemtap-runtime	4.8	4.8
tar	1.34	1.34
tbb	2020.3	2020.3

套件	AMI	超 V 型 VHDX
tcpdump	4.99.1	4.99.1
tcsh	6.24.07	6.24.07
time	1.9	1.9
traceroute	2.1.3	2.1.3
tzdata	2024a	2024a
unzip	6.0	6.0
update-motd	2.2	2.2
userspace-rcu	0.12.1	0.12.1
util-linux	2.37.4	2.37.4
util-linux-core	2.37.4	2.37.4
vim-common	9.0.2153	9.0.2153
vim-data	9.0.2153	9.0.2153
vim-enhanced	9.0.2153	9.0.2153
vim-filesystem	9.0.2153	9.0.2153
vim-minimal	9.0.2153	9.0.2153
wget	1.21.3	1.21.3
which	2.21	2.21
words	3.0	3.0
xfsdump	3.1.11	3.1.11
xfspgrog	5.18.0	5.18.0

套件	AMI	超 V 型 VHDX
xxd	9.0.2153	9.0.2153
xxhash-libs	0.8.0	0.8.0
xz	5.2.5	5.2.5
xz-libs	5.2.5	5.2.5
yum	4.14.0	4.14.0
zip	3.0	3.0
zlib	1.2.11	1.2.11
zram-generator	1.1.2	
zram-generator-def aults	1.1.2	
zstd	1.5.5	1.5.5

更新 AL2023

與 AL2023 版本保持最新狀態很重要，以便您可以從安全性更新和新功能中受益。使用 AL2023 時，您可以透過 [透過 AL2023 上的版本化儲存庫使用確定性升級](#) 在整個環境中確保套件版本與更新之間的一致性。

主題

- [接收有關新更新的通知](#)
- [在 AL2023 中管理套件和作業系統更新](#)
- [透過 AL2023 上的版本化儲存庫使用確定性升級](#)
- [AL2023 上的核心即時修補程式](#)

接收有關新更新的通知

每當發布新的 AL2023 AMI 時，您都可以收到通知。通知將透過 [Amazon SNS](#) 並使用下列主題發布。

```
arn:aws:sns:us-east-1:137112412989:amazon-linux-2023-ami-updates
```

當發布新的 AL2023 AMI 時，會在此處貼出訊息。AMI 的版本將包含在訊息中。

這些訊息可以使用幾種不同的方法來接收。我們建議您使用以下方法。

1. 開啟 [Amazon SNS 主控台](#)。
2. 如有必要，請在導覽列中 AWS 區域 將變更為美國東部 (維吉尼亞北部)。您必須選取已訂閱且建立 SNS 通知的區域。
3. 在導覽面板中依序選擇 Subscriptions (訂閱) 與 Create subscription (建立訂閱)。
4. 針對 Create subscription (建立訂閱) 對話方塊，執行下列作業：
 - a. 對於主題 ARN，複製並粘貼以下 Amazon 資源名稱 (ARN)：。**arn:aws:sns:us-east-1:137112412989:amazon-linux-2023-ami-updates**
 - b. 對於 通訊協定，選擇 電子郵件。
 - c. 針對 Endpoint (端點)，請輸入可用於接收通知的電子郵件地址。
 - d. 選擇 Create subscription (建立訂閱)。
5. 您會收到一封含有「AWS 通知-訂閱確認」主旨行的確認電子郵件。開啟電子郵件並選擇 Confirm subscription (確認訂閱) 完成訂閱。

在 AL2023 中管理套件和作業系統更新

與以前版本的 Amazon Linux 不同，AL2023 AMI 被鎖定到 Amazon Linux 存儲庫的特定版本。若要將安全和錯誤修正套用至 AL2023 執行個體，請更新 DNF 組態。或者，啟動較新的 AL2023 執行個體。

本節說明如何在執行中的執行個體管理 DNF 套件和儲存庫。另外說明如何從使用者資料指令碼設定 DNF，以在啟動時啟用最新的 Amazon Linux 儲存庫。如需更多詳細資訊，請參閱 [DNF 命令參考](#)。

主題

- [檢查可用的套件更新](#)
- [使用 DNF 和儲存庫版本套用安全更新](#)
- [\(安全性 \) 更新後自動重新啟動服務](#)
- [啟動已啟用最新儲存庫版本的執行個體](#)
- [取得套件支援資訊](#)
- [檢查較新的儲存庫版本](#)
- [新增、啟用或停用新儲存庫](#)
- [使用 cloud-init 新增儲存庫](#)

檢查可用的套件更新

您可以使用 `dnf check-update` 命令來檢查系統是否有任何更新。對於 AL2023，建議您將此 `--releasever=version-number` 選項新增至命令。

當您新增此選項時，DNF 也會檢查儲存庫較新版本的更新。例如，執行 `dnf check-update` 命令後，請使用最新傳回的版本作為 `version-number` 的值。

如果執行個體已更新為使用最新版本的存放庫，則輸出會包含所有要更新之套件的清單。

Note

如果您沒有為 `dnf check-update` 命令指定有選用標誌的發行版本，則僅檢查目前設定的儲存庫版本。這表示不會檢查較新版本儲存庫中的套件。

```
$ sudo dnf check-update --releasever=2023.0.20230210
Last metadata expiration check: 0:06:13 ago on Mon 13 Feb 2023 10:39:32 PM UTC.
```

bind-libs.x86_64	32:9.16.27-1.amzn2023	amazonlinux
bind-license.noarch	32:9.16.27-1.amzn2023	amazonlinux
bind-utils.x86_64	32:9.16.27-1.amzn2023	amazonlinux
cloud-init.noarch	22.2.2-1.amzn2023.1.4	amazonlinux
dnf.noarch	4.12.0-2.amzn2023.0.1	amazonlinux
dnf-data.noarch	4.12.0-2.amzn2023.0.1	amazonlinux
dracut.x86_64	055-6.amzn2023.0.4	amazonlinux
dracut-config-generic.x86_64	055-6.amzn2023.0.4	amazonlinux
glib2.x86_64	2.73.2-678.amzn2023	amazonlinux
gmp.x86_64	1:6.2.1-2.amzn2023	amazonlinux
grep.x86_64	3.8-1.amzn2023.0.1	amazonlinux
kpatch-runtime.noarch	0.9.4-7.amzn2023	amazonlinux
libgcc.x86_64	11.3.1-2.amzn2023.0.6	amazonlinux
libgomp.x86_64	11.3.1-2.amzn2023.0.6	amazonlinux
libpkgconf.x86_64	1.7.3-7.amzn2023.0.1	amazonlinux
libstdc++.x86_64	11.3.1-2.amzn2023.0.6	amazonlinux
lz4-libs.x86_64	1.9.4-1.amzn2023	amazonlinux
pkgconf.x86_64	1.7.3-7.amzn2023.0.1	amazonlinux
pkgconf-m4.noarch	1.7.3-7.amzn2023.0.1	amazonlinux
pkgconf-pkg-config.x86_64	1.7.3-7.amzn2023.0.1	amazonlinux
python3-dnf.noarch	4.12.0-2.amzn2023.0.1	amazonlinux
python3-rpm.x86_64	4.16.1.3-12.amzn2023.0.2	amazonlinux
rpm.x86_64	4.16.1.3-12.amzn2023.0.2	amazonlinux
rpm-build-libs.x86_64	4.16.1.3-12.amzn2023.0.2	amazonlinux
rpm-libs.x86_64	4.16.1.3-12.amzn2023.0.2	amazonlinux
rpm-plugin-selinux.x86_64	4.16.1.3-12.amzn2023.0.2	amazonlinux
rpm-plugin-systemd-inhibit.x86_64	4.16.1.3-12.amzn2023.0.2	amazonlinux
rpm-sign-libs.x86_64	4.16.1.3-12.amzn2023.0.2	amazonlinux
slang.x86_64	2.3.2-9.amzn2023.0.1	amazonlinux
system-release.noarch	2023.0.20230210-0.amzn2023	amazonlinux
systemd.x86_64	250.8-1.amzn2023.0.1	amazonlinux
systemd-libs.x86_64	250.8-1.amzn2023.0.1	amazonlinux
systemd-networkd.x86_64	250.8-1.amzn2023.0.1	amazonlinux
systemd-pam.x86_64	250.8-1.amzn2023.0.1	amazonlinux
systemd-resolved.x86_64	250.8-1.amzn2023.0.1	amazonlinux
systemd-udev.x86_64	250.8-1.amzn2023.0.1	amazonlinux
vim-common.x86_64	2:9.0.327-1.amzn2023.0.1	amazonlinux
vim-data.noarch	2:9.0.327-1.amzn2023.0.1	amazonlinux
vim-enhanced.x86_64	2:9.0.327-1.amzn2023.0.1	amazonlinux
vim-filessystem.noarch	2:9.0.327-1.amzn2023.0.1	amazonlinux
vim-minimal.x86_64	2:9.0.327-1.amzn2023.0.1	amazonlinux
wget.x86_64	1.21.3-1.amzn2023	amazonlinux
yum.noarch	4.12.0-2.amzn2023.0.1	amazonlinux

對於這個命令，如果有更新的套件可用，傳回碼是 100。如果沒有更新的套件可用，傳回碼是 0。此外，輸出也會列出所有要更新的套件。

使用 DNF 和儲存庫版本套用安全更新

新套件更新和安全更新僅適用於新的儲存庫版本。對於從舊版 AL2023 AMI 啟動的執行個體，您必須先更新儲存庫版本，然後才能安裝安全更新。dnf check-release-update 命令包含範例更新命令，可將系統上安裝的所有套件更新為較新儲存庫中的版本。

```
$ sudo dnf update --releasever=2023.0.20230210
Last metadata expiration check: 0:01:40 ago on Mon 13 Feb 2023 10:39:32 PM UTC.
Dependencies resolved.
=====
Package                               Arch   Version                               Repository   Size
=====
Upgrading:
bind-libs                             x86_64 32:9.16.27-1.amzn2023                amazonlinux 1.2 M
bind-license                           noarch 32:9.16.27-1.amzn2023                amazonlinux 16 k
bind-utils                             x86_64 32:9.16.27-1.amzn2023                amazonlinux 202 k
cloud-init                             noarch 22.2.2-1.amzn2023.1.4                amazonlinux 1.1 M
dnf                                     noarch 4.12.0-2.amzn2023.0.1                amazonlinux 454 k
dnf-data                               noarch 4.12.0-2.amzn2023.0.1                amazonlinux 42 k
dracut                                  x86_64 055-6.amzn2023.0.4                   amazonlinux 345 k
dracut-config-generic                  x86_64 055-6.amzn2023.0.4                   amazonlinux 8.5 k
glib2                                   x86_64 2.73.2-678.amzn2023                  amazonlinux 2.7 M
gmp                                     x86_64 1:6.2.1-2.amzn2023                   amazonlinux 324 k
grep                                    x86_64 3.8-1.amzn2023.0.1                   amazonlinux 316 k
kpatch-runtime                         noarch 0.9.4-7.amzn2023                     amazonlinux 30 k
libgcc                                  x86_64 11.3.1-2.amzn2023.0.6                amazonlinux 121 k
libgomp                                 x86_64 11.3.1-2.amzn2023.0.6                amazonlinux 296 k
libpkgconf                             x86_64 1.7.3-7.amzn2023.0.1                 amazonlinux 37 k
libstdc++                              x86_64 11.3.1-2.amzn2023.0.6                amazonlinux 758 k
lz4-libs                               x86_64 1.9.4-1.amzn2023                     amazonlinux 81 k
pkgconf                                 x86_64 1.7.3-7.amzn2023.0.1                 amazonlinux 41 k
pkgconf-m4                             noarch 1.7.3-7.amzn2023.0.1                 amazonlinux 15 k
pkgconf-pkg-config                     x86_64 1.7.3-7.amzn2023.0.1                 amazonlinux 11 k
python3-dnf                             noarch 4.12.0-2.amzn2023.0.1                amazonlinux 415 k
python3-rpm                             x86_64 4.16.1.3-12.amzn2023.0.2             amazonlinux 89 k
rpm                                      x86_64 4.16.1.3-12.amzn2023.0.2             amazonlinux 487 k
rpm-build-libs                         x86_64 4.16.1.3-12.amzn2023.0.2             amazonlinux 92 k
rpm-libs                               x86_64 4.16.1.3-12.amzn2023.0.2             amazonlinux 311 k
rpm-plugin-selinux                     x86_64 4.16.1.3-12.amzn2023.0.2             amazonlinux 18 k
rpm-plugin-systemd-inhibit              x86_64 4.16.1.3-12.amzn2023.0.2             amazonlinux 19 k
```

```

rpm-sign-libs          x86_64 4.16.1.3-12.amzn2023.0.2  amazonlinux  22 k
slang                  x86_64 2.3.2-9.amzn2023.0.1      amazonlinux 410 k
system-release        noarch 2023.0.20230210-0.amzn2023 amazonlinux  25 k
systemd               x86_64 250.8-1.amzn2023.0.1     amazonlinux 4.2 M
systemd-libs          x86_64 250.8-1.amzn2023.0.1     amazonlinux 615 k
systemd-networkd      x86_64 250.8-1.amzn2023.0.1     amazonlinux 614 k
systemd-pam           x86_64 250.8-1.amzn2023.0.1     amazonlinux 335 k
systemd-resolved      x86_64 250.8-1.amzn2023.0.1     amazonlinux 277 k
systemd-udev          x86_64 250.8-1.amzn2023.0.1     amazonlinux 1.9 M
vim-common            x86_64 2:9.0.327-1.amzn2023.0.1 amazonlinux 7.2 M
vim-data             noarch 2:9.0.327-1.amzn2023.0.1 amazonlinux  27 k
vim-enhanced         x86_64 2:9.0.327-1.amzn2023.0.1 amazonlinux 1.8 M
vim-filesystem       noarch 2:9.0.327-1.amzn2023.0.1 amazonlinux  21 k
vim-minimal          x86_64 2:9.0.327-1.amzn2023.0.1 amazonlinux 764 k
wget                 x86_64 1.21.3-1.amzn2023        amazonlinux 813 k
yum                  noarch 4.12.0-2.amzn2023.0.1    amazonlinux  39 k

```

Transaction Summary

```

=====
Upgrade 43 Packages
...

```

您可以新增僅使用安全功能更新套件的 `--security` 選項。

```

$ sudo dnf update --releasever=2023.0.20230210 --security
Amazon Linux 2023 repository          18 MB/s | 11 MB    00:00
Last metadata expiration check: 0:00:02 ago on Mon 13 Feb 2023 10:39:32 PM UTC.
Dependencies resolved.
=====
Package                Arch      Version                                Repository      Size
=====
Upgrading:
bind-libs              x86_64   32:9.16.27-1.amzn2023                amazonlinux    1.2 M
bind-license           noarch   32:9.16.27-1.amzn2023                amazonlinux    16 k
bind-utils             x86_64   32:9.16.27-1.amzn2023                amazonlinux    202 k
gmp                   x86_64   1:6.2.1-2.amzn2023                   amazonlinux    324 k
lz4-libs              x86_64   1.9.4-1.amzn2023                     amazonlinux    81 k
vim-common            x86_64   2:9.0.327-1.amzn2023.0.1            amazonlinux    7.2 M
vim-data             noarch   2:9.0.327-1.amzn2023.0.1            amazonlinux    27 k
vim-enhanced         x86_64   2:9.0.327-1.amzn2023.0.1            amazonlinux    1.8 M
vim-filesystem       noarch   2:9.0.327-1.amzn2023.0.1            amazonlinux    21 k
vim-minimal          x86_64   2:9.0.327-1.amzn2023.0.1            amazonlinux    764 k
wget                 x86_64   1.21.3-1.amzn2023                    amazonlinux    813 k

```

Transaction Summary

=====

Upgrade 11 Packages

...

若要探索 AL2023 套件版本，請執行以下一或多個項目：

- 執行 `dnf check-update` 命令。
- 訂閱 Amazon Linux 儲存庫更新 SNS 主題 (`arn:aws:sns:us-east-1:137112412989:amazon-linux-2023-ami-updates`)。如需詳細資訊，請參閱《Amazon Simple Notification Service 開發人員指南》中的[訂閱 Amazon SNS 主題](#)。
- 定期參閱 [AL2023 版本備註](#)。

⚠ Important

當您將安全性更新套用至執行中的執行個體時，請確定 DNF 指向最新的儲存庫版本。

(安全性) 更新後自動重新啟動服務

Amazon Linux 現在隨附[智能重啟軟件包](#)。Smart-restart 每當使用系統套件管理員安裝或刪除套件時，系統更新時都會重新啟動 `systemd` 服務。每當執行時 `dnf (update|upgrade|downgrade)` 會發生這種情況。

Smart-restart 使用來源 `needs-restarting` 套件 `dnf-utils` 和自訂 `denylist` 機制來決定哪些服務需要重新啟動，以及是否建議重新啟動系統。如果建議重新啟動系統，則會產生重新啟動提示標記檔案 (`/run/smart-restart/reboot-hint-marker`)。

安裝 `smart-restart`

執行下列 DNF 命令 (如同使用任何其他套件一樣)。

```
$ sudo dnf install smart-restart
```

安裝之後，後續的事務將觸發 `smart-restart` 邏輯。

拒絕列表

Smart-restart 可以指示阻止某些服務重新啟動。被阻止的服務不會有助於決定是否需要重新啟動。若要封鎖其他服務，請新增具有尾碼 `-denylist` 的 `/etc/smart-restart-conf.d/` 檔案，如下列範例所示。

```
$ cat /etc/smart-restart-conf.d/custom-denylist
# Some comments
myservice.service
```

Note

決定是否需要重新開機時，會讀取和評估所有 `*-denylist` 檔案。

自定義掛鉤

除了 `denylist` 之外，還 `smart-restart` 提供了在嘗試重新啟動服務之前和之後執行自訂指令碼的機制。自訂指令碼可用來手動執行準備步驟，或通知其他元件剩餘或已完成的重新啟動。

`/etc/smart-restart-conf.d/` 帶有尾碼 `-pre-restart` 或的所有程序檔 `-post-restart` 都會被執行。如果順序很重要，請在所有指令碼前面加上數字，以確保執行順序，如下列範例所示。

```
$ ls /etc/smart-restart-conf.d/*-pre-restart
001-my-script-pre-restart
002-some-other-script-pre-restart
```

啟動已啟用最新儲存庫版本的執行個體

您可以將 DNF 命令新增至使用者資料指令碼，以控制啟動 Amazon Linux AMI 時要安裝哪些 RPM 套件。在下列範例中，使用者資料指令碼是用來確認以使用者資料指令碼啟動的任何執行個體都已安裝相同的套件更新。

```
#!/bin/bash
dnf update --releasever=2023.0.20230210
# Additional setup and install commands below
dnf install httpd php7.4 mysql80
```

您必須以超級使用者 (根使用者) 的身分執行此指令碼。若要進行這項動作，請執行以下命令。

```
$ sudo sh -c "bash nameofscript.sh"
```

如需詳細資訊，請參閱 Amazon EC2 使用者指南中的使用者資料和殼層指**令碼**。

Note

不要使用使用者資料指令碼，而是啟動最新的 Amazon Linux AMI 或以 Amazon Linux AMI 為基礎的自訂 AMI。最新的 Amazon Linux AMI 已安裝所有必要的更新，並且設定為指向特定的儲存庫版本。

取得套件支援資訊

AL2023 包含許多不同的開放原始碼軟體專案。這些專案中的每一個都是獨立於 Amazon Linux 進行管理，並具有不同的版本和 end-of-support 排程。為了向您提供有關這些不同套件的 Amazon Linux 特定資訊，DNFsupportinfo 外掛程式會提供有關套件的中繼資料。在下列範例中，**dnf supportinfo** 命令會傳回 glibc 套件的中繼資料。

```
$ sudo dnf supportinfo --pkg glibc
Last metadata expiration check: 0:07:56 ago on Wed Mar 1 23:21:49 2023.
Name           : glibc
Version        : 2.34-52.amzn2023.0.2
State          : installed
Support Status : supported
Support Periods : from 2023-03-15      : supported
                : from 2028-03-15      : unsupported
Support Statement : Amazon Linux 2023 End Of Life
Link           : https://aws.amazon.com/amazon-linux-ami/faqs/
Other Info      : This is the support statement for AL2023. The
                ...: end of life of Amazon Linux 2023 would be March 2028.
                ...: From this point, the Amazon Linux 2023 packages (listed
                ...: below) will no longer, receive any updates from AWS.
```

檢查較新的儲存庫版本

在 AL2023 執行個體中，您可以使用 DNF 公用程式來管理儲存庫和套用更新的 RPM 套件。這些套件可在 Amazon Linux 儲存庫中取得。您可以使用 DNF 命令 `dnf check-release-update` 來檢查 DNF 儲存庫的新版本。

```
$ sudo dnf check-release-update
WARNING:
  A newer release of "Amazon Linux" is available.
```

Available Versions:**Version 2023.0.20230210:**

Run the following command to update to 2023.0.20230210:

```
dnf update --releasever=2023.0.20230210
```

Release notes:

<https://docs.aws.amazon.com/linux/al2023/release-notes/relnotes.html>

這將傳回所有可用 DNF 儲存庫新版本的完整清單。如果沒有傳回任何內容，這代表 DNF 目前設定為使用最新的可用版本。目前安裝的 `system-release` 套件版本設定 `releasever` DNF 變數。若要檢查目前儲存庫版本，請執行以下命令。

```
$ rpm -q system-release --qf "%{VERSION}\n"
```

當您執行 DNF 套件交易 (例如安裝、更新或移除命令) 時，會出現警告訊息，通知您是否有任何新的儲存庫版本。例如，如果您在從舊版 AL2023 啟動的執行個體上安裝 `httpd` 套件，則會傳回下列輸出。

```
$ sudo dnf install httpd -y
```

```
Last metadata expiration check: 0:16:52 ago on Wed Mar 1 23:21:49 2023.
```

```
Dependencies resolved.
```

```
=====
Package                Arch   Version                               Repository   Size
=====
Installing:
httpd                   x86_64 2.4.54-3.amzn2023.0.4                amazonlinux  46 k
Installing dependencies:
apr                     x86_64 1.7.2-2.amzn2023.0.2                amazonlinux  129 k
apr-util                x86_64 1.6.3-1.amzn2023.0.1                amazonlinux  98 k
generic-logos-httpd
noarch                 18.0.0-12.amzn2023.0.3              amazonlinux  19 k
httpd-core              x86_64 2.4.54-3.amzn2023.0.4                amazonlinux  1.3 M
httpd-filesystem       noarch 2.4.54-3.amzn2023.0.4                amazonlinux  13 k
httpd-tools             x86_64 2.4.54-3.amzn2023.0.4                amazonlinux  80 k
libbrotli               x86_64 1.0.9-4.amzn2023.0.2                amazonlinux  315 k
mailcap                 noarch 2.1.49-3.amzn2023.0.3                amazonlinux  33 k
Installing weak dependencies:
apr-util-openssl       x86_64 1.6.3-1.amzn2023.0.1                amazonlinux  17 k
mod_http2              x86_64 1.15.24-1.amzn2023.0.3              amazonlinux  152 k
mod_lua                 x86_64 2.4.54-3.amzn2023.0.4                amazonlinux  60 k
```

Transaction Summary

=====

Install 12 Packages

Total download size: 2.3 M

Installed size: 6.8 M

Downloading Packages:

(1/12): apr-util-openssl-1.6.3-1.am	212 kB/s		17 kB	00:00
(2/12): apr-1.7.2-2.amzn2023.0.2.x8	1.1 MB/s		129 kB	00:00
(3/12): httpd-core-2.4.54-3.amzn202	8.9 MB/s		1.3 MB	00:00
(4/12): mod_http2-1.15.24-1.amzn202	1.9 MB/s		152 kB	00:00
(5/12): apr-util-1.6.3-1.amzn2023.0	1.7 MB/s		98 kB	00:00
(6/12): mod_lua-2.4.54-3.amzn2023.0	1.4 MB/s		60 kB	00:00
(7/12): httpd-2.4.54-3.amzn2023.0.4	1.5 MB/s		46 kB	00:00
(8/12): libbrotli-1.0.9-4.amzn2023.	4.4 MB/s		315 kB	00:00
(9/12): mailcap-2.1.49-3.amzn2023.0	753 kB/s		33 kB	00:00
(10/12): httpd-tools-2.4.54-3.amzn2	978 kB/s		80 kB	00:00
(11/12): httpd-filesystem-2.4.54-3.	210 kB/s		13 kB	00:00
(12/12): generic-logos-httpd-18.0.0	439 kB/s		19 kB	00:00

Total	6.6 MB/s		2.3 MB	00:00
-------	----------	--	--------	-------

Running transaction check

Transaction check succeeded.

Running transaction test

Transaction test succeeded.

Running transaction

Preparing	:		1/1
Installing	:	apr-1.7.2-2.amzn2023.0.2.x86_64	1/12
Installing	:	apr-util-openssl-1.6.3-1.amzn2023.0.1.	2/12
Installing	:	apr-util-1.6.3-1.amzn2023.0.1.x86_64	3/12
Installing	:	mailcap-2.1.49-3.amzn2023.0.3.noarch	4/12
Installing	:	httpd-tools-2.4.54-3.amzn2023.0.4.x86_	5/12
Installing	:	generic-logos-httpd-18.0.0-12.amzn2023	6/12
Running scriptlet:		httpd-filesystem-2.4.54-3.amzn2023.0.4	7/12
Installing	:	httpd-filesystem-2.4.54-3.amzn2023.0.4	7/12
Installing	:	httpd-core-2.4.54-3.amzn2023.0.4.x86_6	8/12
Installing	:	mod_http2-1.15.24-1.amzn2023.0.3.x86_6	9/12
Installing	:	libbrotli-1.0.9-4.amzn2023.0.2.x86_64	10/12
Installing	:	mod_lua-2.4.54-3.amzn2023.0.4.x86_64	11/12
Installing	:	httpd-2.4.54-3.amzn2023.0.4.x86_64	12/12
Running scriptlet:		httpd-2.4.54-3.amzn2023.0.4.x86_64	12/12
Verifying	:	apr-1.7.2-2.amzn2023.0.2.x86_64	1/12
Verifying	:	apr-util-openssl-1.6.3-1.amzn2023.0.1.	2/12
Verifying	:	httpd-core-2.4.54-3.amzn2023.0.4.x86_6	3/12

```

Verifying      : mod_http2-1.15.24-1.amzn2023.0.3.x86_64 4/12
Verifying      : apr-util-1.6.3-1.amzn2023.0.1.x86_64 5/12
Verifying      : mod_lua-2.4.54-3.amzn2023.0.4.x86_64 6/12
Verifying      : libbrotli-1.0.9-4.amzn2023.0.2.x86_64 7/12
Verifying      : httpd-2.4.54-3.amzn2023.0.4.x86_64 8/12
Verifying      : httpd-tools-2.4.54-3.amzn2023.0.4.x86_ 9/12
Verifying      : mailcap-2.1.49-3.amzn2023.0.3.noarch 10/12
Verifying      : httpd-filesystem-2.4.54-3.amzn2023.0.4 11/12
Verifying      : generic-logos-httpd-18.0.0-12.amzn2023 12/12

```

Installed:

```

apr-1.7.2-2.amzn2023.0.2.x86_64
apr-util-1.6.3-1.amzn2023.0.1.x86_64
apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
httpd-2.4.54-3.amzn2023.0.4.x86_64
httpd-core-2.4.54-3.amzn2023.0.4.x86_64
httpd-filesystem-2.4.54-3.amzn2023.0.4.noarch
httpd-tools-2.4.54-3.amzn2023.0.4.x86_64
libbrotli-1.0.9-4.amzn2023.0.2.x86_64
mailcap-2.1.49-3.amzn2023.0.3.noarch
mod_http2-1.15.24-1.amzn2023.0.3.x86_64
mod_lua-2.4.54-3.amzn2023.0.4.x86_64

```

Complete!

新增、啟用或停用新儲存庫

若要使用 DNF 套件管理系統從不同的儲存庫安裝套件，您需要為 `/etc/dnf/dnf.conf` 檔案或其自身位於 `repository.repo` 目錄中的 `/etc/yum.repos.d` 檔案新增儲存庫的資訊。您可以手動執行此作業。但是，大多數 DNF 儲存庫會在其儲存庫 URL 中提供自身的 `repository.repo` 檔案。

Note

目前沒有可新增至 AL2023 的其他儲存庫。這可能會在未來改變。此外，您可以編寫自有套件，並將這些套件提供給 AL2023 企業環境。您必須先新增並啟用儲存套件的儲存庫，才能使用套件。

若要了解目前啟用的儲存庫，可以執行以下命令：

```
$ dnf repolist all --verbose
```



```
Loaded plugins: builddep, changelog, config-manager, copr, debug, debuginfo-install,
download, generate_completion_cache, groups-manager, needs-restarting, playground,
release-notification, repoclosure, repodiff, repograph, repomanage, reposync,
supportinfo
```

```
DNF version: 4.12.0
```

```
cachedir: /var/cache/dnf
```

```
Last metadata expiration check: 0:00:02 ago on Wed Mar 1 23:40:15 2023.
```

```
Repo-id           : amazonlinux
Repo-name         : Amazon Linux 2023 repository
Repo-status      : enabled
Repo-revision    : 1677203368
Repo-updated     : Fri Feb 24 01:49:28 2023
Repo-pkgs        : 12632
Repo-available-pkgs: 12632
Repo-size        : 12 G
Repo-mirrors     : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/core/mirrors/2023.0.20230222/x86_64/mirror.list
Repo-baseurl     : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/core/guids/
cf9296325a6c46ff40c775a8e2d632c4c3fd9d9164014ce3304715d61b90ca8e/x86_64/
                  : (0 more)
Repo-expire      : 172800 second(s) (last: Wed Mar 1 23:40:15
                  : 2023)
Repo-filename    : /etc/yum.repos.d/amazonlinux.repo
```

```
Repo-id           : amazonlinux-debuginfo
Repo-name         : Amazon Linux 2023 repository - Debug
Repo-status      : disabled
Repo-mirrors     : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/core/mirrors/2023.0.20230222/debuginfo/x86_64/mirror.list
Repo-expire      : 21600 second(s) (last: unknown)
Repo-filename    : /etc/yum.repos.d/amazonlinux.repo
```

```
Repo-id           : amazonlinux-source
Repo-name         : Amazon Linux 2023 repository - Source packages
Repo-status      : disabled
Repo-mirrors     : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/core/mirrors/2023.0.20230222/SRPMS/mirror.list
Repo-expire      : 21600 second(s) (last: unknown)
Repo-filename    : /etc/yum.repos.d/amazonlinux.repo
```

```
Repo-id           : kernel-livepatch
Repo-name         : Amazon Linux 2023 Kernel Livepatch repository
Repo-status      : disabled
```

```
Repo-mirrors      : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/kernel-livepatch/mirrors/al2023/x86_64/mirror.list
Repo-expire       : 172800 second(s) (last: unknown)
Repo-filename     : /etc/yum.repos.d/kernel-livepatch.repo

Repo-id           : kernel-livepatch-source
Repo-name         : Amazon Linux 2023 Kernel Livepatch repository -
                  : Source packages
Repo-status       : disabled
Repo-mirrors      : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/kernel-livepatch/mirrors/al2023/SRPMS/mirror.list
Repo-expire       : 21600 second(s) (last: unknown)
Repo-filename     : /etc/yum.repos.d/kernel-livepatch.repo
Total packages: 12632
```

Note

如果您未新增 `--verbose` 選項旗標，輸出只會包含 `Repo-id`、`Repo-name` 和 `Repo-status` 資訊。

若要將 `yum` 儲存庫新增至 `/etc/yum.repos.d` 目錄：

1. 尋找 `.repo` 檔案的位置。在此範例中，`.repo` 檔案位於 <https://www.example.com/repository.repo>。
2. 使用 `dnf config-manager` 命令新增儲存庫。

```
$ sudo dnf config-manager --add-repo https://www.example.com/repository.repo
Loaded plugins: priorities, update-motd, upgrade-helper
adding repo from: https://www.example.com/repository.repo
grabbing file https://www.example.com/repository.repo to /etc/
yum.repos.d/repository.repo
repository.repo | 4.0 kB    00:00
repo saved to /etc/yum.repos.d/repository.repo
```

在您安裝儲存庫後，您必須啟用它，如下一個程序中所說明。

若要在 `/etc/yum.repos.d` 中啟用 `yum` 儲存庫，請使用有 `--enable` 旗標和 `###`名稱的 `dnf config-manager` 命令。

```
$ sudo dnf config-manager --enable repository
```

Note

若要停用儲存庫，請使用相同的命令語法，但在命令中以 `--disable` 取代 `--enable`。

使用 cloud-init 新增儲存庫

除了使用上一個方法新增儲存庫外，您還可以使用 cloud-init 架構來新增儲存庫。

若要新增套件儲存庫，建議您使用下列範本。請考慮在本機儲存此檔案。

```
#cloud-config
yum_repos:
  repository.repo:
    baseurl: https://www.example.com/
    enabled: true
    gpgcheck: true
    gpgkey: file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EXAMPLE
    name: Example Repository
```

Note

使用 cloud-init 的其中一個優點是您可以在設定檔案中加入 `packages:` 部分。在本節中，您可以加入要安裝的套件名稱。您可以從預設儲存庫或在 cloud-config 檔案中新增的新儲存庫安裝套件。

如需 YAML 檔案結構的詳細資訊，請參閱 `cloud-init` 文件中的 [新增 YUM 儲存庫](#)。

設定 YAML 格式檔案後，您可以在 cloud-init 的 AWS CLI 架構中執行。務必加入 `--userdata` 選項和 `.yaml` 檔案的名稱，以呼叫所需的操作。

```
$ aws ec2 run-instances \
  --image-id \
  resolve:ssm:/aws/service/ami-amazon-linux-latest/a12023-ami-kernel-default-x86_64 \
  --instance-type m5.xlarge \
  --region us-east-1 \
  --key-name aws-key-us-east-1 \
  --security-group-ids sg-004a7650 \
```

```
--user-data file://cloud-config.yml
```

透過 AL2023 上的版本化儲存庫使用確定性升級

Note

根據預設，AL2023 執行個體不會在啟動時自動接收其他關鍵和重要的安全性更新。執行個體最初包含 AL2023 和所選 AMI 中可用的更新。

控制從主要和次要版本收到的更新

使用 AL2023 時，您可以在整個環境中確保套件版本與更新之間的一致性。您也可以確保相同 Amazon Machine Image (AMI) 的多個執行個體的一致性。透過版本化儲存庫功能進行確定性升級 (預設為開啟) 時，您可以根據所需排程套用更新。

每當我們發布新的套件更新時，都會有一個新版本要鎖定，以及鎖定到該版本的新 AMI。

AL2023 會鎖定到儲存庫的特定版本。主要或次要版本都支援此功能。透過 SSM 參數公開的 AL2023 AMI 永遠是最新版本。它擁有最多的 up-to-date 軟件包和更新，包括關鍵和重要的安全性更新。

如果您從現有 AMI 啟動執行個體，則不會自動套用更新。在佈建時安裝的任何其他套件都會對應至現有 AMI 的儲存庫版本。

使用此功能時，您可以確保整個環境中的套件版本和更新之間的一致性。如果您要從同一 AMI 啟動多個執行個體，則尤其如此。您可以根據所需排程來套用更新。您也可以在此時套用一組特定的更新，因為這些更新也可以鎖定至特定的儲存庫版本。

主要和次要版本升級之間的差異

AL2023 的主要版本包含大規模更新，且可能會新增、刪除或更新套件。為確保相容性，請在您在該版本上測試應用程式後，再將執行個體升級至新的主要版本。

AL2023 的次要版本包含功能和安全性更新，但不包括套件變更。這樣可以確保 Linux 功能和系統程式庫 API 在新版本上保持可用。不需要在更新前測試您的應用程式。

控制 AL2023 儲存庫中可用的套件更新

當我們發布 AL2023 儲存庫的新版本時，所有以前的版本仍然可用。預設情況下，用於管理儲存庫版本的外掛程式會鎖定到用來建立 AMI 的相同版本。如果您想要控制套件更新，請遵循下列步驟。

1. 執行下列命令來探索可用的儲存庫版本。

```
$ sudo dnf check-release-update
```

2. 執行以下命令來選取版本。

```
$ sudo dnf --releasever=version update
```

此命令會使用 `dnf` 並從目前 Amazon Linux 發行版本，開始更新到命令列指定的發行版本。套件更新的清單會由顯示 `dnf`。在處理更新之前，您必須確認更新。更新完成後，新發行版本會成為 `dnf` 用於所有未來活動的預設發行版本。

如需更多詳細資訊，請參閱 [在 AL2023 中管理套件和作業系統更新](#)。

透過使用版本化儲存庫進行確定性升級

主題

- [使用確定性升級的系統](#)
- [確定性升級系統的選擇性更新](#)
- [使用具決定性升級的持續覆寫](#)

使用確定性升級的系統

當您執行 `dnf upgrade` 命令時，系統會檢查 `releasever` 變數指定的儲存庫中是否有升級。####
`#releasever##### 2023.4.20240513#`

您可以使用下列其中一種方法，來變更 `releasever` 的值。這些方法以降序系統優先順序列出。這表示方法 1 會覆寫方法 2 和方法 3，而方法 2 會覆寫方法 3。

1. 命令行旗標中的值 `--releasever=latest` (如有使用)。
2. 在覆寫變數檔案中指定的值 `/etc/dnf/vars/releasever` (如已設定)。
3. 目前安裝的 `system-release` 套件版本。

在下列範例中，版本為 `2023.0.20230210`：

```
$ rpm -q system-release
system-release-2023.0.20230210-0.amzn2023.noarch
```

在新安裝的系統中，覆寫變數不存在。沒有可用的升級，因為系統已鎖定至 `system-release` 的已安裝版本。

```
$ cat /etc/dnf/vars/releasever
cat: /etc/dnf/vars/releasever: No such file or directory
```

```
$ sudo dnf upgrade
Last metadata expiration check: 0:00:02 ago on Wed 15 Feb 2023 06:14:12 PM UTC.
Dependencies resolved.
Nothing to do.
Complete!
```

您可以使用 `releasever` 旗標來提供您想要的版本，以取得特定版本的套件。

```
$ rpm -q system-release
system-release-2023.0.20230222-0.amzn2023.noarch
```

```
$ sudo dnf upgrade --releasever=2023.0.20230329
Amazon Linux 2023 repository                26 MB/s | 12 MB    00:00
Dependencies resolved.
=====
Package                Arch    Version                                Repository    Size
=====
Installing:
kernel                  aarch64 6.1.21-1.45.amzn2023                  amazonlinux  26 M
Upgrading:
amazon-linux-repo-s3    noarch  2023.0.20230329-0.amzn2023            amazonlinux  18 k
ca-certificates        noarch  2023.2.60-1.0.amzn2023.0.1           amazonlinux  828 k
cloud-init              noarch  22.2.2-1.amzn2023.1.7                 amazonlinux  1.1 M

... [ list edited for clarity ]

system-release          noarch  2023.0.20230329-0.amzn2023            amazonlinux  29 k

... [ list edited for clarity ]

vim-data                noarch  2:9.0.1403-1.amzn2023.0.1            amazonlinux  25 k
vim-minimal             aarch64 2:9.0.1403-1.amzn2023.0.1            amazonlinux  753 k

Transaction Summary
=====
```

```
Install    1 Package
Upgrade   42 Packages
```

```
Total download size: 56 M
```

由於 `--releasever` 選項會同時覆寫 `system-release` 和 `/etc/dnf/vars/releasever`，因此此升級的結果如下：

1. 升級會取代所有在舊版和新版本之間變更的已安裝套件。
2. 升級會將系統鎖定至儲存庫的新版本 `system-release`。

確定性升級系統的選擇性更新

您可能想要安裝最新版本的特定套件，並將系統鎖定為原始發行版本。

您可以使用 `dnf check-update` 來找出要升級的套件。

```
$ sudo dnf check-update --releasever=latest --security
Amazon Linux 2023 repository                13 MB/s | 10 MB    00:00
Last metadata expiration check: 0:00:02 ago on Wed 15 Feb 2023 02:52:21 AM UTC.

bind-libs.aarch64                          32:9.16.27-1.amzn2023.0.1      amazonlinux
bind-license.noarch                        32:9.16.27-1.amzn2023.0.1      amazonlinux
bind-utils.aarch64                        32:9.16.27-1.amzn2023.0.1      amazonlinux
cryptsetup.aarch64                        2.4.3-2.amzn2023.0.1          amazonlinux
cryptsetup-libs.aarch64                   2.4.3-2.amzn2023.0.1          amazonlinux
curl-minimal.aarch64                      7.85.0-1.amzn2023.0.1         amazonlinux
glibc.aarch64                              2.34-40.amzn2023.0.2          amazonlinux
glibc-all-langpacks.aarch64               2.34-40.amzn2023.0.2          amazonlinux
glibc-common.aarch64                      2.34-40.amzn2023.0.2          amazonlinux
glibc-locale-source.aarch64              2.34-40.amzn2023.0.2          amazonlinux
gmp.aarch64                                1:6.2.1-2.amzn2023.0.1        amazonlinux
gnupg2-minimal.aarch64                    2.3.7-1.amzn2023.0.2          amazonlinux
gzip.aarch64                              1.10-5.amzn2023.0.1           amazonlinux
kernel.aarch64                             6.1.12-17.42.amzn2023         amazonlinux
kernel-tools.aarch64                      6.1.12-17.42.amzn2023         amazonlinux
libarchive.aarch64                        3.5.3-2.amzn2023.0.1          amazonlinux
libcurl-minimal.aarch64                   7.85.0-1.amzn2023.0.1         amazonlinux
libsepol.aarch64                          3.4-3.amzn2023.0.2            amazonlinux
libsolv.aarch64                           0.7.22-1.amzn2023.0.1         amazonlinux
libxml2.aarch64                           2.9.14-1.amzn2023.0.1         amazonlinux
```

logrotate.aarch64	3.20.1-2.amzn2023.0.2	amazonlinux
lua-libs.aarch64	5.4.4-3.amzn2023.0.1	amazonlinux
lz4-libs.aarch64	1.9.4-1.amzn2023.0.1	amazonlinux
openssl.aarch64	1:3.0.5-1.amzn2023.0.3	amazonlinux
openssl-libs.aarch64	1:3.0.5-1.amzn2023.0.3	amazonlinux
pcre2.aarch64	10.40-1.amzn2023.0.1	amazonlinux
pcre2-syntax.noarch	10.40-1.amzn2023.0.1	amazonlinux
rsync.aarch64	3.2.6-1.amzn2023.0.2	amazonlinux
vim-common.aarch64	2:9.0.475-1.amzn2023.0.1	amazonlinux
vim-data.noarch	2:9.0.475-1.amzn2023.0.1	amazonlinux
vim-enhanced.aarch64	2:9.0.475-1.amzn2023.0.1	amazonlinux
vim-filesystem.noarch	2:9.0.475-1.amzn2023.0.1	amazonlinux
vim-minimal.aarch64	2:9.0.475-1.amzn2023.0.1	amazonlinux
xz.aarch64	5.2.5-9.amzn2023.0.1	amazonlinux
xz-libs.aarch64	5.2.5-9.amzn2023.0.1	amazonlinux
zlib.aarch64	1.2.11-32.amzn2023.0.3	amazonlinux

安裝您要升級的套件。使用 `sudo dnf upgrade --releasever=latest` 和套件名稱來確保 `system-release` 套件維持不變。

```
$ sudo dnf upgrade --releasever=latest openssl openssl-libs
```

```
Last metadata expiration check: 0:01:28 ago on Wed 15 Feb 2023 02:52:21 AM UTC.
Dependencies resolved.
```

```
=====
Package           Arch           Version                Repository           Size
=====
Upgrading:
openssl           aarch64       1:3.0.5-1.amzn2023.0.3  amazonlinux         1.1 M
openssl-libs     aarch64       1:3.0.5-1.amzn2023.0.3  amazonlinux         2.1 M
```

```
Transaction Summary
```

```
=====
Upgrade 2 Packages
```

```
Total download size: 3.2 M
```

Note

使用 `sudo dnf upgrade --releasever=latest` 更新所有套件，包括 `system-release`。然後，除非您設定持續取代，否則版本會保持鎖定為新的 `system-release`。

使用具決定性升級的持續覆寫

透過將變數值設定為`##`，您就能使用持續覆寫來解鎖系統，而不是加入`--releasever=latest`。

```
$ echo latest | sudo tee /etc/dnf/vars/releasever
latest
```

```
$ sudo dnf upgrade
```

```
Last metadata expiration check: 0:03:36 ago on Wed 15 Feb 2023 02:52:21 AM UTC.
Dependencies resolved.
```

```
=====
Package                Arch    Version                                Repository    Size
=====
Installing:
kernel                  aarch64 6.1.73-45.135.amzn2023                amazonlinux   24 M
Upgrading:
acl                     aarch64 2.3.1-2.amzn2023.0.1                  amazonlinux   72 k
alternatives            aarch64 1.15-2.amzn2023.0.1                   amazonlinux   36 k
amazon-ec2-net-utils   noarch  2.3.0-1.amzn2023.0.1                  amazonlinux   16 k
at                      aarch64 3.1.23-6.amzn2023.0.1                 amazonlinux   60 k
attr                   aarch64 2.5.1-3.amzn2023.0.1                  amazonlinux   59 k
audit                  aarch64 3.0.6-1.amzn2023.0.1                  amazonlinux   249 k
audit-libs             aarch64 3.0.6-1.amzn2023.0.1                  amazonlinux   116 k
aws-c-auth-libs        aarch64 0.6.5-6.amzn2023.0.2                  amazonlinux   79 k
aws-c-cal-libs         aarch64 0.5.12-7.amzn2023.0.2                 amazonlinux   34 k
aws-c-common-libs      aarch64 0.6.14-6.amzn2023.0.2                 amazonlinux   119 k
aws-c-compression-libs aarch64 0.2.14-5.amzn2023.0.2                 amazonlinux   22 k
aws-c-event-stream-libs aarch64 0.2.7-5.amzn2023.0.2                 amazonlinux   47 k
aws-c-http-libs        aarch64 0.6.8-6.amzn2023.0.2                 amazonlinux   147 k
aws-c-io-libs          aarch64 0.10.12-5.amzn2023.0.6                amazonlinux   109 k
aws-c-mqtt-libs        aarch64 0.7.8-7.amzn2023.0.2                 amazonlinux   61 k
aws-c-s3-libs          aarch64 0.1.27-5.amzn2023.0.3                 amazonlinux   54 k
aws-c-sdkutils-libs    aarch64 0.1.1-5.amzn2023.0.2                 amazonlinux   26 k
aws-checksums-libs     aarch64 0.1.12-5.amzn2023.0.2                 amazonlinux   50 k
awscli-2               noarch  2.7.8-1.amzn2023.0.4                  amazonlinux   7.3 M
basesystem             noarch  11-11.amzn2023.0.1                    amazonlinux   7.8 k
bash                   aarch64 5.1.8-2.amzn2023.0.1                  amazonlinux   1.6 M
bash-completion        noarch  1:2.11-2.amzn2023.0.1                 amazonlinux   292 k
bc                     aarch64 1.07.1-14.amzn2023.0.1                amazonlinux   120 k
bind-libs              aarch64 32:9.16.27-1.amzn2023.0.1            amazonlinux   1.2 M
bind-license           noarch  32:9.16.27-1.amzn2023.0.1            amazonlinux   14 k
bind-utils             aarch64 32:9.16.27-1.amzn2023.0.1            amazonlinux   206 k
binutils               aarch64 2.38-20.amzn2023.0.3                  amazonlinux   4.6 M
=====
```

```

boost-filesystem      aarch64 1.75.0-4.amzn2023.0.1      amazonlinux 55 k
boost-system          aarch64 1.75.0-4.amzn2023.0.1      amazonlinux 14 k
boost-thread          aarch64 1.75.0-4.amzn2023.0.1      amazonlinux 54 k
bzip2                 aarch64 1.0.8-6.amzn2023.0.1      amazonlinux 53 k
bzip2-libs            aarch64 1.0.8-6.amzn2023.0.1      amazonlinux 44 k
c-ares                aarch64 1.17.2-1.amzn2023.0.1      amazonlinux 107 k
ca-certificates       noarch 2021.2.50-1.0.amzn2023.0.3  amazonlinux 343 k
checkpolicy           aarch64 3.4-3.amzn2023.0.1         amazonlinux 345 k
chkconfig             aarch64 1.15-2.amzn2023.0.1        amazonlinux 162 k
chrony                aarch64 4.2-7.amzn2023.0.4         amazonlinux 314 k
cloud-init            noarch 22.2.2-1.amzn2023.1.7      amazonlinux 1.1 M
cloud-utils-growpart aarch64 0.31-8.amzn2023.0.2        amazonlinux 31 k
coreutils             aarch64 8.32-30.amzn2023.0.2        amazonlinux 1.1 M
coreutils-common      aarch64 8.32-30.amzn2023.0.2        amazonlinux 2.0 M
cpio                  aarch64 2.13-10.amzn2023.0.1       amazonlinux 269 k
cracklib              aarch64 2.9.6-27.amzn2023.0.1      amazonlinux 83 k
cracklib-dicts        aarch64 2.9.6-27.amzn2023.0.1      amazonlinux 3.6 M
crontabs              noarch 1.11-24.20190603git.amzn2023.0.1
                                                                amazonlinux 19 k
crypto-policies       noarch 20230128-1.gitdfb10ea.amzn2023.0.1
                                                                amazonlinux 61 k
crypto-policies-scripts noarch 20230128-1.gitdfb10ea.amzn2023.0.1
                                                                amazonlinux 81 k
...
Installing dependencies:
amazon-linux-repo-cdn noarch 2023.0.20230210-0.amzn2023  amazonlinux 16 k
xxhash-libs          aarch64 0.8.0-3.amzn2023.0.1        amazonlinux 32 k
Installing weak dependencies:
amazon-chrony-config noarch 4.2-7.amzn2023.0.4         amazonlinux 14 k
gawk-all-langpacks  aarch64 5.1.0-3.amzn2023.0.1      amazonlinux 207 k

```

Transaction Summary

```

=====
Install    5 Packages
Upgrade   413 Packages

```

Total download size: 199 M

Note

如果您使用覆寫變數 `/etc/dnf/vars/releasever`，請使用以下命令，以透過刪除覆寫值來還原預設鎖定行為。

```
$ sudo rm /etc/dnf/vars/releasever
```

AL2023 上的核心即時修補程式

您可以使用 AL2023 的核心即時修補，將安全性弱點和重大錯誤修補程式套用至執行中的 Linux 核心，而不必重新開機或中斷執行中的應用程式。此外，Kernel Live Patching 可以協助改善應用程式的可用性，同時保持基礎設施的安全和最新狀態。

AWS 為 AL2023 發布兩種類型的內核實時補丁：

- 安全性更新 – 包含 Linux 常見漏洞和入侵程式 (CVE) 的更新。這些更新通常會使用 Amazon Linux 安全建議分級評定為 important (重要) 或 critical (嚴重)。它們通常對應到通用漏洞評分系統 (CVSS) 分數的 7 及以上。在某些情況下，AWS 可能會在指派 CVE 之前提供更新。在這些情況下，修補程式可能會顯示為錯誤修正。
- 錯誤修正 – 包含與 CVE 無關的重大錯誤和穩定性問題修正。

AWS 提供 AL2023 核心版本的核心即時修補程式，最長可在其發行後 3 個月。經過此期間後，您必須更新至更新的核心版本，才能繼續接收核心即時修補程式。

AL2023 核心即時修補程式會在現有 AL2023 軟體庫中以已簽署的 RPM 套件形式提供。這些修補程式可以透過現有 DNF 套件管理員工作流程安裝至個別執行個體。或者，您也可以使用 AWS Systems Manager 員將它們安裝在一組代管執行個體上。

AL2023 的 Kernel Live Patching 以無需額外費用的方式提供。

主題

- [限制](#)
- [支援的組態和先決條件](#)
- [使用 Kernel Live Patching](#)

限制

套用核心即時修補程式時，您無法執行休眠、使用進階偵錯工具 (例如 SystemTap、kprobes 和 eBPF 為基礎的工具)，或存取 Kernel Live Patching 基礎架構所使用的 ftrace 輸出檔案。

支援的組態和先決條件

Amazon EC2 執行個體和執行 AL2023 的內部部署虛擬機器支援 Kernel Live Patching。

若要在 AL2023 上使用 Kernel Live Patching，您必須使用下列：

- 64 位元 x86_64 或 ARM64 架構
- 核心版本 6.1

政策要求

若要從 AL2023 儲存庫下載套件，Amazon EC2 需要存取服務擁有的 Amazon S3 儲存貯體。如果您在環境中使用適用於 Amazon S3 的 Amazon 虛擬私有雲端 (VPC) 端點，請確保您的 VPC 端點政策允許存取這些公用儲存貯體。下表說明 Amazon EC2 在進行核心即時修補時可能需要存取的 Amazon S3 儲存貯體。

S3 儲存貯體 ARN	描述
<i>ARN: AWS::3::###-##-de612dc2/*</i>	包含 AL2023 儲存庫的 Amazon S3 貯體

使用 Kernel Live Patching

您可以透過執行個體本身的命令列，在個別執行個體啟用和使用 Kernel Live Patching。或者，您也可以使用 AWS 系統管理員，在受管執行個體群組啟用和使用 Kernel Live Patching。

下列各節說明如何透過命令列在個別執行個體上啟用並使用 Kernel Live Patching。

如需有關在受管執行個體群組上啟用並使用 Kernel Live Patching 的詳細資訊，請參閱 AWS Systems Manager 使用者指南中的 [在 AL2023 執行個體上使用 Kernel Live Patching](#)。

主題

- [啟用 Kernel Live Patching](#)
- [檢視可用的核心即時修補程式](#)
- [套用核心即時修補程式](#)
- [檢視套用的核心即時修補程式](#)
- [停用核心即時修補](#)

啟用 Kernel Live Patching

Kernel Live Patching 在 AL2023 上預設為停用狀態。若要使用即時修補，您必須安裝 Kernel Live Patching 的 DNF 外掛程式，並啟用即時修補功能。

啟用 Kernel Live Patching

1. 核心即時修補程式適用於搭載核心版本 6.1 的 AL2023。若要檢查您的核心版本，請執行下列命令。

```
$ sudo dnf list kernel
```

2. 安裝 Kernel Live Patching 的 DNF 外掛程式。

```
$ sudo dnf install -y kpatch-dnf
```

3. 啟用 Kernel Live Patching 的 DNF 外掛程式。

```
$ sudo dnf kernel-livepatch -y auto
```

此命令也會從設定的軟體庫安裝最新版本的核心即時修補程式 RPM。

4. 若要確認核心即時修補的 DNF 外掛程式是否成功安裝，請執行下列命令。

當您啟用 Kernel Live Patching 時，系統會自動套用空白的核心即時修補程式 RPM。如果 Kernel Live Patching 已順利啟用，此命令會傳回包含初始空白核心即時修補程式 RPM 的清單。

```
$ sudo rpm -qa | grep kernel-livepatch
dnf-plugin-kernel-livepatch-1.0-0.11.amzn2023.noarch
kernel-livepatch-6.1.12-17.42-1.0-0.amzn2023.x86_64
```

5. 安裝 kpatch 套件。

```
$ sudo dnf install -y kpatch-runtime
```

6. 如果之前已安裝 kpatch 服務，請更新此服務。

```
$ sudo dnf update kpatch-runtime
```

7. 啟動 kpatch 服務。此服務會在初始化或開機時載入所有核心即時修補程式。

```
$ sudo systemctl enable kpatch.service && sudo systemctl start kpatch.service
```

檢視可用的核心即時修補程式

Amazon Linux 安全性警示會發佈至資訊 Amazon Linux 安全中心。如需有關 AL2023 安全性警示 (包括核心即時修補程式警示) 的詳細資訊，請參閱 [Amazon Linux 安全中心](#)。核心即時修補程式的字首為 ALASLIVEPATCH。Amazon Linux 安全中心可能不會列出解決錯誤的核心即時修補程式。

您也可以使用命令列來探索建議和 CVE 的可用核心即時修補程式。

列出所有可用的核心即時修補程式以供建議使用

使用下列 命令。

```
$ sudo dnf updateinfo list
Last metadata expiration check: 1:06:23 ago on Mon 13 Feb 2023 09:28:19 PM UTC.
ALAS2LIVEPATCH-2021-123    important/Sec. kernel-
livepatch-6.1.12-17.42-1.0-4.amzn2023.x86_64
ALAS2LIVEPATCH-2022-124    important/Sec. kernel-
livepatch-6.1.12-17.42-1.0-3.amzn2023.x86_64
```

列出 CVE 的所有可用核心即時修補程式

使用下列 命令。

```
$ sudo dnf updateinfo list cves
Last metadata expiration check: 1:07:26 ago on Mon 13 Feb 2023 09:28:19 PM UTC.
CVE-2022-0123    important/Sec. kernel-livepatch-6.1.12-17.42-1.0-4.amzn2023.x86_64
CVE-2022-3210    important/Sec. kernel-livepatch-6.1.12-17.42-1.0-3.amzn2023.x86_64
```

套用核心即時修補程式

您可以使用 DNF 套件管理員來套用核心即時修補程式，方法與套用定期更新相同。Kernel Live Patching 的 DNF 外掛程式會管理您要套用的核心即時修補程式，而不需要重新開機。

Tip

建議您定期使用 Kernel Live Patching 來更新您的核心，以確保安全與最新狀態。

您可以選擇套用特定的核心即時修補程式，或套用任何可用的核心即時修補程式，以及定期的安全性更新。

套用特定核心即時修補程式

1. 使用 [檢視可用的核心即時修補程式](#) 中描述的其中一個命令取得核心即時修補程式版本。
2. 為您的 AL2023 核心套用核心即時修補程式。

```
$ sudo dnf install kernel-livepatch-kernel_version-package_version.amzn2023.x86_64
```

例如，下列命令會針對 AL2023 核心版本 6.1.12-17.42 套用核心即時修補程式。

```
$ sudo dnf install kernel-livepatch-6.1.12-17.42-1.0-4.amzn2023.x86_64
```

套用任何可用的核心即時修補程式，以及您的定期安全性更新

使用下列命令。

```
$ sudo dnf update --security
```

省略包含錯誤修正的 `--security` 選項。

Important

- 套用核心即時修補程式後，核心版本不會更新。只有在執行個體重新啟動後，版本才會更新為新版本。
- AL2023 核心會接收核心即時修補程式，為期三個月。經過此期間後，該核心版本不會發行任何新的核心即時修補程式。
- 若要在 3 個月後繼續接收核心即時修補程式，您必須重新啟動執行個體，才能轉換到新的核心版本。執行個體會在您更新核心後的 3 個月內繼續收到核心即時修補程式。
- 若要檢查核心版本的支援視窗，請執行下列命令：

```
$ sudo dnf kernel-livepatch support
```

檢視套用的核心即時修補程式

檢視套用的核心即時修補程式

使用下列命令。

```
$ sudo kpatch list
Loaded patch modules:
livepatch_CVE_2022_36946 [enabled]

Installed patch modules:
livepatch_CVE_2022_36946 (6.1.57-29.131.amzn2023.x86_64)
livepatch_CVE_2022_36946 (6.1.57-30.131.amzn2023.x86_64)
```

此命令會傳回已載入及已安裝的安全性更新核心即時修補程式清單。下列為範例輸出。

Note

單一核心即時修補程式可包含並安裝多個即時修補程式。

停用核心即時修補

如果您不再需要使用 Kernel Live Patching，您可以隨時停用它。

- 停用使用 livepatches：

1. 停用外掛程式：

```
$ sudo dnf kernel-livepatch manual
```

2. 停用 kpatch 服務：

```
$ sudo systemctl disable --now kpatch.service
```

- 完全移除 livepatch 工具：

1. 移除外掛程式：

```
$ sudo dnf remove kpatch-dnf
```

2. 移除 kpatch-runtime：


```
$ sudo dnf remove kpatch-runtime
```

3. 移除任何已安裝的 livepatches :

```
$ sudo dnf remove kernel-livepatch\*
```

開始使用 AL2023 上的程式設計執行階段

AL2023 提供了某些語言運行時的不同版本。我們與同時支持多個版本的上游項目合作。了解有關如何使用 `dnf` 命令來搜尋和安裝這些套件，以安裝和管理這些名稱版本化套件的資訊。

以下主題概述了 AL2023 中每種語言生態系統的存在方式。

主題

- [AL2023 中的 C、C++ 和 Fortran](#)
- [AL2023 中的 Go](#)
- [AL2023 中的 Java](#)
- [AL2023 中的 Perl](#)
- [AL2023 中的 PHP](#)
- [AL2023 中的 Python](#)
- [AL2023 中的 Rust](#)

AL2023 中的 C、C++ 和 Fortran

AL2023 包括 GNU 編譯器集合 (GCC) 和 Clang 前端 LLVM (低階虛擬機器)。

GCC 的主要版本將在整個 AL2023 的壽命週期保持不變。次要版本導入錯誤修正，並可能包含在 AL2023 版本中。其他錯誤、效能和安全修正可能會反向移植至 AL2023 隨附的 GCC 主要版本。

AL2023 包括第 11 版 GCC 的 C (`gcc`)，C++ (`g++`) 和 Fortran (`gfortran`) 前端。

AL2023 不會啟用 (`gobjc`)，Ada (`gnat`)，Go 目標 C 或目標 C++ 前端 (`gcc-go`)。

預設的編譯器會標記 AL2023 RPM 是使用包含最佳化和強化旗標建置。要使用 GCC 構建自己的代碼，我們建議您包括優化和強化標誌。

Note

調用 `gcc --version` 時，會顯示 `gcc (GCC) 11.3.1 20221121 (Red Hat 11.3.1-4)` 等的版本字串。Red Hat 指的是 Amazon Linux GCC 套件所基於的 [GCC 供應商分支](#)。根據顯示的錯誤報告網址 `gcc --help`，所有錯誤報告和支援請求都應導向至 Amazon Linux。

如需有關此廠商分支中某些長期變更的詳細資訊，例如__GNUC_RH_RELEASE__巨集，請參閱 [Fedora 套件原始碼](#)。

如需核心工具鏈的詳細資訊，請參閱[核心工具鏈套件 glibc、gcc、binutils](#)。

如需有關 AL2023 及其與其他 Linux 發行版本之間的關係的詳細資訊，請參閱[與 Fedora 的關係](#)。

如需 AL2023 中與 AL2 相比，編譯器三元組變更的詳細資訊，請參閱。[編譯器三元組](#)

AL2023 中的 Go

您可能需要構建在 Amazon Linux [Go](#)上編寫的自己的代碼，並且可能需要使用 AL2023 提供的工具鏈。與 AL2 類似，AL2023 將在操作系統的整個生命週期中更新Go工具鏈。這可能是為了回應隨附工具鏈中的任何 CVE，或是包含在季度版本中。

Go是一種相對快速移動的語言。在某種情況下，編寫的現有應用程序Go必須適應新版本的Go工具鏈。如需有關的詳細資訊Go，請參閱 [Go1 和Go方案的未來](#)。

儘管 AL2023 將在其生命週期內合併新版本的Go工具鏈，但這不會與上游Go版本一起處於鎖定步驟中。因此，如果您想要使用Go語言和標準庫的尖端功能來構建Go代碼，則使用 AL2023 中提供的Go工具鏈可能不適合。

在 AL2023 的存留期間，先前的套件版本不會從儲存庫中移除。如果需要先前的Go工具鏈，您可以選擇放棄較新Go工具鏈的錯誤和安全性修正，並使用適用於任何 RPM 的相同機制從軟件庫安裝舊版本。

如果您想在 AL2023 上構建自己的Go代碼，則可以使用 AL2023 中包含的Go工具鏈，並知道該工具鏈可能會在 AL2023 的生命週期中向前發展。

AL2023 Lambda 函數寫入 Go

當Go編譯為原生程式碼時，Lambda 會視Go為自訂執行階段。您可以使用provided.al2023執行階段將 AL2023 上的Go函數部署到 Lambda。

如需詳細資訊，請參閱AWS Lambda 開發人員指南Go中[的使用建置 Lambda 函數](#)。

AL2023 中的 Java

AL2023 提供多種版本的 [Amazon Corretto](#) 來支援Java基於工作負載。包括在 AL2023 中的所有Java基礎軟件包都建立了Amazon Corretto 17 17。

Corretto 是一個開放 Java 開發工具包 (OpenJDK) 的構建與長期支持從. Amazon Corretto 已使用 Java 技術兼容性工具包 (TCK) 進行認證，以確保其符合 Java SE 標準，並且可以在 Linux，Windows和。macOS

[Amazon Corretto](#) 套件可用於每個版本的 Corretto 1.8.0、Corretto 11 和 Corretto 17。

AL2023 各個 Corretto 版本享有與 Corretto 版本相同的支援期間，或是到 AL2023 的壽命結束 (以較早者為準)。如需詳細資訊，請參閱 [Amazon Linux 套件支援聲明](#)和 [Amazon 套件常見問答集](#)。

AL2023 中的 Perl

AL2023 提供了[Perl](#)編程語言的 5.32 版本。

儘管在過去幾十年中，作為 Perl 5 個版本的一部分，提供Perl了高度的語言兼容性，但在 AL2023 發布期間，Amazon Linux 預計不會從 Perl 5.32 開始移動。Amazon Linux 將根據我們的[套件支援聲明](#)，在 AL2023 的生命週期內繼續Perl進行安全性修補程式。

AL2023 中的 Perl 模組

在 AL2023 中，各種Perl模組都以 RPM 的形式封裝。雖然有許多以 RPM 形式提供的Perl模組，但 Amazon Linux 並不旨在封裝所有可能Perl的模組。封裝為 RPM 的模組可能會被其他作業系統 RPM 套件所依賴，因此 Amazon Linux 會優先考慮這些安全修補程式，而不是純粹的功能更新。

AL2023 還包括，以CPAN便Perl開發人員可以將慣用的包管理器用於Perl模塊。

AL2023 中的 PHP

AL2023 目前提供了兩種[PHP](#)編程語言版本，每種版本都支持與上游相同的時間段PHP。如需詳細資訊，請參閱 [Package 支援陳述式](#)。

使用 AL2023，您可以使用 PHP 8.2 的新功能，同時仍支援需要 PHP 8.1 的應用程式。

從舊 PHP 版本遷移

上游PHP社區匯總了[全面的遷移文檔](#)，以便從 [PHP 8.1 遷移到 PHP 8.2](#)。還有[從 PHP 8.0 遷移到 8.1](#) 的文件。

AL2 在amazon-linux-extras啟用 AL2023 的簡易升級路徑方面，包括 PHP 8.0、8.1 和 8.2。

從 PHP 7.x 版本遷移

Note

該 [PHP](#) 項目維護受 [支持版本](#) 的列表和時間表，以及不支持的 [分支](#) 列表。
當 AL2023 發行時，PHP 社群不支援所有的 [PHP 7.x](#) 和 [5.x](#) 版本，而且未包含在 AL2023 中作為選項。

上游 PHP 社區匯總了 [全面的遷移文檔](#)，以便從 [PHP 7.4 遷移到 PHP 8.0](#)。結合上一節有關遷移到 PHP 8.1 和 PHP 8.2 的文檔，您可以將 PHP 基於應用程序遷移到現代應用程序 PHP。

Note

AL2 包括 PHP 7.1、7.2、7.3 和 7.4 英寸。amazon-linux-extras 重要的是要注意，所有這些 Extras 都是 end-of-life 並且不能保證獲得任何進一步的安全更新。

AL2023 中的 PHP 模組

AL2023 包括被包含在 PHP 核心許多 PHP 模塊。AL2023 的目的並不是將所有套件納入 [PHP 推廣社區圖書館 \(PECL\)](#)。

AL2023 中的 Python

AL2023 刪除 Python 2.7，現在需要的任何組件 Python 都被寫入與 Python 3 一起工作。

AL2023 使 Python 3 可用於 `/usr/bin/python3` 保持與客戶代碼的兼容性，以及與 AL2023 一起提供的 Python 代碼，這將在 AL2023 的使用壽命中保持為 Python 3.9。

`/usr/bin/python3` 指向的蟒蛇的版本被認為是系統 Python，對於 AL2023，這是 Python 3.9。

較新版本的 Python (例如 Python 3.11) 會在 AL2023 中以套件的形式提供，並在上游版本的存留期內受到支援。有關 Python 3.11 支持多長時間的信息，請參閱 [Python 3.11](#)。

AL2023 可以同時安裝多個版本的 Python。雖然 `/usr/bin/python3` 總是 Python 3.9，但每個版本的 Python 都是命名空間，並且可以通過其版本號找到。例如，如果已安裝 `python3.11`，則 `/usr/bin/python3.11` 將與 `/usr/bin/python3.9` 同時存在，且 `/usr/bin/python3` symlink 至 `/usr/bin/python3.9`。

Note

不要更改 `/usr/bin/python3` 符號鏈接指向的內容，因為這可能會破壞 AL2023 的核心功能。

AL2023 中的 Python 模組

在 AL2023 中，各種 Python 模組都以 RPM 的形式封裝。一般而言，Python 模組的 RPM 只會針對 Python 的系統版本建置。

AL2023 中的 Rust

您可能希望構建他們在 Amazon Linux [Rust](#) 上編寫的代碼，並且可能需要使用 AL2023 提供的工具鏈。

與 AL2 類似，AL2023 將在操作系統的整個生命週期中更新 Rust 工具鏈。這可能是為了回應隨附工具鏈中的任何 CVE，或是包含在季度版本中。

[Rust](#) 是一種相對快速的語言，新版本約每六週推出一次。這些版本可能會新增語言或標準資料庫功能。儘管 AL2023 將在其生命週期內合併新版本的 Rust 工具鏈，但這不會與上游 Rust 版本一起處於鎖定步驟中。因此，如果您想使用該 Rust 語言的尖端功能構建 Rust 代碼，則使用 AL2023 中提供的 Rust 工具鏈可能不適合。

在 AL2023 的壽命週期內，舊的套件版本不會從儲存庫中移除。如果需要較舊的 Rust 工具鏈，您可以選擇放棄對較新 Rust 工具鏈的錯誤和安全性修正，並使用適用於任何 RPM 的相同機制從軟件庫安裝舊版本。

如果您想在 AL2023 上構建自己的 Rust 代碼，則可以使用 AL2023 中包含的 Rust 工具鏈，並知道該工具鏈可能會在 AL2023 的生命週期中向前發展。

AL2023 Lambda 函數寫入 Rust

由於 Rust 編譯為原生程式碼，Lambda 會視 Rust 為自訂執行階段。您可以使用 `provided.al2023` 執行階段將 AL2023 上的 Rust 函數部署到 Lambda。

如需詳細資訊，請參閱 AWS Lambda 開發人員指南 Rust 中的 [使用建置 Lambda 函數](#)。

Amazon 2 中的安全性和合規性

Important

如果您想報告漏洞或對 AWS 雲服務或開源項目有安全疑慮，請使用[漏洞報告頁面](#)與 AWS 安全性聯繫

雲安全 AWS 是最高的優先級。身為 AWS 客戶，您可以從資料中心和網路架構中獲益，該架構專為滿足對安全性最敏感的組織的需求而打造。

安全是 AWS 與您之間共同的責任。[共同責任模型](#)將其描述為雲端的安全性和雲端中的安全性：

- 雲端的安全性 — AWS 負責保護在 AWS 雲端中執行 AWS 服務的基礎架構。AWS 還為您提供可以安全使用的服務。若要瞭解適用於 AL2 的規範遵循計劃，請參閱[合規計劃AWS 服務範圍內的AWS 服務，遵循規服務](#)。
- 雲端內部的安全：您的責任取決於所使用的 AWS 服務。您也必須對其他因素負責，包括資料的機密性、您的公司的要求和適用法律和法規。

主題

- [適用於 AL2023 的 Amazon 安全建議](#)
- [設定 AL2023 的環境管理系統模式](#)
- [在 AL2023 上啟用 FIPS 模式](#)
- [AL2023 核心強化處理](#)
- [AL2023 上的 UEFI 安全開機](#)

適用於 AL2023 的 Amazon 安全建議

儘管我們努力確保 Amazon Linux 的安全，但有時會出現需要修正的安全問題。當有修正程式可用時，會發出建議。我們發佈建議的主要位置是 Amazon Linux 安全中心 (ALAS)。如需詳細資訊，請參閱[Amazon Linux 安全中心](#)。

⚠ Important

如果您想要報告弱點或對 AWS 雲端服務或開放原始碼專案有安全疑慮，請使用[弱點報告頁面](#)聯絡 AWS 安全性。

有關問題的資訊以及影響 AL2023 的相關更新，由 Amazon Linux 團隊在多個位置發佈。安全工具通常會從這些主要來源擷取資訊並將結果呈現給您。因此，您可能不會直接與 Amazon Linux 發佈的主要來源互動，而是直接與您偏好的工具 (例如 [Amazon Inspector](#)) 所提供的介面互動。

Amazon Linux 安全中心公告

Amazon Linux 公告是針對不符合諮詢的項目提供的。本節包含 ALAS 本身的公告，以及不符合諮詢的資訊。如需詳細資訊，請參閱 [Amazon Linux 安全中心 \(ALAS\) 公告](#)。

例如，[2021-001-Amazon Linux 修補程序公告阿帕奇 Log4j 適合的公告](#)，而不是一個諮詢。在本公告中，Amazon Linux 新增了一個套件，以協助客戶減輕非 Amazon Linux 一部分的軟體中的安全問題。

[Amazon Linux 安全中心 CVE 資源管理器](#) 也在 ALAS 公告中宣布。如需詳細資訊，請參閱 [CVE 的新網站](#)。

Amazon Linux 安全中心常見問題

如需有關 ALAS 以及 Amazon Linux 如何評估 CVE 的一些常見問題解答，請參閱 [Amazon Linux 安全中心 \(ALAS\) 常見問答集 \(FAQ\)](#)。

設定 AL2023 的環境管理系統模式

根據預設，「安全性增強型 Linux」(SELinux) 會被 enabled 設定為 AL2023 的 permissive 模式。在寬容模式中，會記錄權限遭拒的情形，但不會強制執行。SELinux 是核心功能和公用程式的集合，為核心的主要子系統提供了強大、靈活且強制的存取控制 (MAC) 架構。

SELinux 提供增強的機制，可以根據機密性和完整性需求強制執行資訊分離。這種資訊分離可減少竄改和略過應用程式安全機制的威脅。另外還限制惡意或有缺陷的應用程式可能造成的損壞。

SELinux 包含一組樣本安全政策組態檔案，旨在滿足日常安全目標。

如需 SELinux 特性和功能的詳細資訊，請參閱 [SELinux Notebook](#) 和 [政策語言](#)。

主題

- [AL2023 的預設環境管理系統狀態和模式](#)
- [變更為 enforcing 模式](#)
- [為 AL2023 停用 SELinux 的選項](#)

AL2023 的預設環境管理系統狀態和模式

對於 AL2023，SELinux 默認情況下是 enabled 並設置為 permissive 模式。在 permissive 模式中，會記錄權限遭拒的情形，但不會強制執行。

getenforce 或 **sestatus** 命令會告知您目前的 SELinux 狀態、政策和模式。

將預設狀態設為 enabled 和 permissive 時，**getenforce** 命令會傳回 permissive。

此命**sestatus**令會傳回 SELinux 狀態和目前的 SELinux 政策，如下列範例所示：

```
$ sestatus
SELinux status:                enabled
SELinuxfs mount:              /sys/fs/selinux
SELinux root directory:      /etc/selinux
Loaded policy name:          targeted
Current mode:                 permissive
Mode from config file:       permissive
Policy MLS status:           enabled
Policy deny_unknown status:   allowed
Memory protection checking:   actual (secure)
Max kernel policy version:    33
```

當您以 permissive 模式執行 SELinux 時，使用者可能會錯誤地標示檔案。當您以 disabled 狀態執行 SELinux 時，檔案不會被標記。當您變更為 enforcing 模式時，不正確或未標記的檔案都可能導致問題。

SELinux 會自動重新標記檔案，以避免此問題。SELinux 會在您將狀態變更為 enabled 時，透過自動重新標籤來防止標籤問題。

變更為 enforcing 模式

當您以 enforcing 模 SELinux 式執行時，SELinux 公用程式就是設 enforcing 定的原則。SELinux 根據策略的規則允許或拒絕存取，來控制特定應用程式的功能。

若要尋找目前的SELinux模式，請執行`getenforce`指令。

```
getenforce
Permissive
```

編輯設定檔案以啟用 **enforcing** 模式

若要將模式變更為`enforcing`，請使用下列步驟。

1. 編輯 `/etc/selinux/config` 檔案以變更為 `enforcing` 模式。SELINUX設定看起來應如下列範例所示。

```
SELINUX=enforcing
```

2. 重新啟動系統以完成對 `enforcing` 模式的變更。

```
$ sudo reboot
```

下次開機時，SELinux重新標示系統中的所有檔案和目錄。SELinux還添加了當時SELinux創建的SELinux文件和目錄的上下文`disabled`。

變更為`enforcing`模式之後，SELinux可能會因為SELinux原則規則不正確或遺漏而拒絕某些處理行動。您可以使用以下命令檢視SELinux拒絕的動作。

```
$ sudo ausearch -m AVC,USER_AVC,SELINUX_ERR,USER_SELINUX_ERR -ts recent
```

使用 `cloud-init` 來啟用 **enforcing** 模式

或者，當您啟動執行個體時，請將下列 `cloud-config` 作為使用者資料傳遞以啟用 `enforcing` 模式。

```
#cloud-config
selinux:
  mode: enforcing
```

預設情況下，此設定會導致執行個體重新啟動。為了提高穩定性，建議您重新啟動執行個體 但是，您可以依照偏好輸入下列 `cloud-config`，以跳過重新開機。

```
#cloud-config
```

```
selinux:  
  mode: enforcing  
  selinux_no_reboot: 1
```

為 AL2023 停用 SELinux 的選項

停用時SELinux，不會載入或強制執行SELinux原則，而且不會記錄存取向量快取 (AVC) 訊息。你失去了跑步的所有好處SELinux。

我們建議您使用permissive模式SELinux，而不是停用。在permissive模式下運行的成本僅比SELinux完全禁用要多一點。與停用後轉換回permissiveenforcing模式相比，從模式切換至enforcing模式所需的組態調整要少得多。SELinux您可以標記檔案，然後系統可以追蹤和記錄作用中原則可能已拒絕的動作。

變更SELinux為permissive模式

當您以permissive模式執行SELinux時，不會強制執行SELinux原則。在permissive模式中，會SELinux記錄 AVC 訊息，但不會拒絕作業。您可以使用這些 AVC 訊息進行疑難排解、偵錯和SELinux原則改進。

若要變更SELinux為寬容模式，請使用下列步驟。

1. 編輯 `/etc/selinux/config` 檔案以變更為 permissive 模式。該SELINUX值應該看起來像下面的例子。

```
SELINUX=permissive
```

2. 重新啟動系統以完成對 permissive 模式的變更。

```
sudo reboot
```

停用 SELinux

停用時SELinux，不會載入或強制執行SELinux原則，而且不會記錄 AVC 訊息。你失去了跑步的所有好處SELinux。

若要停用SELinux，請使用下列步驟。

1. 確定已安裝grubby套件。

```
rpm -q grubby
grubby-version
```

- 設定開機載入器，以將 `selinux=0` 新增到核心命令行。

```
sudo grubby --update-kernel ALL --args selinux=0
```

- 重新啟動系統。

```
sudo reboot
```

- 運行 `getenforce` 命令以確認 SELinux 是 Disabled。

```
$ getenforce
Disabled
```

如需有關的詳細資訊 SELinux，請參閱 [SELinux 記事本](#) 和 [SELinux 組態](#)。

在 AL2023 上啟用 FIPS 模式

本節說明如何在 AL2023 上啟用聯邦資訊處理標準 (FIPS)。如需 FIPS 的詳細資訊，請參閱：

- [美國聯邦資訊處理標準 \(FIPS\)](#)
- [法規遵循常見問題集：美國聯邦資訊處理標準](#)

Note

本節說明如何在 AL2023 中啟用 FIPS 模式，但不涵蓋 AL2023 加密模組的認證狀態。

必要條件

- 現有的 AL2023 (AL2023.2 或更新版本) Amazon EC2 執行個體可存取網際網路，以下載所需套件。如需啟動 AL2023 Amazon EC2 執行個體的詳細資訊，請參閱 [使用亞馬遜主控台啟動 AL2023](#)。
- 您必須使用 SSH 或 AWS Systems Manager 連線至 Amazon EC2 執行個體。如需詳細資訊，請參閱 [連線至 AL2023 執行個體](#)。

⚠ Important

FIPS 模式不支援 ED25519 SSH 使用者金鑰。如果使用 ED25519 SSH 金鑰對啟動 Amazon EC2 執行個體，您必須使用其他演算法 (例如 RSA) 產生新金鑰，否則可能無法在啟用 FIPS 模式後存取執行個體。如需詳細資訊，請參閱 Amazon EC2 使用者指南中的[建立金鑰配對](#)。

啟用 FIPS 模式

1. 使用 SSH 或 AWS Systems Manager 連線至 AL2023 執行個體。
2. 請確保系統為最新版本。如需詳細資訊，請參閱 [在 AL2023 中管理套件和作業系統更新](#)。
3. 請確定已安裝和 crypto-policies 公用程式 up-to-date。

```
sudo dnf -y install crypto-policies crypto-policies-scripts
```

4. 執行下列命令來啟用 FIPS 模式。

```
sudo fips-mode-setup --enable
```

5. 使用下列命令重新啟動執行個體。

```
sudo reboot
```

6. 若要驗證 FIPS 模式是否已啟用，請重新連線到執行個體，然後執行下列命令：

```
sudo fips-mode-setup --check
```

下列範例的輸出結果顯示 FIPS 模式已啟用：

```
FIPS mode is enabled.  
Initramfs fips module is enabled.  
The current crypto policy (FIPS) is based on the FIPS policy.
```

AL2023 核心強化處理

AL2023 中的 6.1 Linux 核心經過設定與建置，包含數個強化選項和功能。

核心強化選項 (獨立於架構)

CONFIG 選項	AL2023/6.1/aarch64	AL2023/6.1/x86_64
<u>CONFIG_ACPI_CUSTOM_METHOD</u>	n	n
<u>CONFIG_BINFMT_MISC</u>	m	m
<u>CONFIG_BUG</u>	y	y
<u>CONFIG_BUG_ON_DATA_CORRUPTION</u>	y	y
<u>CONFIG_CFI_CLANG</u>	N/A	N/A
<u>CONFIG_CFI_PERMISSIVE</u>	N/A	N/A
<u>CONFIG_COMPAT</u>	y	y
<u>CONFIG_COMPAT_BRK</u>	n	n
<u>CONFIG_COMPAT_VDSO</u>	N/A	n
<u>CONFIG_DEBUG_CREDENTIALS</u>	n	n
<u>CONFIG_DEBUG_LIST</u>	y	y
<u>CONFIG_DEBUG_NOTIFIERS</u>	n	n
<u>CONFIG_DEBUG_SG</u>	n	n
<u>CONFIG_DEBUG_VIRTUAL</u>	n	n
<u>CONFIG_DEBUG_WX</u>	n	n

CONFIG 選項	AL2023/6.1/aarch64	AL2023/6.1/x86_64
<u>CONFIG_DEFAULT_MMAP_MIN_ADDR</u>	65536	65536
<u>CONFIG_DEVMEM</u>	N/A	N/A
<u>CONFIG_DEVMEM</u>	n	n
<u>CONFIG_EFI_DISABLE_PCI_DMA</u>	n	n
<u>CONFIG_FORTIFY_SOURCE</u>	y	y
<u>CONFIG_HARDENED_USERCOPY</u>	y	y
<u>CONFIG_HARDENED_USERCOPY_FALLBACK</u>	N/A	N/A
<u>CONFIG_HARDENED_USERCOPY_PAGESPAN</u>	N/A	N/A
<u>CONFIG_HIBERNATION</u>	y	y
<u>CONFIG_HW_RANDOM_TPM</u>	N/A	N/A
<u>CONFIG_INET_DIAG</u>	m	m
<u>CONFIG_INIT_ON_ALLOC_DEFAULT_ON</u>	n	n
<u>CONFIG_INIT_ON_FREE_DEFAULT_ON</u>	n	n
<u>CONFIG_INIT_STACK_ALL_ZERO</u>	N/A	N/A

CONFIG 選項	AL2023/6.1/aarch64	AL2023/6.1/x86_64
<u>CONFIG_IOMMU_DEFAULT_DMA_STRICT</u>	n	n
<u>CONFIG_IOMMU_SUPPORT</u>	y	y
<u>CONFIG_IO_STRICT_DEVMEM</u>	N/A	N/A
<u>CONFIG_KEXEC</u>	y	y
<u>CONFIG_KFENCE</u>	n	n
<u>CONFIG_LDISC_AUTOLOAD</u>	n	n
<u>CONFIG_LEGACY_PTYS</u>	n	n
<u>CONFIG_LOCK_DOWN_KERNEL_FORCE_CONFIDENTIALITY</u>	n	n
<u>CONFIG_MODULES</u>	y	y
<u>CONFIG_MODULE_SIG</u>	y	y
<u>CONFIG_MODULE_SIG_ALL</u>	y	y
<u>CONFIG_MODULE_SIG_FORCE</u>	n	n
<u>CONFIG_MODULE_SIG_HASH</u>	sha512	sha512
<u>CONFIG_MODULE_SIG_KEY</u>	certs/signing_key.pem	certs/signing_key.pem

CONFIG 選項	AL2023/6.1/aarch64	AL2023/6.1/x86_64
<u>CONFIG_MODULE_SIG_SHA512</u>	y	y
<u>CONFIG_PAGE_POISONING</u>	n	n
<u>CONFIG_PAGE_POISONING_NO_SANITY</u>	N/A	N/A
<u>CONFIG_PAGE_POISONING_ZERO</u>	N/A	N/A
<u>CONFIG_PANIC_ON_OOPS</u>	y	y
<u>CONFIG_PANIC_TIMEOUT</u>	0	0
<u>CONFIG_PROC_KCORE</u>	y	y
<u>CONFIG_RANDOMIZE_KSTACK_OFFSET_DEFAULT</u>	n	n
<u>CONFIG_RANDOM_TRUST_BOOTLOADER</u>	y	y
<u>CONFIG_RANDOM_TRUST_CPU</u>	y	y
<u>CONFIG_REFCOUNT_FULL</u>	N/A	N/A
<u>CONFIG_SCHED_CORE</u>	N/A	y
<u>CONFIG_SCHED_STACK_END_CHECK</u>	y	y
<u>CONFIG_SECCOMP</u>	y	y

CONFIG 選項	AL2023/6.1/aarch64	AL2023/6.1/x86_64
<u>CONFIG_SECCOMP_FILTER</u>	y	y
<u>CONFIG_SECURITY</u>	y	y
<u>CONFIG_SECURITY_DMESG_RESTRICT</u>	y	y
<u>CONFIG_SECURITY_LANDLOCK</u>	n	n
<u>CONFIG_SECURITY_LOCKDOWN_LSM</u>	y	y
<u>CONFIG_SECURITY_LOCKDOWN_LSM_EARLY</u>	y	y
<u>CONFIG_SECURITY_SELINUX_BOOTPARAM</u>	y	y
<u>CONFIG_SECURITY_SELINUX_DEVELOP</u>	y	y
<u>CONFIG_SECURITY_SELINUX_DISABLE</u>	n	n
<u>CONFIG_SECURITY_WRITABLE_HOOKS</u>	N/A	N/A
<u>CONFIG_SECURITY_YAMA</u>	y	y
<u>CONFIG_SHUFFLE_PAGE_ALLOCATOR</u>	y	y
<u>CONFIG_SLAB_FREELIST_HARDENED</u>	y	y

CONFIG 選項	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_SLAB_FREELIST_RANDOM	y	y
CONFIG_SLUB_DEBUG	y	y
CONFIG_STACKPROTECTOR	y	y
CONFIG_STACKPROTECTOR_STRONG	y	y
CONFIG_STATIC_USERMODEHELPER	n	n
CONFIG_STRICT_DEVMEM	n	n
CONFIG_STRICT_KERNEL_RWX	y	y
CONFIG_STRICT_MODULE_RWX	y	y
CONFIG_SYN_COOKIES	y	y
CONFIG_VMAP_STACK	y	y
CONFIG_WERROR	n	n
CONFIG_ZERO_CALL_USED_REGS	n	n

允許在執行時期插入/取代ACPI 方法 (組態 ACPI _ 自訂方法)

Amazon Linux 會停用此選項，因為它允許根使用者寫入任意核心記憶體。

此選項是[核心自我保護專案建議設定](#)之一。

其他二進位格式 (`binfmt_misc`)

雖然此選項是[核心自我保護專案 \(KSP\) 建議設定](#)之一，但 AL2023 並未將此組態選項設為 KSP 建議的選項。在 AL2023 中，此功能為選擇性，並建構作為核心模組。

BUG() 支援

此選項是[核心自我保護專案建議設定](#)之一。

BUG() 如果核心在檢查核心記憶體結構的有效性時發生資料損毀

Linux 核心的某些部分將檢查資料結構的內部一致性，並可在偵測到資料損毀時 BUG()。

此選項是[核心自我保護專案建議設定](#)之一。

COMPAT_BRK

停用此選項 (這是 Amazon Linux 設定核心的方式) 時，`randomize_va_space` `sysctl` 設定預設為 2，這也會以 `mmap` 基礎、堆疊和 VDSO 頁面隨機化為基礎啟用堆積隨機化。

此選項存在於核心中，以提供與 1996 及更舊版本的久遠 `libc.so.5` 二進位檔案的相容性。

此選項是[核心自我保護專案建議設定](#)之一。

COMPAT_VDSO

此組態選項與 x86-64 相關，但與 aarch64 無關。將此項設定為 `n` 時，Amazon Linux 核心不會讓 32 位元虛擬動態共用物件 (VDSO) 顯示在可預測的位址。已知可透過將此選項設為 `n` 來破壞的最新 `glibc` 自 2004 年開始設為 `glibc 2.3.3`。

此選項是[核心自我保護專案建議設定](#)之一。

CONFIG_DEBUG 控制強化

由 `CONFIG_DEBUG` 控制的 Linux 核心組態選項通常是設計用來在核心建置中進行問題除錯，因此效能等項目不是優先事項。AL2023 可啟用 `CONFIG_DEBUG_LIST` 強化選項。

設定 IOMMU 前，在 EFI 虛設常式中停用 PCI 裝置的 DMA

雖然此選項是[核心自我保護專案 \(KSP\) 建議設定](#)之一，但 AL2023 並未將此組態選項設為 KSP 建議的選項。

在核心和使用者空間之間複製記憶體強化

當核心需要將記憶體複製到使用者空間或從使用者空間複製記憶體時，此選項會啟用一些檢查，以防止某些類別的堆積溢位問題。

核心 4.16 到 5.15 中存在 `CONFIG_HARDENED_USERCOPY_FALLBACK` 選項，可協助核心開發人員透過 `WARN()` 找出任何遺失的 `allowlist` 項目。由於 AL2023 隨附 6.1 核心，因此此選項不再與 AL2023 相關。

該 `CONFIG_HARDENED_USERCOPY_PAGESPAN` 選項存在於內核中，主要是作為開發人員的調試選項，不再適用於 AL2023 中的 6.1 內核。

此選項是[核心自我保護專案建議設定](#)之一。

休眠支援

雖然此選項是[核心自我保護專案 \(KSP\) 建議設定](#)之一，但 AL2023 並未將此組態選項設為 KSP 建議的選項。必須啟用此選項，才能支援[隨需執行個體休眠](#)的能力，並支援[休眠中斷 Spot 執行個體](#)的功能。

隨機數生成

AL2023 核心經過設定，以確保 EC2 內部有足夠的熵可供使用。

CONFIG_INET_DIAG

雖然此選項是[核心自我保護專案 \(KSP\) 建議設定](#)之一，但 AL2023 並未將此組態選項設為 KSP 建議的選項。在 AL2023 中，此功能為選擇性，並建構作為核心模組。

將分配和解除分配的所有核心頁面和 Slab 分配器記憶體歸零

雖然此選項是[核心自我保護專案 \(KSP\) 建議設定](#)之一，但 AL2023 並未將此組態選項設為 KSP 建議的選項。這些選項在 AL2023 中會停用，因為預設啟用此功能可能會對效能造成影響。`CONFIG_INIT_ON_ALLOC_DEFAULT_ON` 行為可透過將 `init_on_alloc=1` 加入核心命令行來啟用，並可透過加入 `init_on_free=1` 來啟用 `CONFIG_INIT_ON_FREE_DEFAULT_ON` 行為。

將所有堆疊變數初始化為零 (CONFIG_INIT_STACK_ALL_ZERO)

雖然此選項是[核心自我保護專案 \(KSP\) 建議設定](#)之一，但 AL2023 並未將此組態選項設為 KSP 建議的選項。此選項需要 GCC 12 或更新版本，而 AL2023 則隨附 GCC 11。

核心模組簽署

AL2023 簽署並驗證核心模組的簽章。為了保持與建置第三方模組之使用者的相容性，CONFIG_MODULE_SIG_FORCE 選項未啟用，因為這會要求模組具有有效的簽名。對於想要確保所有核心模組都已簽署的使用者，[鎖定 Linux 安全模組 \(LSM\)](#) 可設為強制執行此操作。

kexec

雖然此選項是[核心自我保護專案 \(KSP\) 建議設定](#)之一，但 AL2023 並未將此組態選項設為 KSP 建議的選項。此選項已啟用，以便使用 kdump 功能。

IOMMU 支援

AL2023 能夠支援異物管理系統。預設情況下，不會啟用 CONFIG_IOMMU_DEFAULT_DMA_STRICT 選項，但可以透過將 `iommu.passthrough=0 iommu.strict=1` 加入核心命令行來設定此功能。

kfence

雖然此選項是[核心自我保護專案 \(KSP\) 建議設定](#)之一，但 AL2023 並未將此組態選項設為 KSP 建議的選項。

舊版 pty 支援

AL2023 使用現代化的PTY介面 (devpts)。

此選項是[核心自我保護專案建議設定](#)之一。

鎖定 Linux 安全模組 (LSM)

AL2023 建置 lockdown LSM，當使用「安全開機」時，會自動鎖定核心。

CONFIG_LOCK_DOWN_KERNEL_FORCE_CONFIDENTIALITY 選項未啟用。雖然此選項是[核心自我保護專案 \(KSP\) 建議設定](#)之一，但 AL2023 並未將此組態選項設為 KSP 建議的選項。未使用安全開機時，可以啟用鎖定 LSM 並視需要進行設定。

頁面中毒

雖然此選項是[核心自我保護專案 \(KSP\) 建議設定](#)之一，但 AL2023 並未將此組態選項設為 KSP 建議的選項。與此類似 [將分配和解除分配的所有核心頁面和 Slab 分配器記憶體歸零](#)，由於可能會影響效能，因此在 AL2023 核心中停用此功能。

堆疊保護器

AL2023 核心是使GCC用啟用的堆疊保護器功能所建置，並搭配選項-fstack-protector-strong。

此選項是[核心自我保護專案建議設定](#)之一。

seccomp BPF API

seccomp 強化功能會由 systemd 和容器執行期等的軟體使用，以加強使用者空間應用程式。

此選項是[核心自我保護專案建議設定](#)之一。

panic() 逾時

AL2023 核心會將此值設定為0，表示核心在發生錯誤後不會重新開機。雖然此選項是[核心自我保護專案 \(KSP\) 建議設定](#)之一，但 AL2023 並未將此組態選項設為 KSP 建議的選項。這可透過 sysctl、/proc/sys/kernel/panic 和在核心命令行上進行設定。

安全模型

AL2023 默認情況下，在允許模式下啟用 SELinux。如需詳細資訊，請參閱 [設定 AL2023 的環境管理系統模式](#)。

[鎖定 Linux 安全模組 \(LSM\)](#) 和 yama 模組也會啟用。

/proc/kcore

雖然此選項是[核心自我保護專案 \(KSP\) 建議設定](#)之一，但 AL2023 並未將此組態選項設為 KSP 建議的選項。

syscall 項目的核心堆疊位移隨機化

雖然此選項是[核心自我保護專案 \(KSP\) 建議設定](#)之一，但 AL2023 並未將此組態選項設為 KSP 建議的選項。這可透過在核心命令行上設定 randomize_kstack_offset=on 來啟用。

參考計數檢查 (CONFIG_REFCOUNT_FULL)

雖然此選項是[核心自我保護專案 \(KSP\) 建議設定](#)之一，但 AL2023 並未將此組態選項設為 KSP 建議的選項。由於此選項可能會對效能造成影響，因此無法正確啟用。

SMT 核心的排程器感知 (**CONFIG_SCHED_CORE**)

AL2023 核心內建CONFIG_SCHED_CORE，可讓使用者空間應用程式使prctl(PR_SCHED_CORE)用。此選項是[核心自我保護專案建議設定](#)之一。

檢查呼叫 **schedule()** (**CONFIG_SCHED_STACK_END_CHECK**) 時的堆疊損毀

AL2023 核心是在CONFIG_SCHED_STACK_END_CHECK啟用的情況下建置的。此選項是[核心自我保護專案建議設定](#)之一。

記憶體分配器強化

AL2023 核心可透

過CONFIG_SHUFFLE_PAGE_ALLOCATOR、CONFIG_SLAB_FREELIST_HARDENED和CONFIG_SLAB_FREELIST_HARDENED項強化核心記憶體配置器。此選項是[核心自我保護專案建議設定](#)之一。

SLUB 除錯支援

AL2023 核心會啟用，CONFIG_SLUB_DEBUG因為此選項會為可在核心命令列上啟用的分配器啟用選用的除錯功能。此選項是[核心自我保護專案建議設定](#)之一。

CONFIG_STATIC_USERMODEHELPER

雖然此選項是[核心自我保護專案 \(KSP\) 建議設定](#)之一，但 AL2023 並未將此組態選項設為 KSP 建議的選項。這是因為 CONFIG_STATIC_USERMODEHELPER 需要發行版的特殊支援，且目前不存在 Amazon Linux 中。

唯讀核心文字和 rodata (**CONFIG_STRICT_KERNEL_RWX** 和 **CONFIG_STRICT_MODULE_RWX**)

AL2023 核心設定為將核心和核心模組的文字和rodata記憶體標記為唯讀，並將非文字記憶體標記為無法執行。此選項是[核心自我保護專案建議設定](#)之一。

TCP syncookie 支援 (**CONFIG_SYN_COOKIES**)

AL2023 內核是在支持 TCP 同步餅的情況下構建的。此選項是[核心自我保護專案建議設定](#)之一。

有保護頁面的虛擬映射堆疊 (**CONFIG_VMAP_STACK**)

AL2023 核心內建CONFIG_VMAP_STACK，可透過保護頁面啟用虛擬對應的核心堆疊。此選項是[核心自我保護專案建議設定](#)之一。

使用編譯器警告作為錯誤建置 (CONFIG_WERROR)

雖然此選項是[核心自我保護專案 \(KSP\) 建議設定](#)之一，但 AL2023 並未將此組態選項設為 KSP 建議的選項。

在函數退出登錄歸零 (CONFIG_ZERO_CALL_USED_REGS)

雖然此選項是[核心自我保護專案 \(KSP\) 建議設定](#)之一，但 AL2023 並未將此組態選項設為 KSP 建議的選項。

使用者空間分配的最小地址

此強化選項有助於減輕核心 NULL 指標錯誤的影響。此選項是[核心自我保護專案建議設定](#)之一。

clang 特定的強化選項

AL2023 核心是使用GCC而非建置的clang，因此無法啟用CONFIG_CFI_CLANG強化選項，這也CONFIG_CFI_PERMISSIVE不適用。雖然此選項是[核心自我保護專案 \(KSP\) 建議設定](#)之一，但 AL2023 並未將此組態選項設為 KSP 建議的選項。

x86-64 特定核心強化選項

CONFIG 選項	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_AMD_IOMMU	N/A	y
CONFIG_AMD_IOMMU_V2	N/A	y
CONFIG_IA32_EMULAT ION	N/A	y
CONFIG_INTEL_IOMMU	N/A	y
CONFIG_INTEL_IOMMU _DEFAULT_ON	N/A	n
CONFIG_INTEL_IOMMU _SVM	N/A	n
CONFIG_LEGACY_VSYS CALL_NONE	N/A	n

CONFIG 選項	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_MODIFY_LDT_SYSCALL	N/A	n
CONFIG_PAGE_TABLE_ISOLATION	N/A	y
CONFIG_RANDOMIZE_MEMORY	N/A	y
CONFIG_X86_64	N/A	y
CONFIG_X86_MSR	N/A	y
CONFIG_X86_VSYSCALL_EMULATION	N/A	y
CONFIG_X86_X32	N/A	N/A
CONFIG_X86_X32_ABI	N/A	n

x86-64 支援

基本 x86-64 支援包括實體位址延伸 (PAE) 和無執行 (NX) 位元支援。此選項是[核心自我保護專案建議設定](#)之一。

AMD 和 Intel IOMMU 支援

AL2023 內核在支持 AMD 和英特爾的情況下構建 IOMMUs。此選項是[核心自我保護專案建議設定](#)之一。

未設定 `CONFIG_INTEL_IOMMU_DEFAULT_ON` 選項，但可透過將 `intel_iommu=on` 傳遞至核心命令行來啟用。雖然此選項是[核心自我保護專案 \(KSPP\) 建議設定](#)之一，但 AL2023 並未將此組態選項設為 KSPP 建議的選項。

此選 `CONFIG_INTEL_IOMMU_SVM` 項目目前未在 AL2023 中啟用。雖然此選項是[核心自我保護專案 \(KSPP\) 建議設定](#)之一，但 AL2023 並未將此組態選項設為 KSPP 建議的選項。

32 位元使用者空間的支援

⚠ Important

對 32 位元 x86 使用者空間的支援已棄用，在未來的 Amazon Linux 主要版本中，可能會移除對執行 32 位元使用者空間二進位檔案的支援。

📘 Note

雖然 AL2023 不再包含任何 32 位元套件，但核心仍會支援執行 32 位元使用者空間。如需詳細資訊，請參閱[32 位元 x86 \(i686\) 套件](#)。

若要支援執行 32 位元使用者空間應用程式，AL2023 不會啟用此 `CONFIG_X86_VSYSCALL_EMULATION` 選項，並啟用 `CONFIG_IA32_EMULATION`、`CONFIG_COMPAT`、和 `CONFIG_X86_VSYSCALL_EMULATION` 選項。雖然此選項是[核心自我保護專案 \(KSP\) 建議設定](#)之一，但 AL2023 並未將此組態選項設為 KSP 建議的選項。

未啟用 64 位元處理器的 x32 原生 32 位元 ABI (`CONFIG_X86_X32` 和 `CONFIG_X86_X32_ABI`)。此選項是[核心自我保護專案建議設定](#)之一。

x86 模型特定暫存器 (MSR) 支援

`CONFIG_X86_MSR` 選項已啟用以支援 `turbostat`。雖然此選項是[核心自我保護專案 \(KSP\) 建議設定](#)之一，但 AL2023 並未將此組態選項設為 KSP 建議的選項。

`modify_ldt` syscall

AL2023 不允許使用者程式使用 `modify_ldt` 系統調用修改 x86 本機描述元表 (LDT)。執行 16 位元或分段程式碼需要此呼叫，缺少此呼叫可能會破壞 `dosemu` 等的軟體、在 WINE 下執行某些程式，以及非常舊的執行緒庫。此選項是[核心自我保護專案建議設定](#)之一。

在使用者模式下移除核心對應

AL2023 會設定核心，讓大部分的核位址都不會對應到使用者空間中。此選項是[核心自我保護專案建議設定](#)之一。

隨機化核心記憶體區段

AL2023 配置內核以隨機化內核內存部分的基本虛擬地址。此選項是[核心自我保護專案建議設定](#)之一。

aarch64 特定核心強化選項

CONFIG 選項	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_ARM64_BTI	y	N/A
CONFIG_ARM64_BTI_KERNEL	N/A	N/A
CONFIG_ARM64_PTR_AUTH	y	N/A
CONFIG_ARM64_PTR_AUTH_KERNEL	y	N/A
CONFIG_ARM64_SW_TTBR0_PAN	y	N/A
CONFIG_UNMAP_KERNEL_AT_EL0	y	N/A

分支目標識別

AL2023 核心支援分支目標識別 (CONFIG_ARM64_BTI)。此選項是[核心自我保護專案建議設定](#)之一。

CONFIG_ARM64_BTI_KERNEL 選項在 AL2023 中未啟用，因為它是使用 GCC 建置，並且由於 [gcc 錯誤](#)，[上游核心目前已停用](#)對使用此選項建置核心的支援。雖然此選項是[核心自我保護專案 \(KSPP\) 建議設定](#)之一，但 AL2023 並未將此組態選項設為 KSPP 建議的選項。

指標驗證 (CONFIG_ARM64_PTR_AUTH)

AL2023 核心是以指標驗證延伸模組 (ARMv8.3 延伸模組的一部分) 的支援而建置，可用來協助減輕傳回導向程式設計 (ROP) 技術。Graviton 3 已推出對 [Graviton](#) 進行指標驗證所需的硬體支援。

CONFIG_ARM64_PTR_AUTH 選項已啟用，並為使用者空間提供指標驗證的支援。由於此CONFIG_ARM64_PTR_AUTH_KERNEL選項也已啟用，因此 AL2023 核心能夠自行使用傳回位址保護。

此選項是[核心自我保護專案建議設定](#)之一。

模擬特權存取永不使用 TTBR0_EL1 切換

此選項可防止核心直接存取使用者空間記憶體，只會由使用者存取常式將 TTBR0_EL1 暫時設定為有效值。

此選項是[核心自我保護專案建議設定](#)之一。

在使用者空間中執行時取消對應核心

AL2023 核心設定為在使用者空間中執行時取消對應核心 () CONFIG_UNMAP_KERNEL_AT_EL0。此選項是[核心自我保護專案建議設定](#)之一。

AL2023 上的 UEFI 安全開機

AL2023 從 2023.1 版開始支援 UEFI 安全開機。您必須將 AL2023 與同時支援 UEFI 和 UEFI 安全開機的 Amazon EC2 執行個體搭配使用。[如需詳細資訊，請參閱 Amazon EC2 使用者指南中的啟動執行個體。](#)

啟用 UEFI 安全開機的 AL2023 執行個體僅接受由簽署的核心層級程式碼 (包括 Linux 核心和模組)，Amazon 因此您可以確保執行個體只執行由 AWS 簽署的核心層級程式碼。

如需有關 Amazon EC2 執行個體和 UEFI 安全開機的詳細資訊，請參閱 Amazon EC2 使用者指南中的[UEFI 安全開機](#)。

必要條件

- 您必須使用 AL2023 版本 2023.1 或更新版本的 AMI。
- 執行個體類型必須支援 UEFI 安全開機。[如需詳細資訊，請參閱 Amazon EC2 使用者指南中的啟動執行個體。](#)

在 AL2023 上啟用 UEFI 安全開機

標準 AL2023 AMI 包含開機載入器和由我方金鑰簽章的核心。您可以透過註冊現有執行個體或透過預先啟用 UEFI 安全開機建立 AMI，方法是從快照註冊映像來啟用 UEFI 安全開機。標準 AL2023 AMI 預設不會啟用 UEFI 安全開機。

AL2023 AMI 的開機模式已設定為 `uefi-preferred`，如果執行個體類型支援 UEFI，則可確保透過這些 AMI 啟動的執行個體將使用 UEFI 韌體。如果執行個體類型不支援 UEFI，則會在舊版 BIOS 上啟動執行個體。執行個體在舊版 BIOS 模式中啟動時，不會強制執行 UEFI 安全開機。

如需 Amazon EC2 執行個體上 AMI 開機模式的詳細資訊，請參閱 Amazon EC2 使用者指南中的 [開機模式](#)。

主題

- [註冊現有執行個體](#)
- [從快照註冊映像](#)
- [撤銷更新](#)
- [UEFI 安全開機如何在 AL2023 上運作](#)
- [註冊自有金鑰](#)

註冊現有執行個體

若要註冊現有的執行個體，請在特定 UEFI 韌體變數中填入一組金鑰，讓韌體能夠驗證開機載入器，以便開機載入器在下次開機時驗證核心。

1. Amazon Linux 提供簡化註冊過程的工具。執行下列命令以佈建執行個體，並使用必要的金鑰和憑證集合來佈建執行個體。

```
sudo amazon-linux-sb enroll
```

2. 執行下列命令重新開機執行個體。執行個體重新開機後，將會啟用 UEFI 安全開機。

```
sudo reboot
```

Note

Amazon Linux AMI 目前不支援 Nitro 信任平台模組 (NitroTPM)。如果除了 UEFI 安全開機之外，您還需要 NitroTPM，請使用下一節中的資訊。

從快照註冊映像

使用 Amazon EC2 `register-image` API 並從 Amazon EBS 根磁碟區的快照註冊 AMI 時，您可以使用包含 UEFI 變數存放區狀態的二進位 Blob 佈建 AMI。透過提供 AL2023 `UefiData`，即可啟用 UEFI 安全開機，而且不需要遵循上一節中的步驟。

如需有關建立和使用二進位 blob 的詳細資訊，請參閱 Amazon EC2 使用者指南中的[選項 B：建立包含預先填入的變數存放區的二進位 blob](#)。

AL2023 提供可直接在 Amazon EC2 執行個體上使用且預先建置的二進位 Blob。二進位 Blob 位於執行中執行個體的 `/usr/share/amazon-linux-sb-keys/uefi.vars`。此 Blob 是由 `amazon-linux-sb-keys` RPM 套件提供，此套件從 2023.1 版開始預設安裝在 AL2023 AMI 上。

Note

若要確保您使用的是最新版本的金鑰和撤銷，請使用您用來建立 AMI 的相同 AL2023 版本中的 Blob。

註冊映像時，建議使用設定為 `uefi` 的 `RegisterImage` API `BootMode` 參數。這可讓您透過將 `TpmSupport` 參數設定為 `v2.0` 來啟用 NitroTPM。此外，將 `BootMode` 設定為 `uefi` 可確保啟用 UEFI 安全開機，且在切換至不支援 UEFI 的執行個體類型時，無法意外停用。

如需有關 NitroTPM 的詳細資訊，請參閱 Amazon EC2 使用者指南中的[NitroTPM](#)。

撤銷更新

Amazon Linux 可能需要發行新版本的開機載入器 `grub2`，或使用更新金鑰簽署的 Linux 核心。在這種情況下，可能需要撤銷舊金鑰，以防止有人藉由舊版開機載入器的可利用錯誤繞過 UEFI 安全開機驗證程序。

`grub2` 或 `kernel` 套件的套件更新一律會自動將撤銷清單更新至執行中執行個體的 UEFI 變數存放區。這表示啟用 UEFI 安全開機並安裝套件的安全更新後，您將無法再執行舊版的套件。

UEFI 安全開機如何在 AL2023 上運作

與其他 Linux 發行版不同，Amazon Linux 不提供稱為 Shim 的額外元件來作為第一階段的開機載入器。Shim 通常使用 Microsoft 金鑰簽署。例如，在具有 Shim 的 Linux 發行版上，Shim 會載入 grub2 開機載入器並 Shim 自己的程式碼來驗證 Linux 核心。此外，Shim 會在位於 UEFI 變數存放區的機器擁有者金鑰 (MOK) 資料庫中維護自有金鑰和撤銷組合，並透過 mokutil 工具控制。

Amazon Linux 不提供 Shim。由於 AMI 擁有者控制 UEFI 變數，因此不需要此中繼步驟，而且會對啟動和開機時間產生不利影響。此外，我們選擇預設不包括對任何供應商金鑰的信任，以減少不需要的二進位檔案可以執行的機會。與往常一樣，如果客戶選擇這樣做，則可以包含二進位檔案。

使用 Amazon Linux 時，UEFI 可以直接載入並驗證 grub2 開機載入器。grub2 開機載入器已修改為在載入 Linux 核心後使用 UEFI 對其進行驗證。因此，系統會使用儲存在一般 UEFI db 變數 (授權金鑰資料庫) 中的相同憑證來驗證 Linux 核心，並針對與開機載入器和其他 UEFI 二進位檔案相同的 dbx 變數 (撤銷資料庫) 進行測試。由於我們提供自有 PK 和 KEK 金鑰，用於控制對 db 資料庫和 dbx 資料庫的存取，因此我們可以根據需要發行已簽章的更新和撤銷，而無需中介 (如 Shim)。

[如需 UEFI 安全開機的詳細資訊，請參閱 Amazon EC2 使用者指南中的 UEFI 安全開機運作方式。](#)

註冊自有金鑰

如上一節所述，Amazon Linux 不需要在 Amazon EC2 上使用 shim 以進行 UEFI 安全開機。當您正在閱讀其他 Linux 發行版的文件時，您可能會發現使用 mokutil 管理機器擁有者金鑰 (MOK) 資料庫的說明文件，且這些資料庫不在 AL2023 中。shim 和 MOK 環境解決在 UEFI 韌體中註冊金鑰的特定限制，這些限制不適用於 Amazon EC2 實作 UEFI 安全開機的方式。使用 Amazon EC2 時，有一些機制可以輕鬆地直接操作 UEFI 變數存放區中的金鑰。

如果您想要註冊自有金鑰，您可以在現有執行個體內操作變數存放區 (請參閱[從執行個體內將金鑰新增至變數存放區](#))，或建置預先填入的二進位 Blob (請參閱[建立包含預先填入變數存放區的二進位 Blob](#))。

本文為英文版的機器翻譯版本，如內容有任何歧義或不一致之處，概以英文版為準。