使用者指南

AWS Certificate Manager

版本 1.0

AWS Certificate Manager: 使用者指南

Table of Contents

什麼是 AWS Certificate Manager？

AWS Certificate Manager (ACM) 處理建立、儲存及更新公開與私密SSL/TLSX.509 憑證和金鑰的複雜性，以保護您的 AWS 網站和應用程式。您可以透過直接發行ACM或將協力廠商憑證匯入ACM管理系統，來為您的整合式 AWS 服務提供憑證。ACM憑證可以保護單一網域名稱、多個特定網域名稱、萬用字元網域或其組合。ACM萬用字元憑證可保護無限數量的子網域。您也可以匯出簽署的ACM憑證，以 AWS 私有 CA 便在內部的任何位置使用PKI。

> **Note**
>
> ACM不適用於獨立網絡服務器。如果您想要在 Amazon EC2 執行個體上設定獨立的安全伺服器，以下教學提供說明：TLS在 Amazon Linux 2023 上設定SSL/。

主題

適ACM合我的服務嗎？

AWS 為部署受管理 X.509 憑證的客戶提供了兩個選項。選擇最適合您需求的選項。

1. AWS Certificate Manager (ACM) — 此服務適用TLS於需要使用安全網路存在的企業客戶。ACM憑證是透過 Elastic Load Balancing、Amazon CloudFront、Amazon API 閘道和其他整合式 AWS 服務來部署。最常見的這類應用是具有龐大流量要求的安全公有網站。ACM此外，透過自動更新即將到期的憑證，簡化安全性管理。您正位於此服務的正確位置。
2. AWS 私有 CA— 此服務適用於在 AWS 雲端內建立公開金鑰基礎結構 (PKI)，並專供組織內部私人使用的企業客戶。使用時 AWS 私有 CA，您可以建立自己的憑證授權單位 (CA) 階層，並發行憑

證，以驗證使用者、電腦、應用程式、服務、伺服器和其他裝置。私有 CA 發行的憑證無法在網際網路上使用。如需詳細資訊，請參閱 AWS 私有 CA 使用者指南。

ACM憑證特性

提供的公用憑證ACM具有本節中所述的特性。

> **Note**
>
> 這些特性僅適用於由提供的憑證ACM。它們可能不適用於您匯入的憑證ACM。

憑證授權機構和階層

您透過申請的公有憑證ACM是從 Amazon 受管公共憑證授權單位 (CA) Amazon 信任服務取得的。Amazon 根 CAs 1 到 4 由名為星空 G2 根證書頒發機構-G2 的舊根交叉簽名。Starfield 是 Android 裝置信任的根，Android Gingerbread 和 iOS 4.1 以後的版本，均將其視為信任的根。Amazon 根受 iOS 11 以後的版本所信任。任何包含 Amazon 或 Starfield 根目錄的瀏覽器、應用程式或作業系統都會信任從中ACM取得的公用憑證。

發行給客戶的葉或終端實體憑證，ACM透過數個中CAs繼資料中的任何一個，從 Amazon 信任服務根 CA 衍生其授權。ACM根據要求的憑證類型 (RSA或ECDSA) 隨機指派中繼 CA。由於中繼 CA 是在產生要求之後隨機選取的，因ACM此不會提供中繼 CA 資訊。

瀏覽器和應用程式信任

ACM證書是由所有主流瀏覽器，包括谷歌瀏覽器，Microsoft 互聯網瀏覽器和 Microsoft 邊緣，火狐瀏覽器和蘋果 Safari 瀏覽器信任。信任ACM憑證的瀏覽器透過SSL/連線TLS至使用ACM憑證的網站時，會在狀態列或網址列中顯示鎖定圖示。ACM憑證也受到 Java 的信任。

中繼和根 CA 輪換

為維護彈性且靈活的憑證基礎設施，Amazon 可在未提前通知的情況下，隨時選擇停止中繼 CA。這種變動不會對客戶造成任何影響。如需詳細資訊，請參閱部落格文章 Amazon introduces dynamic intermediate certificate authorities (Amazon 推出動態中繼憑證授權機構)。

雖然不太可能發生，但萬一 Amazon 必須停止根 CA，這種變動會在有必要時立即發生。由於此類變更的巨大影響，Amazon 將使用所有可用的機制來通知 AWS 客戶，包括將電子郵件傳送給帳戶擁有者，以及向技術客戶經理拓展。 AWS Health Dashboard

存取防火牆以撤銷憑證

如果終端實體憑證不再值得信任，該憑證就會遭到撤銷。OCSP和CRLs是用來驗證憑證是否已撤銷的標準機制。OCSP和CRLs是用來發佈撤銷資訊的標準機制。部分客戶的防火牆可能需要額外的規則來允許這些機制運作。

下列範例URL萬用字元模式可用來識別撤銷流量。星號 (*) 萬用字元代表一或多個英數字元，問號 (?) 代表一個英數字元，井字號 (#) 代表一個數字。

- OCSP

  ```
  http://ocsp.?????.amazontrust.com
  ```

  ```
  http://ocsp.*.amazontrust.com
  ```

- CRL

  ```
  http://crl.?????.amazontrust.com/?????.crl
  ```

  ```
  http://crl.*.amazontrust.com/*.crl
  ```

網域驗證 (DV)

ACM 憑證是由網域驗證。也就是說，ACM證書的主題字段標識了域名，僅此而已。當您要求ACM憑證時，您必須驗證您擁有或控制您在要求中指定的所有網域。您可以使用電子郵件或DNS. 如需更多詳細資訊，請參閱「電子郵件驗證」及「DNS驗證」。

有效期間

ACM憑證的有效期為 13 個月 (395 天)。

受管續約和部署

ACM管理ACM憑證續約程序，並在更新憑證後佈建憑證。自動續約可協助您避免因憑證設定錯誤、撤銷或過期引起的停機時間。如需詳細資訊，請參閱 ACM憑證的受管理續約。

多個網域名稱

每個ACM憑證必須包含至少一個完整網域名稱 (FQDN)，而且您可以視需要新增其他名稱。例如，當您建立ACM憑證時`www.example.com`，`www.example.net`如果客戶可以使用其中一個名稱存取您的網站，您也可以新增名稱。這也適用的 bare 網域 (也稱為 Zone Apex 或裸網域)。也就是說，您可以為 www.example.com 申請ACM憑證，然後新增名稱 example.com。如需詳細資訊，請參閱 請求公有憑證。

萬用字元名稱

ACM可讓您在網域名稱中使用星號 (*) 來建立包含萬用字元名稱的ACM憑證，以保護相同網域中的多個網站。例如，*.example.com 可保護 www.example.com 和 images.example.com。

> **Note**
>
> 請求萬用字元憑證時，星號 (*) 必須在網域名稱的最左方，而且僅能保護一個子網域等級。例如，***.example.com** 可以保護 **login.example.com** 和 **test.example.com**，但不能保護 **test.login.example.com**。另請注意，***.example.com** 只可以保護 **example.com** 的子網域，無法保護 bare 或 apex 網域 (**example.com**)。不過，您可以在申請中指定多個網域名稱，以申請保護 bare 或 apex 網域及其子網域的憑證。例如，您可以申請保護 **example.com** 和 ***.example.com** 的憑證。

金鑰演算法

憑證必須指定演算法和金鑰大小。目前，支援下列RSA和橢圓曲線數位簽章演算法 (ECDSA) 公開金鑰演算法ACM。ACM可以使用星號 (*) 標記的演算法要求發行新憑證。其餘演算法僅支援匯入的憑證。

> **Note**
>
> 當您要求 CA 簽署的私有PKI憑證時 AWS Private CA，指定的簽署演算法系列 (RSA或ECDSA) 必須符合 CA 秘密金鑰的演算法系列。

- RSA位RSA_1024元
- RSA位元 (RSA_2048) *
- RSA位元 RSA_3072
- RSA位元 RSA_4096
- ECDSA256 位元 (EC_prime256v1) *
- ECDSA位元 (EC_secp384r1) *
- ECDSA位元 EC_secp521r1

ECDSA金鑰較小，提供與RSA金鑰相當的安全性，但運算效率更高。不過，並ECDSA非所有網路用戶端都支援。下表從改編而來 NIST，顯示了具有各種尺寸密鑰ECDSA的代表性安全強度RSA和密鑰。所有值均以位元為單位。

比較演算法和金鑰的安全性

安全性強度	RSA金鑰大小	ECDSA金鑰大小
128	3072	256
192	7680	384
256	15360	512

安全性強度可理解為 2 的次方，與破壞加密所需的猜測次數有關。例如，3072 位元RSA金鑰和 256 ECDSA 位元金鑰都可以擷取，而不會超過 2 128 次猜測。

如需協助您選擇演算法的資訊，請參閱如何在中評估和使用ECDSA憑證的 AWS 部落格文章 AWS Certificate Manager。

> **Important**
>
> 請注意，整合服務僅允許支援的演算法和金鑰大小與其資源相關聯。此外，它們的支援會因憑證匯入IAM或匯入憑證而有所不同ACM。如需更多詳細資訊，請參閱各服務的文件。
>
> - 如需 Elastic Load Balancing，請參閱應用程式負載平衡器的HTTPS接聽程式。
> - 如需詳細資訊 CloudFront，請參閱支援的SSL/TLS通訊協定和密碼。

Punycode

必須滿足以下與國際化網域名稱有關的 Punycode 要求：

1. 以 "<character><character>--" 模式開頭的網域名稱必須匹配 "xn--"。
2. 以 "xn--" 開頭的網域名稱也必須是有效的國際化網域名稱。

Punycode 範例

網域名稱	滿足 #1	滿足 #2	允許	注意
example。com	N/A	無	✓	不以 "<character><character>--" 開頭

網域名稱	滿足 #1	滿足 #2	允許	注意
a--example.com	N/A	無	✓	不以 "<character><character>--" 開頭
abc--example.com	N/A	無	✓	不以 "<character><character>--" 開頭
xn--xyz.com	是	是	✓	有效的國際化網域名稱 (解析為簡.com)
xn--example.com	是	否	✗	不是有效的國際化網域名稱
ab--example.com	否	否	✗	必須以 "xn--" 開頭

例外狀況

注意下列事項：

- ACM不提供延伸驗證 (EV) 憑證或組織驗證 (OV) 憑證。
- ACM不為SSL/TLS協議以外的任何東西提供證書。
- 您無法使用ACM憑證進行電子郵件加密。
- ACM目前不允許您選擇退出憑證的受管理ACM憑證續約。此外，您匯入的憑證無法使用受管續約ACM。
- 您無法為 Amazon 擁有的網域名稱申請憑證，例如結尾為 amazonaws.com、cloudfront.net 或 elasticbeanstalk.com 的網域名稱。
- 您無法下載ACM憑證的私密金鑰。
- 您無法在 Amazon 彈性運算雲端 (AmazonEC2) 網站或應用程式上直接安裝ACM憑證。不過，您可以搭配任何整合的服務使用憑證。如需詳細資訊，請參閱 服務整合 AWS Certificate Manager。

支援地區

請瀏覽AWS AWS 一般參考或AWS 區域表中的區域和端點，以查看的區域可用性ACM。

ACM 中的憑證為區域資源。若要將具有 Elastic Load Balancing 的憑證用於相同的完整網域名稱 (FQDN) 或多個 AWS 區域FQDNs中的一組，您必須要求或匯入每個區域的憑證。對於由提供的憑證 ACM，這表示您必須針對每個區域重新驗證憑證中的每個網域名稱。您無法在區域間複製憑證。

若要在 Amazon 使用ACM憑證 CloudFront，您必須在美國東部 (維吉尼亞北部) 區域申請或匯入憑證。ACM此區域中與發佈相關聯的憑證會 CloudFront 散發至針對該發佈所設定的所有地理位置。

服務整合 AWS Certificate Manager

AWS Certificate Manager 支持越來越多的 AWS 服務。您無法直接在您的ACM網站或應用程式上安裝 AWS 私有 CA 憑證或私人憑證。 AWS

> **Note**
>
> 公有ACM憑證可以安裝在連接到硝基 Enclave 的 Amazon EC2 執行個體上，但無法安裝到其他 Amazon 執行個體。EC2如需在未連線至 Nitro Enclave 的 Amazon EC2 執行個體上設定獨立網頁伺服器的詳細資訊，請參閱教學課程：在 Amazon Linux 2 上安裝LAMP網路伺服器或教學：使用 Amazon Linux 安裝LAMP網路伺服器。AMI

ACM下列服務支援憑證：

Elastic Load Balancing

Elastic Load Balancing 會自動將傳入的應用程式流量分配到多個 Amazon EC2 執行個體。它會偵測運作狀態不良的執行個體，並將流量重新路由至運作狀態良好的執行個體，直到運作狀態不良的執行個體恢復為止。Elastic Load Balancing 會自動擴展其處理容量的請求，以回應傳入的流量。如需負載平衡的詳細資訊，請參閱 Elastic Load Balancing 使用者指南。

一般而言，為了透過SSL/提供安全內容TLS，負載平衡器需要在負載平衡器或後端 Amazon EC2 執行個體上安裝SSL/TLS憑證。ACM與 Elastic Load Balancing 整合，可在負載平衡器上部署ACM憑證。如需詳細資訊，請參閱建立 Application Load Balancer。

Amazon CloudFront

Amazon CloudFront 是一種 Web 服務，可透過從全球節點網路傳遞您的內容，加快向最終使用者分發動態和靜態 Web 內容的速度。當使用者要求您提供服務的內容時 CloudFront，會將使用者路由至提供最低延遲的節點位置。這可確保盡可能以最佳效能交付內容。如果內容目前位於該節點，請立即 CloudFront 傳送。如果內容目前不在該節點，請從您識別為最終內容來源的 Amazon S3 儲

存貯體或 Web 伺服器 CloudFront 擷取該內容。如需有關的詳細資訊 CloudFront，請參閱 Amazon CloudFront 開發人員指南。

若要透過SSL/提供安全的內容TLS， CloudFront 需要在 CloudFront 發行版或支援的內容來源上安裝SSL/TLS憑證。ACM與整合 CloudFront 以在發行版上部 CloudFront 署ACM憑證。如需詳細資訊，請參閱取得SSL/TLS憑證。

> **Note**
>
> 若要在中使用ACM憑證 CloudFront，您必須在美國東部 (維吉尼亞北部) 區域申請或匯入憑證。

Amazon Cognito

Amazon Cognito 為您的 Web 和行動應用程式提供身分驗證、授權和使用者管理。用戶可以直接使用您的 AWS 帳戶 憑據或通過第三方（例如 Facebook，Amazon，谷歌或蘋果）登錄。如需有關 Amazon Cognito 的詳細資訊，請參閱《Amazon Cognito 開發人員指南》。

當您將 Cognito 使用者集區設定為使用 Amazon CloudFront 代理時， CloudFront 可能會放置ACM憑證以保護自訂網域的安全。在這種情況下，請注意，您必須先移除憑證的關聯， CloudFront 然後才能刪除憑證。

AWS Elastic Beanstalk

Elastic Beanstalk 可協助您在 AWS 雲端部署和管理應用程式，而不必擔心執行這些應用程式的基礎架構。 AWS Elastic Beanstalk 降低管理複雜性。您只需上傳應用程式，Elastic Beanstalk 就會自動處理容量佈建、負載平衡、擴展和應用程式運作狀態監控的細節。Elastic Beanstalk 使用 Elastic Load Balancing 服務來建立負載平衡器。如需 Elastic Beanstalk 的詳細資訊，請參閱 AWS Elastic Beanstalk 開發人員指南。

若要選擇憑證，您必須在 Elastic Beanstalk 主控台中為您的應用程式設定負載平衡器。如需詳細資訊，請參閱將 Elastic Beanstalk 環境的 Load Balancer 設定為終止。HTTPS

AWS App Runner

App Runner 是一種 AWS 服務，可提供快速、簡單且具成本效益的方式，從原始程式碼或容器映像直接部署到 AWS 雲端中可擴充且安全的 Web 應用程式。您不需要學習新技術、決定要使用哪個運算服務，也不需要知道如何佈建和設定 AWS 資源。如需 App Runner 的詳細資訊，請參閱 AWS App Runner 開發人員指南。

當您為自訂網域名稱與應用程式執行者服務建立關聯時，App Runner 會在內部建立可追蹤網域有效性的憑證。它們存儲在ACM。取消網域與服務的關聯或刪除服務後的七天內，App Runner 不會刪除這些憑證。這整套程序都會自動執行，您不需要自行新增或管理任何憑證。如需詳細資訊，請參閱 AWS App Runner 開發人員指南中的管理 App Runner 服務的自訂網域名稱。

Amazon API 网关

隨著行動裝置的擴散以及物聯網 (IoT) 的成長，建立可用於存取資料並與後端系統互動APIs的方式變得越來越普遍。 AWS您可以使用API閘道來發佈、維護、監控和保護您的APIs. 部署API到API閘道後，您可以設定自訂網域名稱以簡化對其的存取。若要設定自訂網域名稱，您必須提供SSL/TLS憑證。您可以使用ACM來產生或匯入憑證。如需 Amazon API 閘道的詳細資訊，請參閱 Amazon API 閘道開發人員指南。

AWS 硝基飛地

AWS 硝基隔離區是 Amazon 的一EC2項功能，可讓您從 Amazon 執行個體建立隔離的執行環境 (稱為隔離區)。EC2隔離區是獨立、強化且高度受限的虛擬機器。它們只提供與其上層執行個體的安全本機通訊端連線。不具有持久性儲存、互動式存取或外部聯網功能。使用者無法SSH進入 Enclave，而且上層執行個體的處理序、應用程式或使用者 (包括 root 或 admin) 也無法存取 Enclave 內的資料和應用程式。

EC2連接到硝基飛碟區的執行個體支援憑證。ACM如需詳細資訊，請參閱適用於 Nitro Enclaves 的 AWS Certificate Manager。

> **Note**
>
> 您無法將ACM憑證與未連線至 Nitro Enclave 的EC2執行個體建立關聯。

AWS CloudFormation

AWS CloudFormation 協助您建立 Amazon Web Services 資源的模型和設定。您可以建立描述要使用之 AWS 資源的範本，例如「Elastic Load Balancing」或「API閘道」。然後， AWS CloudFormation 會負責佈建和設定這些資源。您不需要單獨創建和配置 AWS 資源，並找出依賴於什麼; AWS CloudFormation 處理所有這些。ACM憑證包含為範本資源，這表示 AWS CloudFormation 可以要求ACM憑證，您可以與 AWS 服務搭配使用以啟用安全連線。此外，您可以使用許多可設定的 AWS 資源也隨附ACM憑證 AWS CloudFormation。

有關的一般資訊 CloudFormation，請參閱《AWS CloudFormation 使用者指南》。如需支援之ACM資源的相關資訊 CloudFormation，請參閱AWS::CertificateManager: 憑證。

透過提供的強大自動化功能 AWS CloudFormation，很容易超過您的憑證配額，尤其是新 AWS 帳戶。我們建議您遵循的ACM最佳做法 AWS CloudFormation。

> **Note**
>
> 如果您使用建立ACM憑證 AWS CloudFormation， AWS CloudFormation 堆疊會保持在 CREATE_IN_ PROGRESS 狀態。任何進一步的堆疊操作都將被延遲，直到您根據憑證驗證電子郵件中的指示操作為止。如需詳細資訊，請參閱在建立、更新或刪除堆疊操作期間，資源無法穩定。

AWS Amplify

Amplify 是一組專門打造的工具和功能，可讓前端 Web 和行動開發人員快速輕鬆地在其上建置完整堆疊的應用程式。 AWS Amplify 提供兩種服務：Amplify Hosting 和 Amplify Studio。Amplify Hosting 提供了一個 Git 型的工作流程，可用來託管具有連續部署的全堆疊無伺服器 Web 應用程式。Amplify Studio 是視覺化的開發環境，可簡化可擴展、全堆疊 Web 和行動應用程式的建立作業。使用 Studio 來構建具有一組 UI 組件的前端 ready-to-use UI，創建應用程序後端，然後將兩者連接在一起。如需有關 Amplify 的詳細資訊，請參閱《AWS Amplify 使用者指南》。

如果您將自訂網域連接到應用程式，Amplify 主控台會發出ACM憑證以保護它。

Amazon OpenSearch 服務

Amazon Ser OpenSearch vice 是一種搜尋和分析引擎，適用於日誌分析、即時應用程式監控和點擊串流分析等使用案例。如需詳細資訊，請參閱 Amazon OpenSearch 服務開發人員指南。

當您建立包含自訂網域和端點的 OpenSearch 服務叢集時，您可以使用ACM憑證佈建關聯的應用程式負載平衡器。

AWS Network Firewall

AWS Network Firewall 這是一項受管服務，可讓您輕鬆為所有 Amazon 虛擬私有雲部署基本網路保護 (VPCs)。如需詳細資訊，請參閱 AWS Network Firewall 開發人員指南。

Network Firewall 防火牆集成了用ACM於TLS檢查。如果您在 Network Firewall 中使用TLS檢查，則必須配置ACM憑證，以解密和重新加密TLS通過防火牆的SSL/流量。如需有關 Network Firewall 如何使用以進行TLS檢查的ACM詳細資訊，請參閱AWS Network Firewall 開發人員指南中的使用SSL/TLS憑證搭配TLS檢查組態的要求。

網站標章和信任標誌

Amazon 不提供網站標章，也不允許其商標做為網站標章：

- AWS Certificate Manager (ACM) 不提供您可在網站上使用的安全網站標章。如果您想要使用網站標章，可以從第三方供應商處獲得。我們建議您選擇供應商來評估並確定您的網站或商業行為的安全性。
- Amazon 不允許其商標或標誌做為憑證徽章、網站標章或信任標誌。這種類型的印章和徽章可以複製到不使用該ACM服務的站點，並且可以不適當地使用以虛假偽裝建立信任。為了保護我們的客戶和 Amazon 的聲譽，我們不允許以這種方式使用我們的商標和標誌。

配額

以下 AWS Certificate Manager (ACM) 服務配額適用於每個 AWS 帳戶的每個 AWS 區域。

若要查看可以調整的配額，請參閱AWS 一般參考指南中的ACM配額表。如需申請提高配額，請在 AWS Support Center (支援中心) 建立案例。

一般配額

項目	預設配額
ACM 憑證的數量 已過期和已撤銷的憑證會繼續計入此總額。 由 CA 簽署的憑證 AWS 私有 CA 不會計入此總數。	2500
每年的ACM憑證數量 (過去 365 天) 您每年、地區和帳戶最多可以要求兩倍的ACM憑證配額。例如，如果您的配額為 2,500，您可以在指定的區域和帳戶中每年要求最多 5,000 個ACM憑證。您同時最多只能有 2,500 個憑證。若要在一年內請求 5,000 個憑證，您必須在當年刪除 2,500 個憑證，以維持在配額數量內。如果	加倍您的帳戶配額

項目	預設配額
您在某段時間內需要超過 2,500 個憑證，您必須聯絡 AWS Support Center。 由 CA 簽署的憑證 AWS 私有 CA 不會計入此總數。	
匯入憑證的數量	2,500
每年 (過去 365 天) 匯入的憑證數量	加倍您的帳戶配額
每個ACM憑證的網域名稱數目 每個ACM憑證的預設配額為 10 個網域名稱。您的配額可能較佳。 您提交的第一個網域名稱會包含為憑證的主體常見名稱 (CN)。所有名稱皆包含於主體別名副檔名。 您可以申請多達 100 個網域名稱。若要要求增加配額，請在服務的「Service Quotas」主控台中ACM建立要求。不過，建立案例前，請確保了解如果使用電子郵件驗證，新增更多網域名稱可能會產生更多管理工作。如需詳細資訊，請參閱 網域驗證。 每個憑ACM證的網域名稱數量配額僅適用於由提供的憑證ACM。此配額不適用於您匯入的憑證ACM。下列各節僅適用於ACM憑證。	10

項目	預設配額
私人數目 CAs ACM與 AWS Private Certificate Authority (AWS 私有 CA) 集成。您可以使用ACM主控台 AWS CLI、或從託管ACMAPI的現有私有憑證授權單位 (CA) 要求私有憑證 AWS 私有 CA。這些憑證會在ACM環境中進行管理，並且與由發行的公用憑證具有相同的限制ACM。如需詳細資訊，請參閱 要求私有PKI憑證。您也可以使用獨立 AWS 私有 CA 服務來發行私有憑證。如需詳細資訊，請參閱發行私有最終實體憑證。 已刪除的私有 CA 將計入您的配額，直到其還原期間結束為止。如需詳細資訊，請參閱刪除您的私有 CA。	200
每個 CA 的私有憑證數量 (生命週期)	1,000,000

API費率配額

下列配額適用ACMAPI於每個區域和帳戶。ACM根據操作API，以不同的配額節流API請求。節流表示ACM拒絕其他有效的要求，因為要求超過作業的每秒要求數量配額。當請求被限制時，ACM返回一個`ThrottlingException`錯誤。下表列出每個API作業，以及ACM節流該作業要求的配額。

> **Note**
>
> 除了下表中列出的API操作之外，還ACM可以從調用外部操`IssueCertificate`作 AWS 私有 CA。如需上的 up-to-date 速率配額資訊`IssueCertificate`，請參閱的端點和配額 AWS 私有 CA。

每項ACMAPI作業的 R equests-per-second 配額

API呼叫	每秒請求數
`AddTagsToCertificate`	5

API呼叫	每秒請求數
DeleteCertificate	10
DescribeCertificate	10
ExportCertificate	5
GetAccountConfiguration	1
GetCertificate	10
ImportCertificate	1
ListCertificates	8
ListTagsForCertificate	10
PutAccountConfiguration	1
RemoveTagsFromCertificate	5
RenewCertificate	5
RequestCertificate	5
ResendValidationEmail	1
UpdateCertificateOptions	5

如需詳細資訊，請AWS Certificate Manager API參閱參考。

定價 AWS Certificate Manager

對於您管理的SSL/TLS憑證，您不需要支付額外費用 AWS Certificate Manager。您只需為執行網站或應用程式而建立的 AWS 資源付費。如需最新ACM定價資訊，請參閱 AWS 網站上的AWS Certificate Manager 服務定價頁面。

中的安全性 AWS Certificate Manager

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，這些架構是為了滿足對安全性最敏感的組織的需求而建置的。

安全是 AWS 與您之間共同承擔的責任。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。若要深入瞭解適用於的規範遵循計劃 AWS Certificate Manager，請參閱合規計劃的AWS 服務範圍範圍)。
- 雲端中的安全性 — 您的責任取決於您使用的 AWS 服務。您也必須對其他因素負責，包括資料的機密性、您公司的要求和適用法律和法規。

本文件可協助您瞭解如何在使用 AWS Certificate Manager (ACM) 時套用共同責任模型。下列主題將示範如何設定 ACM 以達到您的安全和合規目標。您也會學到如何使用其他可 AWS 協助您監控和保護 ACM 資源的服務。

主題

- 資料保護 AWS Certificate Manager
- 的 Identity and Access Management AWS Certificate Manager
- 韌性 AWS Certificate Manager
- AWS Certificate Manager中的基礎設施安全
- 最佳實務

資料保護 AWS Certificate Manager

AWS 共用責任模型適用於中的資料保護 AWS Certificate Manager。如此模型中所述， AWS 負責保護執行所有 AWS 雲端. 您負責維護在此基礎設施上託管內容的控制權。您也同時負責所使用 AWS 服務的安全組態和管理任務。如需資料隱私權的詳細資訊，請參閱資料隱私權常見問答集。如需有關歐洲資料保護的相關資訊，請參閱 AWS 安全性部落格上的 AWS 共同的責任模型和 GDPR 部落格文章。

基於資料保護目的，我們建議您使用 AWS IAM Identity Center 或 AWS Identity and Access Management (IAM) 保護 AWS 帳戶 登入資料並設定個別使用者。如此一來，每個使用者都只會獲得授與完成其任務所必須的許可。我們也建議您採用下列方式保護資料：

- 每個帳戶均要使用多重要素驗證 (MFA)。
- 使用 SSL/TLS 與 AWS 資源進行通訊。我們需要 TLS 1.2 並建議使用 TLS 1.3。
- 使用設定 API 和使用者活動記錄 AWS CloudTrail。
- 使用 AWS 加密解決方案以及其中的所有默認安全控制 AWS 服務。
- 使用進階的受管安全服務 (例如 Amazon Macie)，協助探索和保護儲存在 Amazon S3 的敏感資料。
- 如果您在透過命令列介面或 API 存取時需要經 AWS 過 FIPS 140-2 驗證的加密模組，請使用 FIPS 端點。如需有關 FIPS 和 FIPS 端點的更多相關資訊，請參閱聯邦資訊處理標準 (FIPS) 140-2 概觀。

我們強烈建議您絕對不要將客戶的電子郵件地址等機密或敏感資訊，放在標籤或自由格式的文字欄位中，例如名稱欄位。這包括當您使用控制台、API 或 AWS SDK AWS 服務 使用 ACM 或其他工作時。AWS CLI您在標籤或自由格式文字欄位中輸入的任何資料都可能用於計費或診斷日誌。如果您提供外部伺服器的 URL，我們強烈建議請勿在驗證您對該伺服器請求的 URL 中包含憑證資訊。

憑證私有金鑰的安全性

當您要求公用憑證時， AWS Certificate Manager (ACM) 會產生公開/私密 key pair。您針對匯入的憑證產生金鑰對。公有金鑰會成為憑證的一部分。ACM 會儲存憑證及其對應的私密金鑰，並使用 AWS Key Management Service (AWS KMS) 來協助保護私密金鑰。運作程序如下：

1. 當您第一次在 AWS 區域中要求或匯入憑證時，ACM 會建立一個受管理 AWS KMS key 的別名 aws/acm。此 KMS 金鑰在每個 AWS 帳戶和每個 AWS 區域中都是唯一的。
2. ACM 會使用此 KMS 金鑰來加密憑證的私有金鑰。ACM 只會存放加密版本的私有金鑰；ACM 不會以純文字形式存放私有金鑰。ACM 使用相同的 KMS 金鑰來加密特定 AWS 帳戶和特定 AWS 區域中所有憑證的私密金鑰。
3. 將憑證與整合 AWS Certificate Manager的服務相關聯時，ACM 會將憑證和加密的私有金鑰傳送到該服務。也會在中建立授權 AWS KMS ，允許服務使用 KMS 金鑰來解密憑證的私密金鑰。如需授權的詳細資訊，請參閱 AWS Key Management Service 開發人員指南中的使用授權。如需 ACM 支援之服務的詳細資訊，請參閱 服務整合 AWS Certificate Manager。

   > **Note**
   >
   > 您可以控制自動建立的 AWS KMS 授權。如果您因任何原因刪除此授權，就會失去該整合服務的 ACM 功能。

4. 整合的服務會使用 KMS 金鑰解密私有金鑰。然後，服務會使用憑證和解密的 (純文字) 私有金鑰與其用戶端建立安全的通訊管道 (SSL/TLS 工作階段)。

5. 憑證與整合的服務取消關聯時，步驟 3 建立的授予便會淘汰。這表示服務不能再使用 KMS 金鑰解密憑證的私有金鑰。

的 Identity and Access Management AWS Certificate Manager

AWS Identity and Access Management (IAM) 可協助系統管理員安全地控制 AWS 資源存取權。 AWS 服務 IAM管理員控制誰可以驗證（登錄）和授權（有權限）使用ACM資源。IAM是您 AWS 服務 可以免費使用的。

主題

- 物件
- 使用身分驗證
- 使用政策管理存取權
- 如何 AWS Certificate Manager 使用 IAM
- 以身分識別為基礎的原則範例 AWS Certificate Manager
- ACM API 許可：動作和資源參考
- AWS Certificate Manager 的 AWS 受管政策
- 搭配 ACM 使用條件索引鍵
- 搭配 ACM 使用服務連結角色 (SLR)
- 疑難排解 AWS Certificate Manager 身分和存取

物件

你如何使用 AWS Identity and Access Management （IAM）不同，具體取決於你在做的工作ACM。

服務使用者 — 如果您使用ACM服務執行工作，則管理員會為您提供所需的認證和權限。當您使用更多ACM功能來完成工作時，您可能需要其他權限。了解存取許可的管理方式可協助您向管理員請求正確的許可。如果無法存取中的圖徵ACM，請參閱疑難排解 AWS Certificate Manager 身分和存取。

服務管理員 — 如果您負責公司的ACM資源，您可能擁有完整的存取權ACM。決定您的服務使用者應該存取哪些ACM功能和資源是您的工作。然後，您必須向IAM管理員提交請求，才能變更服務使用者的權限。檢閱此頁面上的資訊，以瞭解的基本概念IAM。若要深入瞭解貴公司如何IAM搭配使用ACM，請參閱如何 AWS Certificate Manager 使用 IAM。

IAM系統管理員 — 如果您是IAM系統管理員，您可能想要瞭解如何撰寫原則來管理存取權的詳細資訊ACM。若要檢視可在中使用ACM的識別型原則範例IAM，請參閱。以身分識別為基礎的原則範例 AWS Certificate Manager

使用身分驗證

驗證是您 AWS 使用身分認證登入的方式。您必須以IAM使用者身分或假設IAM角色來驗證 (登入AWS)。 AWS 帳戶根使用者

您可以使用透過 AWS 身分識別來源提供的認證，以聯合身分識別身分登入。 AWS IAM Identity Center (IAM身分識別中心) 使用者、貴公司的單一登入驗證，以及您的 Google 或 Facebook 認證都是聯合身分識別的範例。當您以同盟身分登入時，您的管理員先前會使用IAM角色設定聯合身分識別。當您使 AWS 用同盟存取時，您會間接擔任角色。

根據您的使用者類型，您可以登入 AWS Management Console 或 AWS 存取入口網站。如需有關登入的詳細資訊 AWS，請參閱《AWS 登入 使用指南》 AWS 帳戶中的如何登入您的。

如果您 AWS 以程式設計方式存取，請 AWS 提供軟體開發套件 (SDK) 和命令列介面 (CLI)，以使用您的認證以密碼編譯方式簽署您的要求。如果您不使用 AWS 工具，則必須自行簽署要求。如需使用建議的方法自行簽署要求的詳細資訊，請參閱使用IAM者指南中的簽署 AWS API要求。

無論您使用何種身分驗證方法，您可能都需要提供額外的安全性資訊。例如， AWS 建議您使用多重要素驗證 (MFA) 來增加帳戶的安全性。若要深入瞭解，請參閱使用AWS IAM Identity Center 者指南中的多重要素驗證和使用多重要素驗證 (MFA) AWS的使用IAM者指南。

AWS 帳戶 根使用者

當您建立時 AWS 帳戶，您會從一個登入身分開始，該身分可完整存取該帳戶中的所有資源 AWS 服務和資源。此身分稱為 AWS 帳戶 root 使用者，可透過使用您用來建立帳戶的電子郵件地址和密碼登入來存取。強烈建議您不要以根使用者處理日常任務。保護您的根使用者憑證，並將其用來執行只能由根使用者執行的任務。如需需要您以 root 使用者身分登入的完整工作清單，請參閱《使用指南》中的〈需要 root 使用者認證的IAM工作〉。

聯合身分

最佳作法是要求人類使用者 (包括需要系統管理員存取權的使用者) 使用與身分識別提供者的同盟，才能使用臨時認證 AWS 服務 來存取。

聯合身分識別是來自企業使用者目錄的使用者、Web 身分識別提供者、Identi ty Center 目錄，或使用透過身分識別來源提供的認證進行存取 AWS 服務 的任何使用者。 AWS Directory Service同盟身分存取時 AWS 帳戶，他們會假設角色，而角色則提供臨時認證。

對於集中式存取權管理，我們建議您使用 AWS IAM Identity Center。您可以在 IAM Identity Center 中建立使用者和群組，也可以連線並同步至您自己身分識別來源中的一組使用者和群組，以便在所有應用程式 AWS 帳戶 和應用程式中使用。如需IAM身分識別中心的相關資訊，請參閱IAM識別中心是什麼？在《AWS IAM Identity Center 使用者指南》中。

IAM 使用者和群組

IAM使用者是您內部的身分，具 AWS 帳戶 有單一人員或應用程式的特定權限。在可能的情況下，我們建議您仰賴臨時登入資料，而不要建立具有長期認證 (例如密碼和存取金鑰) 的IAM使用者。不過，如果您的特定使用案例需要使用IAM者的長期認證，建議您輪換存取金鑰。如需詳細資訊，請參閱《使用指南》中的「IAM定期輪換存取金鑰」以瞭解需要長期認證的使用案例。

IAM群組是指定IAM使用者集合的身分識別。您無法以群組身分簽署。您可以使用群組來一次為多名使用者指定許可。群組可讓管理大量使用者許可的程序變得更為容易。例如，您可以擁有一個名為的群組，IAMAdmins並授與該群組管理IAM資源的權限。

使用者與角色不同。使用者只會與單一人員或應用程式建立關聯，但角色的目的是在由任何需要它的人員取得。使用者擁有永久的長期憑證，但角色僅提供暫時憑證。要了解更多信息，請參閱《IAM用戶指南》中的創建用戶（而不是角色）的IAM時間。

IAM 角色

IAM角色是您 AWS 帳戶 中具有特定權限的身份。它類似於用IAM戶，但不與特定人員相關聯。您可以 AWS Management Console 透過切換角色來暫時擔任中的角色。IAM您可以透過呼叫 AWS CLI 或 AWS API作業或使用自訂來擔任角色URL。如需有關使用角色方法的詳細資訊，請參閱《使用指南》中的IAM〈使用IAM角色〉。

IAM具有臨時認證的角色在下列情況下很有用：

- 聯合身分使用者存取 — 如需向聯合身分指派許可，請建立角色，並為角色定義許可。當聯合身分進行身分驗證時，該身分會與角色建立關聯，並獲授予由角色定義的許可。如需聯合角色的相關資訊，請參閱《使用指南》中的〈建立第三方身分識別提供IAM者的角色〉。如果您使用IAM身分識別中心，則需要設定權限集。為了控制身分驗證後可以存取的內IAM容，IAMIdentity Center 會將權限集與中的角色相關聯。如需有關許可集的資訊，請參閱 AWS IAM Identity Center 使用者指南中的許可集。
- 暫時IAM使用者權限 — IAM 使用者或角色可以假定某個IAM角色，暫時取得特定工作的不同權限。
- 跨帳戶存取 — 您可以使用IAM角色允許不同帳戶中的某個人 (受信任的主體) 存取您帳戶中的資源。角色是授予跨帳戶存取權的主要方式。但是，對於某些策略 AWS 服務，您可以將策略直接附加到資

源（而不是使用角色作為代理）。若要瞭解跨帳戶存取角色與以資源為基礎的政策之間的差異，請參閱《IAM使用指南》IAM中的〈跨帳號資源存取〉。

- 跨服務訪問 — 有些 AWS 服務 使用其他 AWS 服務功能。例如，當您在服務中撥打電話時，該服務通常會在 Amazon 中執行應用程式EC2或將物件存放在 Amazon S3 中。服務可能會使用呼叫主體的許可、使用服務角色或使用服務連結角色來執行此作業。
  - 轉寄存取工作階段 (FAS) — 當您使用使用IAM者或角色執行中的動作時 AWS，您會被視為主參與者。使用某些服務時，您可能會執行某個動作，進而在不同服務中啟動另一個動作。FAS會使用主參與者呼叫的權限 AWS 服務，並結合要求 AWS 服務 向下游服務發出要求。FAS只有當服務收到需要與其他 AWS 服務 資源互動才能完成的請求時，才會發出請求。在此情況下，您必須具有執行這兩個動作的許可。有關提出FAS請求時的策略詳細信息，請參閱轉發訪問會話。
  - 服務角色 — 服務角色是指服務代表您執行動作所代表的IAM角色。IAM管理員可以從中建立、修改和刪除服務角色IAM。如需詳細資訊，請參閱《IAM使用指南》 AWS 服務中的建立角色以將權限委派給
  - 服務連結角色 — 服務連結角色是連結至. AWS 服務服務可以擔任代表您執行動作的角色。服務連結角色會顯示在您的中， AWS 帳戶 且屬於服務所有。IAM管理員可以檢視 (但無法編輯服務連結角色) 的權限。
- 在 Amazon 上執行的應用程式 EC2 — 您可以使用IAM角色來管理在執行個體上EC2執行的應用程式以及發出 AWS CLI 或 AWS API請求的臨時登入資料。這比在EC2實例中存儲訪問密鑰更好。若要將 AWS 角色指派給EC2執行個體並讓其所有應用程式都能使用，請建立附加至執行個體的執行個體設定檔。執行個體設定檔包含角色，可讓執行個體上EC2執行的程式取得臨時登入資料。如需詳細資訊，請參閱使用者指南中的使用IAM角色將許可授與在 Amazon EC2 執行個體上執行的應IAM用程式。

要了解是否使用IAM角色還是用IAM戶，請參閱《用戶指南》中的「IAM創建IAM角色的時機（而不是用戶）」。

使用政策管理存取權

您可以透 AWS 過建立原則並將其附加至 AWS 身分識別或資源來控制中的存取。原則是一個物件 AWS ，當與身分識別或資源相關聯時，會定義其權限。 AWS 當主參與者 (使用者、root 使用者或角色工作階段) 提出要求時，評估這些原則。政策中的許可決定是否允許或拒絕請求。大多數原則會 AWS 以JSON文件的形式儲存在中。如需有關JSON原則文件結構和內容的詳細資訊，請參閱《IAM使用指南》中的策略概觀。JSON

管理員可以使用 AWS JSON策略來指定誰可以存取什麼內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

預設情況下，使用者和角色沒有許可。若要授與使用者對所需資源執行動作的權限，IAM管理員可以建立IAM策略。然後，系統管理員可以將IAM原則新增至角色，使用者可以擔任這些角色。

IAM原則會定義動作的權限，不論您用來執行作業的方法為何。例如，假設您有一個允許 `iam:GetRole` 動作的政策。具有該原則的使用者可以從 AWS Management Console AWS CLI、或取得角色資訊 AWS API。

身分型政策

以身分識別為基礎的原則是您可以附加至身分識別 (例如使用者、使用IAM者群組或角色) 的JSON權限原則文件。這些政策可控制身分在何種條件下能對哪些資源執行哪些動作。若要瞭解如何建立以身分識別為基礎的策略，請參閱《IAM使用指南》中的〈建立IAM策略〉。

身分型政策可進一步分類成內嵌政策或受管政策。內嵌政策會直接內嵌到單一使用者、群組或角色。受管理的策略是獨立策略，您可以將其附加到您的 AWS 帳戶. 受管政策包括 AWS 受管政策和客戶管理的策略。若要了解如何在受管策略或內嵌策略之間進行選擇，請參閱《IAM使用手冊》中的「在受管策略和內嵌策略之間進行選擇」。

資源型政策

以資源為基礎的JSON策略是您附加至資源的政策文件。以資源為基礎的政策範例包括IAM角色信任政策和 Amazon S3 儲存貯體政策。在支援資源型政策的服務中，服務管理員可以使用它們來控制對特定資源的存取權限。對於附加政策的資源，政策會定義指定的主體可以對該資源執行的動作以及在何種條件下執行的動作。您必須在資源型政策中指定主體。主參與者可以包括帳戶、使用者、角色、同盟使用者或。 AWS 服務

資源型政策是位於該服務中的內嵌政策。您無法在以資源為基礎的策略IAM中使用 AWS 受管政策。

存取控制清單 (ACLs)

存取控制清單 (ACLs) 控制哪些主參與者 (帳戶成員、使用者或角色) 具有存取資源的權限。ACLs類似於以資源為基礎的策略，雖然它們不使用JSON政策文件格式。

Amazon S3 和 Amazon VPC 是支持服務的示例ACLs。 AWS WAF若要進一步了解ACLs，請參閱 Amazon 簡單儲存服務開發人員指南中的存取控制清單 (ACL) 概觀。

其他政策類型

AWS 支援其他較不常見的原則類型。這些政策類型可設定較常見政策類型授予您的最大許可。

- 權限界限 — 權限界限是一項進階功能，您可以在其中設定以身分識別為基礎的原則可授與給IAM實體 (IAM使用者或角色) 的最大權限。您可以為實體設定許可界限。所產生的許可會是實體的身分型政策和其許可界限的交集。會在 `Principal` 欄位中指定使用者或角色的資源型政策則不會受到許可界限限制。所有這類政策中的明確拒絕都會覆寫該允許。如需有關權限界限的詳細資訊，請參閱《IAM使用指南》中的IAM實體的權限界限。
- 服務控制策略 (SCPs) — SCPs 是指定中組織或組織單位 (OU) 最大權限的JSON策略 AWS Organizations。 AWS Organizations 是一種用於分組和集中管理您企業擁 AWS 帳戶 有的多個服務。如果您啟用組織中的所有功能，則可以將服務控制策略 (SCPs) 套用至您的任何或所有帳戶。SCP限制成員帳戶中實體的權限，包括每個帳戶 AWS 帳戶根使用者。若要取得有關 Organizations 的更多資訊SCPs，請參閱《AWS Organizations 使用指南》中的〈SCPs運作方式〉
- 工作階段政策 – 工作階段政策是一種進階政策，您可以在透過編寫程式的方式建立角色或聯合使用者的暫時工作階段時，作為參數傳遞。所產生工作階段的許可會是使用者或角色的身分型政策和工作階段政策的交集。許可也可以來自資源型政策。所有這類政策中的明確拒絕都會覆寫該允許。如需詳細資訊，請參閱《IAM使用指南》中的工作階段原則。

多種政策類型

將多種政策類型套用到請求時，其結果形成的許可會更為複雜、更加難以理解。若要瞭解如何在涉及多個原則類型時 AWS 決定是否允許要求，請參閱IAM使用指南中的原則評估邏輯。

如何 AWS Certificate Manager 使用 IAM

在您用IAM來管理存取權之前ACM，請先瞭解哪些IAM功能可搭配使用ACM。

IAM您可以搭配使用的功能 AWS Certificate Manager

IAM 功能	ACM支持
身分型政策	是
資源型政策	否
政策動作	是
政策資源	是
政策條件索引鍵 (服務特定)	是

IAM 功能	ACM支持
ACLs	否
ABAC(策略中的標籤)	部分
臨時憑證	是
主體許可	是
服務角色	否
服務連結角色	是

若要深入瞭解其他 AWS 服務如何ACM與大部分IAM功能搭配使用，請參閱IAM使用者指南IAM中的使用AWS 服務。

適用於 ACM 的身分型政策

支援以身分識別為基礎的原則：是

以身分識別為基礎的原則是您可以附加至身分識別 (例如使用者、使用IAM者群組或角色) 的JSON權限原則文件。這些政策可控制身分在何種條件下能對哪些資源執行哪些動作。若要瞭解如何建立以身分識別為基礎的策略，請參閱《IAM使用指南》中的〈建立IAM策略〉。

使用以IAM身分識別為基礎的策略，您可以指定允許或拒絕的動作和資源，以及允許或拒絕動作的條件。您無法在身分型政策中指定主體，因為這會套用至連接的使用者或角色。若要瞭解可在JSON策略中使用的所有元素，請參閱《使用IAM者指南》中的IAMJSON策略元素參考資料。

ACM 的身分型政策範例

若要檢視以ACM身分為基礎的原則範例，請參閱。以身分識別為基礎的原則範例 AWS Certificate Manager

ACM 內的資源型政策

支援以資源為基礎的政策：否

以資源為基礎的JSON策略是您附加至資源的政策文件。以資源為基礎的政策範例包括IAM角色信任政策和 Amazon S3 儲存貯體政策。在支援資源型政策的服務中，服務管理員可以使用它們來控制對特定

資源的存取權限。對於附加政策的資源，政策會定義指定的主體可以對該資源執行的動作以及在何種條件下執行的動作。您必須在資源型政策中指定主體。主參與者可以包括帳戶、使用者、角色、同盟使用者或。 AWS 服務

若要啟用跨帳戶存取，您可以在以資源為基礎的策略中指定一個或多個帳戶中的一個或多個帳戶中的IAM實體作為主體。新增跨帳戶主體至資源型政策，只是建立信任關係的一半。當主參與者和資源不同時 AWS 帳戶，受信任帳戶中的IAM管理員也必須授與主參與者實體 (使用者或角色) 權限，才能存取資源。其透過將身分型政策連接到實體來授與許可。不過，如果資源型政策會為相同帳戶中的主體授予存取，這時就不需要額外的身分型政策。如需詳細資訊，請參閱《IAM使用指南》IAM中的〈跨帳號資源存取〉。

適用於 ACM 的政策動作

支援原則動作：是

管理員可以使用 AWS JSON策略來指定誰可以存取什麼內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

JSON策略`Action`元素描述了您可以用來允許或拒絕策略中存取的動作。策略動作通常與關聯的 AWS API操作具有相同的名稱。有一些例外情況，例如沒有匹配API操作的僅限權限的操作。也有一些作業需要政策中的多個動作。這些額外的動作稱為相依動作。

政策會使用動作來授予執行相關聯動作的許可。

若要查看ACM動作清單，請參閱服務授權參考 AWS Certificate Manager中所定義的動作。

中的策略動作在動作之前ACM使用下列前置詞：

```
acm
```

若要在單一陳述式中指定多個動作，請用逗號分隔。

```
"Action": [
      "acm:action1",
      "acm:action2"
         ]
```

若要檢視以ACM身分為基礎的原則範例，請參閱。以身分識別為基礎的原則範例 AWS Certificate Manager

ACM 的政策資源

支援原則資源：是

管理員可以使用 AWS JSON策略來指定誰可以存取什麼內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

ResourceJSON原則元素會指定要套用動作的一個或多個物件。陳述式必須包含 Resource 或 NotResource 元素。最佳做法是使用其 Amazon 資源名稱 (ARN) 指定資源。您可以針對支援特定資源類型的動作 (稱為資源層級許可) 來這麼做。

對於不支援資源層級許可的動作 (例如列出操作)，請使用萬用字元 (*) 來表示陳述式適用於所有資源。

```
"Resource": "*"
```

若要查看ACM資源類型及其清單ARNs，請參閱服務授權參考 AWS Certificate Manager中所定義的資源。若要瞭解您可以針對每個資源指定哪些動作，請參閱由定義ARN的動作 AWS Certificate Manager。

若要檢視以ACM身分為基礎的原則範例，請參閱。以身分識別為基礎的原則範例 AWS Certificate Manager

ACM 的政策條件索引鍵

支援服務特定政策條件金鑰：是

管理員可以使用 AWS JSON策略來指定誰可以存取什麼內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

Condition 元素 (或 Condition 區塊) 可讓您指定使陳述式生效的條件。Condition 元素是選用項目。您可以建立使用條件運算子的條件運算式 (例如等於或小於)，來比對政策中的條件和請求中的值。

若您在陳述式中指定多個 Condition 元素，或是在單一 Condition 元素中指定多個索引鍵， AWS 會使用邏輯 AND 操作評估他們。如果您為單一條件索引鍵指定多個值，請使用邏輯OR運算來 AWS 評估條件。必須符合所有條件，才會授與陳述式的許可。

您也可以在指定條件時使用預留位置變數。例如，只有在IAM使用者名稱標記資源時，您才可以授與IAM使用者存取資源的權限。如需詳細資訊，請參閱《IAM使用指南》中的IAM政策元素：變數和標籤。

AWS 支援全域條件金鑰和服務特定條件金鑰。若要查看所有 AWS 全域條件索引鍵，請參閱《使用指南》中的AWS 全域條件內IAM容索引鍵。

若要查看ACM條件索引鍵清單，請參閱服務授權參考 AWS Certificate Manager中的條件金鑰。若要瞭解您可以使用條件索引鍵的動作和資源，請參閱定義的動作 AWS Certificate Manager。

若要檢視以ACM身分為基礎的原則範例，請參閱。以身分識別為基礎的原則範例 AWS Certificate Manager

ACM 中的 ACLs

支持ACLs：無

存取控制清單 (ACLs) 控制哪些主參與者 (帳戶成員、使用者或角色) 具有存取資源的權限。ACLs類似於以資源為基礎的策略，雖然它們不使用JSON政策文件格式。

ABAC與 ACM

支援 ABAC (策略中的標籤): 部分

以屬性為基礎的存取控制 (ABAC) 是一種授權策略，可根據屬性定義權限。在中 AWS，這些屬性稱為標籤。您可以將標籤附加至IAM實體 (使用者或角色) 和許多 AWS 資源。標記實體和資源是的第一步ABAC。然後，您可以設計ABAC策略，以便在主參與者的標籤與他們嘗試存取的資源上的標籤相符時允許作業。

ABAC在快速成長的環境中很有幫助，並且有助於原則管理變得繁瑣的情況。

如需根據標籤控制存取，請使用 `aws:ResourceTag/key-name`、`aws:RequestTag/key-name` 或 `aws:TagKeys` 條件索引鍵，在政策的條件元素中，提供標籤資訊。

如果服務支援每個資源類型的全部三個條件金鑰，則對該服務而言，值為 Yes。如果服務僅支援某些資源類型的全部三個條件金鑰，則值為 Partial。

如需有關的詳細資訊ABAC，請參閱什麼是ABAC？ 在《IAM使用者指南》中。若要檢視包含設定步驟的自學課程ABAC，請參閱《使用指南》中的〈使用以屬性為基礎的存取控制 (ABAC) IAM 〉。

將臨時憑證與 ACM 搭配使用

支持臨時憑據：是

當您使用臨時憑據登錄時，某些 AWS 服務 不起作用。如需其他資訊，包括哪些 AWS 服務 與臨時登入資料搭配使用 AWS 服務 ，請參閱《IAM使用指南》IAM中的使用方式。

如果您使用除了使用者名稱和密碼以外的任何方法登入，則您正在 AWS Management Console 使用臨時認證。例如，當您 AWS 使用公司的單一登入 (SSO) 連結存取時，該程序會自動建立臨時認證。當您以使用者身分登入主控台，然後切換角色時，也會自動建立臨時憑證。如需有關切換角色的詳細資訊，請參閱《IAM使用者指南》中的〈切換到角色 (主控台)〉。

您可以使用 AWS CLI 或手動建立臨時認證 AWS API。然後，您可以使用這些臨時登入資料來存取 AWS。 AWS 建議您動態產生臨時登入資料，而不是使用長期存取金鑰。如需詳細資訊，請參閱IAM。

ACM 的跨服務主體權限

支援轉寄存取工作階段 (FAS)：是

當您使用使用IAM者或角色在中執行動作時 AWS，您會被視為主參與者。使用某些服務時，您可能會執行某個動作，進而在不同服務中啟動另一個動作。FAS會使用主參與者呼叫的權限 AWS 服務，並結合要求 AWS 服務 向下游服務發出要求。FAS只有當服務收到需要與其他 AWS 服務 資源互動才能完成的請求時，才會發出請求。在此情況下，您必須具有執行這兩個動作的許可。有關提出FAS請求時的策略詳細信息，請參閱轉發訪問會話。

ACM 的服務角色

支援服務角色：否

服務角色是服務假定代表您執行動作的IAM角色。IAM管理員可以從中建立、修改和刪除服務角色 IAM。如需詳細資訊，請參閱《IAM使用指南》 AWS 服務中的建立角色以將權限委派給

> **Warning**
>
> 變更服務角色的權限可能會中斷ACM功能。只有在ACM提供指引時才編輯服務角色。

ACM 的服務連結角色

支援服務連結角色：是

服務連結角色是一種連結至. AWS 服務服務可以擔任代表您執行動作的角色。服務連結角色會顯示在您的中， AWS 帳戶 且屬於服務所有。IAM管理員可以檢視 (但無法編輯服務連結角色) 的權限。

如需有關建立或管理服務連結角色的詳細資訊，請參閱使用IAM的AWS 服務。在表格中尋找服務，其中包含服務連結角色欄中的 Yes。選擇是連結，以檢視該服務的服務連結角色文件。

以身分識別為基礎的原則範例 AWS Certificate Manager

依預設，使用者和角色沒有建立或修改ACM資源的權限。他們也無法使用 AWS Management Console、 AWS Command Line Interface (AWS CLI) 或執行工作 AWS API。若要授與使用者對所需資源執行動作的權限，IAM管理員可以建立IAM策略。然後，系統管理員可以將IAM原則新增至角色，使用者可以擔任這些角色。

若要瞭解如何使用這些範例原則文件來建立以IAM身分識別為基礎的JSON策略，請參閱使用指南中的IAM建立IAM策略。

如需有關由定義的動作和資源類型的詳細資訊ACM，包括每個ARNs資源類型的格式，請參閱服務授權參考 AWS Certificate Manager中的動作、資源和條件索引鍵。

主題

- 政策最佳實務
- 使用 ACM 主控台
- 允許使用者檢視他們自己的許可
- 列出憑證
- 擷取憑證
- 匯入憑證
- 刪除憑證

政策最佳實務

以身分識別為基礎的政策會決定某人是否可以建立、存取或刪除您帳戶中的ACM資源。這些動作可能會讓您的 AWS 帳戶產生費用。當您建立或編輯身分型政策時，請遵循下列準則及建議事項：

- 開始使用 AWS 受管原則並邁向最低權限權限 — 若要開始將權限授與使用者和工作負載，請使用可授與許多常見使用案例權限的AWS 受管理原則。它們在您的 AWS 帳戶. 建議您透過定義特定於您使用案例的 AWS 客戶管理政策，進一步降低使用權限。如需詳細資訊，請參閱AWS 《IAM使用指南》中針對工作職能的AWS 受管理策略或受管理的策略。
- 套用最低權限權限 — 當您使用原則設定權限時，IAM只授與執行工作所需的權限。為實現此目的，您可以定義在特定條件下可以對特定資源採取的動作，這也稱為最低權限許可。如需有關使用套用權限IAM的詳細資訊，請參閱《使用指南》IAM中的IAM《策略與權限》。
- 使用IAM策略中的條件進一步限制存取 — 您可以在策略中新增條件，以限制對動作和資源的存取。例如，您可以撰寫政策條件，以指定必須使用傳送所有要求SSL。您也可以使用條件來授與對服務動

作的存取權 (如透過特定) 使用這些動作 AWS 服務，例如 AWS CloudFormation。如需詳細資訊，請參閱《IAM使用指南》中的IAMJSON策略元素：條件。

- 使用 IAM Access Analyzer 驗證您的原IAM則，以確保安全和功能性的權限 — IAM Access Analyzer 會驗證新的和現有的原則，以便原則遵循IAM原則語言 (JSON) 和IAM最佳做法。IAMAccess Analyzer 提供超過 100 項原則檢查和可行的建議，協助您撰寫安全且功能正常的原則。如需詳細資訊，請參閱IAM使IAM用指南中的存取分析器原則驗證。
- 需要多因素驗證 (MFA) — 如果您的案例需要使IAM用者或 root 使用者 AWS 帳戶，請開啟以取得額外MFA的安全性。若要在呼叫API作業MFA時需要，請在原則中新增MFA條件。如需詳細資訊，請參閱《IAM使用指南》中的 < 設定MFA受保護的API存取 > 。

如需中最佳作法的詳細資訊IAM，請參閱《IAM使用指南》IAM中的「安全性最佳作法」。

使用 ACM 主控台

若要存取 AWS Certificate Manager 主控台，您必須擁有最少一組權限。這些權限必須允許您列出和檢視有關 AWS 帳戶. ACM 如果您建立比最基本必要許可更嚴格的身分型政策，則對於具有該政策的實體 (使用者或角色) 而言，主控台就無法如預期運作。

您不需要為只對 AWS CLI 或撥打電話的使用者允許最低主控台權限 AWS API。相反地，只允許存取符合他們嘗試執行之API作業的動作。

若要確保使用者和角色仍可使用ACM主控台，請同時將ACM*AWSCertificateManagerReadOnly* AWS 受管理的原則附加至實體。如需詳細資訊，請參閱《使用指南》中的〈將權限新增至IAM使用者〉。

允許使用者檢視他們自己的許可

此範例顯示如何建立原則，讓使IAM用者檢視附加至其使用者身分識別的內嵌和受管理原則。此原則包含在主控台上或以程式設計方式使用或完成此動作的 AWS CLI 權限 AWS API。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
```

```
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

列出憑證

下列原則可讓使用者列出使用者帳戶中的所有ACM憑證。

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

> **Note**
>
> ACM憑證出現在 Elastic Load Balancing 和 CloudFront 主控台中需要此權限。

擷取憑證

下列原則可讓使用者擷取特定ACM憑證。

```
{
            "Version":"2012-10-17",
            "Statement":{
            "Effect":"Allow",
            "Action":"acm:GetCertificate",
            "Resource":"arn:aws:acm:region:account:certificate/certificate_ID"
            }
            }
```

匯入憑證

以下政策可讓使用者匯入憑證。

```
{
            "Version":"2012-10-17",
            "Statement":{
            "Effect":"Allow",
            "Action":"acm:ImportCertificate",
            "Resource":"arn:aws:acm:region:account:certificate/certificate_ID"
            }
            }
```

刪除憑證

下列原則可讓使用者刪除特定ACM憑證。

```
{
            "Version":"2012-10-17",
            "Statement":{
            "Effect":"Allow",
            "Action":"acm:DeleteCertificate",
            "Resource":"arn:aws:acm:region:account:certificate/certificate_ID"
            }
            }
```

ACM API 許可：動作和資源參考

當您設定存取控制並撰寫可連接到 IAM 使用者或角色的許可政策時，可以使用以下表格做為參考。表格中的第一欄會列出每個 AWS Certificate Manager API 操作。您可以在政策的 `Action` 元素中指定動作。其餘欄位提供其他資訊：

您可以在 ACM 政策中使用 IAM 政策元素來表達條件。如需完整的清單，請參閱 IAM 使用者指南中的可用金鑰。

> **Note**
>
> 若要指定動作，請使用後接 API 操作名稱的 `acm:` 字首 (例如，`acm:RequestCertificate`)。

ACM API 作業與許可

ACM API 作業	必要許可 (API 操作)	資源
AddTagsToCertificate	`acm:AddTagsToCertificate`	`arn:aws:acm:`*region*`:`*account*`:certificate/`*certificate_ID*
DeleteCertificate	`acm:DeleteCertificate`	`arn:aws:acm:`*region*`:`*account*`:certificate/`*certificate_ID*
DescribeCertificate	`acm:DescribeCertificate`	`arn:aws:acm:`*region*`:`*account*`:certificate/`*certificate_ID*
ExportCertificate	`acm:ExportCertificate`	`arn:aws:acm:`*region*`:`*account*`:certificate/`*certificate_ID*
GetAccountConfiguration	`acm:GetAccountConfiguration`	*

ACM API 作業	必要許可 (API 操作)	資源
GetCertificate	acm:GetCertificate	arn:aws:acm: *region*:*account*:certificate/ *certificate_ID*
ImportCertificate	acm:ImportCertificate	arn:aws:acm: *region*:*account*:certificate/* 或 *
ListCertificates	acm:ListCertificates	*
ListTagsForCertificate	acm:ListTagsForCertificate	arn:aws:acm: *region*:*account*:certificate/ *certificate_ID*
PutAccountConfiguration	acm:PutAccountConfiguration	*
RemoveTagsFromCertificate	acm:RemoveTagsFromCertificate	arn:aws:acm: *region*:*account*:certificate/ *certificate_ID*
RequestCertificate	acm:RequestCertificate	arn:aws:acm: *region*:*account*:certificate/* 或 *
ResendValidationEmail	acm:ResendValidationEmail	arn:aws:acm: *region*:*account*:certificate/ *certificate_ID*

ACM API 作業	必要許可 (API 操作)	資源
UpdateCertificateOptions	acm:UpdateCertificateOptions	arn:aws:acm: *region*:*account*:certificate/ *certificate_ID*

AWS Certificate Manager 的 AWS 受管政策

AWS 受管政策是由 AWS 建立和管理的獨立政策。AWS 受管政策的設計在於為許多常見使用案例提供許可，如此您就可以開始將許可指派給使用者、群組和角色。

請記住，AWS 受管政策可能不會授與您特定使用案例的最低權限許可，因為它們可供所有 AWS 客戶使用。我們建議您定義使用案例專屬的客戶管理政策，以便進一步減少許可。

您無法更改 AWS 受管政策中定義的許可。如果 AWS 更新 AWS 受管政策中定義的許可，更新會影響政策連接的所有主體身分 (使用者、群組和角色)。在推出新的 AWS 服務 或有新的 API 操作可供現有服務使用時，AWS 很可能會更新 AWS 受管政策。

如需詳細資訊，請參閱《IAM 使用者指南》https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies_managed-vs-inline.html#aws-managed-policies中的 AWS 受管政策。

AWSCertificateManagerReadOnly

此政策提供 ACM 憑證唯讀存取權，可讓使用者描述、列出及擷取 ACM 憑證。

```
{
"Version":"2012-10-17",
"Statement":{
    "Effect":"Allow",
    "Action":[
        "acm:DescribeCertificate",
        "acm:ListCertificates",
        "acm:GetCertificate",
        "acm:ListTagsForCertificate",
        "acm:GetAccountConfiguration"
```

```
    ],
    "Resource":"*"
}
}
```

若要在主控台中檢視此 AWS 受管政策，請前往 https://console.aws.amazon.com/iam/home#policies/arn:aws:iam::aws:policy/AWSCertificateManagerReadOnly。

AWSCertificateManagerFullAccess

此政策提供所有 ACM 動作和資源的完整存取權限。

```
{
"Version":"2012-10-17",
"Statement":[
    {
        "Effect":"Allow",
        "Action":[
            "acm:*"
        ],
        "Resource":"*"
    },
    {
        "Effect":"Allow",
        "Action":"iam:CreateServiceLinkedRole",
        "Resource":"arn:aws:iam::*:role/aws-service-role/acm.amazonaws.com/
AWSServiceRoleForCertificateManager*",
        "Condition":{
            "StringEquals":{
                "iam:AWSServiceName":"acm.amazonaws.com"
            }
        }
    },
    {
        "Effect":"Allow",
        "Action":[
            "iam:DeleteServiceLinkedRole",
            "iam:GetServiceLinkedRoleDeletionStatus",
            "iam:GetRole"
        ],
        "Resource":"arn:aws:iam::*:role/aws-service-role/acm.amazonaws.com/
AWSServiceRoleForCertificateManager*"
```

```
    }
    ]
}
```

若要在主控台中檢視此 AWS 受管政策，請前往 https://console.aws.amazon.com/iam/home#policies/arn:aws:iam::aws:policy/AWSCertificateManagerFullAccess。

AWS 受管政策的 ACM 更新項目

檢視自 ACM 開始追蹤 AWS 受管政策變更以來的更新詳細資訊。如需有關此頁面變更的自動提醒，請訂閱 ACM 文件歷史紀錄 頁面的 RSS 摘要。

變更	描述	日期
為 AWSCertificateManagerReadOnly 政策新增 `GetAccountConfiguration` 支援。	所以此 `AWSCertificateManagerReadOnly` 政策現在包含呼叫 `GetAccountConfiguration` API 動作的許可。	2021 年 3 月 3 日
ACM 開始追蹤變更	ACM 開始追蹤其 AWS 受管政策的變更。	2021 年 3 月 3 日

搭配 ACM 使用條件索引鍵

AWS Certificate Manager 使用 AWS Identity and Access Management (IAM) 條件索引鍵來限制憑證請求的存取權。藉由 IAM 政策或服務控制政策 (SCP) 中的條件索引鍵，您可以建立符合組織準則的憑證請求。

> **Note**
>
> 將 ACM 條件索引鍵與 AWS 全域條件索引鍵 (例如 `aws:PrincipalArn`) 結合，可進一步限制對特定使用者或角色執行的動作。

ACM 的支援條件

ACM API 作業與支援條件

條件索引鍵	支援的 ACM API 作業	類型	描述
acm:ValidationMethod	RequestCertificate	字串 (EMAIL、DNS)	根據 ACM 驗證方法篩選請求
acm:DomainNames	RequestCertificate	ArrayOfString	根據 ACM 請求中的網域名稱篩選
acm:KeyAlgorithm	RequestCertificate	字串	根據 ACM 索引鍵演算法和大小篩選請求
acm:CertificateTransparencyLogging	RequestCertificate	字串 (ENABLED、DISABLED)	根據 ACM 憑證透明度記錄偏好設定篩選請求
acm:CertificateAuthority	RequestCertificate	ARN	根據 ACM 請求中的憑證授權機構篩選請求

範例 1：限制驗證方法

除了使用 arn:aws:iam::123456789012:role/AllowedEmailValidation 角色發送的請求之外，以下政策會拒絕使用電子郵件驗證方法傳送的新憑證請求。

```
{
    "Version":"2012-10-17",
    "Statement":{
        "Effect":"Deny",
        "Action":"acm:RequestCertificate",
        "Resource":"*",
        "Condition":{
            "StringLike" : {
                "acm:ValidationMethod":"EMAIL"
            },
```

```
            "ArnNotLike": {
                "aws:PrincipalArn": [ "arn:aws:iam::123456789012:role/
AllowedEmailValidation"]
            }
        }
    }
}
```

範例 2：防範萬用字元網域

以下政策會拒絕使用萬用字元網域的所有新 ACM 憑證請求。

```
{
    "Version":"2012-10-17",
    "Statement":{
        "Effect":"Deny",
        "Action":"acm:RequestCertificate",
        "Resource":"*",
        "Condition": {
            "ForAnyValue:StringLike": {
                "acm:DomainNames": [
                    "${*}.*"
                ]
            }
        }
    }
}
```

範例 3：限制憑證網域

以下政策會拒絕網域結尾不是 *.amazonaws.com 的所有新 ACM 憑證請求。

```
{
    "Version":"2012-10-17",
    "Statement":{
        "Effect":"Deny",
        "Action":"acm:RequestCertificate",
        "Resource":"*",
```

```
        "Condition": {
            "ForAnyValue:StringNotLike": {
                "acm:DomainNames": ["*.amazonaws.com"]
            }
        }
    }
}
```

政策可以進一步限制為特定的子網域。此政策只會允許每個網域符合至少一個網域名稱條件的請求。

```
{
    "Version":"2012-10-17",
    "Statement":{
        "Effect":"Deny",
        "Action":"acm:RequestCertificate",
        "Resource":"*",
        "Condition": {
            "ForAllValues:StringNotLike": {
                "acm:DomainNames": ["support.amazonaws.com", "developer.amazonaws.com"]
            }
        }
    }
}
```

範例 4：限制索引鍵演算法

以下政策使用條件索引鍵 StringNotLike，只允許使用 ECDSA 384 位元 (EC_secp384r1) 索引鍵演算法請求取得憑證。

```
{
    "Version":"2012-10-17",
        "Statement":{
        "Effect":"Deny",
        "Action":"acm:RequestCertificate",
        "Resource":"*",
        "Condition":{
            "StringNotLike" : {
                "acm:KeyAlgorithm":"EC_secp384r1"
```

```
            }
        }
    }
}
```

以下政策使用條件索引鍵 StringLike 和萬用字元 * 比對功能，防範 ACM 中出現使用任何 RSA 索引鍵演算法的新憑證請求。

```
{
    "Version":"2012-10-17",
    "Statement":{
        "Effect":"Deny",
        "Action":"acm:RequestCertificate",
        "Resource":"*",
        "Condition":{
            "StringLike" : {
                "acm:KeyAlgorithm":"RSA*"
            }
        }
    }
}
```

範例 5：限制憑證授權機構

以下政策只允許使用所提供私有憑證授權機構 (PCA) ARN 的私有憑證請求。

```
{
    "Version":"2012-10-17",
    "Statement":{
        "Effect":"Deny",
        "Action":"acm:RequestCertificate",
        "Resource":"*",
        "Condition":{
            "StringNotLike": {
                "acm:CertificateAuthority":" arn:aws:acm-
pca:region:account:certificate-authority/CA_ID"
            }
        }
```

```
    }
}
```

此政策使用 acm:CertificateAuthority 條件：僅允許 Amazon 信任服務發出的公開信任憑證請求。將憑證授權機構 ARN 設定為 false 可防範來自 PCA 的私有憑證請求。

```
{
"Version":"2012-10-17",
    "Statement":{
        "Effect":"Deny",
        "Action":"acm:RequestCertificate",
        "Resource":"*",
        "Condition":{
            "Null" : {
                "acm:CertificateAuthority":"false"
            }
        }
    }
}
```

搭配 ACM 使用服務連結角色 (SLR)

AWS Certificate Manager 使用 AWS Identity and Access Management (IAM) 服務連結角色來啟用受管理 ACM 憑證的自動續約。服務連結角色 (SLR) 是一種直接連結至 ACM 服務的 IAM 角色。此角色由 ACM 預先定義，包含本服務代您呼叫其他 AWS 服務需要的所有許可。

SLR 可讓 ACM 設定程序更為簡單，因為您不必手動新增必要的自動憑證簽署許可。ACM 會定義期 SLR 的許可，除非另外定義，否則只有 ACM 才能擔任此角色。定義的許可包括信任政策和許可政策，且該許可政策無法附加至其他 IAM 實體。

如需關於支援 SLR 的其他服務的資訊，請參閱可搭配 IAM 運作的AWS 服務，並且在服務連結角色直欄中，尋找顯示為是的服務。選擇具有連結的 Yes (是)，以檢視該服務的 SLR 說明文件。

ACM 的 SLR 許可

ACM 使用名為 Amazon Certificate Manager 服務角色政策的 SLR。

AWSServiceRoleForCertificateManager SLR 信任以下服務擔任該角色：

- acm.amazonaws.com

此角色許可政策允許 ACM 對指定資源完成下列動作：

- 動作：acm-pca:IssueCertificate, acm-pca:GetCertificate on "*"

您必須設定許可，IAM 實體 (如使用者、群組或角色) 才可建立、編輯或刪除 SLR。如需詳細資訊，請參閱《IAM 使用者指南》中的服務連結角色許可。

> **Important**
>
> ACM 可能會提醒您無法判斷您的帳戶中是否存在 SLR。如果必要的 iam:GetRole 許可已授與給您帳戶的 ACM SLR，則 SLR 建立後就不會再次發出提醒。如果再次發出提醒，表示您或您的帳戶管理員可能需要授與 iam:GetRole 許可給 ACM，或為您的帳戶與 ACM 受管政策 AWSCertificateManagerFullAccess 建立關聯。

為 ACM 建立 SLR

您不需要手動建立 ACM 使用的 SLR。當您使用 AWS Management Console、或 AWS API 發行 ACM 憑證時 AWS CLI，ACM 會在您第一次選擇私有 CA 來簽署憑證時，為您建立 SLR。

如果您遇到訊息，指出 ACM 無法判斷您的帳戶中是否存在 SLR，這可能表示您的帳戶未授予需要的讀取權限。 AWS 私有 CA 這並不會阻止安裝 SLR，而且您仍然可以發行憑證，但 ACM 將無法自動續約憑證，直到您解決問題為止。如需詳細資訊，請參閱 ACM服務連結角色的問題 () SLR。

> **Important**
>
> 此 SLR 可以顯示在您的帳戶，如果您於其他服務中完成一項動作時，可以使用支援此角色的功能。此外，如果您在 2017 年 1 月 1 日之前使用 ACM 服務，則當它開始支援 SLR 時，ACM 就會在您的帳戶中建立 AWSServiceRoleForCertificateManager 角色。若要進一步了解，請參閱我的 IAM 帳戶中出現的新角色。

若您刪除了此 SLR 而之後需要重新建立，可以使用下列其中一種方法：

- 在 IAM 主控台中，選擇 [角色]、[建立角色]、[Cer tificate Manager]，以CertificateManagerServiceRolePolicy使用案例建立新角色。

- 使用 IAM API CreateServiceLinkedRole或對應的 AWS CLI 命令 create-service-linked-role，建立包含acm.amazonaws.com服務名稱的單反相機。

如需詳細資訊，請參閱 IAM 使用者指南中的建立服務連結角色。

為 ACM 編輯 SLR

ACM 不允許您編輯 AWSServiceRoleForCertificateManager 服務連結角色。建立 SLR 後，因為各種實體皆會參考該角色，所以無法變更該角色的名稱。然而，您可使用 IAM 來編輯角色描述。如需更多資訊，請參閱 IAM 使用者指南中的編輯服務連結角色。

為 ACM 刪除 SLR

您通常不需要刪除 AWSServiceRoleForCertificateManager 單鏡反光相機。不過，您可以使用 IAM 主控台 AWS CLI 或 AWS API 手動刪除角色。如需詳細資訊，請參閱《IAM 使用者指南》中的刪除服務連結角色。

ACM SLR 的支援區域

ACM 支援在 ACM 和 AWS 私有 CA 可用的所有區域中使用 SLR。如需詳細資訊，請參閱 AWS 區域與端點。

區域名稱	區域身分	ACM 中的支援
美國東部 (維吉尼亞北部)	us-east-1	是
美國東部 (俄亥俄)	us-east-2	是
美國西部 (加利佛尼亞北部)	us-west-1	是
美國西部 (奧勒岡)	us-west-2	是
亞太區域 (孟買)	ap-south-1	是
亞太區域 (大阪)	ap-northeast-3	是
亞太區域 (首爾)	ap-northeast-2	是
亞太區域 (新加坡)	ap-southeast-1	是
亞太區域 (雪梨)	ap-southeast-2	是

區域名稱	區域身分	ACM 中的支援
亞太區域 (東京)	ap-northeast-1	是
加拿大 (中部)	ca-central-1	是
歐洲 (法蘭克福)	eu-central-1	是
歐洲 (蘇黎世)	eu-central-2	是
歐洲 (愛爾蘭)	eu-west-1	是
歐洲 (倫敦)	eu-west-2	是
歐洲 (巴黎)	eu-west-3	是
南美洲 (聖保羅)	sa-east-1	是
AWS GovCloud (美國西部)	us-gov-west-1	是
AWS GovCloud (美國東部) 東	us-gov-east-1	是

疑難排解 AWS Certificate Manager 身分和存取

使用下列資訊可協助您診斷及修正使用和時可能會遇到的ACM常見問題IAM。

主題

- 我沒有執行操作的授權 ACM
- 我沒有授權要求證書 ACM
- 我沒有授權執行 iam：PassRole
- 我想允許我以外的人訪 AWS 帳戶 問我的ACM資源

我沒有執行操作的授權 ACM

如果您收到錯誤，告知您未獲授權執行動作，您的政策必須更新，允許您執行動作。

當使用mateojacksonIAM者嘗試使用主控台來檢視虛構*my-example-widget*資源的詳細資料，但沒有虛構的acm:*GetWidget*權限時，就會發生下列範例錯誤。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 acm:GetWidget on resource: my-example-widget
```

在此情況下，必須更新 mateojackson 使用者的政策，允許使用 acm:*GetWidget* 動作存取 *my-example-widget* 資源。

如果您需要協助，請聯絡您的 AWS 系統管理員。您的管理員提供您的簽署憑證。

我沒有授權要求證書 ACM

如果您收到此錯誤，表示您ACM或PKI管理員已設定規則，防止您以目前的狀態要求憑證。

當使用IAM者嘗試使用主控台使用組織管理員設定的選項來要求憑證時，就會發生下列範例錯誤。DENY

```
User: arn:aws:sts::account::ID: is not authorized to perform: acm:RequestCertificate
on resource: arn:aws:acm:region:account:certificate/*
with an explicit deny in a service control policy
```

在這種情況下，使用者應透過遵守管理員所設政策的方式重新提出請求。或者也能請管理員更新政策，允許使用者請求取得憑證。

我沒有授權執行 iam：PassRole

如果您收到未獲授權執行iam:PassRole動作的錯誤訊息，則必須更新您的原則以允許您將角色傳遞給ACM。

有些 AWS 服務 允許您將現有角色傳遞給該服務，而不是建立新的服務角色或服務連結角色。如需執行此作業，您必須擁有將角色傳遞至該服務的許可。

當名為的使用IAM者marymajor嘗試使用主控台執行中的動作時，就會發生下列範例錯誤ACM。但是，動作請求服務具備服務角色授予的許可。Mary 沒有將角色傳遞至該服務的許可。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

在這種情況下，Mary 的政策必須更新，允許她執行 iam:PassRole 動作。

如果您需要協助，請聯絡您的 AWS 系統管理員。您的管理員提供您的簽署憑證。

我想允許我以外的人訪 AWS 帳戶 問我的ACM資源

您可以建立一個角色，讓其他帳戶中的使用者或您組織外部的人員存取您的資源。您可以指定要允許哪些信任物件取得該角色。對於支援以資源為基礎的政策或存取控制清單 (ACLs) 的服務，您可以使用這些政策授與人員存取您的資源。

如需進一步了解，請參閱以下內容：

- 若要瞭解是否ACM支援這些功能，請參閱如何 AWS Certificate Manager 使用 IAM。
- 若要瞭解如何提供您所擁有資源 AWS 帳戶 的存取權，請參閱《IAM使用者指南》中 AWS 帳戶 的〈提供存取權給其他IAM使用者〉。
- 若要瞭解如何將資源存取權提供給第三方 AWS 帳戶，請參閱《IAM使用指南》中的提供第三方 AWS 帳戶 擁有的存取權。
- 若要瞭解如何透過身分聯盟提供存取權，請參閱使用指南中的提供對外部驗證使用IAM者的存取權 (身分聯合)。
- 若要瞭解針對跨帳號存取使用角色與以資源為基礎的政策之間的差異，請參閱《使用IAM者指南》IAM中的〈跨帳號資源存取〉。

韌性 AWS Certificate Manager

AWS 全球基礎架構是圍繞區 AWS 域和可用區域建立的。 AWS 區域提供多個實體分離和隔離的可用區域，這些區域透過低延遲、高輸送量和高度備援的網路連線。透過可用區域，您可以設計與操作的應用程式和資料庫，在可用區域之間自動容錯移轉而不會發生中斷。可用區域的可用性、容錯能力和擴展能力，均較單一或多個資料中心的傳統基礎設施還高。

如需區域和可用區域的相關 AWS 資訊，請參閱AWS 全域基礎結構。

AWS Certificate Manager中的基礎設施安全

作為託管服務， AWS Certificate Manager 受到 AWS 全球網絡安全的保護。有關 AWS 安全服務以及如何 AWS 保護基礎架構的詳細資訊，請參閱AWS 雲端安全 若要使用基礎架構安全性的最佳做法來設計您的 AWS 環境，請參閱安全性支柱架構良 AWS 好的架構中的基礎結構保護。

您可以使用 AWS 已發佈的 API 呼叫透過網路存取 ACM。使用者端必須支援下列專案：

- Transport Layer Security (TLS)。我們需要 TLS 1.2 並建議使用 TLS 1.3。

- 具備完美轉送私密(PFS)的密碼套件，例如 DHE (Ephemeral Diffie-Hellman)或 ECDHE (Elliptic Curve Ephemeral Diffie-Hellman)。現代系統(如 Java 7 和更新版本)大多會支援這些模式。

此外，請求必須使用存取金鑰 ID 和與 IAM 主體相關聯的私密存取金鑰來簽署。或者，您可以透過 AWS Security Token Service (AWS STS) 來產生暫時安全憑證來簽署請求。

授予對 ACM 的程式存取

如果使用者想要與 AWS 之外的 AWS Management Console. 授與程式設計存 AWS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。	請依照您要使用的介面所提供的指示操作。 • 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。
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。 • 如需相關資訊 AWS CLI，請參閱使用指南中的使用

哪個使用者需要程式設計存取權？	到	By
		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。

最佳實務

最佳做法是可以幫助您更有效地使用 AWS Certificate Manager (AWS Certificate Manager) 的建議。以下最佳實務是根據目前 ACM 客戶的實際體驗。

主題

- 帳戶層級分隔
- AWS CloudFormation
- 憑證關聯
- 網域驗證
- 新增或刪除網域名稱
- 取消使用憑證透明度記錄功能
- 開啟 AWS CloudTrail

帳戶層級分隔

在您的政策中使用帳戶層級分隔來控制誰可以在帳戶層級存取憑證。將您的生產憑證保存在與測試和開發憑證不同的帳戶中。如果您無法使用帳戶層級分隔，可以透過拒絕策略中的`kms:CreateGrant`動作來限制特定角色的存取權。這會限制帳戶中哪些角色可以高層級簽署憑證。如需有關贈款的資訊，包括授權術語，請參閱AWS Key Management Service 開發人員指南 AWS KMS中的授權。

如果您想要更精細的控制，而不是限制kms:CreateGrant依帳戶的使用，您可以使用 kms:EncryptionContext 條件金鑰限kms:CreateGrant制特定憑證。指定arn:aws:acm為要限制的關鍵字和 ARN 的值。下列範例原則會阻止使用特定憑證，但允許其他憑證。

```
{
   "Version": "2012-10-17",
   "Statement": [
       {
           "Sid": "VisualEditor0",
           "Effect": "Deny",
           "Action": "kms:CreateGrant",
           "Resource": "*",
           "Condition": {
               "StringEquals": {
                   "kms:EncryptionContext:aws:acm:arn": "arn:aws:acm:us-
east-1:111122223333:certificate/b26def74-1234-4321-9876-951d4c07b197"
               }
           }
       }
   ]
}
```

AWS CloudFormation

AWS CloudFormation 您可以使用建立描述要使用之 AWS 資源的範本。 AWS CloudFormation 然後為您佈建和配置這些資源。 AWS CloudFormation 可以佈建 ACM 支援的資源，例如 Elastic Load Balancing CloudFront、亞馬遜和 Amazon API 閘道。如需詳細資訊，請參閱 服務整合 AWS Certificate Manager。

如果您使 AWS CloudFormation 用快速建立和刪除多個測試環境，建議您不要為每個環境建立個別的 ACM 憑證。這樣做會快速用盡您的憑證配額。如需詳細資訊，請參閱 配額。反之，建立一個涵蓋所有用於測試之網域名稱的萬用字元憑證。例如，如果您要重複為只有版本編號不同的網域名稱建立 ACM 憑證，像是 *<version>*.service.example.com，則請改為 *<*>*.service.example.com 建立單一萬用字元憑證。在用來建立測試環境的範本中包含萬 AWS CloudFormation 用字元憑證。

憑證關聯

憑證關聯 (有時稱為 SSL 關聯) 是一個程序，可讓您在應用程式中直接與 X.509 憑證或公開金鑰關聯遠端主機來驗證該主機，而不是使用憑證階層。因此，應用程式會使用關聯來繞過 SSL/TLS 憑證鏈驗證。典型的 SSL 驗證程序會檢查整個憑證鏈的簽章，從根憑證授權機構 (CA) 憑證到次級 CA 憑證 (如

果有)。還會在階層底部檢查遠端主機的憑證。反之，您的應用程式可以關聯至遠端主機的憑證，以表示只有該憑證受信任，而不信任根憑證或任何其他憑證鏈中的憑證。您可以在開發期間將遠端主機的憑證或公開金鑰新增至應用程式。應用程式也可以在第一次連線到主機時新增憑證或金鑰。

> **Warning**
>
> 我們建議應用程式不要關聯 ACM 憑證。ACM 會執行 ACM憑證的受管理續約 以在憑證過期前自動續約 Amazon 發行的 SSL/TLS 憑證。為了續約憑證，ACM 會產生新的公私有金鑰對。如果您的應用程式關聯 ACM 憑證，而且成功使用新的公有金鑰續約憑證，則應用程式可能會無法連線到網域。

如果您決定關聯憑證，以下選項不會阻礙應用程式連線到您的網域：

- 將自己的憑證匯入 ACM，然後將應用程式關聯至匯入的憑證。ACM 不會嘗試自動續約匯入的憑證。
- 如果您使用的是公有憑證，請將應用程式釘選到所有可用的 Amazon 根憑證。如果您使用的是私有憑證，請將您的應用程式釘選到 CA 根憑證。

網域驗證

Amazon 憑證授權單位 (CA) 才能為您的網站發行憑證， AWS Certificate Manager (ACM) 必須先確認您擁有或控制您在請求中指定的所有網域。您可以使用電子郵件或 DNS 執行驗證。如需更多詳細資訊，請參閱「DNS驗證」及「電子郵件驗證」。

新增或刪除網域名稱

您無法從現有 ACM 憑證新增或移除網域名稱。反之，您必須使用修訂的網域名稱清單申請新憑證。例如，如果您的憑證有五個網域名稱，而且需要新增四個網域名稱，則必須使用九個網域名稱申請新憑證。如同使用任何新憑證，您必須驗證申請中所有網域名稱的所有權，包括先前為原始憑證驗證的名稱。

如果您使用電子郵件驗證，便會針對每個網域收到多達 8 封驗證電子郵件，至少其中 1 封必須在 72 個小時內執行。例如，使用五個網域名稱申請憑證時，您會收到多達 40 個驗證訊息，至少其中 5 封必須在 72 個小時內執行。隨著憑證申請的網域名稱數量增加，使用電子郵件驗證網域所有權的必要工作也因此增加。

如果使用 DNS 驗證，則必須為您要驗證的 FQDN 寫入一個新的 DNS 記錄到資料庫。ACM 會將要建立的記錄傳送給您，然後查詢資料庫以判斷是否已新增記錄。新增記錄會宣告您擁有或控制網域。在

上述範例中，如果使用五個網域名稱申請憑證，則必須建立五個 DNS 記錄。我們建議您盡可能使用 DNS 驗證。

取消使用憑證透明度記錄功能

> **Important**
>
> 無論您採取什麼動作來取消憑證透明度記錄，任何可存取繫結憑證的公有或私有端點的用戶端或個人仍可能會記錄您的憑證。不過，憑證不會包含已簽署的憑證時間戳記 (SCT)。只有發行的 CA 可將 SCT 嵌入至憑證。

從 2018 年 4 月 30 日開始，Google Chrome 不再信任未記錄在憑證透明度日誌的公有 SSL/TLS 憑證。因此，從 2018 年 4 月 24 日開始，Amazon CA 開始將所有新憑證和續約發行到至少兩個公有日誌。憑證記錄後便無法移除。如需詳細資訊，請參閱 憑證透明度記錄。

記錄會在您申請憑證或續約憑證時自動執行，但您可以選擇不自動執行。這樣做的常見原因包括安全性和隱私權方面的考量。例如，記錄內部主機網域名稱會提供潛在攻擊者平常不公開的內部網路相關資訊。此外，記錄可能洩漏新的或未發佈的產品和網站的名稱。

若要在要求憑證時選擇退出透明度記錄，請使用要求憑證 AWS CLI 命令或 RequestCertificateAPI 作業的`options`參數。如果您的憑證是在 2018 年 4 月 24 日之前簽發的，而且您想要確定在續訂期間未記錄憑證，您可以使用update-certificate-options命令或 UpdateCertificateOptionsAPI 作業來選擇退出。

限制

- 您無法使用主控台來啟用或停用透明度記錄。
- 憑證進入續約期之後就無法變更記錄狀態，通常是憑證過期前 60 天。如果狀態變更失敗，並不會產生錯誤訊息。

憑證記錄後便無法從日誌移除。在該時間點取消將不會生效。如果您在申請憑證時取消記錄，然後選擇之後記錄，則在憑證續約前，不會記錄憑證。如果您要立即記錄憑證，我們建議您發行新憑證。

以下範例示範如何使用 request-certificate 命令在申請新憑證時停用憑證透明度。

```
aws acm request-certificate \
--domain-name www.example.com \
--validation-method DNS \
```

```
--options CertificateTransparencyLoggingPreference=DISABLED \
```

上述命令會輸出新憑證的 ARN。

```
{
    "CertificateArn": "arn:aws:acm:region:account:certificate/certificate_ID"
}
```

如果您已經擁有憑證，並且不希望在更新時將其記錄下來，請使用update-certificate-options指令。此命令不會傳回數值。

```
aws acm update-certificate-options \
--certificate-arn arn:aws:acm:region:account:\
certificate/certificate_ID \
--options CertificateTransparencyLoggingPreference=DISABLED
```

開啟 AWS CloudTrail

開始使用 ACM 之前，請先開啟 CloudTrail 記錄功能。 CloudTrail 透過擷取帳戶 AWS API 呼叫的歷史記錄，包括透過 AWS 管理主控台、 AWS SDK、和更高層級的 Amazon Web Services 進行的 API 呼叫 AWS Command Line Interface，可讓您監控 AWS 部署。您也可以找出哪些使用者和帳戶呼叫過 ACM API、發出呼叫的來源 IP 地址，以及呼叫的發生時間。您可以使用 API 整合 CloudTrail 到應用程式中、為您的組織自動建立追蹤、檢查追蹤的狀態，以及控制系統管理員開啟和關閉 CloudTrail 登入的方式。如需詳細資訊，請參閱建立追蹤記錄。前往 CloudTrail 搭配使用 AWS Certificate Manager 查看 ACM 動作的追蹤記錄範例。

設定

使用 AWS Certificate Manager (ACM) 您可以為 AWS 基礎的網站和應用程式佈建和管理SSL/TLS憑證。您可ACM以用來建立或匯入憑證，然後管理憑證。您必須使用其他 AWS 服務將憑證部署到您的網站或應用程式。如需與整合之服務的詳細資訊ACM，請參閱服務整合 AWS Certificate Manager。以下各節將討論使用前需要執行的步驟ACM。

主題

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. 打開https://portal.aws.amazon.com/billing/註冊。
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。作為安全最佳實務，請將管理存取權指派給使用者，並且僅使用根使用者來執行需要根使用者存取權的任務。

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。

建立具有管理存取權的使用者

註冊後，請保護您的 AWS 帳戶 AWS 帳戶根使用者 AWS IAM Identity Center、啟用和建立系統管理使用者，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. 為您的 root 使用者開啟多因素驗證 (MFA)。

   如需指示，請參閱《使用指南》中的「IAM為 AWS 帳戶 root 使用者啟用虛擬MFA裝置 (主控台)」。

建立具有管理存取權的使用者

1. 啟用IAM身分識別中心。

   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身分識別中心中，將管理存取權授與使用者。

   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」。

以具有管理存取權的使用者身分登入

- 若要使用您的 IAM Identity Center 使用者登入URL，請使用建立IAM身分識別中心使用者時傳送至您電子郵件地址的登入資訊。

  如需使用IAM身分識別中心使用者登入的說明，請參閱使用指南中的登入 AWS 存取入口網站。AWS 登入

指派存取權給其他使用者

1. 在 IAM Identity Center 中，建立遵循套用最低權限權限的最佳作法的權限集。

   如需指示，請參閱《AWS IAM Identity Center 使用者指南》中的建立許可集。

2. 將使用者指派至群組，然後對該群組指派單一登入存取權。

   如需指示，請參閱《AWS IAM Identity Center 使用者指南》中的新增群組。

註冊網域名稱

完整網域名稱 (FQDN) 是網際網路上組織或個人的唯一名稱，後面接著頂層網域後綴，例如 .com 或 .org。如果您沒有已註冊的網域名稱，可以透過 Amazon Route 53 或眾多其他商業註冊商註冊。通常您會在註冊商的網站申請網域名稱。註冊商查詢WHOIS以確定請求FQDN是否可用。如果可用，註冊商通常會列出具有不同網域域名的相關名稱，讓您取得任何可用的名稱。註冊通常會持續一段期間 (例如一或兩年) 才需要續約。

如需透過 Amazon Route 53 註冊網域名稱的詳細資訊，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。

(選用) 為您的網域設定電子郵件

> **Note**
>
> 只有當您使用電子郵件驗證聲明您擁有或控制憑證要求中指定的 FQDN (完整網域名稱) 時，才需要執行下列步驟。ACM要求您在發行憑證之前先驗證擁有權或控制權。您可以使用電子郵件驗證或DNS驗證。
>
> 如果您可以編輯DNS設定，建議您使用DNS網域驗證而非電子郵件驗證。DNS驗證不需要為網域名稱設定電子郵件。如需DNS驗證的詳細資訊，請參閱DNS驗證。

網域驗證

若要為您的網域設定電子郵件驗證，請使用主控台或DomainValidationOption在通話中進行設定RequestCertificate。ACM會將驗證電子郵件傳送至您要求的網域名稱。如果您希望在該域接收這些電子郵件，也可以指定一個超級域作為驗證域。任何最小網站位址的子網域皆為有效，之後@會用作電子郵件地址的網域做為尾碼。例如，如果您將 example.com 指定為子網域 .example.com 的驗證網域，您可以收到電子郵件至 admin@example.com。如果您有驗證電子郵件的相關問題，請參閱 針對電子郵件驗證問題進行疑難排解。

(選擇性) 設定CAA記錄

您可以選擇性地設定憑證授權單位授權 (CAA) DNS 記錄，以指定允許 AWS Certificate Manager (ACM) 為您的網域或子網域發行憑證。驗證您的網域之後，請ACM檢查記錄是否存在，以確定該CAA記錄可以為您核發憑證。如果您不想啟用CAA檢查功能，可以選擇不設定網域的CAA記錄。

記CAA錄包含下列資料欄位：

flags

指定標籤欄位的值是否受支援ACM。將此值設定為 0。

標籤

tag 欄位可以是以下其中一個值。請注意，iodef 欄位目前已被忽略。

issue

表示您在值欄位中指定的 ACM CA 已獲授權，可為您的網域或子網域發行憑證。

issuewild

表示您在值欄位中指定的 ACM CA 已獲授權，可為您的網域或子網域發行萬用字元憑證。萬用字元憑證適用於網域或子網域及其子網域。

值

此欄位的值取決於 tag 欄位的值。您必須用引號 ("") 括住此值。

當 tag 是 issue 時

value 欄位包含 CA 網域名稱。此欄位可以包含 Amazon CA 以外的 CA 的名稱。但是，如果您沒有指定以下四種 Amazon 之一的CAA記錄CAs，則ACM無法向您的域或子域發行證書：

- amazon.com
- amazontrust.com
- awstrust.com
- amazonaws.com

value 欄位也可以包含分號 (;)，表示不應允許任何 CA 為您的網域或子網域發行憑證。如果您在某個時間點決定不再需要為特定網域發行的憑證，請使用此欄位。

當 tag 是 issuewild 時

value 欄位與 tag 為 issue 時的相同，只是值適用於萬用字元憑證。

如果存在不包含 ACM CA 值的問題範圍CAA記錄，則無法由發行萬用字元。ACM如果沒有 issuewild 存在，但有發行CAA記錄ACM，則通配符可能由發行。ACM

Example CAA記錄範例

在下列範例中，您的網域名稱首先是記錄類型 (CAA)。flags 欄位一律為 0。tags 欄位可以是 issue 或 issuewild。如果欄位有問題，而您在值欄位中輸入 CA 伺服器的網域名稱，則CAA記錄會指出您指定

的伺服器可核發您要求的憑證。如果您在值欄位中輸入分號「;」，CAA記錄表示不允許任何 CA 發行憑證。CAA記錄的配置因DNS提供者而異。

```
Domain       Record type  Flags  Tag      Value
example.com.   CAA            0        issue      "SomeCA.com"
```

```
Domain       Record type  Flags  Tag      Value
example.com.   CAA            0        issue      "amazon.com"
```

```
Domain       Record type  Flags  Tag      Value
example.com.   CAA            0        issue      "amazontrust.com"
```

```
Domain       Record type  Flags  Tag      Value
example.com.   CAA            0        issue      "awstrust.com"
```

```
Domain       Record type  Flags  Tag      Value
example.com.   CAA            0        issue      "amazonaws.com"
```

```
Domain       Record type  Flags  Tag      Value
example.com    CAA            0        issue      ";"
```

如需有關如何新增或修改DNS記錄的詳細資訊，請洽詢您的DNS提供者。53 號路線支持CAA記錄。如果 Route 53 是您的DNS提供者，請參閱CAA格式以取得有關建立記錄的詳細資訊。

發行和管理憑證

ACM憑證可用於透過網際網路或內部網路建立安全通訊。您可以直接從 ACM (「憑證」) 要求公開信任的ACM憑證，或匯入由第三方發行的公開信任憑證。另外也支援自我簽署的憑證。若要佈建組織的內部PKI，您可以發行由建立和管理的私有ACM憑證授權單位 (CA) 簽署的憑證AWS 私有 CA。CA 可能位於您的帳戶中，或透過其他帳戶與您共用。

> **Note**
>
> 公有ACM憑證可以安裝在連接到硝基 Enclave 的 Amazon EC2 執行個體上，但無法安裝到其他 Amazon 執行個體。EC2如需在未連線至 Nitro Enclave 的 Amazon EC2 執行個體上設定獨立網頁伺服器的詳細資訊，請參閱教學課程：在 Amazon Linux 2 上安裝LAMP網路伺服器或教學：使用 Amazon Linux 安裝LAMP網路伺服器。AMI

> **Note**
>
> 由於私有 CA 簽署的憑證預設不受信任，因此系統管理員必須將它們安裝在用戶端信任存放區中。

若要開始發行憑證，請登入 AWS 管理主控台，然後在https://console.aws.amazon.com/acm/家中開啟ACM主控台。出現簡介頁面時，請選擇 Get Started (開始使用)。否則，請在左側導覽窗格CAs中選擇 [Cer tificate Manager] 或 [私人]。

主題

請求公有憑證

以下各節將討論如何使用ACM主控台或 AWS CLI 要求公用ACM憑證。請求公有憑證之後，您必須完成驗證網域所有權所述的任何一個程序。

公用ACM憑證遵循 X.509 標準，並受到下列限制：

- 名稱：您必須使用DNS符合規範的主旨名稱。如需詳細資訊，請參閱 網域名稱。
- 演算法：對於加密，憑證私密金鑰演算法必須是 2048 位元RSA、256 位元或 384 位ECDSA 元。ECDSA
- 有效期限：每個憑證的有效期限皆為 13 個月 (395 天)。
- 續約：ACM嘗試在 11 個月後自動續訂私有憑證。

如果您在要求憑證時遇到問題，請參閱憑證請求疑難排解。

若要為私人PKI使用要求憑證 AWS 私有 CA，請參閱要求私有PKI憑證。

> **Note**
>
> 管理員可以使用ACM條件式金鑰原則來控制終端使用者發行新憑證的方式。透過這些條件索引鍵，您可對與憑證請求相關的網域、驗證方法和其他屬性設下限制。

> **Note**
>
> 除非您選擇退出，否則公開信任的ACM憑證會自動記錄在至少兩個憑證透明度資料庫中。目前，您不能使用主控台來選擇退出。您必須使用 AWS CLI 或ACMAPI。如需詳細資訊，請參閱取消使用憑證透明度記錄功能。如需透明度日誌的一般資訊，請參閱憑證透明度記錄。

主題

- 使用主控台請求公有憑證
- 請求使用公用憑證 CLI

使用主控台請求公有憑證

要求ACM公用憑證 (主控台)

1. 登入 AWS 管理主控台，然後在https://console.aws.amazon.com/acm/家中開啟ACM主控台。

   選擇 Request a certificate (請求憑證)。

2. 在 Domain names (網域名稱) 部分，輸入您的網域名稱。

   您可以使用完整網域名稱 (FQDN)，例如**www.example.com**，或裸網域名稱或頂點網域名稱，例如**example.com**. 您也可以在最左方使用星號 (*) 做為萬用字元，以保護相同網域中的多個網站名稱。例如，***.example.com** 可保護 **corp.example.com** 和 **images.example.com**。萬用字元名稱將顯示在憑證的「主旨」欄位和「主旨替代名稱」副檔名中ACM。

   請求萬用字元憑證時，星號 (*) 必須在網域名稱的最左方，而且僅能保護一個子網域層級。例如，***.example.com** 可以保護 **login.example.com** 和 **test.example.com**，但不能保護 **test.login.example.com**。另請注意，***.example.com** 只可以保護 **example.com** 的子網域，無法保護 bare 或 apex 網域 (**example.com**)。若要保護兩者，請參閱下一個步驟。

   > **Note**
   >
   > 根據 RFC5280 規定，您在此步驟中輸入的網域名稱 (技術上是一般名稱) 的長度不得超過 64 個八位元組 (字元)，包括句點。您提供的每個後續主旨替代名稱 (SAN)，如下一個步驟所提供的長度最多可達 253 個八位元組。

   若要新增其他名稱，請選擇 Add another name to this certificate (將其他名稱新增至此憑證)，然後在文字方塊中輸入名稱。若要同時保護 bare 或 apex 網域 (例如 **example.com**) 及其子網域 (例如 ***.example.com**)，此功能非常實用。

3. 在 [驗證方法] 區段中，根據您的需求選擇 [DNS驗證]-[建議] 或 [電子郵件驗證]。

   > **Note**
   >
   > 如果您可以編輯DNS設定，建議您使用DNS網域驗證而非電子郵件驗證。DNS驗證比電子郵件驗證有多個好處。請參閱DNS驗證。

在ACM發行憑證之前，它會驗證您是否擁有或控制憑證要求中的網域名稱。您可以使用電子郵件驗證或DNS驗證。

如果您選擇電子郵件驗證，請ACM將驗證電子郵件傳送至您在網域名稱欄位中指定的網域。如果您指定驗證網域，請改為將電子郵件ACM傳送至該驗證網域。如需電子郵件驗證的詳細資訊，請參閱「電子郵件驗證」。

如果您使用DNS驗證，您只需將提供的CNAME記錄新增ACM至您的DNS組態即可。如需DNS驗證的詳細資訊，請參閱DNS驗證。

4. 在金鑰演算法區段中，選擇三種可用演算法之一：

   - RSA預設值
   - ECDSAP 256
   - ECDSA頁 384

   如需協助您選擇演算法的資訊，請參閱金鑰演算法和 AWS 部落格文章如何評估ECDSA和使用 AWS Certificate Manager.

5. 在 Tags（標籤）頁面上，您可以選擇標記您的憑證。標籤是索引鍵值配對，可做為識別和組織 AWS 資源的中繼資料。如需ACM標籤參數的清單，以及如何在建立後將標籤新增至憑證的指示，請參閱標記 AWS Certificate Manager 憑證。

   完成新增標籤後，請選擇 Request（請求）。

6. 處理要求之後，主控台會將您返回憑證清單，其中會顯示新憑證相關的資訊。

   憑證被請求後會進入待驗證狀態，除非憑證請求因為出現「憑證請求失敗」故障排除主題中列出的情況而失敗。ACM重複嘗試驗證憑證 72 小時，然後逾時。如果憑證顯示「失敗」或「驗證逾時」狀態，請刪除要求，使用DNS驗證或電子郵件驗證更正問題，然後再試一次。如果驗證成功，憑證會進入已發行狀態。

> **Note**
>
> 視您排序清單的方式而定，您要尋找的憑證可能無法立即顯示。您可以點選右邊的黑色三角形來變更順序。您也可以使用右上角的頁碼在多張憑證頁面之間瀏覽。

請求使用公用憑證 CLI

使用要求憑證命令在命令列上要求新的公用ACM憑證。驗證方法的選用值為DNS和EMAIL。金鑰演算法的選用值為 RSA _2048 (如果未明確提供參數，則為預設值)、EC_PRIME256v1 和 EC_SECP384R1。

```
aws acm request-certificate \
--domain-name www.example.com \
--key-algorithm EC_Prime256v1 \
--validation-method DNS \
--idempotency-token 1234 \
--options CertificateTransparencyLoggingPreference=DISABLED
```

此命令會輸出新公用憑證的 Amazon 資源名稱 (ARN)。

```
{
    "CertificateArn": "arn:aws:acm:Region:444455556666:certificate/certificate_ID"
}
```

要求私有PKI憑證

如果您可以存取由建立的現有私有 CA AWS 私有 CA，ACM可以要求適合在您的私人中使用的憑證 PKI。CA 可能位於您的帳戶中，或透過其他帳戶與您共用。如需建立私有 CA 的相關資訊，請參閱建立私有憑證授權機構。

根據預設，私有 CA 簽署的憑證不受信任，也ACM不支援任何形式的驗證。因此，管理員必須採取行動，將其安裝至組織的用戶端信任存放區。

私有ACM憑證遵循 X.509 標準，並受到下列限制：

- 名稱：您必須使用DNS符合規範的主旨名稱。如需詳細資訊，請參閱 網域名稱。
- 演算法：對於加密，憑證私密金鑰演算法必須是 2048 位元RSA、256 位元或 384 位ECDSA 元。ECDSA

  > **Note**
  >
  > 指定的簽章演算法系列 (RSA或ECDSA) 必須符合 CA 秘密金鑰的演算法系列。

- 有效期限：每個憑證的有效期限皆為 13 個月 (395 天)。簽署 CA 憑證的結束日期必須超過所請求憑證的結束日期，否則憑證請求將會失敗。

- 續約：ACM嘗試在 11 個月後自動續訂私有憑證。

用來簽署終端實體憑證的私有 CA 必須遵守其本身的限制：

- CA 必須處於作用中狀態。
- CA 私密金鑰演算法必須是 RSA 2048 或 RSA 4096。

> **Note**
>
> 與公開信任的憑證不同，私有 CA 簽署的憑證不需要驗證。

主題

設定私有 CA 的存取權

您可以使用 AWS 私有 CA 以下兩種情況之一來簽署ACM憑證：

- 單一帳戶：簽署的 CA 和核發的ACM憑證位於相同的 AWS 帳戶中。

  若要啟用單一帳戶發行和續約， AWS 私有 CA 系統管理員必須授與ACM服務主體的權限，才能建立、擷取及列出憑證。這是使用 AWS 私有 CA API動作CreatePermission或 AWS CLI 命令創建權限來完成的。帳戶擁有者會將這些權限指派給負責核發憑證的IAM使用者、群組或角色。

- 跨帳戶：簽署的 CA 和發行的ACM憑證位於不同的 AWS 帳戶中，而且 CA 的存取權已授與憑證所在的帳戶。

  若要啟用跨帳戶核發和續約， AWS 私有 CA 系統管理員必須使用 AWS 私有 CA API動作PutPolicy或命令放入原則，將以資源為基礎的政策附加至 CA。 AWS CLI政策會指定其他帳戶中允許有限存取 CA 的委託人。如需詳細資訊，請參閱搭配ACM私有 CA 使用以資源為基礎的原則。

  跨帳戶案例也需ACM要設定服務連結角色 (SLR)，以作為主體與策略互動。PCAACM在發行第一個憑證時SLR自動建立。

ACM可能會提醒您，它無法確定您的帳戶中是否SLR存在。如果您的帳戶已授與所需的`iam:GetRole`ACMSLR權限，則在建立之後不會再次發生警示。SLR如果重複發生，則您或您的帳戶管理員可能需要授與`iam:GetRole`權限ACM，或將您的帳戶與 ACM-managed 策略建立關聯。`AWSCertificateManagerFullAccess`

如需詳細資訊，請參閱搭配使用服務連結角色ACM。

> **Important**
>
> 您的ACM憑證必須與支援的 AWS 服務主動關聯，才能自動續訂憑證。如需ACM支援資源的相關資訊，請參閱服務整合 AWS Certificate Manager。

使用ACM主控台要求私人PKI憑證

1. 登入 AWS 管理主控台，然後在https://console.aws.amazon.com/acm/家中開啟ACM主控台。

   選擇 Request a certificate (請求憑證)。
2. 在 Request certificate（請求憑證）頁面上，選擇 Request a private certificate（請求私有憑證），然後選擇 Next（下一步）以繼續進行。
3. 在「憑證授權單位詳細資料」區段中，按一下「憑證授權單位」功能表，然後選擇一個可用CAs的如果 CA 是從其他帳戶共用，則會ARN以擁有權資訊開頭。

   系統隨即會顯示 CA 相關詳細資訊，協助您驗證是否已選擇正確者：

   - 擁有者
   - 類型
   - Common name (CN) (通用名稱 (CN))
   - Organization (O) (組織 (O))
   - Organization unit (OU) (組織單位 (OU))
   - Country name (C) (國家/地區名稱 (C))
   - State or province (州或省)
   - Locality name (地區名稱)
4. 在 Domain names（網域名稱）部分，輸入您的網域名稱。您可以使用完整網域名稱 (FQDN)，例如**`www.example.com`**，或裸網域名稱或頂點網域名稱，例如**`example.com`**. 您也可以在最左方

使用星號 (*) 做為萬用字元，以保護相同網域中的多個網站名稱。例如，***.example.com** 可保護 **corp.example.com** 和 **images.example.com**。萬用字元名稱將顯示在憑證的「主旨」欄位和「主旨替代名稱」副檔名中ACM。

> **Note**
>
> 請求萬用字元憑證時，星號 (*) 必須在網域名稱的最左方，而且僅能保護一個子網域層級。例如，***.example.com** 可以保護 **login.example.com** 和 **test.example.com**，但不能保護 **test.login.example.com**。另請注意，***.example.com** 只可以保護 **example.com** 的子網域，無法保護 bare 或 apex 網域 (**example.com**)。若要保護兩者，請參閱下一個步驟

或者，請選擇 Add another name to this certificate (將其他名稱新增至此憑證)，然後在文字方塊中輸入名稱。若要同時驗證 bare 或 apex 網域 (例如 **example.com**) 及其子網域 (例如 ***.example.com**)，此功能非常實用。

5. 在金鑰演算法區段中，選擇三種可用演算法之一：

   - RSA預設值
   - ECDSAP 256
   - ECDSA頁 384

   如需協助您選擇演算法的資訊，請參閱 金鑰演算法。

6. 在 Tags（標籤）部分，您可以選擇標記您的憑證。標籤是索引鍵值配對，可做為識別和組織 AWS 資源的中繼資料。如需ACM標籤參數的清單，以及如何在建立後將標籤新增至憑證的指示，請參閱標記 AWS Certificate Manager 憑證。

7. 在 Certificate renewal permissions（憑證續約權限）部分中，確認有關憑證更新權限的通知。這些權限允許自動更新您使用所選 CA 簽署的私有PKI憑證。如需詳細資訊，請參閱搭配使用服務連結角色ACM。

8. 在提供所有必要資訊後，選擇 Request（請求）。主控台會傳回憑證清單給您，讓您檢視新的憑證。

> **Note**
>
> 視您排序清單的方式而定，您要尋找的憑證可能無法立即顯示。您可以點選右邊的黑色三角形來變更順序。您也可以使用右上角的頁碼在多張憑證頁面之間瀏覽。

請求私人PKI憑證 CLI

使用要求憑證命令要求中的私人憑證。ACM

> **Note**
>
> 當您要求 CA 簽署的私有PKI憑證時 AWS Private CA，指定的簽署演算法系列 (RSA或 ECDSA) 必須符合 CA 秘密金鑰的演算法系列。

```
aws acm request-certificate \
--domain-name www.example.com \
--idempotency-token 12563 \
--certificate-authority-arn arn:aws:acm-pca:Region:444455556666:\
certificate-authority/CA_ID
```

此命令會輸出新私有憑證的 Amazon 資源名稱 (ARN)。

```
{
    "CertificateArn": "arn:aws:acm:Region:444455556666:certificate/certificate_ID"
}
```

在大多數情況下，在您第一次使用共用 CA 時，ACM會自動將服務連結角色 (SLR) 附加至您的帳戶。SLR啟用自動更新您發行的最終實體憑證。要檢查SLR是否存在，可以IAM使用以下命令進行查詢：

```
aws iam get-role --role-name AWSServiceRoleForCertificateManager
```

如果存SLR在，命令輸出應如下所示：

```
{
```

```
   "Role":{
      "Path":"/aws-service-role/acm.amazonaws.com/",
      "RoleName":"AWSServiceRoleForCertificateManager",
      "RoleId":"AAAAAAA0000000BBBBBBB",
      "Arn":"arn:aws:iam::{account_no}:role/aws-service-role/acm.amazonaws.com/
AWSServiceRoleForCertificateManager",
      "CreateDate":"2020-08-01T23:10:41Z",
      "AssumeRolePolicyDocument":{
         "Version":"2012-10-17",
         "Statement":[
            {
               "Effect":"Allow",
               "Principal":{
                  "Service":"acm.amazonaws.com"
               },
               "Action":"sts:AssumeRole"
            }
         ]
      },
      "Description":"SLR for ACM Service for accessing cross-account Private CA",
      "MaxSessionDuration":3600,
      "RoleLastUsed":{
         "LastUsedDate":"2020-08-01T23:11:04Z",
         "Region":"ap-southeast-1"
      }
   }
}
```

如果遺失SLR，請參閱搭配使用服務連結角色ACM。

驗證網域所有權

Amazon 憑證授權單位 (CA) 才能為您的網站發行憑證， AWS Certificate Manager (ACM) 必須證明您擁有或控制您在請求中指定的所有網域名稱。您可以選擇在申請憑證時，透過網域名稱系統 (DNS) 驗證或電子郵件驗證來證明您的擁有權。

> **Note**
>
> 驗證僅適用於由發行的公開信任憑證ACM。ACM不會驗證匯入的憑證或私有 CA 簽署的憑證的網域擁有權。ACM無法驗證 Amazon VPC 私有託管區域或任何其他私有網域中的資源。如需詳細資訊，請參閱 針對憑證驗證進行疑難排解。

一般而言，我們建議您使用DNS驗證而不是電子郵件驗證，原因如下：

- 如果您使用 Amazon Route 53 管理公開DNS記錄，您可以ACM直接透過以下方式更新記錄。
- ACM只要憑DNS證仍在使用中且DNS記錄到位，就會自動續訂驗證的憑證。
- 若要續約，需要網域擁有者對經電子郵件驗證的憑證執行動作。ACM在到期前 45 天開始傳送續約通知。這些通知會傳送至網域的WHOIS信箱地址，以及最多五個一般系統管理員地址。通知中包含網域擁有者可點選的連結，方便進行續約。驗證所有列出的網域後，會以相同的方式ACM發行續約憑證ARN。

如果您缺乏編輯網域DNS資料庫的授權，則必須改用電子郵件驗證。

> **Note**
>
> 建立具有電子郵件驗證的憑證之後，您無法切換為使DNS用驗證憑證。若要使用DNS驗證，請刪除憑證，然後建立使用驗證的新憑DNS證。

主題

DNS驗證

網域名稱系統 (DNS) 是連線至網路之資源的目錄服務。您的DNS提供商維護一個包含定義域的記錄的數據庫。當您選擇DNS驗證時，會ACM提供一或多個必須新增至此資料庫的CNAME記錄。這些記錄包含唯一的鍵值組，可作為您控制網域的證明。

> **Note**
>
> 建立具有電子郵件驗證的憑證之後，您無法切換為使DNS用驗證憑證。若要使用DNS驗證，請刪除憑證，然後建立使用驗證的新憑DNS證。

例如，如果您要求使用其他名稱的`example.com`網域憑證，則會`www.example.com`為您ACM建立兩CNAME筆記錄。每個專為您的網域和帳戶建立的記錄皆包含名稱和值。該值是指向用於自動續約憑證的 AWS ACM網域的別名。這些CNAME記錄只能添加到您的DNS數據庫一次。ACM只要憑證正在使用中且您的CNAME記錄仍在原處，就會自動更新您的憑證。

> **Important**
>
> 如果您不使用 Amazon Route 53 管理公開DNS記錄，請聯絡您的DNS供應商以瞭解如何新增記錄。如果您沒有編輯域名DNS數據庫的權限，則必須改用電子郵件驗證。

不需要重複驗證，只要CNAME記錄保留，您就可以為您的完整網域名稱 (FQDN) 要求額外的ACM憑證。也就是說，您可以建立具有相同網域名稱的替代憑證，或建立涵蓋不同子網域的憑證。由於CNAME驗證令牌適用於任何 AWS 區域，因此您可以在多個區域中重新創建相同的證書。您也可以取代已刪除的憑證。

您可以從與其相關聯的 AWS 服務中移除憑證，或刪除CNAME記錄來停止自動續訂。如果 Route 53 不是您的DNS提供者，請聯絡您的供應商以瞭解如何刪除記錄。如果您的供應商是 Route 53，請參閱 Route 53 開發人員指南中的刪除資源記錄集。如需受管的憑證續約的詳細資訊，請參閱「ACM憑證的受管理續約」。

> **Note**
>
> CNAME如果您的DNS組態中有超過五個以上的連結CNAMEs在一起，解析度將會失敗。如果您需要更長的鏈結，建議使用電子郵件驗證。

如何CNAME記錄工ACM作

> **Note**
>
> 本節適用於不使用 Route 53 作為其DNS供應商的客戶。

如果您不使用 Route 53 作為您的提DNS供商，則需要手動將提供的CNAME記錄輸入ACM到您的提供商的數據庫中，通常是通過網站。CNAME記錄用於多種目的，包括作為重定向機制和供應商特定元數據的容器。對於ACM，這些記錄允許初始網域擁有權驗證和持續的自動憑證續約。

下表顯示六個網域名稱的範例CNAME記錄。每筆記錄的記錄名稱-記錄值配對用於驗證網域名稱所有權。

請注意，在表格中，前兩個記錄名稱-記錄值配對是相同的。這說明了對於萬用字元網域 (例如`*.example.com`，由ACM建立的字串) 與為其基礎網域建立的字串相同。`example.com`若非如此，每個網域名稱的配對記錄名稱和記錄值會有所不同。

範例CNAME記錄

網域名稱	記錄名稱	記錄值	註解
*.example.com	_*x1*. 例子.	_*x2*. ACM-驗證.	Identical (相同)
example。com	_*x1*. 例子.	_*x2*. ACM-驗證.	
www.example.com	_*x3*. 例子.	_*x4*. ACM-驗證.	唯一
host.example.com	_*x5*. 主機. 例如.	_*x6*. ACM-驗證.	唯一
subdomain.example.com	_*x7*. 子域名. 例子.	_*x8*. ACM-驗證.	唯一
host.subdomain.example.com	_*x9*. 主機. 子域名.	_*x10*. ACM-驗證.	唯一

所以此 *xN* 下劃線 (_) 後面的值是由生成的長字符串ACM。例如：

```
_3639ac514e785e898d2646601fa951d5.example.com.
```

代表產生的記錄名稱。相關聯的記錄值可能是

```
_98d2646601fa951d53639ac514e785e8.acm-validation.aws.
```

對於相同的DNS記錄。

> **Note**
>
> 如果您的DNS提供者不支援帶有前導底線的CNAME值，請參閱疑難排解DNS驗證問題。

當您要求憑證並指定DNS驗證時，會以下列格式ACM提供CNAME資訊：

網域名稱	記錄名稱	記錄類型	記錄值
example.com	_a79865eb4cd1a6ab990a45779b4e0b96.example.com。	CNAME	_424c7224e9b0146f9a8808af955727d0.acm-validations.aws。

網域名稱是與憑證FQDN相關聯的。記錄名稱為鍵值組的索引鍵，能唯一識別記錄。記錄值為鍵值組的值。

這三個值（網域名稱、記錄名稱和記錄值）都必須輸入到DNS供應商 Web 介面的適當欄位中，才能新增DNS記錄。供應商處理記錄名稱 (或單純「名稱」) 欄位的做法不一致。在某些情況下，您應該提供整個字符串，如上所示。其他供應商會自動將網域名稱附加到您輸入的任何字串中，這表示 (在本例中) 您應該只輸入

```
_a79865eb4cd1a6ab990a45779b4e0b96
```

到名稱欄位中。如果您猜錯了，並輸入包含網域名稱 (例如 *.example.com*) 的記錄名稱，最終可能會產生以下結果：

```
_a79865eb4cd1a6ab990a45779b4e0b96.example.com.example.com.
```

在這種情況下，驗證將會失敗。因此，您應該試著事先確定您的供應商所期望的輸入類型。

設定DNS驗證

本節說明如何設定公用憑證以使用DNS驗證。

在主控台中設定DNS驗證

> **Note**
>
> 此程序假設您至少已建立一個憑證，而且您正在建立憑證的 AWS 區域中工作。如果您嘗試開啟主機，而是看到第一次使用的畫面，或是成功開啟主機，但在清單中看不到您的憑證，請確認您已指定正確的區域。

1. 開啟位於 ACM 的 https://console.aws.amazon.com/acm/ 主控台。

2. 在憑證清單中，選擇具有您想要設定的 Pending validation（待定驗證）狀態的憑證之 Certificate ID（憑證 ID）。此動作會開啟憑證的詳細資料頁面。
3. 在 Domains（網域）部分中，完成下列兩個程序之一：

   a. (選用) 使用 Route 53 進行驗證。

      Create records in Route 53（在 Route 53 中建立記錄）按鈕會在符合以下情況時顯示：

      - 您使用 Route 53 作為您的DNS提供者。
      - 您擁有寫入 Route 53 託管區域的許可。
      - 您FQDN尚未通過驗證。

      > **Note**
      >
      > 如果您實際上正在使用 Route 53，但在 Route 53 中建立記錄按鈕遺失或停用，請參閱 ACM控制台不顯示「在 Route 53 中創建記錄」按鈕。

      選擇 Create records in Route 53（在 Route 53 中建立記錄）按鈕，然後選擇 Create records（建立記錄）。[憑證狀態] 頁面應該會開啟，並顯示狀態標題報告已成功建立DNS記錄。

      您的新憑證可能會繼續顯示 Pending validation（待定驗證）狀態最多 30 分鐘。

      > **Tip**
      >
      > 您無法以編程方式請求在 Route 53 中ACM自動創建記錄。但是，您可以API撥打 AWS CLI 或呼叫 Route 53，以在 Route 53 DNS 資料庫中建立記錄。如需有關 Route 53 記錄集的詳細資訊，請參閱使用 Route 53使用資源記錄集。

   b. （可選）如果您不使用 Route 53 作為您的DNS提供者，則必須檢索CNAME信息並將其添加到DNS數據庫中。在新憑證的詳細資訊頁面上，您可透過以下兩種方式執行此操作：

      - 複製「網域」區段中顯示的CNAME元件。此資訊必須手動新增至您的資DNS料庫。
      - 或者，選擇「匯出至」CSV。結果檔案中的資訊需要手動新增至資DNS料庫。

> **Important**
>
> 若要避免驗證問題，請如何CNAME記錄工ACM作在將資訊新增至DNS提供者的資料庫之前先檢閱。如果您遇到問題，請參閱「排解DNS驗證問題」。

如果ACM無法在產生CNAME值之後的 72 小時內驗證網域名稱，請將憑證狀態ACM變更為驗證逾時。產生此結果的最可能原因是您未使用ACM產生的值成功更新DNS組態。若要解決此問題，您必須在檢閱CNAME指示後要求新憑證。

電子郵件驗證

Amazon 憑證授權單位 (CA) 才能為您的網站發行憑證， AWS Certificate Manager (ACM) 必須先確認您擁有或控制您在請求中指定的所有網域。您可以使用電子郵件或執行驗證DNS。此主題討論電子郵件驗證。

如果您在使用電子郵件驗證時遇到問題，請參閱針對電子郵件驗證問題進行疑難排解。

電子郵件驗證的作用

ACM會針對每個網域，傳送驗證電子郵件至下列五個常見的系統電子郵件。或者，如果您希望在該域接收這些電子郵件，則可以指定一個超級域作為驗證域。任何最小網站位址的子網域皆為有效，之後@會用作電子郵件地址的網域做為尾碼。例如，如果您將 example.com 指定為子網域 .example.com 的驗證網域，您可以收到電子郵件至 admin@example.com。

- administrator@your_domain_name
- hostmaster@your_domain_name
- postmaster@your_domain_name
- webmaster@your_domain_name
- admin@your_domain_name

若要證明您擁有網域，您必須選取這些電子郵件中包含的驗證連結。ACM還會傳送驗證電子郵件到這些相同的地址，以便在憑證到期後 45 天續訂憑證。

使用 ACM API or 的多網域憑證要求的電子郵件驗證會CLI導致每個要求的網域傳送電子郵件訊息，即使要求中包含其他網域的子網域也一樣。網域擁有者必須先驗證每個網域的電子郵件訊息，才ACM能核發憑證。

此程序的例外情況

如果您要求以**www**或萬用字元星號 (*) 開頭的網域名稱的ACM憑證，請ACM移除前導**www**或星號，並將電子郵件傳送至管理位址。這些地址是由預先掛起的管理員 @、管理員 @、主持人 @、郵政管理員 @ 和網站管理員 @ 到域名的剩餘部分組成。例如，如果您要求申請 www.example.com 的ACM憑證，電子郵件會傳送到 admin@example.com，而不是傳送給 admin@www.example.com。同樣地，如果您要求 *.test.example.com 的ACM憑證，電子郵件也會傳送至 admin@test.example.com。其餘的常用管理地址也是以類似方式形成。

> **Important**
>
> 自 2024 年 6 月起，ACM不再支援透過WHOIS聯絡人地址進行新的電子郵件驗證。從 2024 年 10 月開始，現有憑證不ACM會傳送續約通知到網域的WHOIS聯絡地址。ACM將繼續發送驗證電子郵件到所請求域的五個常用系統地址。如需詳細資訊，請參閱AWS Certificate Manager將停止WHOIS查詢電子郵件驗證的憑證

考量事項

請遵循以下有關電子郵件驗證的注意事項。

- 您需要在您的網域中註冊一個有效的電子郵件地址，才能使用電子郵件驗證。設定電子郵件地址的程序不在本指南的說明範圍內。
- 驗證僅適用於由發行的公開信任憑證ACM。ACM不會驗證匯入的憑證或私有 CA 簽署的憑證的網域擁有權。ACM無法驗證 Amazon VPC 私有託管區域或任何其他私有網域中的資源。如需詳細資訊，請參閱 針對憑證驗證進行疑難排解。
- 建立具有電子郵件驗證的憑證之後，您無法切換為使DNS用驗證憑證。若要使用DNS驗證，請刪除憑證，然後建立使用驗證的新憑DNS證。

憑證過期日期和續約

ACM憑證的有效期為 13 個月 (395 天)。續訂憑證需要網域擁有者採取行動。ACM會在到期前 45 天開始傳送續約通知至與網域相關聯的電子郵件地址。通知包含網域擁有者可以按一下進行續約的連結。驗證所有列出的網域後，會以相同的方式ACM發行續約憑證ARN。

(選擇性) 重新傳送驗證郵件

每封驗證電子郵件皆包含符記，可用於核准憑證要求。不過，因為核准流程所需的驗證電子郵件可能會遭垃圾郵件篩選器封鎖，或在傳輸中遺失，因此符記會在 72 小時後自動過期。如果您沒有收到原始電子郵件或符記已過期，您可以要求重新傳送電子郵件。如需如何重新傳送驗證電子郵件的詳細資訊，請參閱 重新傳送驗證電子郵件

若電子郵件驗證持續發生問題，請參閱故障診斷中的針對電子郵件驗證問題進行疑難排解一節。

列出由管理的憑證 ACM

您可以使用ACM主控台或 AWS CLI 列出由管理的憑證ACM。控制台可以在一個頁面中列出多達 500 個證書，最多可列CLI出 1000 個證書。

使用主控台列出憑證

1. 開啟位於 ACM 的 https://console.aws.amazon.com/acm/ 主控台。
2. 檢閱憑證清單中的資訊。您可以使用右上角的頁碼在多張憑證頁面之間瀏覽。每個憑證都會占用一列，依預設針對每個憑證顯示下列欄：

- 網域名稱 — 憑證的完整網域名稱 (FQDN)。
- Type（類型） - 憑證類型。可能值為：Amazon issued（Amazon 已發行) | Private（私有) | Imported（已匯入)
- Status (狀態) - 憑證狀態。可能值為：Pending validation (待定驗證) | Issued (發行) | Inactive (非作用中) | Expired (已過期) | Revoked (已撤銷) | Failed (失敗) | Validation timed out (驗證逾時)
- 使用中？ — ACM 憑證是否與「Elastic Load Balancing」或之類的 AWS 服務主動關聯 CloudFront。此值可以是 No (否) 或 Yes (是)。
- 續約資格 — 憑證是否可以在接近到期ACM時自動續訂。可能值為：Eligible (符合資格) | Ineligible (不符合資格)。如需資格規則，請參閱 ACM憑證的受管理續約。

透過選擇主控台右上角的設定圖示，您可以自訂頁面上顯示的憑證數目、指定儲存格內容的換行行為，以及顯示其他資訊欄位。可用的選填欄位如下：

- Additional domain names (其他網域名稱) – 憑證中包含的一或多個網域名稱 (主體別名)。
- 要求時間 — ACM 要求憑證的時間。

- Issued at (發行時間) – 發行憑證的時間。此資訊僅適用於 Amazon 發行的憑證，不適用於匯入的憑證。
- Not before (生效時間) – 憑證生效的時間。
- Not after (失效時間) – 憑證失效的時間。
- Revoked at (撤銷時間) – 已撤銷憑證的撤銷時間。
- Name tag (名稱標籤) – 此憑證上 Name (名稱) 標籤的值 (如果有這個標籤的話)。
- Renewal status (續約狀態) – 所要求憑證續約的狀態。只有在要求續約後，此欄位才會顯示並具有值。可能的值為：Pending automatic renewal (等待自動續約) | Pending validation (等待驗證) | Success (成功續約) | Failure (未能續約)。

> **Note**
>
> 憑證狀態的變更可能需要數小時才會變成可用。若遇到問題，憑證要求會在 72 小時後逾時，並且必須從頭開始重複發行或續約程序。

Page size (頁面大小) 偏好設定會指定每個主控台頁面上傳回的憑證數量。

如需可用憑證詳細資訊的更多資訊，請參閱 說明ACM憑證。

若要使用 AWS CLI

使用 list 憑證命令列出您的ACM管理憑證，如下列範例所示：

```
$ aws acm list-certificates --max-items 10
```

此命令會傳回與以下內容相似的資訊：

```
{
    "CertificateSummaryList": [
        {
            "CertificateArn":
 "arn:aws:acm:Region:444455556666:certificate/certificate_ID",
            "DomainName": "example.com"
  "SubjectAlternativeNameSummaries": [
                "example.com",
                "other.example.com"
            ],
            "HasAdditionalSubjectAlternativeNames": false,
            "Status": "ISSUED",
```

```
            "Type": "IMPORTED",
            "KeyAlgorithm": "RSA-2048",
            "KeyUsages": [
                "DIGITAL_SIGNATURE",
                "KEY_ENCIPHERMENT"
            ],
            "ExtendedKeyUsages": [
                "NONE"
            ],
            "InUse": false,
            "RenewalEligibility": "INELIGIBLE",
            "NotBefore": "2022-06-14T23:42:49+00:00",
            "NotAfter": "2032-06-11T23:42:49+00:00",
            "CreatedAt": "2022-08-25T19:28:05.531000+00:00",
            "ImportedAt": "2022-08-25T19:28:05.544000+00:00"
        },...
    ]
}
```

依預設，只會傳回具有keyTypesRSA_1024或RSA_2048且具有至少一個指定網域的憑證。若要查看您控制的其他憑證 (例如無網域憑證或使用不同演算法或位元大小的憑證)，請提供下列範例所示的 `--includes` 參數。此參數可讓您指定篩選條件結構的成員。

```
$ aws acm list-certificates --max-items 10 --includes keyTypes=RSA_4096
```

說明ACM憑證

您可以使用主ACM控台或列出有關憑證的詳細中繼資料。 AWS CLI

在主控台中檢視憑證詳細資訊

1. 開啟主ACM控台，位https://console.aws.amazon.com/acm/於顯示您的憑證。您可以使用右上角的頁碼在多張憑證頁面之間瀏覽。
2. 若要顯示所列憑證的詳細中繼資料，請選擇 Certificate ID (憑證識別碼)。頁面會開啟，顯示下列資訊：

   - Certificate status (憑證狀態)
     - Identifier (識別符) - 憑證的 32 位元組十六進位唯一識別碼
     - ARN— 表單中的 Amazon 資源名稱 (ARN) `arn:aws:acm:Region:444455556666:certificate/certificate_ID`

  - 類型 — 識別ACM憑證的管理類別。可能值為：Amazon Issued (Amazon 已發行) | Private (私有) | Imported (已匯入)。如需詳細資訊，請參閱「請求公有憑證」、「要求私有PKI憑證」或「將憑證匯入 AWS Certificate Manager」。
  - Status（狀態) - 憑證狀態。可能值為：Pending validation (待定驗證) | Issued (發行) | Inactive (非作用中) | Expired (已過期) | Revoked (已撤銷) | Failed (失敗) | Validation timed out (驗證逾時)
  - Detailed status（詳細狀態) - 發行或匯入憑證的日期與時間
- 網域
  - 網域 — 憑證的完整網域名稱 (FQDN)。
  - Status（狀態) - 網域驗證狀態。可能值為：Pending validation (待定驗證) | Revoked (已撤銷) | Failed (失敗) | Validation timed out (驗證逾時) | Success (成功)
- 詳細資訊
  - 使用中？ - 憑證是否與 AWS 整合服務相關聯 可能值為：Yes (是) | No (否)
  - 網域名稱 — 憑證的第一個完整網域名稱 (FQDN)。
  - Number of additional names（其他名稱的數量) - 憑證有效的網域名稱數量
  - Serial number（序號) - 憑證的 16 位元組十六進位序號
  - 公有金鑰資訊 - 產生金鑰對的密碼編譯演算法
  - Signature algorithm (簽章演算法) - 用於簽署憑證的密碼編譯演算法。
  - 可搭配使用 — 支援具有這些參數之憑證的ACM整合式服務清單
  - Requested at（請求於) - 發出請求的日期和時間
  - Issued at（發行日期) - 如適用，發行日期及時間
  - Imported at（匯入) - 如適用，匯入的日期和時間
  - Not before（生效時間) - 憑證有效期間開始
  - Not after（失效時間) - 憑證的過期日期和時間
  - Renewal eligibility (續約資格) - 可能的值為：Eligible (符合資格)| Ineligible (不符合資格) 如需資格規則，請參閱 ACM憑證的受管理續約。
  - Renewal status (續約狀態) – 所要求憑證續約的狀態。只有在要求續約後，此欄位才會顯示並具有值。可能的值為：Pending automatic renewal (等待自動續約) | Pending validation (等待驗證) | Success (成功續約) | Failure (未能續約)。

> **Note**
>
> 憑證狀態的變更可能需要數小時才會變成可用。若遇到問題，憑證要求會在 72 小時後逾時，並且必須從頭開始重複發行或續約程序。

  - CA — 簽署ARN的 CA
- Tags (標籤)
  - 索引鍵
  - 值
- Validation state (驗證狀態) - 如果適用，可能值如下：
  - Pending (待定) - 已請求驗證且尚未完成。
  - Validation timed out (驗證逾時) - 請求的驗證已逾時，但您可以重複該請求。
  - 無 — 憑證適用於私人PKI或是自我簽署的憑證，不需要驗證。

使用檢視憑證詳細資料 AWS CLI

使用中的描述憑證 AWS CLI 來顯示憑證詳細資料，如下列命令所示：

```
$ aws acm describe-certificate --certificate-arn
 arn:aws:acm:Region:444455556666:certificate/certificate_ID
```

此命令會傳回與以下內容相似的資訊：

```
{
    "Certificate": {
        "CertificateArn": "arn:aws:acm:Region:444455556666:certificate/certificate_ID",
        "Status": "EXPIRED",
        "Options": {
            "CertificateTransparencyLoggingPreference": "ENABLED"
        },
        "SubjectAlternativeNames": [
            "example.com",
            "www.example.com"
        ],
        "DomainName": "gregpe.com",
        "NotBefore": 1450137600.0,
        "RenewalEligibility": "INELIGIBLE",
        "NotAfter": 1484481600.0,
```

```
        "KeyAlgorithm": "RSA-2048",
        "InUseBy": [
            "arn:aws:cloudfront::account:distribution/E12KXPQHVLSYVC"
        ],
        "SignatureAlgorithm": "SHA256WITHRSA",
        "CreatedAt": 1450212224.0,
        "IssuedAt": 1450212292.0,
        "KeyUsages": [
            {
                "Name": "DIGITAL_SIGNATURE"
            },
            {
                "Name": "KEY_ENCIPHERMENT"
            }
        ],
        "Serial": "07:71:71:f4:6b:e7:bf:63:87:e6:ad:3c:b2:0f:d0:5b",
        "Issuer": "Amazon",
        "Type": "AMAZON_ISSUED",
        "ExtendedKeyUsages": [
            {
                "OID": "1.3.6.1.5.5.7.3.1",
                "Name": "TLS_WEB_SERVER_AUTHENTICATION"
            },
            {
                "OID": "1.3.6.1.5.5.7.3.2",
                "Name": "TLS_WEB_CLIENT_AUTHENTICATION"
            }
        ],
        "DomainValidationOptions": [
            {
                "ValidationEmails": [
                    "hostmaster@example.com",
                    "admin@example.com",
                    "postmaster@example.com",
                    "webmaster@example.com",
                    "administrator@example.com"
                ],
                "ValidationDomain": "example.com",
                "DomainName": "example.com"
            },
            {
                "ValidationEmails": [
                    "hostmaster@example.com",
                    "admin@example.com",
```

```
                    "postmaster@example.com",
                    "webmaster@example.com",
                    "administrator@example.com"
                ],
                "ValidationDomain": "www.example.com",
                "DomainName": "www.example.com"
            }
        ],
        "Subject": "CN=example.com"
    }
}
```

刪除由管理的憑證 ACM

您可以使用ACM控制台或 AWS CLI 刪除憑證。

Important

- 您無法刪除其他 AWS 服務正在使用的ACM憑證。若要刪除使用中的憑證，您必須先移除憑證關聯。這是使用控制台或CLI相關服務完成的。
- 刪除私人憑證授權機構 (CA) 發行的憑證對 CA 沒有任何影響。您將繼續向 CA 收費，直到其遭刪除為止。如需詳細資訊，請參閱 AWS Private Certificate Authority 使用者指南中的刪除私有 CA。

使用主控台刪除憑證

1. 開啟位於 ACM 的 https://console.aws.amazon.com/acm/ 主控台。
2. 在憑證清單中，選取憑ACM證的核取方塊，然後選擇刪除。

> **Note**
>
> 視您排序清單的方式而定，您要尋找的憑證可能無法立即顯示。您可以點選右邊的黑色三角形來變更順序。您也可以使用右上角的頁碼在多張憑證頁面之間瀏覽。

若要刪除憑證 AWS CLI

使用 delete-certificate 命令來刪除憑證，如以下命令所示：

```
$ aws acm delete-certificate --certificate-arn
 arn:aws:acm:Region:444455556666:certificate/certificate_ID
```

安裝ACM憑證

您無法使用ACM直接在您的網站或應用程 AWS 式上安裝公用憑證。您必須使用與之整合的其中一項服務ACM。如需詳細資訊，請參閱 服務整合 AWS Certificate Manager。

ACM由 CA 在中簽署 AWS 私有 CA 且用於私人的憑證PKI可以在您擁有管理權限的任何系統上匯出和安裝手動安裝。公有網際網路上不信任這些憑證。

ACM憑證的受管理續約

ACM為您的亞馬遜頒發的SSL/TLS證書提供託管續訂。這意味著，ACM將自動更新您的證書（如果您正在使用DNS驗證），或者它將在到期時向您發送電子郵件通知。這些服務是針對公有憑證和私有ACM憑證提供的。

根據下列考量，憑證符合自動續約的資格：

- ELIGIBLE如果與其他 AWS 服務相關聯，例如 Elastic Load Balancing 或 CloudFront.
- ELIGIBLE如果自發行或上次續約後匯出。
- ELIGIBLE如果它是通過調用發行的私有證書，ACMRequestCertificateAPI然後導出或與其他 AWS 服務相關聯。
- ELIGIBLE如果它是通過管理控制台發行的私有證書，然後導出或與其他 AWS 服務相關聯。
- NOTELIGIBLE如果它是通過調用 AWS 私有 CA IssueCertificateAPI.
- NOTELIGIBLE如果匯入。
- NOTELIGIBLE如果已過期。

此外，必須滿足以下與國際化網域名稱有關的 Punycode 要求：

1. 以 "<character><character>--" 模式開頭的網域名稱必須匹配 "xn--"。
2. 以 "xn--" 開頭的網域名稱也必須是有效的國際化網域名稱。

Punycode 範例

網域名稱	滿足 #1	滿足 #2	允許	注意
example。com	N/A	無	✓	不以 "<character><character>--" 開頭
a--example.com	N/A	無	✓	不以 "<character><character>--" 開頭
abc--example.com	N/A	無	✓	不以 "<character><character>--" 開頭

網域名稱	滿足 #1	滿足 #2	允許	注意
xn--xyz.com	是	是	✓	有效的國際化網域名稱 (解析為簡.com)
xn--examp le.com	是	否	✗	不是有效的國際化網域名稱
ab--examp le.com	否	否	✗	必須以 "xn--" 開頭

ACM續訂憑證時，憑證的 Amazon 資源名稱 (ARN) 會保持不變。另外，ACM 憑證為區域資源。如果您在多個 AWS 區域中擁有相同網域名稱的憑證，則每個憑證都必須獨立續約。

主題

- 續約公開信任的憑證
- 在私人中更新憑證 PKI
- 檢查憑證的續約狀態

續約公開信任的憑證

發行受管理、公開信任的憑證時，您 AWS Certificate Manager 必須證明您是網域擁有者。這通過DNS驗證或電子郵件驗證來發生。當憑證出現要續約時，ACM會使用您先前選擇的相同方法來重新驗證您的擁有權。下列主題描述了續約程序在每一種案例裡運作的方式。

主題

- 已驗證的網域續約 DNS
- 續約透過電子郵件驗證的網域

已驗證的網域續約 DNS

對於最初使用DNS驗證發行的ACM憑證，受管續訂是完全自動化的。

在到期前 60 天，ACM檢查以下續訂條件：

- 服務目前正在使用憑 AWS 證。

- 所有必要ACM提供的DNSCNAME記錄（每個唯一的主體替代名稱一個）都存在並可通過公共DNS訪問。

如果符合這些條件，請ACM考慮已驗證的網域名稱並更新憑證。

ACM在續約期間無法自動驗證網域時 (例如，因為存在CAA記錄)，傳送 AWS Health 事件和 Amazon EventBridge 事件。這些活動會在過期前 45 天、30 天、15 天、7 天、3 天和 1 天傳送。如需詳細資訊，請參閱 對 ACM 的 Amazon EventBridge 支持。

續約透過電子郵件驗證的網域

ACM憑證的有效期為 13 個月 (395 天)。續訂憑證需要網域擁有者採取行動。ACM會在到期前 45 天開始傳送續約通知至與網域相關聯的電子郵件地址。通知包含網域擁有者可以按一下進行續約的連結。驗證所有列出的網域後，會以相同的方式ACM發行續約憑證ARN。

如需有關驗證電子郵件的詳細資訊，請參閱「電子郵件驗證」。

若要瞭解如何以程式設計方式來回應驗證電子郵件，請參閱 自動化電子郵件驗證的流程 。

重新傳送驗證電子郵件

在您要求憑證時為網域設定電子郵件驗證後 (請參閱(選用) 為您的網域設定電子郵件)，您可以使用 AWS Certificate Manager API來要求ACM傳送憑證續約的網域驗證電子郵件給您。您應在以下情況執行此動作：

- 您在初次要求憑證時使用了電子郵件驗ACM證。
- 您的憑證的續約狀態為待定驗證。如需有關判斷憑證續約狀態的詳細資訊，請參閱 檢查憑證的續約狀態。
- 您沒有收到或找不到ACM傳送憑證續訂的原始網域驗證電子郵件訊息。

若要將驗證電子郵件傳送至與您原先在憑證要求中設定的網域不同的網域，您可以使用ACMAPI AWS CLI、或中的ResendValidationEmail作業 AWS SDKs。ACM將發送電子郵件到指定的驗證域。您可以在支援的區域 AWS CLI 中使用 AWS CloudShell 在瀏覽器中存取。

要求ACM重新傳送網域驗證電子郵件訊息 (主控台)

1. 請在https://console.aws.amazon.com/acm/家中開啟 AWS Certificate Manager 主機。
2. 選擇需要驗證憑證的憑證 ID。

3. 選擇重新傳送驗證電子郵件。

要求ACM重新傳送網域驗證電子郵件 (ACMAPI)

使用中的ResendValidationEmail作業ACMAPI。如此一來，請傳遞憑證、需要手動驗證的網域，以及您想要接收網域驗證電子郵件的網域。ARN以下範例顯示如何使用 AWS CLI執行此作業。此範例含分行符號以利閱讀。

```
$ aws acm resend-validation-email \
 --certificate-arn arn:aws:acm:region:account:certificate/certificate_ID \
 --domain subdomain.example.com \
 --validation-domain example.com
```

在私人中更新憑證 PKI

ACM由私有 CA 簽署的憑證符合 AWS 私有 CA 受管理續約的資格。與公開信任的ACM憑證不同，私有憑證不PKI需要驗證。系統管理員在用戶端信任存放區中安裝適當的根憑證授權機構憑證時，就會建立信任。

> **Note**
>
> 只有使用ACM主控台或的RequestCertificate動作取得的憑證才ACMAPI有資格進行受管理續約。直接 AWS 私有 CA 使用的IssueCertificate動作發行的憑證不 AWS 私有 CA API受管理ACM。

當受管理憑證離到期日 60 天後，ACM會自動嘗試續約。這包括手動匯出和安裝的憑證 (例如在內部部署資料中心裡)。客戶也可以隨時使用的RenewCertificate動作強制續訂ACMAPI。如需強制續約的 Java 實作範例，請參閱 續約憑證。

續約後，會依下列其中一種方式將憑證部署至服務：

- 如果憑證與ACM整合式服務相關聯，則新憑證會取代舊憑證，而不需要額外的客戶動作。
- 如果憑證未與ACM整合式服務相關聯，則需要客戶採取動作才能匯出並安裝更新的憑證。您可以手動執行這些動作，也可以在 Amazon 的協助下執行這些動作 EventBridge，AWS Lambda如下所示。AWS Health如需詳細資訊，請參閱 自動化續約憑證的匯出作業

自動化續約憑證的匯出作業

下列程序提供範例解決方案，可在續訂私有PKI憑證時ACM自動匯出私有憑證。此範例只會將憑證及其私密金鑰匯出ACM；匯出後，憑證仍必須安裝在其目標裝置上。

若要使用主控台自動化憑證匯出作業

1. 遵循 AWS Lambda 開發人員指南中的程序，建立並設定呼叫ACM匯出的 Lambda 函數API。

   a. 建立 Lambda 函數。

   b. 為您的函數建立 Lambda 執行角色，並將下列信任政策加入函數。此原則會透過呼叫的ExportCertificate動作，授予函數中程式碼的權限，以擷取更新的憑證和私密金鑰ACMAPI。

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

2. 在 Amazon 中建立規則 EventBridge以偵聽ACM健康事件，並在偵測到 Lambda 函數時呼叫您的 Lambda 函數。ACM每次嘗試更新憑證時，都會寫入 AWS Health 事件。如需這些通知的詳細資訊，請參閱「使用 Personal Health Dashboard 檢查狀態（PHD）」。

   加入下列事件模式來設定規則。

```
{
   "source":[
      "aws.health"
   ],
   "detail-type":[
      "AWS Health Event"
   ],
   "detail":{
      "service":[
         "ACM"
```

```
      ],
      "eventTypeCategory":[
         "scheduledChange"
      ],
      "eventTypeCode":[
         "AWS_ACM_RENEWAL_STATE_CHANGE"
      ]
   },
   "resources":[
      "arn:aws:acm:region:account:certificate/certificate_ID"
   ]
}
```

3. 在目標系統上手動安裝憑證以完成續約程序。

測試私有PKI憑證的受管更新

您可以使用ACMAPI或 AWS CLI 手動測試ACM受管續訂工作流程的組態。這樣，您可以在到期ACM前確認您的憑證將自動更新。

> **Note**
>
> 您只能測試由發行和匯出之憑證的續約 AWS 私有 CA。

當您使用下述API動作或CLI命令時，會ACM嘗試更新憑證。如果續約成功，請ACM更新管理主控台或API輸出中顯示的憑證中繼資料。如果憑證與ACM整合式服務相關聯，則會部署新憑證，並在 Amazon E CloudWatch vents 中產生更新事件。如果續約失敗，會ACM傳回錯誤並建議修復動作。(您可以使用 describe-certificate 命令。) 如果憑證未透過整合服務部署，您仍需要將憑證匯出並手動安裝到您的資源上。

> **Important**
>
> 若要更新 AWS 私有 CA 憑證ACM，您必須先授與ACM服務主體權限，才能執行這項操作。如需詳細資訊，請參閱將憑證續訂權限指派給ACM。

若要手動測試憑證續約 (AWS CLI)

1. 使用 renew-certificate 命令來續約私有匯出憑證。

```
aws acm renew-certificate \
 --certificate-arn arn:aws:acm:region:account:certificate/certificate_ID
```

2. 然後，使用 describe-certificate 命令來確認憑證的續約詳細資訊已更新。

```
aws acm describe-certificate \
 --certificate-arn arn:aws:acm:region:account:certificate/certificate_ID
```

手動測試憑證更新 (ACMAPI)

- 傳送要RenewCertificate求，指定要續約ARN的私人憑證。然後使用DescribeCertificate作業確認憑證的續約詳細資料已更新。

檢查憑證的續約狀態

當您嘗試更新憑證時，請在憑證詳細資料中ACM提供續訂狀態資訊欄位。您可以使用 AWS Certificate Manager 控制台ACMAPI、 AWS CLI、或 AWS Health Dashboard 來檢查ACM憑證的續約狀態。如果您使用主控台 AWS CLI、或 ACMAPI，續訂狀態可能會有下列四個可能的狀態值之一。如果您使用 AWS Health Dashboard，便會顯示類似值。

待定自動續約

ACM正在嘗試自動驗證憑證中的網域名稱。如需詳細資訊，請參閱 已驗證的網域續約 DNS。無需採取進一步動作。

待定驗證

ACM無法自動驗證憑證中的一或多個網域名稱。您必須採取動作驗證這些網域名稱，否則憑證不會續約。如果您最初使用電子郵件驗證憑證，請尋找來自的電子郵件，ACM然後按照該電子郵件中的連結來執行驗證。如果您使用DNS驗證，請檢查以確定您的DNS記錄存在，並且您的憑證仍在使用中。

Success (成功)

憑證中的所有網域名稱都會經過驗證，並ACM更新憑證。無需採取進一步動作。

失敗

一或多個網域名稱未在憑證到期前驗證，也ACM沒有續約憑證。您可以要求新的憑證。

如果憑證與其他 AWS 服務 (例如 Elastic Load Balancing) 相關聯，或者如果憑證在發行或上次更新後已匯出 CloudFront，則該憑證有資格進行續約。

> **Note**
>
> 續約狀態的變更可能需要數小時才能提供。若出現問題，續約要求會在 72 小時後逾時，您必須從頭開始續約程序。如需故障診斷協助，請參閱憑證請求疑難排解。

主題

檢查狀態 (主控台)

下列程序討論如何使用ACM主控台來檢查ACM憑證的續約狀態。

1. 請在https://console.aws.amazon.com/acm/家中開啟 AWS Certificate Manager 主機。
2. 展開憑證以檢視其詳細資訊。
3. 在 Details (詳細資訊) 區段中找到 Renewal Status (續約狀態)。如果您沒有看到狀態，表示ACM尚未啟動此憑證的受管理續約程序。

檢查狀態 (API)

如需說明如何使用DescribeCertificate動作檢查狀態的 Java 範例，請參閱描述憑證。

檢查狀態 (CLI)

下列範例顯示如何使用 AWS Command Line Interface (AWS CLI) 檢查ACM憑證續約狀態。

```
$ aws acm describe-certificate \
 --certificate-arn arn:aws:acm:region:account:certificate/certificate_ID
```

在回應中，請注意 RenewalStatus 欄位中的值。如果您沒有看到此RenewalStatus欄位，表示ACM尚未啟動憑證的受管理續約程序。

使用 Personal Health Dashboard 檢查狀態（PHD）

ACM嘗試在到期前 60 天自動續訂您的ACM憑證。如果ACM無法自動續訂您的憑證，它會 AWS Health Dashboard 在 45 天、30 天、15 天、7 天、3 天和到期後的 1 天間隔傳送憑證續訂事件通知給您，以通知您需要採取行動。 AWS Health Dashboard 是 AWS Health 服務的一部分。它不需要設定，而且您帳戶中經過驗證的任何使用者皆可檢視。如需詳細資訊，請參閱 AWS Health 使用者指南。

> **Note**
>
> ACM將連續的續約事件通知寫入PHD時間表中的單一事件。每個通知都會覆寫前一個通知，直到續約成功為止。

使用 AWS Health Dashboard：

1. 登入到https://phd.aws.amazon.com/phd/家 AWS Health Dashboard 裡 #/。
2. 選擇 Event log (事件日誌)。
3. 在 Filter by tags or attributes (依標籤或屬性篩選) 選擇 Service (服務)。
4. 選擇 Certificate Manager。
5. 選擇套用。
6. 在 Event category (事件類別) 選擇 Scheduled Change (排定的變更)。
7. 選擇套用。

自動化電子郵件驗證的流程

電子郵件驗證的ACM憑證通常需要網域擁有者手動處理。組織如需處理大量透過電子郵件驗證的憑證，可能會偏好建立可自動執行所需回應的剖析器。為了協助客戶使用電子郵件驗證，本節中的資訊說明用於網域驗證電子郵件訊息範本，以及完成驗證程序所涉及的工作流程。

驗證電子郵件範本

驗證電子郵件訊息具有下列兩種格式之一，具體取決於要求新憑證還是續約現有憑證。反白顯示字串的內容應取代為待驗證網域的專屬值。

驗證新憑證

電子郵件範本文字：

```
Greetings from Amazon Web Services,

We received a request to issue an SSL/TLS certificate for requested_domain.

Verify that the following domain, AWS account ID, and certificate identifier
 correspond
to a request from you or someone in your organization.

Domain: fqdn
AWS account ID: account_id
AWS Region name: region_name
Certificate Identifier: certificate_identifier

To approve this request, go to Amazon Certificate Approvals
(https://region_name.acm-certificates.amazon.com/approvals?
code=validation_code&context=validation_context)
and follow the instructions on the page.

This email is intended solely for authorized individuals for fqdn. To express any
 concerns
about this email or if this email has reached you in error, forward it along with a
 brief
explanation of your concern to validation-questions@amazon.com.

Sincerely,
```

```
Amazon Web Services
```

驗證憑證以進行續約

電子郵件範本文字：

```
Greetings from Amazon Web Services,

We received a request to issue an SSL/TLS certificate for requested_domain.
This email is a request to validate ownership of the domain in order to renew
the existing, currently in use, certificate. Certificates have defined
validity periods and email validated certificates, like this one, require you
to re-validate for the certificate to renew.

Verify that the following domain, AWS account ID, and certificate identifier
correspond to a request from you or someone in your organization.

Domain: fqdn
AWS account ID: account_id
AWS Region name: region_name
Certificate Identifier: certificate_identifier

To approve this request, go to Amazon Certificate Approvals at
https://region_name.acm-certificates.amazon.com/approvals?code=
$validation_code&context=$validation_context
and follow the instructions on the page.

This email is intended solely for authorized individuals for fqdn. You can see
more about how AWS Certificate Manager validation works here -
https://docs.aws.amazon.com/acm/latest/userguide/email-validation.html.
To express any concerns about this email or if this email has reached you in
error, forward it along with a brief explanation of your concern to
validation-questions@amazon.com.

Sincerely,
Amazon Web Services

--
Amazon Web Services, Inc. is a subsidiary of Amazon.com, Inc. Amazon.com is a
registered trademark of Amazon.com, Inc.

This message produced and distributed by Amazon Web Services, Inc.,
410 Terry Ave. North, Seattle, WA 98109-5210.
```

```
(c)2015-2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Our privacy policy is posted at https://aws.amazon.com/privacy
```

一旦你收到一個新的驗證消息 AWS，我們建議您使用它作為您的解析器最 up-to-date 權威的模板。客戶若使用 2020 年 11 月之前設計的訊息剖析器，應注意可能已對範本進行下列變更：

- 電子郵件主旨行現在為「Certificate request for *domain name*」而不是「"Certificate approval for *domain name*」。
- AWS account ID 現在會以不含破折號或連字號的形式呈現。
- Certificate Identifier現在會顯示整個憑證，而ARN不是簡短的表單，例如，*arn:aws:acm:us-east-1:000000000000:certificate/3b4d78e1-0882-4f51-954a-298ee44ff369*而不是*3b4d78e1-0882-4f51-954a-298ee44ff369*。
- 憑證核准URL現在包含acm-certificates.amazon.com而不是certificates.amazon.com。
- 按一下憑證核准開啟的核准表單URL現在包含核准按鈕。核准按鈕 div 的名稱現在是 approve-button 而不是 approval_button。
- 新請求的憑證和續約憑證的驗證訊息使用相同的電子郵件格式。

驗證工作流程

本節提供電子郵件驗證憑證的續約工作流程的相關資訊。

- 當ACM主控台處理多網域憑證要求時，會將驗證電子郵件訊息傳送至您要求公用憑證時指定的網域名稱或驗證網域。網域擁有者必須先驗證每個網域的電子郵件訊息，才ACM能核發憑證。如需詳細資訊，請參閱使用電子郵件驗證網域所有權。
- 使用 ACM API or 的多網域憑證要求的電子郵件驗證會CLI導致每個要求的網域傳送電子郵件訊息，即使要求中包含其他網域的子網域也一樣。網域擁有者必須先驗證每個網域的電子郵件訊息，才ACM能核發憑證。

  如果您透過ACM主控台重新傳送現有憑證的電子郵件，電子郵件將傳送至原始憑證要求中指定的驗證網域，或傳送確切的網域 (如果未指定驗證網域)。若要接收不同網域的驗證電子郵件，您可以要求新憑證，並指定要用於驗證的驗證網域。或者，您也可以使用APISDK、或來呼叫ResendValidationEmailValidationDomain參數CLI。但是，ResendValidationEmail請求中指定的驗證網域僅用於該呼叫，不會儲存到憑證 Amazon 資源名稱 (ARN) 以供 future 驗證電子郵

件使用。ResendValidationEmail每次您希望透過原始憑證要求中未指定的網域名稱收到驗證電子郵件時，都必須撥打電話。

> **Note**
>
> 2020 年 11 月之前，客戶只需要驗證頂點網域，並核發ACM也涵蓋任何子網域的憑證。在該時間之前設計訊息剖析器的客戶，應注意電子郵件驗證工作流程有所變更。

- 使用ACMAPI或CLI，您可以強制將多網域憑證要求的所有驗證電子郵件訊息傳送至 Apex 網域。在中API，使用RequestCertificate動作的DomainValidationOptions參數來指定屬於ValidationDomainDomainValidationOption類型成員的值。在中CLI，使用要求憑證命令的--domain-validation-options參數來指定的值。ValidationDomain

將憑證匯入 AWS Certificate Manager

除了由 AWS Certificate Manager (ACM) 提供的要求SSL/TLS憑證之外，您還可以匯入在以外取得的憑證 AWS。您可能會這麼做，是因為您已經擁有來自協力廠商憑證授權單位 (CA) 的憑證，或是因為您的應用程式特定需求不符合已ACM發行的憑證。

您可以將匯入的憑證與任何整合的AWS 服務搭配使用ACM。您匯入的憑證的運作方式與提供的憑證相同ACM，但有一個重要的例外：ACM不為匯入的憑證提供受管理的續約。

若要更新匯入的憑證，您可以從憑證簽發者取得新憑證，然後手動將ACM其重新匯入。此動作會保留憑證的關聯及其 Amazon 資源名稱 (ARN)。或者，您也可以匯入全新的憑證。您可以匯入具有相同網域名稱的多個憑證，但必須一次匯入一個。

> **Important**
>
> 您須負責監控匯入憑證的過期日期，並在憑證過期前續約。您可以使用 Amazon E CloudWatch vents 在匯入的憑證即將到期時傳送通知，以簡化此任務。如需詳細資訊，請參閱使用 Amazon EventBridge。

中的所有憑證ACM都是地區資源，包括您匯入的憑證。若要在不同區 AWS 域中搭配 Elastic Load Balancing 器使用相同憑證，您必須將憑證匯入每個要使用憑證的區域。若要在 Amazon 使用憑證 CloudFront，您必須將憑證匯入美國東部 (維吉尼亞北部) 區域。如需詳細資訊，請參閱 支援地區。

如需如何將憑證匯入的相關資訊ACM，請參閱下列主題。如果您在匯入憑證時遇到問題，請參閱 憑證匯入問題。

主題

匯入憑證的先決條件

若要將自我簽署SSL/TLS憑證匯入ACM，您必須同時提供憑證及其私密金鑰。若要匯入非AWS 憑證授權機構 (CA) 簽署的憑證，您也必須納入憑證的私有和公有金鑰。您的憑證必須滿足此主題中所描述的所有條件。

對所有匯入的憑證，您必須指定密碼編譯演算法和金鑰大小。ACM支持以下算法（括號中的API名稱）：

- RSARSA_1024
- RSA位元 RSA_2048
- RSA位元 RSA_3072
- RSA位元 RSA_4096
- ECDSA二EC_prime256v1十六位元
- ECDSA位元 EC_secp384r1
- ECDSA位元 EC_secp521r1

另請注意以下額外要求：

- ACM整合式服務只允許其支援的演算法和金鑰大小與其資源建立關聯。例如， CloudFront 僅支援 1024 位元、2048 位元RSARSA、3072 位元和橢圓主要曲線 256 位元金鑰RSA，而「Application Load Balancer」則支援所有可用的演算法。ACM如需詳細資訊，請參閱您所使用服務的說明文件。
- 憑證必須是SSL/TLSX.509 第 3 版憑證。它必須包含公開金鑰、網站的完整網域名稱 (FQDN) 或 IP 位址，以及發行者的相關資訊。
- 憑證可以由您擁有的私有金鑰自行簽署，或由核發 CA 的私有金鑰簽署。您必須提供不超過 5 KB (5,120 個位元組) 的私有金鑰，且必須未加密。
- 如果憑證是由 CA 簽署，而您選擇提供憑證鏈結，則該鏈結必須是 PEM —coded。
- 憑證在匯入時必須有效。您無法在憑證有效期間開始前或過期後匯入憑證。NotBefore 憑證欄位包含有效期間開始日期和包含結束日期的 NotAfter 欄位。
- 所有必要的憑證材料 (憑證、私密金鑰和憑證鏈) 都必須經過 PEM —coded 編碼。上傳 DER — 編碼的材料會導致錯誤。如需詳細資訊和範例，請參閱 用於匯入的憑證和金鑰格式。
- 當您更新 (重新匯入) 憑證時，您無法新增KeyUsage或ExtendedKeyUsage副檔名（如果副檔名不存在於先前匯入的憑證中）。
- AWS CloudFormation 不支援將憑證匯入ACM。

用於匯入的憑證和金鑰格式

ACM要求您分別匯入憑證、憑證鏈結和私密金鑰 (如果有的話)，並以PEM格式對每個元件進行編碼。PEM代表隱私增強郵件。此PEM格式通常用於表示憑證、憑證要求、憑證鏈結和金鑰。PEM—format 檔案的典型副檔名為 `.pem`，但不需要。

> **Note**
>
> AWS 不提供用於操作PEM檔案或其他憑證格式的公用程式。下列範例仰賴一般文字編輯器來進行簡單的作業。如果您需要執行更複雜的任務（例如轉換文件格式或提取密鑰），可以隨時使用免費的開源工具，例SSL如 Open。

下列範例說明要匯入的檔案格式。如果單一檔案中出現多個元件，請 (小心地) 使用文字編輯器將它們分成三個檔案。請注意，如果您不正確地編輯PEM檔案中的任何字元，或是在任何行尾加入一或多個空格，則憑證、憑證鏈結或私密金鑰將無效。

Example 1. PEM— 編碼憑證

```
-----BEGIN CERTIFICATE-----
Base64-encoded certificate
-----END CERTIFICATE-----
```

Example 2. PEM— 編碼的憑證鏈

憑證鏈包含一或多個憑證。您可以使用文字編輯器、Windows 的 `copy` 指令，或 Linux `cat` 命令，將憑證檔案串連為憑證鏈。憑證必須依序串連，使每個憑證直接認證上一個憑證。如果匯入私有憑證，請最後複製根憑證。以下範例包含三個憑證，但您的憑證鏈可包含更多或更少憑證。

> **Important**
>
> 不要將您的憑證複製到憑證鏈。

```
-----BEGIN CERTIFICATE-----
Base64-encoded certificate
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
Base64-encoded certificate
```

```
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
Base64-encoded certificate
-----END CERTIFICATE-----
```

Example 3. PEM— 編碼的私密金鑰

X.509 第 3 版憑證使用公有金鑰演算法。建立 X.509 憑證或憑證請求時，您需指定必須用於建立私有/公有金鑰對的演算法和金鑰位元大小。公有金鑰會置於憑證或要求中。您必須將關聯的私有金鑰保密。在匯入憑證時指定私有金鑰。金鑰必須為未加密。下列範例顯示RSA私密金鑰。

```
-----BEGIN RSA PRIVATE KEY-----
Base64-encoded private key
-----END RSA PRIVATE KEY-----
```

下一個範例顯示了 PEM —coded 的橢圓曲線私密金鑰。視您建立金鑰的方式而定，可能不會包含參數區塊。如果包括參數圖塊，請在匯入過程中使用鍵之前將其ACM移除。

```
-----BEGIN EC PARAMETERS-----
Base64-encoded parameters
-----END EC PARAMETERS-----
-----BEGIN EC PRIVATE KEY-----
Base64-encoded private key
-----END EC PRIVATE KEY-----
```

匯入憑證

您可以使用、或將ACM外部取得的憑證 (亦即由協力廠商信任服務提供者提供的 AWS Management Console憑證) 匯入ACMAPI。 AWS CLI下列主題向您展示如何使用 AWS Management Console 和 AWS CLI。向非AWS 發行人取得證書的程序不在本指南的範圍內。

> **Important**
>
> 您選取的簽章演算法必須符合 匯入憑證的先決條件。

主題

- 匯入 (主控台)

- 匯入 (AWS CLI)

匯入 (主控台)

以下範例顯示如何使用 AWS Management Console匯入憑證。

1. 請在https://console.aws.amazon.com/acm/家中開啟ACM主機。如果這是您第一次使用ACM，請查找標AWS Certificate Manager題並選擇其下的「開始使用」按鈕。
2. 選擇 Import a certificate (匯入憑證)。
3. 請執行下列操作：
   a. 對於憑證主體，貼上要匯入的PEM已編碼憑證。開頭應為 `-----BEGIN CERTIFICATE-----` 而結尾是 `-----END CERTIFICATE-----`。
   b. 對於憑證私密金鑰，貼上憑證的已PEM編碼、未加密的私密金鑰。開頭應為 `-----BEGIN PRIVATE KEY-----` 而結尾是 `-----END PRIVATE KEY-----`。
   c. (選擇性) 對於「憑證」鏈結，貼上PEM已編碼的憑證鏈結。
4. (選擇性) 若要將標籤新增至匯入的憑證，請選擇「標籤」。標籤是指派給 AWS 資源的標籤。每個標籤皆包含由您定義的一個金鑰與一個選用值。您可以使用標籤來整理資源或追蹤 AWS 成本。
5. 選擇匯入。

匯入 (AWS CLI)

以下範例顯示如何使用 AWS Command Line Interface (AWS CLI) 匯入憑證。該範例假設如下：

- PEM編碼的憑證會儲存在名為`Certificate.pem`的檔案中。
- PEM編碼的憑證鏈結會儲存在名為`CertificateChain.pem`的檔案中。
- 經過PEM編碼的未加密私密金鑰會儲存在名為的檔案中。`PrivateKey.pem`

若要使用以下範例，請將檔案名稱取代為您自己的檔案名稱，並在連續的一行中輸入命令。為方便閱讀，以下範例包含分行符號和多餘的空格。

```
$ aws acm import-certificate --certificate fileb://Certificate.pem \
      --certificate-chain fileb://CertificateChain.pem \
      --private-key fileb://PrivateKey.pem
```

如果命`import-certificate`令成功，它會傳回已匯入憑證的 Amazon 資源名稱 (ARN)。

重新匯入憑證

如果您已匯入憑證並將其與其他 AWS 服務產生關聯，則可以在憑證到期前重新匯入該憑證，同時保留原始憑證的 AWS 服務關聯。如需整合 AWS 服務的詳細資訊ACM，請參閱服務整合 AWS Certificate Manager。

重新匯入憑證時，適用以下條件：

- 您可以新增或移除網域名稱。
- 您不能移除憑證中的所有網域名稱。
- 如果金鑰用量延伸在最初匯入的憑證中存在，您就可以加入新的延伸值，但不能移除現有值。
- 如果延伸的金鑰用量延伸在最初匯入的憑證中存在，您就可以加入新的延伸值，但不能移除現有值。
- 金鑰類型和大小無法變更。
- 您無法在重新匯入憑證時套用資源標籤。

主題

重新匯入 (主控台)

以下範例顯示如何使用 AWS Management Console重新匯入憑證。

1. 請在https://console.aws.amazon.com/acm/家中開啟ACM主機。
2. 選擇或展開要重新匯入的憑證。
3. 開啟憑證的詳細資訊窗格，然後選擇 Reimport certificate (重新匯入憑證) 按鈕。如果您是透過勾選憑證名稱旁的方塊來選擇憑證，請選擇 Actions (動作) 功能表上的 Reimport certificate (重新匯入憑證)。
4. 對於憑證主體，貼上PEM編碼的最終實體憑證。
5. 對於憑證私密金鑰，貼上與憑證公開金鑰相關聯的未加PEM密私密金鑰。
6. (選擇性) 對於「憑證」鏈結，貼上PEM已編碼的憑證鏈結。憑證鏈包含所有中繼發行認證授權單位的一個或多個憑證，以及根憑證。如果要匯入的憑證是自動指派的，就不需要憑證鏈。
7. 檢閱憑證的資訊。如果沒有任何錯誤，請選擇 Reimport (重新匯入)。

重新匯入 (AWS CLI)

以下範例顯示如何使用 AWS Command Line Interface (AWS CLI) 重新匯入憑證。該範例假設如下：

- PEM編碼的憑證會儲存在名為Certificate.pem的檔案中。
- PEM編碼的憑證鏈結會儲存在名為CertificateChain.pem的檔案中。
- (僅限私人憑證) 經過PEM編碼的未加密私密金鑰會儲存在名為的檔案中。PrivateKey.pem
- 您擁有要重新匯入ARN的憑證。

若要使用下列範例，請使用您自己的檔案名稱和取ARN代檔案名稱，然後在一行連續輸入指令。為方便閱讀，以下範例包含分行符號和多餘的空格。

> **Note**
>
> 若要重新匯入憑證，您必須指定憑證ARN。

```
$ aws acm import-certificate --certificate fileb://Certificate.pem \
      --certificate-chain fileb://CertificateChain.pem \
      --private-key fileb://PrivateKey.pem \
      --certificate-
arn arn:aws:acm:region:123456789012:certificate/12345678-1234-1234-1234-12345678901
```

如果命import-certificate令成功，它會傳回憑證的 Amazon 資源名稱 (ARN)。

匯出私有憑證

您可以匯出由發行的憑證，以 AWS 私有 CA 便在私人 PKI 環境中的任何位置使用。匯出的檔案包含憑證、憑證鏈，以及加密的私有金鑰。此檔案必須安全地存放。若要取得有關的更多資訊 AWS 私有 CA，請參閱AWS Private Certificate Authority 使用指南

> **Note**
>
> 無論是由 ACM 發行還是已匯入，您都無法匯出公開信任憑證或其私密金鑰。

主題

匯出私有憑證 (主控台)

1. 登入 AWS 管理主控台並開啟 ACM 主控台，網址為 https://console.aws.amazon.com/acm/home。
2. 選擇 Certificate Manager
3. 選擇您要匯出的憑證的連結。
4. 選擇 Export (匯出)。
5. 輸入並確認私有金鑰的密碼短語。

   > **Note**
   >
   > 建立複雜密碼時，您可以使用除 #、$ 或 % 以外的任何 ASCII 字元。

6. 選擇 Generate PEM Encoding (產生 PEM 編碼)。
7. 您可以將憑證、憑證鏈和加密金鑰複製到記憶體中，或為每個項目選擇 Export to a file (匯出到檔案)。
8. 選擇完成。

匯出私有憑證 (CLI)

使用 export-certificate 命令匯出私有憑證和私有金鑰。執行命令時，您必須指定複雜密碼。為了增加安全性，請使用檔案編輯器將您的複雜密碼存放在檔案中，然後透過提供檔案來提供複雜密碼。這可防止將密碼短語存放在命令歷史記錄中，並防止其他人在您輸入時看到密碼短語。

> **Note**
>
> 包含複雜密碼的檔案不得以行結束字元結尾。您可以依如下方式檢查您的密碼檔案：
>
> ```
> $ file -k passphrase.txt
> passphrase.txt: ASCII text, with no line terminators
> ```

以下範例使用管道將命令輸出至 jq，以套用 PEM 格式。

```
[Linux]
$ aws acm export-certificate \
     --certificate-arn arn:aws:acm:Region:444455556666:certificate/certificate_ID \
     --passphrase fileb://path-to-passphrase-file  \
     | jq -r '"\(.Certificate)\(.CertificateChain)\(.PrivateKey)"'

[Windows]
$ aws acm export-certificate \
     --certificate-arn arn:aws:acm:Region:444455556666:certificate/certificate_ID \
     --passphrase fileb://path-to-passphrase-file  \
     | jq -r '\"(.Certificate)(.CertificateChain)(.PrivateKey)\"'
```

這個輸出是 base64 編碼、PEM 格式的憑證，也包含憑證鏈和加密私有金鑰，如下列縮短的範例所示。

```
-----BEGIN CERTIFICATE-----
MIIDTDCCAjSgAwIBAgIRANWuFpqA16g3IwStE3vVpTwwDQYJKoZIhvcNAQELBQAw
EzERMA8GA1UECgwIdHJvbG9sb2wwHhcNMTkwNzE5MTYxNTU1WhcNMjAwODE5MTcx
NTU1WjAXMRUwEwYDVQQDDAx3d3cuc3B1ZHMuaW8wggEiMA0GCSqGSIb3DQEBAQUA
...
8UNFQvNoo1VtICL4cwWOdLOkxpwkkKWtcEkQuHE1v5Vn6HpbfFmxkdPEasoDhthH
FFWIf4/+VOlbDLgjU4HgtmV4IJDtqM9rGOZ42eFYmmc3eQO0GmigBBwwXp3j6hoi
74YM+igvtILnbYkPYhY9qz8h7lHUmannS8j6YxmtpPY=
-----END CERTIFICATE-----
```

```
-----BEGIN CERTIFICATE-----
MIIC8zCCAdugAwIBAgIRAM/jQ/6h2/MI1NYWX3dDaZswDQYJKoZIhvcNAQELBQAw
EzERMA8GA1UECgwIdHJvbG9sb2wwHhcNMTkwNjE5MTk0NTE2WhcNMjkwNjE5MjA0
NTE2WjATMREwDwYDVQQKDAh0cm9sb2xvbDCCASIwDQYJKoZIhvcNAQEBBQADggEP
...
j2PAOviqIXjwr08Zo/rTy/8m6LAsmm3LVVYKLyPdl+KB6M/+H93Z1/Bs8ERqqga/
6lfM6iw2JHtkW+q4WexvQSoqRXFhCZWbWPZTUpBS0d4/Y5q92S3iJLRa/JQ0d4U1
tWZyqJ2rj2RL+h7CE71XIAM//oHGcDDPaQBFD2DTisB/+ppGeDuB
-----END CERTIFICATE-----
-----BEGIN ENCRYPTED PRIVATE KEY-----
MIIFKzBVBgkqhkiG9w0BBQ0wSDAnBgkqhkiG9w0BBQwwGgQUMrZb7kZJ8nTZg7aB
1zmaQh4vwloCAggAMB0GCWCGSAFlAwQBKgQQDViroIHStQgNOjR6nTUnuwSCBNAN
JM4SG202YPUiddWeWmX/RKGg3lIdE+A0WLTPskNCdCAHqdhOSqBwt65qUTZe3gBt
...
ZGipF/DobHDMkpwiaRR5sz6nG4wcki0ryYjAQrdGsR6EVvUUXADkrnrrxuHTWjFl
wEuqyd8X/ApkQsYFX/nhepOEIGWf8Xu0nrjQo77/evhG0sHXborGzgCJwKuimPVy
Fs5kw5mvEoe5DAe3rSKsSUJ1tM4RagJj2WH+BC04SZWNH8kxfOC1E/GSLBCixv3v
+Lwq38CEJRQJLdpta8NcLKnFBwmmVs9OV/VXzNuHYg==
-----END ENCRYPTED PRIVATE KEY-----
```

若要輸出一切項目到檔案，請將 > 重定向器附加到上述範例中，以產生下列內容。

```
$ aws acm export-certificate \
     --certificate-arn arn:aws:acm:Region:444455556666:certificate/certificate_ID \
     --passphrase fileb://path-to-passphrase-file \
     | jq -r '"\(.Certificate)\(.CertificateChain)\(.PrivateKey)"' \
     > /tmp/export.txt
```

標記 AWS Certificate Manager 憑證

標籤是可以指派給 ACM 憑證的標記。每個標籤皆包含鍵與值。您可以使用 AWS Certificate Manager 主控台、AWS Command Line Interface (AWS CLI) 或 ACM API 來新增、檢視或移除 ACM 憑證的標籤。您可以選擇要在 ACM 主控台中顯示的標籤。

您可以建立符合需求的自訂標籤。例如，您可以使用 `Environment = Prod` 或 `Environment = Beta` 標籤來標記多個 ACM 憑證，以識別每個 ACM 憑證適用的環境。以下清單包含幾個其他自訂標籤的範例：

- `Admin = Alice`
- `Purpose = Website`
- `Protocol = TLS`
- `Registrar = Route53`

其他 AWS 資源也支援標記功能。因此，您可以將相同標籤指派至不同資源，以指出資源是否相關。例如，您可以指派 `Website = example.com` 等標籤至 ACM 憑證、負載平衡器以及用於 example.com 網站的其他資源。

主題

標籤限制

以下基本限制適用於 ACM 憑證標籤：

- 每個 ACM 憑證的標籤數上限為 50。
- 標籤金鑰的長度上限為 127 個字元。
- 標籤值的長度上限為 255 個字元。
- 標籤鍵與值皆區分大小寫。
- `aws:` 字首是保留供 AWS 使用，您無法新增、編輯或刪除索引鍵開頭為 `aws:` 的標籤。開頭為 `aws:` 的標籤不算在根據資源配額的標籤計數內。
- 若您計畫在多項服務和資源使用標記結構描述，請記住，其他服務可能有其他字元使用限制。請參閱文件以了解該服務。

- ACM 憑證標籤不可用於 AWS Management Console 的 Resource Groups and Tag Editor (資源群組和標籤編輯器)。

如需 AWS 標記慣例的一般資訊，請參閱標記 AWS 資源。

管理標籤

您可以使用 AWS 管理主控台、AWS Command Line Interface 或 AWS Certificate Manager API 新增、編輯及刪除標籤。

管理標籤 (主控台)

您可以使用 AWS Management Console 新增、刪除或編輯標籤。您也可以在欄顯中顯示標籤。

新增標籤

依照以下程序使用 ACM 主控台新增標籤。

將標籤新增至憑證 (主控台)

1. 前往 https://console.aws.amazon.com/acm/home 登入 AWS Management Console 並開啟 AWS Certificate Manager 主控台。
2. 在您要標記的憑證旁選擇箭頭。
3. 在詳細資訊窗格中，向下捲動至 Tags (標籤)。
4. 選擇 Edit (編輯)，然後選擇 Add Tag (新增標籤)。
5. 為標籤輸入金鑰和值。
6. 選擇 Save (儲存)。

刪除標籤

依照以下程序使用 ACM 主控台刪除標籤。

刪除標籤 (主控台)

1. 前往 https://console.aws.amazon.com/acm/home 登入 AWS Management Console 並開啟 AWS Certificate Manager 主控台。
2. 在具有您要刪除的標籤的憑證旁選擇箭頭。

3. 在詳細資訊窗格中，向下捲動至 Tags (標籤)。
4. 選擇 編輯 。
5. 在您要刪除的標籤旁，選擇 X。
6. 選擇 Save (儲存)。

編輯標籤

依照以下程序使用 ACM 主控台編輯標籤。

編輯標籤 (主控台)

1. 前往 https://console.aws.amazon.com/acm/home 登入 AWS Management Console 並開啟 AWS Certificate Manager 主控台。
2. 在您要編輯的憑證旁選擇箭頭。
3. 在詳細資訊窗格中，向下捲動至 Tags (標籤)。
4. 選擇 編輯 。
5. 修改您想要變更的標籤金鑰或值。
6. 選擇 Save (儲存)。

在欄中顯示標籤

依照以下程序，在 ACM 主控台以欄顯示標籤。

以欄顯示標籤 (主控台)

1. 前往 https://console.aws.amazon.com/acm/home 登入 AWS Management Console 並開啟 AWS Certificate Manager 主控台。
2. 透過選擇主控台右上角的齒輪圖示

   選擇您要以欄顯示的標籤。
3. 在想要以欄顯示的標籤旁，選取核取方塊。

管理標籤 (CLI)

請參閱下列主題，了解如何使用 AWS CLI 新增、列出及刪除標籤。

- add-tags-to-certificate
- list-tags-for-certificate
- remove-tags-from-certificate

管理標籤 (ACM API)

請參閱下列主題，了解如何使用 API 新增、列出及刪除標籤。

- AddTagsToCertificate
- ListTagsForCertificate
- RemoveTagsFromCertificate

監控和記錄 AWS Certificate Manager

監控是維持 AWS 解決方案的可靠性、可用性和效能的 AWS Certificate Manager 重要組成部分。您應該從 AWS 解決方案的所有部分收集監視資料，以便在發生多點失敗時更輕鬆地偵錯。

下列主題說明可與 ACM 搭配使用的 AWS 雲端監控工具。

主題

使用 Amazon EventBridge

您可以使用 Amazon EventBridge (先前稱為 E CloudWatch vents) 自動化 AWS 服務，並自動回應系統事件，例如應用程式可用性問題或資源變更。包括 ACM EventBridge 在內的 AWS 服務活動會以近乎即時的方式傳送至 Amazon。您可以使用事件觸發目標，包括 AWS Lambda 功能、任 AWS Batch 務、Amazon SNS 主題和許多其他目標。有關更多信息，請參閱什麼是 Amazon EventBridge？

主題

對 ACM 的 Amazon EventBridge 支持

本主題列出並說明 Amazon EventBridge 支援的 ACM 相關事件。

ACM Certificate Approaching Expiration (ACM 憑證即將到期) 事件

ACM 會從過期前 45 天開始，每天傳送所有作用中憑證 (公有、私有和匯入) 的過期事件。您可以使用 ACM API 的PutAccountConfiguration動作來變更此時間。

ACM 會自動啟動其發行的合格憑證續約，但匯入的憑證必須在到期前重新發行並重新匯入，以避免中斷。如需詳細資訊，請參閱重新匯入憑證。您可以使用過期事件來設定自動化以將憑證重新匯入 ACM。如需使用自動化的範例 AWS Lambda，請參閱 EventBridge 在 ACM 中使用 Amazon 觸發動作。

ACM Certificate Approaching Expiration (ACM 憑證即將到期) 事件的結構如下。

```
{
  "version": "0",
  "id": "id",
  "detail-type": "ACM Certificate Approaching Expiration",
  "source": "aws.acm",
  "account": "account",
  "time": "2020-09-30T06:51:08Z",
  "region": "region",
  "resources": [
    "arn:aws:acm:region:account:certificate/certificate_ID"
  ],
  "detail": {
    "DaysToExpiry": 31,
    "CommonName": "example.com"
  }
}
```

ACM Certificate Expired (ACM 憑證已過期) 事件

> **Note**
>
> 憑證過期事件不適用於匯入的憑證。

客戶可以接聽此事件，以在其帳戶中的 ACM 已核發的公有或私有憑證到期時收到提醒。

ACM Certificate Expired (ACM 憑證已過期) 事件的結構如下。

```
{
    "version": "0",
    "id": "id",
    "detail-type": "ACM Certificate Expired",
    "source": "aws.acm",
    "account": "account",
    "time": "2019-12-22T18:43:48Z",
    "region": "region",
    "resources": [
        "arn:aws:acm:region:account:certificate/certificate_ID"
     ],
     "detail": {
```

```
        "CertificateType" : "AMAZON_ISSUED" | "PRIVATE",
        "CommonName": "example.com",
        "DomainValidationMethod" : "EMAIL" | "DNS",
        "CertificateCreatedDate" : "2018-12-22T18:43:48Z",
        "CertificateExpirationDate" : "2019-12-22T18:43:48Z",
        "InUse" : TRUE | FALSE,
        "Exported" : TRUE | FALSE
    }
 }
```

ACM Certificate Available (ACM 憑證可用) 事件

客戶可以接聽此事件，以便在受管理的公有或私有憑證可供使用時收到通知。事件會在憑證發行、續約和匯入時發佈。若為私有憑證，一旦可用，仍需要客戶動作才能將其部署至主機。

ACM Certificate Available (ACM 憑證可用) 事件的結構如下。

```
{
    "version": "0",
    "id": "id",
    "detail-type": "ACM Certificate Available",
    "source": "aws.acm",
    "account": "account",
    "time": "2019-12-22T18:43:48Z",
    "region": "region",
    "resources": [
        "arn:aws:acm:region:account:certificate/certificate_ID"
    ],
    "detail": {
       “Action” : "ISSUANCE" | "RENEWAL" | "IMPORT" | "REIMPORT",
       "CertificateType" : "AMAZON_ISSUED" | "PRIVATE" | "IMPORTED",
       "CommonName": "example.com",
       "DomainValidationMethod" : "EMAIL" | "DNS",
       "CertificateCreatedDate" : "2019-12-22T18:43:48Z",
       "CertificateExpirationDate" : "2019-12-22T18:43:48Z",
       "DaysToExpiry" : 395,
       "InUse" : TRUE | FALSE,
       "Exported" : TRUE | FALSE
     }
}
```

ACM Certificate Renewal Action Required (需要 ACM 憑證續約動作) 事件

> **Note**
>
> 憑證續訂動作必要事件不適用於匯入的憑證。

客戶可以接聽此事件，以便在必須採取客戶動作後才能續約憑證時收到警示。例如，若客戶新增了阻止 ACM 續約憑證的 CAA 記錄，則 ACM 會在到期前 45 天自動續約失敗時發佈此事件。若未採取任何客戶動作，ACM 會在 30 天、15 天、3 天和 1 天時進行進一步的續約嘗試，或者直到採取客戶行動、憑證過期或憑證不再符合續約資格為止。這些續約嘗試均會發佈一個事件。

ACM Certificate Renewal Action Required (需要 ACM 憑證續約動作) 事件的結構如下。

```
{
   "version": "0",
   "id": "id",
   "detail-type": "ACM Certificate Renewal Action Required",
   "source": "aws.acm",
   "account": "account",
   "time": "2019-12-22T18:43:48Z",
   "region": "region",
   "resources": [
       "arn:aws:acm:region:account:certificate/certificate_ID"
    ],
    "detail": {
       "CertificateType" : "AMAZON_ISSUED" | "PRIVATE",
       "CommonName": "example.com",
       "DomainValidationMethod" : "EMAIL" | "DNS",
       "RenewalStatusReason" : "CAA_ERROR" | "PENDING_DOMAIN_VALIDATION" |
 "NO_AVAILABLE_CONTACTS" | "ADDITIONAL_VERIFICATION_REQUIRED" | "DOMAIN_NOT_ALLOWED"
 | "INVALID_PUBLIC_DOMAIN" | "DOMAIN_VALIDATION_DENIED" | "PCA_LIMIT_EXCEEDED"
 | "PCA_INVALID_ARN" | "PCA_INVALID_STATE" | "PCA_REQUEST_FAILED" |
 "PCA_NAME_CONSTRAINTS_VALIDATION" | "PCA_RESOURCE_NOT_FOUND" | "PCA_INVALID_ARGS" |
 "PCA_INVALID_DURATION" | "PCA_ACCESS_DENIED" | "SLR_NOT_FOUND" | "OTHER",
       "DaysToExpiry": 30,
       "CertificateExpirationDate" : "2019-12-22T18:43:48Z",
       "InUse" : TRUE | FALSE,
       "Exported" : TRUE | FALSE
   }
}
```

AWS 健康事件

AWS 系統會針對符合續約資格的 ACM 憑證產生健全狀況事件。如需有關續約資格的資訊，請參閱 ACM憑證的受管理續約。

運作狀態事件會在兩種情況下產生：

- 順利續約公有或私有憑證時。
- 客戶必須採取動作才能進行續約時。這可能表示點選電子郵件中的連結 (針對經過電子郵件驗證的憑證)，或者解決錯誤。每個事件都包含下列其中一個事件代碼。代碼會顯示為可用於篩選的變數。
  - AWS_ACM_RENEWAL_STATE_CHANGE (憑證已續約、已過期或即將過期)
  - CAA_CHECK_FAILURE (CAA 檢查失敗)
  - AWS_ACM_RENEWAL_FAILURE (由私有 CA 簽署的憑證)

運作狀態事件的結構如下。在此範例中，已產生 AWS_ACM_RENEWAL_STATE_CHANGE 事件。

```
{
   "source":[
      "aws.health"
   ],
   "detail-type":[
      "AWS Health Event"
   ],
   "detail":{
      "service":[
         "ACM"
      ],
      "eventTypeCategory":[
         "scheduledChange"
      ],
      "eventTypeCode":[
         "AWS_ACM_RENEWAL_STATE_CHANGE"
      ]
   }
}
```

EventBridge 在 ACM 中使用 Amazon 觸發動作

您可以根據這些事件建立 Amazon EventBridge 規則，並使用 Amazon EventBridge 主控台設定偵測到事件時發生的動作。本節提供設定 Amazon EventBridge 規則和產生動作的範例程序。

主題

使用 Amazon SNS 回應事件

本節說明如何設定 Amazon SNS 以便在 ACM 每次產生運作狀態事件時都傳送文字通知。

請完成下列程序來設定回應。

若要建立 Amazon EventBridge 規則並觸發動作

1. 創建一個 Amazon EventBridge 規則。如需詳細資訊，請參閱建立可回應事件的 Amazon EventBridge 規則。

   a. 在 Amazon 主 EventBridge 控台 https://console.aws.amazon.com/events/ 中，導覽至事件 > 規則頁面，然後選擇建立規則。

   b. 在 Create rule (建立規則) 頁面中，選擇 Event Pattern (事件模式)。

   c. 針對 Service Name (服務名稱)，從功能表選擇 Health (運作狀態)。

   d. 針對 Event Type (事件類型)，選擇 Specific Health events (特定運作狀態事件)。

   e. 選擇 Specific service(s) (特定服務)，然後從功能表中選擇 ACM。

   f. 選擇 Specific event type category(s) (特定事件類型類別)，然後選擇 accountNotification。

   g. 選擇 Any event type code (任何事件類型代碼)。

   h. 選擇 Any resource (任何資源)。

   i. 在 Event Pattern Preview (事件模式預覽) 編輯器中，貼上事件發出的 JSON 模式。這個範例會使用來自 AWS 健康事件 區段的模式。

```
{
   "source":[
      "aws.health"
   ],
   "detail-type":[
      "AWS Health Event"
   ],
   "detail":{
      "service":[
         "ACM"
```

```
      ],
      "eventTypeCategory":[
         "scheduledChange"
      ],
      "eventTypeCode":[
         "AWS_ACM_RENEWAL_STATE_CHANGE"
      ]
   }
}
```

2. 設定動作。

   在 Targets (目標) 區段中，您可以從許多能立即使用您事件的服務中進行選擇，例如 Amazon Simple Notification Service (SNS)，或者您可以選擇 Lambda 函數將事件傳遞給自訂的可執行程式碼。如需 AWS Lambda 實作的範例，請參閱「使用 Lambda 函數回應事件」。

使用 Lambda 函數回應事件

此程序示範如 AWS Lambda 何使用在 Amazon 上監聽 EventBridge、使用 Amazon 簡單通知服務 (SNS) 建立通知，以及如何將發現項目發佈到 AWS Security Hub，讓管理員和安全團隊能見度。

設定 Lambda 函數和 IAM 角色

1. 首先設定 AWS Identity and Access Management (IAM) 角色，然後定義 Lambda 函數所需的許可。此安全性最佳實務可讓您彈性地指定誰擁有呼叫函數的授權，以及限制授與該使用者的許可。不建議直接在用戶帳戶下運行大多數 AWS 操作，尤其不是在管理員帳戶下運行。

   前往 https://console.aws.amazon.com/iam/ 開啟 IAM 主控台。

2. 使用 JSON 政策編輯器來建立以下範本中定義的政策。提供您自己的區域和 AWS 帳戶詳細信息。如需詳細資訊，請參閱在 JSON 索引標籤上建立政策。

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"LambdaCertificateExpiryPolicy1",
         "Effect":"Allow",
         "Action":"logs:CreateLogGroup",
         "Resource":"arn:aws:logs:<region>:<AWS-ACCT-NUMBER>:*"
      },
      {
```

```
         "Sid":"LambdaCertificateExpiryPolicy2",
         "Effect":"Allow",
         "Action":[
            "logs:CreateLogStream",
            "logs:PutLogEvents"
         ],
         "Resource":[
            "arn:aws:logs:<region>:<AWS-ACCT-NUMBER>:log-group:/aws/lambda/handle-
expiring-certificates:*"
         ]
      },
      {
         "Sid":"LambdaCertificateExpiryPolicy3",
         "Effect":"Allow",
         "Action":[
            "acm:DescribeCertificate",
            "acm:GetCertificate",
            "acm:ListCertificates",
            "acm:ListTagsForCertificate"
         ],
         "Resource":"*"
      },
      {
         "Sid":"LambdaCertificateExpiryPolicy4",
         "Effect":"Allow",
         "Action":"SNS:Publish",
         "Resource":"*"
      },
      {
         "Sid":"LambdaCertificateExpiryPolicy5",
         "Effect":"Allow",
         "Action":[
            "SecurityHub:BatchImportFindings",
            "SecurityHub:BatchUpdateFindings",
            "SecurityHub:DescribeHub"
         ],
         "Resource":"*"
      },
      {
         "Sid":"LambdaCertificateExpiryPolicy6",
         "Effect":"Allow",
         "Action":"cloudwatch:ListMetrics",
         "Resource":"*"
      }
```

```
    ]
}
```

3. 建立 IAM 角色，並將新政策連接到該角色。有關建立 IAM 角色和附加政策的詳細資訊，請參閱為 AWS 服務建立角色 (主控台)。
4. 開啟主 AWS Lambda 控台，網址為 https://console.aws.amazon.com/lambda/。
5. 建立 Lambda 函數。如需詳細資訊，請參閱使用主控台建立 Lambda 函數。請完成下列步驟：

   a. 在 Create function (建立函數) 頁面上，選擇 Author from scratch (從頭開始撰寫) 選項來建立函數。

   b. 在「函數名稱handle-expiring-certificates」欄位中指定名稱，例如「」。

   c. 在 Runtime (執行時間) 清單中選擇 Python 3.8。

   d. 展開 Change default execution role (變更預設執行角色)，然後選擇 se an existing role (使用現有角色)。

   e. 從 Existing role (現有角色) 清單中選擇您稍早建立的角色。

   f. 選擇建立函數。

   g. 在 Function code (函數程式碼) 底下插入以下程式碼：

```
# Copyright 2021 Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: MIT-0
#
# Permission is hereby granted, free of charge, to any person obtaining a copy
 of this
# software and associated documentation files (the "Software"), to deal in the
 Software
# without restriction, including without limitation the rights to use, copy,
 modify,
# merge, publish, distribute, sublicense, and/or sell copies of the Software,
 and to
# permit persons to whom the Software is furnished to do so.
#
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 IMPLIED,
# INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
# PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
 COPYRIGHT
# HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN
 ACTION
# OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH
 THE
```

```
# SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

import json
import boto3
import os
from datetime import datetime, timedelta, timezone
# ----------------------------------------
# setup global data
# ----------------------------------------
utc = timezone.utc
# make today timezone aware
today = datetime.now().replace(tzinfo=utc)
# set up time window for alert - default to 45 if its missing
if os.environ.get('EXPIRY_DAYS') is None:
    expiry_days = 45
else:
    expiry_days = int(os.environ['EXPIRY_DAYS'])
expiry_window = today + timedelta(days = expiry_days)
def lambda_handler(event, context):
    # if this is coming from the ACM event, its for a single certificate
    if (event['detail-type'] == "ACM Certificate Approaching Expiration"):
        response = handle_single_cert(event, context.invoked_function_arn)
    return {
        'statusCode': 200,
        'body': response
    }
def handle_single_cert(event, context_arn):
    cert_client = boto3.client('acm')
    cert_details =
 cert_client.describe_certificate(CertificateArn=event['resources'][0])
    result = 'The following certificate is expiring within ' + str(expiry_days)
 + ' days: ' + cert_details['Certificate']['DomainName']
    # check the expiry window before logging to Security Hub and sending an SNS
    if cert_details['Certificate']['NotAfter'] < expiry_window:
        # This call is the text going into the SNS notification
        result = result + ' (' + cert_details['Certificate']['CertificateArn']
 + ') '
        # this call is publishing to SH
        result = result + ' - ' + log_finding_to_sh(event, cert_details,
 context_arn)
        # if there's an SNS topic, publish a notification to it
        if os.environ.get('SNS_TOPIC_ARN') is None:
            response = result
        else:
```

```
            sns_client = boto3.client('sns')
            response = sns_client.publish(TopicArn=os.environ['SNS_TOPIC_ARN'],
 Message=result, Subject='Certificate Expiration Notification')
    return result
def log_finding_to_sh(event, cert_details, context_arn):
    # setup for security hub
    sh_region = get_sh_region(event['region'])
    sh_hub_arn = "arn:aws:securityhub:{0}:{1}:hub/default".format(sh_region,
 event['account'])
    sh_product_arn = "arn:aws:securityhub:{0}:{1}:product/{1}/
default".format(sh_region, event['account'])
    # check if security hub is enabled, and if the hub arn exists
    sh_client = boto3.client('securityhub', region_name = sh_region)
    try:
        sh_enabled = sh_client.describe_hub(HubArn = sh_hub_arn)
    # the previous command throws an error indicating the hub doesn't exist or
 lambda doesn't have rights to it so we'll stop attempting to use it
    except Exception as error:
        sh_enabled = None
        print ('Default Security Hub product doesn\'t exist')
        response = 'Security Hub disabled'
    # This is used to generate the URL to the cert in the Security Hub Findings
 to link directly to it
    cert_id = right(cert_details['Certificate']['CertificateArn'], 36)
    if sh_enabled:
        # set up a new findings list
        new_findings = []
            # add expiring certificate to the new findings list
        new_findings.append({
            "SchemaVersion": "2018-10-08",
            "Id": cert_id,
            "ProductArn": sh_product_arn,
            "GeneratorId": context_arn,
            "AwsAccountId": event['account'],
            "Types": [
                "Software and Configuration Checks/AWS Config Analysis"
            ],
            "CreatedAt": event['time'],
            "UpdatedAt": event['time'],
            "Severity": {
                "Original": '89.0',
                "Label": 'HIGH'
            },
            "Title": 'Certificate expiration',
```

```
            "Description": 'cert expiry',
            'Remediation': {
                'Recommendation': {
                    'Text': 'A new certificate for ' +
 cert_details['Certificate']['DomainName'] + ' should be imported to replace
 the existing imported certificate before expiration',
                    'Url': "https://console.aws.amazon.com/acm/home?region=" +
 event['region'] + "#/?id=" + cert_id
                }
            },
            'Resources': [
                {
                    'Id': event['id'],
                    'Type': 'ACM Certificate',
                    'Partition': 'aws',
                    'Region': event['region']
                }
            ],
            'Compliance': {'Status': 'WARNING'}
        })
        # push any new findings to security hub
        if new_findings:
            try:
                response =
 sh_client.batch_import_findings(Findings=new_findings)
                if response['FailedCount'] > 0:
                    print("Failed to import {}
 findings".format(response['FailedCount']))
            except Exception as error:
                print("Error: ", error)
                raise
    return json.dumps(response)
# function to setup the sh region
def get_sh_region(event_region):
    # security hub findings may need to go to a different region so set that
 here
    if os.environ.get('SECURITY_HUB_REGION') is None:
        sh_region_local = event_region
    else:
        sh_region_local = os.environ['SECURITY_HUB_REGION']
    return sh_region_local
# quick function to trim off right side of a string
def right(value, count):
    # To get right part of string, use negative first index in slice.
```

```
    return value[-count:]
```

h. 在 Environment variables (環境變數) 底下，選擇 Edit (編輯) 並選擇性新增以下變數。

- (選用) EXPIRY_DAYS

  指定傳送憑證過期通知的前置時間 (以天為單位)。此函數預設值為 45 天，但您可以指定自訂值。

- (選用) SNS_TOPIC_ARN

  指定 Amazon SNS 的 ARN。用下列格式提供完整的 ARN：arn:aws:sns:*<region>*:*<account-number>*:*<topic-name>*。

- (選用) SECURITY_HUB_REGION

  在不同的區域 AWS Security Hub 中指定一個。如果沒有指定，便會使用執行中 Lambda 函數使用的區域。如果函數在多個區域中執行，則可能需要將所有憑證訊息移至單一區域中的 Security Hub。

i. 在 Basic settings (基本設定) 下，將 Timeout (逾時) 設為 30 秒。

j. 請在頁面頂端選擇 Deploy (部署)。

完成下列程序中的任務，以開始使用此解決方案。

自動執行電子郵件過期通知程序

在此範例中，我們會在透過 Amazon EventBridge 引發活動時，針對每個即將到期的憑證提供單一電子郵件。根據預設，ACM 每天會針對過期前 45 天或以下天數的憑證引發事件。(可以使用 ACM API 的PutAccountConfiguration作業來自訂此期間。) 這些事件都會觸發下列串聯的自動化動作：

```
ACM raises Amazon EventBridge event #
>>>>>>> events

          Event matches Amazon EventBridge rule #

                    Rule calls Lambda function #

                              Function sends SNS email and logs a Finding in Security
 Hub
```

1. 建立 Lambda 函數並設定許可。(已完成 - 請參閱「設定 Lambda 函數和 IAM 角色」)。

2. 為 Lambda 函數建立標準 SNS 主題，用來傳出通知。如需詳細資訊，請參閱建立 Amazon SNS 主題。
3. 任何對訂閱新 SNS 主題感興趣的人。如需詳細資訊，請參閱訂閱 Amazon SNS 主題。
4. 創建一個 Amazon EventBridge 規則以觸發 Lambda 函數。如需詳細資訊，請參閱建立可回應事件的 Amazon EventBridge 規則。

   在 Amazon 主 EventBridge 控台 https://console.aws.amazon.com/events/ 中，導覽至事件 > 規則頁面，然後選擇建立規則。指定 Service Name (服務名稱)、Event Type (事件類型)以及 Lambda function (Lambda 函數)。在 Event Pattern preview (事件模式預覽) 編輯器中，貼上以下程式碼：

```
{
  "source": [
    "aws.acm"
  ],
  "detail-type": [
    "ACM Certificate Approaching Expiration"
  ]
}
```

   事件 (例如 Lambda 接收的事件) 會顯示在 Show sample event(s) (顯示範例事件) 底下：

```
{
  "version": "0",
  "id": "9c95e8e4-96a4-ef3f-b739-b6aa5b193afb",
  "detail-type": "ACM Certificate Approaching Expiration",
  "source": "aws.acm",
  "account": "123456789012",
  "time": "2020-09-30T06:51:08Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:acm:us-east-1:123456789012:certificate/61f50cd4-45b9-4259-b049-
d0a53682fa4b"
  ],
  "detail": {
    "DaysToExpiry": 31,
    "CommonName": "My Awesome Service"
  }
}
```

清理方式

一旦您不再需要範例組態或任何組態，最佳實務是移除該組態的所有軌跡，避免安全問題和未來的非預期費用：

- IAM 政策及角色
- Lambda 函數
- CloudWatch 事件規則
- CloudWatch 與 Lambda 相關的記錄
- SNS 主題

CloudTrail 搭配使用 AWS Certificate Manager

AWS Certificate Manager 與提供 ACM 中使用者 AWS CloudTrail、角色或服務所採取之動作記錄的 AWS 服務整合。 CloudTrail 您的 AWS 帳戶預設為啟用狀態。 CloudTrail 擷取 ACM 的 API 呼叫做為事件，包括來自 ACM 主控台的呼叫和 ACM API 作業的程式碼呼叫。如果您設定追蹤，您可以啟用持續交付 CloudTrail 事件到 Amazon S3 儲存貯體，包括 ACM 的事件。如果您未設定追蹤，您仍然可以在 [事件歷程記錄] 中檢視 CloudTrail 主控台中最近的事件。

使用收集的資訊 CloudTrail，您可以判斷向 ACM 提出的要求、提出要求的 IP 位址、提出要求的人員、提出要求的時間以及其他詳細資訊。如需詳細資訊，請參閱檢視具有事 CloudTrail 件記錄的事件。當 ACM 中發生受支援的事件活動時，該活動會與事件歷史記錄中的其他 AWS 服務 CloudTrail 事件一起記錄在事件中。您可以檢視、搜尋和下載 AWS 帳戶的最新事件。

此外，您還可以設定其他 AWS 服務，以進一步分析 CloudTrail 記錄中收集的事件資料並採取行動。

如需相關資訊 CloudTrail，請參閱下列文件：

- AWS CloudTrail 用戶指南。
- 建立追蹤的概觀
- CloudTrail 支援的服務與整合
- 設定的 Amazon SNS 通知 CloudTrail
- 從多個區域接收 CloudTrail 記錄檔並從多個帳戶接收 CloudTrail 記錄檔

主題

- 記錄支援 ACM API 動作 CloudTrail

- 記錄整合服務的 API 呼叫

記錄支援 ACM API 動作 CloudTrail

ACM 支援將下列動作記錄為記 CloudTrail 錄檔中的事件：

每一筆事件或日誌專案都會包含產生請求者的資訊。身分資訊可協助您判斷下列事項：

- 請求是否使用 AWS 帳戶根使用者 或 AWS Identity and Access Management (IAM) 使用者登入資料提出。
- 提出該請求時，是否使用了特定角色或聯合身分使用者的暫時安全憑證。
- 請求是否由其他 AWS 服務提出

如需詳細資訊，請參閱CloudTrail使 userIdentity 元素。

下列各節提供所支援之 API 操作的範例日誌。

- 將標籤新增到憑證 (AddTagsToCertificate)
- 刪除憑證 (DeleteCertificate)
- 描述憑證 (DescribeCertificate)
- 匯出憑證 (ExportCertificate)
- 匯入憑證 (ImportCertificate)
- 列出憑證 (ListCertificates)
- 列出憑證標籤 (ListTagsForCertificate)
- 從憑證移除標籤 (RemoveTagsFromCertificate)
- 請求憑證 (RequestCertificate)
- 重新傳送驗證電子郵件 (ResendValidationEmail)
- 擷取憑證 (GetCertificate)

將標籤新增到憑證 (AddTagsToCertificate)

下列 CloudTrail 範例顯示呼叫 AddTagsToCertificateAPI 的結果。

```
{

   "Records":[
```

```
      {
         "eventVersion":"1.04",
         "userIdentity":{
            "type":"IAMUser",
            "principalId":"AIDACKCEVSQ6C2EXAMPLE",
            "arn":"arn:aws:iam::123456789012:user/Alice",
            "accountId":"123456789012",
            "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
            "userName":"Alice"
         },
         "eventTime":"2016-04-06T13:53:53Z",
         "eventSource":"acm.amazonaws.com",
         "eventName":"AddTagsToCertificate",
         "awsRegion":"us-east-1",
         "sourceIPAddress":"192.0.2.0",
         "userAgent":"aws-cli/1.10.16",
         "requestParameters":{
            "tags":[
               {
                  "value":"Alice",
                  "key":"Admin"
               }
            ],
            "certificateArn":"arn:aws:acm:us-east-1:123456789012:certificate/
fedcba98-7654-3210-fedc-ba9876543210"
         },
         "responseElements":null,
         "requestID":"fedcba98-7654-3210-fedc-ba9876543210",
         "eventID":"fedcba98-7654-3210-fedc-ba9876543210",
         "eventType":"AwsApiCall",
         "recipientAccountId":"123456789012"
      }
   ]
}
```

刪除憑證 (DeleteCertificate)

下列 CloudTrail 範例顯示呼叫 DeleteCertificateAPI 的結果。

```
{

   "Records":[
      {
```

```
         "eventVersion":"1.04",
         "userIdentity":{
            "type":"IAMUser",
            "principalId":"AIDACKCEVSQ6C2EXAMPLE",
            "arn":"arn:aws:iam::123456789012:user/Alice",
            "accountId":"123456789012",
            "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
            "userName":"Alice"
         },
         "eventTime":"2016-03-18T00:00:26Z",
         "eventSource":"acm.amazonaws.com",
         "eventName":"DeleteCertificate",
         "awsRegion":"us-east-1",
         "sourceIPAddress":"192.0.2.0",
         "userAgent":"aws-cli/1.9.15",
         "requestParameters":{
            "certificateArn":"arn:aws:acm:us-east-1:123456789012:certificate/
fedcba98-7654-3210-fedc-ba9876543210"
         },
         "responseElements":null,
         "requestID":"01234567-89ab-cdef-0123-456789abcdef",
         "eventID":"01234567-89ab-cdef-0123-456789abcdef",
         "eventType":"AwsApiCall",
         "recipientAccountId":"123456789012"
      }
   ]
}
```

描述憑證 (DescribeCertificate)

下列 CloudTrail 範例顯示呼叫 DescribeCertificateAPI 的結果。

> **Note**
>
> DescribeCertificate作業的 CloudTrail 記錄不會顯示您指定之 ACM 憑證的相關資訊。您可以使用主控台、或 DescribeCertificateAPI 來檢視憑證的 AWS Command Line Interface相關資訊。

```
{
   "Records":[
      {
```

```
         "eventVersion":"1.04",
         "userIdentity":{
            "type":"IAMUser",
            "principalId":"AIDACKCEVSQ6C2EXAMPLE",
            "arn":"arn:aws:iam::123456789012:user/Alice",
            "accountId":"123456789012",
            "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
            "userName":"Alice"
         },
         "eventTime":"2016-03-18T00:00:42Z",
         "eventSource":"acm.amazonaws.com",
         "eventName":"DescribeCertificate",
         "awsRegion":"us-east-1",
         "sourceIPAddress":"192.0.2.0",
         "userAgent":"aws-cli/1.9.15",
         "requestParameters":{
            "certificateArn":"arn:aws:acm:us-east-1:123456789012:certificate/
fedcba98-7654-3210-fedc-ba9876543210"
         },
         "responseElements":null,
         "requestID":"fedcba98-7654-3210-fedc-ba9876543210",
         "eventID":"fedcba98-7654-3210-fedc-ba9876543210",
         "eventType":"AwsApiCall",
         "recipientAccountId":"123456789012"
      }
   ]
}
```

匯出憑證 (ExportCertificate)

下列 CloudTrail 範例顯示呼叫 ExportCertificateAPI 的結果。

```
{
   "Records":[
      {
         "version":"0",
         "id":"01234567-89ab-cdef-0123-456789abcdef",
         "detail-type":"AWS API Call via CloudTrail",
         "source":"aws.acm",
         "account":"123456789012",
         "time":"2018-05-24T15:28:11Z",
         "region":"us-east-1",
         "resources":[
```

```
        ],
        "detail":{
           "eventVersion":"1.04",
           "userIdentity":{
              "type":"Root",
              "principalId":"123456789012",
              "arn":"arn:aws:iam::123456789012:user/Alice",
              "accountId":"123456789012",
              "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
              "userName":"Alice"
           },
           "eventTime":"2018-05-24T15:28:11Z",
           "eventSource":"acm.amazonaws.com",
           "eventName":"ExportCertificate",
           "awsRegion":"us-east-1",
           "sourceIPAddress":"192.0.2.0",
           "userAgent":"aws-cli/1.15.4 Python/2.7.9 Windows/8 botocore/1.10.4",
           "requestParameters":{
              "passphrase":{
                 "hb":[
                    42,
                    42,
                    42,
                    42,
                    42,
                    42,
                    42,
                    42,
                    42,
                    42
                 ],
                 "offset":0,
                 "isReadOnly":false,
                 "bigEndian":true,
                 "nativeByteOrder":false,
                 "mark":-1,
                 "position":0,
                 "limit":10,
                 "capacity":10,
                 "address":0
              },
              "certificateArn":"arn:aws:acm:us-east-1:123456789012:certificate/
fedcba98-7654-3210-fedc-ba9876543210"
```

```
            },
            "responseElements":{
                "certificateChain":
                "-----BEGIN CERTIFICATE-----
                base64 certificate
                -----END CERTIFICATE-----
                -----BEGIN CERTIFICATE-----
                base64 certificate
                -----END CERTIFICATE-----",
                "privateKey":"**********",
                "certificate":
                "-----BEGIN CERTIFICATE-----
                base64 certificate
                -----END CERTIFICATE-----"
            },
            "requestID":"01234567-89ab-cdef-0123-456789abcdef",
            "eventID":"fedcba98-7654-3210-fedc-ba9876543210",
            "eventType":"AwsApiCall"
         }
      }
   ]
}
```

匯入憑證 (ImportCertificate)

下列範例顯示 CloudTrail 記錄 ACM ImportCertificateAPI 作業呼叫的記錄項目。

```
{
   "eventVersion":"1.04",
   "userIdentity":{
      "type":"IAMUser",
      "principalId":"AIDACKCEVSQ6C2EXAMPLE",
      "arn":"arn:aws:iam::111122223333:user/Alice",
      "accountId":"111122223333",
      "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
      "userName":"Alice"
   },
   "eventTime":"2016-10-04T16:01:30Z",
   "eventSource":"acm.amazonaws.com",
   "eventName":"ImportCertificate",
   "awsRegion":"ap-southeast-2",
   "sourceIPAddress":"54.240.193.129",
   "userAgent":"Coral/Netty",
```

```
   "requestParameters":{
      "privateKey":{
         "hb":[
            "byte",
            "byte",
            "byte",
            "..."
         ],
         "offset":0,
         "isReadOnly":false,
         "bigEndian":true,
         "nativeByteOrder":false,
         "mark":-1,
         "position":0,
         "limit":1674,
         "capacity":1674,
         "address":0
      },
      "certificateChain":{
         "hb":[
            "byte",
            "byte",
            "byte",
            "..."
         ],
         "offset":0,
         "isReadOnly":false,
         "bigEndian":true,
         "nativeByteOrder":false,
         "mark":-1,
         "position":0,
         "limit":2105,
         "capacity":2105,
         "address":0
      },
      "certificate":{
         "hb":[
            "byte",
            "byte",
            "byte",
            "..."
         ],
         "offset":0,
         "isReadOnly":false,
```

```
            "bigEndian":true,
            "nativeByteOrder":false,
            "mark":-1,
            "position":0,
            "limit":2503,
            "capacity":2503,
            "address":0
         }
      },
      "responseElements":{
         "certificateArn":"arn:aws:acm:ap-
southeast-2:111122223333:certificate/01234567-89ab-cdef-0123-456789abcdef"
      },
      "requestID":"01234567-89ab-cdef-0123-456789abcdef",
      "eventID":"01234567-89ab-cdef-0123-456789abcdef",
      "eventType":"AwsApiCall",
      "recipientAccountId":"111122223333"
}
```


列出憑證 (ListCertificates)

下列 CloudTrail 範例顯示呼叫 ListCertificatesAPI 的結果。

> **Note**
>
> ListCertificates作業的 CloudTrail 記錄檔不會顯示您的 ACM 憑證。您可以使用控制台、或 ListCertificatesAPI 來檢 AWS Command Line Interface視憑證清單。


```
{
   "Records":[
      {
         "eventVersion":"1.04",
         "userIdentity":{
            "type":"IAMUser",
            "principalId":"AIDACKCEVSQ6C2EXAMPLE",
            "arn":"arn:aws:iam::123456789012:user/Alice",
            "accountId":"123456789012",
            "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
            "userName":"Alice"
         },
         "eventTime":"2016-03-18T00:00:43Z",
```

```
         "eventSource":"acm.amazonaws.com",
         "eventName":"ListCertificates",
         "awsRegion":"us-east-1",
         "sourceIPAddress":"192.0.2.0",
         "userAgent":"aws-cli/1.9.15",
         "requestParameters":{
            "maxItems":1000,
            "certificateStatuses":[
               "ISSUED"
            ]
         },
         "responseElements":null,
         "requestID":"74c99844-ec9c-11e5-ac34-d1e4dfe1a11b",
         "eventID":"cdfe1051-88aa-4aa3-8c33-a325270bff21",
         "eventType":"AwsApiCall",
         "recipientAccountId":"123456789012"
      }
   ]
}
```


列出憑證標籤 (ListTagsForCertificate)

下列 CloudTrail 範例顯示呼叫 ListTagsForCertificateAPI 的結果。

> **Note**
>
> ListTagsForCertificate作業的 CloudTrail 記錄不會顯示您的標籤。您可以使用控制台、或 ListTagsForCertificateAPI 來檢 AWS Command Line Interface視標籤清單。


```
{
   "Records":[
      {
         "eventVersion":"1.04",
         "userIdentity":{
            "type":"IAMUser",
            "principalId":"AIDACKCEVSQ6C2EXAMPLE",
            "arn":"arn:aws:iam::123456789012:user/Alice",
            "accountId":"123456789012",
            "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
            "userName":"Alice"
         },
```

```
         "eventTime":"2016-04-06T13:30:11Z",
         "eventSource":"acm.amazonaws.com",
         "eventName":"ListTagsForCertificate",
         "awsRegion":"us-east-1",
         "sourceIPAddress":"192.0.2.0",
         "userAgent":"aws-cli/1.10.16",
         "requestParameters":{
            "certificateArn":"arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
         },
         "responseElements":null,
         "requestID":"b010767f-fbfb-11e5-b596-79e9a97a2544",
         "eventID":"32181be6-a4a0-48d3-8014-c0d972b5163b",
         "eventType":"AwsApiCall",
         "recipientAccountId":"123456789012"
      }
   ]
}
```

從憑證移除標籤 (RemoveTagsFromCertificate)

下列 CloudTrail 範例顯示呼叫 RemoveTagsFromCertificateAPI 的結果。

```
{
   "Records":[
      {
         "eventVersion":"1.04",
         "userIdentity":{
            "type":"IAMUser",
            "principalId":"AIDACKCEVSQ6C2EXAMPLE",
            "arn":"arn:aws:iam::123456789012:user/Alice",
            "accountId":"123456789012",
            "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
            "userName":"Alice"
         },
         "eventTime":"2016-04-06T14:10:01Z",
         "eventSource":"acm.amazonaws.com",
         "eventName":"RemoveTagsFromCertificate",
         "awsRegion":"us-east-1",
         "sourceIPAddress":"192.0.2.0",
         "userAgent":"aws-cli/1.10.16",
         "requestParameters":{
```

```
            "certificateArn":"arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012",
            "tags":[
               {
                  "value":"Bob",
                  "key":"Admin"
               }
            ]
         },
         "responseElements":null,
         "requestID":"40ded461-fc01-11e5-a747-85804766d6c9",
         "eventID":"0cfa142e-ef74-4b21-9515-47197780c424",
         "eventType":"AwsApiCall",
         "recipientAccountId":"123456789012"
      }
   ]
}
```

請求憑證 (RequestCertificate)

下列 CloudTrail 範例顯示呼叫 RequestCertificateAPI 的結果。

```
{
   "Records":[
      {
         "eventVersion":"1.04",
         "userIdentity":{
            "type":"IAMUser",
            "principalId":"AIDACKCEVSQ6C2EXAMPLE",
            "arn":"arn:aws:iam::123456789012:user/Alice",
            "accountId":"123456789012",
            "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
            "userName":"Alice"
         },
         "eventTime":"2016-03-18T00:00:49Z",
         "eventSource":"acm.amazonaws.com",
         "eventName":"RequestCertificate",
         "awsRegion":"us-east-1",
         "sourceIPAddress":"192.0.2.0",
         "userAgent":"aws-cli/1.9.15",
         "requestParameters":{
            "subjectAlternativeNames":[
               "example.net"
```

```
            ],
            "domainName":"example.com",
            "domainValidationOptions":[
               {
                  "domainName":"example.com",
                  "validationDomain":"example.com"
               },
               {
                  "domainName":"example.net",
                  "validationDomain":"example.net"
               }
            ],
            "idempotencyToken":"8186023d89681c3ad5"
         },
         "responseElements":{
            "certificateArn":"arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
         },
         "requestID":"77dacef3-ec9c-11e5-ac34-d1e4dfe1a11b",
         "eventID":"a4954cdb-8f38-44c7-8927-a38ad4be3ac8",
         "eventType":"AwsApiCall",
         "recipientAccountId":"123456789012"
      }
   ]
}
```

重新傳送驗證電子郵件 (ResendValidationEmail)

下列 CloudTrail 範例顯示呼叫 ResendValidationEmailAPI 的結果。

```
{
   "Records":[
      {
         "eventVersion":"1.04",
         "userIdentity":{
            "type":"IAMUser",
            "principalId":"AIDACKCEVSQ6C2EXAMPLE",
            "arn":"arn:aws:iam::123456789012:user/Alice",
            "accountId":"123456789012",
            "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
            "userName":"Alice"
         },
         "eventTime":"2016-03-17T23:58:25Z",
```

```
         "eventSource":"acm.amazonaws.com",
         "eventName":"ResendValidationEmail",
         "awsRegion":"us-east-1",
         "sourceIPAddress":"192.0.2.0",
         "userAgent":"aws-cli/1.9.15",
         "requestParameters":{
            "domain":"example.com",
            "certificateArn":"arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012",
            "validationDomain":"example.com"
         },
         "responseElements":null,
         "requestID":"23760b88-ec9c-11e5-b6f4-cb861a6f0a28",
         "eventID":"41c11b06-ca91-4c1c-8c61-af349ea8bab8",
         "eventType":"AwsApiCall",
         "recipientAccountId":"123456789012"
      }
   ]
}
```

擷取憑證 (GetCertificate)

下列 CloudTrail 範例顯示呼叫 GetCertificateAPI 的結果。

```
{

   "Records":[
      {
         "eventVersion":"1.04",
         "userIdentity":{
            "type":"IAMUser",
            "principalId":"AIDACKCEVSQ6C2EXAMPLE",
            "arn":"arn:aws:iam::123456789012:user/Alice",
            "accountId":"123456789012",
            "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
            "userName":"Alice"
         },
         "eventTime":"2016-03-18T00:00:41Z",
         "eventSource":"acm.amazonaws.com",
         "eventName":"GetCertificate",
         "awsRegion":"us-east-1",
         "sourceIPAddress":"192.0.2.0",
         "userAgent":"aws-cli/1.9.15",
```

```
         "requestParameters":{
            "certificateArn":"arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
         },
         "responseElements":{
            "certificateChain":

            "-----BEGIN CERTIFICATE-----
            Base64-encoded certificate chain
            -----END CERTIFICATE-----",
            "certificate":
            "-----BEGIN CERTIFICATE-----
            Base64-encoded certificate
            -----END CERTIFICATE-----"

         },
         "requestID":"744dd891-ec9c-11e5-ac34-d1e4dfe1a11b",
         "eventID":"7aa4f909-00dd-478a-9a00-b2709bcad2bb",
         "eventType":"AwsApiCall",
         "recipientAccountId":"123456789012"
      }
   ]
}
```

記錄整合服務的 API 呼叫

您可以使 CloudTrail 用稽核與 ACM 整合的服務所發出的 API 呼叫。若要取得有關使用的更多資訊 CloudTrail，請參閱AWS CloudTrail 使用指南。以下範例顯示可產生的日誌類型 (視佈建 ACM 憑證的 AWS 資源而定)。

主題

- 建立負載平衡器

建立負載平衡器

您可以使 CloudTrail 用稽核與 ACM 整合的服務所發出的 API 呼叫。若要取得有關使用的更多資訊 CloudTrail，請參閱AWS CloudTrail 使用指南。下列範例顯示根據您佈建 ACM 憑證的 AWS 資源而定，可產生的記錄類型。

主題

- 建立負載平衡器

- 透過負載平衡器註冊 Amazon EC2 執行個體
- 加密私有金鑰
- 解密私有金鑰

建立負載平衡器

以下範例顯示名為 Alice 的 IAM 使用者呼叫 `CreateLoadBalancer` 函數。負載平衡器的名稱為 `TestLinuxDefault`，而接聽程式是使用 ACM 憑證建立。


```
{

   "eventVersion":"1.03",
   "userIdentity":{
      "type":"IAMUser",
      "principalId":"AIDACKCEVSQ6C2EXAMPLE",
      "arn":"arn:aws:iam::111122223333:user/Alice",
      "accountId":"111122223333",
      "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
      "userName":"Alice"
   },
   "eventTime":"2016-01-01T21:10:36Z",
   "eventSource":"elasticloadbalancing.amazonaws.com",
   "eventName":"CreateLoadBalancer",
   "awsRegion":"us-east-1",
   "sourceIPAddress":"192.0.2.0/24",
   "userAgent":"aws-cli/1.9.15",
   "requestParameters":{
      "availabilityZones":[
         "us-east-1b"
      ],
      "loadBalancerName":"LinuxTest",
      "listeners":[
         {
            "sSLCertificateId":"arn:aws:acm:us-
east-1:111122223333:certificate/12345678-1234-1234-1234-123456789012",
            "protocol":"HTTPS",
            "loadBalancerPort":443,
            "instanceProtocol":"HTTP",
            "instancePort":80
         }
      ]
   },
```

```
   "responseElements":{
      "dNSName":"LinuxTest-1234567890.us-east-1.elb.amazonaws.com"
   },
   "requestID":"19669c3b-b0cc-11e5-85b2-57397210a2e5",
   "eventID":"5d6c00c9-a9b8-46ef-9f3b-4589f5be63f7",
   "eventType":"AwsApiCall",
   "recipientAccountId":"111122223333"
}
```

透過負載平衡器註冊 Amazon EC2 執行個體

當您將網站或應用程式佈建在 Amazon Elastic Compute Cloud (Amazon EC2) 執行個體上時，負載平衡器必須了解該執行個體。這可以透過 Elastic Load Balancing 主控台或 AWS Command Line Interface完成。下列範例顯示`RegisterInstancesWithLoadBalancer`對 AWS 帳戶 123456789012 名稱 LinuxTest 的負載平衡器呼叫。

```
{
   "eventVersion":"1.03",
   "userIdentity":{
      "type":"IAMUser",
      "principalId":"AIDACKCEVSQ6C2EXAMPLE",
      "arn":"arn:aws:iam::123456789012:user/ALice",
      "accountId":"123456789012",
      "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
      "userName":"Alice",
      "sessionContext":{
         "attributes":{
            "mfaAuthenticated":"false",
            "creationDate":"2016-01-01T19:35:52Z"
         }
      },
      "invokedBy":"signin.amazonaws.com"
   },
   "eventTime":"2016-01-01T21:11:45Z",
   "eventSource":"elasticloadbalancing.amazonaws.com",
   "eventName":"RegisterInstancesWithLoadBalancer",
   "awsRegion":"us-east-1",
   "sourceIPAddress":"192.0.2.0/24",
   "userAgent":"signin.amazonaws.com",
   "requestParameters":{
      "loadBalancerName":"LinuxTest",
      "instances":[
         {
```

```
            "instanceId":"i-c67f4e78"
         }
      ]
   },
   "responseElements":{
      "instances":[
         {
            "instanceId":"i-c67f4e78"
         }
      ]
   },
   "requestID":"438b07dc-b0cc-11e5-8afb-cda7ba020551",
   "eventID":"9f284ca6-cbe5-42a1-8251-4f0e6b5739d6",
   "eventType":"AwsApiCall",
   "recipientAccountId":"123456789012"
}
```

加密私有金鑰

以下範例顯示加密私有金鑰 (與 ACM 憑證相關聯) 的 Encrypt 呼叫。加密是在 AWS內執行。

```
{
   "Records":[
      {
         "eventVersion":"1.03",
         "userIdentity":{
            "type":"IAMUser",
            "principalId":"AIDACKCEVSQ6C2EXAMPLE",
            "arn":"arn:aws:iam::111122223333:user/acm",
            "accountId":"111122223333",
            "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
            "userName":"acm"
         },
         "eventTime":"2016-01-05T18:36:29Z",
         "eventSource":"kms.amazonaws.com",
         "eventName":"Encrypt",
         "awsRegion":"us-east-1",
         "sourceIPAddress":"AWS Internal",
         "userAgent":"aws-internal",
         "requestParameters":{
            "keyId":"arn:aws:kms:us-east-1:123456789012:alias/aws/acm",
            "encryptionContext":{
               "aws:acm:arn":"arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
```

```
            }
         },
         "responseElements":null,
         "requestID":"3c417351-b3db-11e5-9a24-7d9457362fcc",
         "eventID":"1794fe70-796a-45f5-811b-6584948f24ac",
         "readOnly":true,
         "resources":[
            {
               "ARN":"arn:aws:kms:us-
east-1:123456789012:key/87654321-4321-4321-4321-210987654321",
               "accountId":"123456789012"
            }
         ],
         "eventType":"AwsServiceEvent",
         "recipientAccountId":"123456789012"
      }
   ]
}
```

解密私有金鑰

以下範例顯示解密私有金鑰 (與 ACM 憑證相關聯) 的 Decrypt 呼叫。解密在內部執行 AWS，解密的密鑰永遠不會離開 AWS。

```
{
   "eventVersion":"1.03",
   "userIdentity":{
      "type":"AssumedRole",
      "principalId":"AIDACKCEVSQ6C2EXAMPLE:1aba0dc8b3a728d6998c234a99178eff",
      "arn":"arn:aws:sts::111122223333:assumed-role/
DecryptACMCertificate/1aba0dc8b3a728d6998c234a99178eff",
      "accountId":"111122223333",
      "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
      "sessionContext":{
         "attributes":{
            "mfaAuthenticated":"false",
            "creationDate":"2016-01-01T21:13:28Z"
         },
         "sessionIssuer":{
            "type":"Role",
            "principalId":"APKAEIBAERJR2EXAMPLE",
            "arn":"arn:aws:iam::111122223333:role/DecryptACMCertificate",
            "accountId":"111122223333",
```

```
            "userName":"DecryptACMCertificate"
         }
      }
   },
   "eventTime":"2016-01-01T21:13:28Z",
   "eventSource":"kms.amazonaws.com",
   "eventName":"Decrypt",
   "awsRegion":"us-east-1",
   "sourceIPAddress":"AWS Internal",
   "userAgent":"aws-internal/3",
   "requestParameters":{
      "encryptionContext":{
         "aws:elasticloadbalancing:arn":"arn:aws:elasticloadbalancing:us-
east-1:123456789012:loadbalancer/LinuxTest",
         "aws:acm:arn":"arn:aws:acm:us-
east-1:123456789012:certificate/87654321-4321-4321-4321-210987654321"
      }
   },
   "responseElements":null,
   "requestID":"809a70ff-b0cc-11e5-8f42-c7fdf1cb6e6a",
   "eventID":"7f89f7a7-baff-4802-8a88-851488607fb9",
   "readOnly":true,
   "resources":[
      {
         "ARN":"arn:aws:kms:us-
east-1:123456789012:key/12345678-1234-1234-1234-123456789012",
         "accountId":"123456789012"
      }
   ],
   "eventType":"AwsServiceEvent",
   "recipientAccountId":"123456789012"
}
```

支援的 CloudWatch 指標

Amazon CloudWatch 是 AWS 資源的監控服務。您可以用 CloudWatch 來收集和追蹤指標、設定警示，以及自動回應 AWS 資源中的變更。ACM 會針對帳戶中的每個憑證每天發佈一次指標，直到到期為止。

AWS/CertificateManager 命名空間包含下列指標。

指標	描述	單位	維度
DaysToExpiry	憑證到期前的天數。ACM 會在憑證過期後停止發佈此指標。	Integer	CertificateArn • 值：憑證的 ARN

如需 CloudWatch 測量結果的相關資訊，請參閱下列主題：

- 使用 Amazon CloudWatch 指標
- 創建 Amazon CloudWatch 警報

使用 API (Java 範例)

您可以使用 AWS Certificate Manager API，透過編寫程式的方式傳送 HTTP 請求，與服務互動。如需詳細資訊，請參閱 AWS Certificate Manager API 參考。

除了 Web API (或 HTTP API)，您還可以使用 AWS 軟體開發套件和命令列工具來與 ACM 和其他服務互動。如需詳細資訊，請參閱 Amazon Web Services 適用工具。

下列主題說明如何使用其中一個 AWS 開發套件 AWS SDK for Java 在 AWS Certificate Manager API 執行一些可用的操作。

主題

將標籤新增到憑證

以下範例說明如何使用 AddTagsToCertificate 函數。

```
package com.amazonaws.samples;

import java.io.IOException;
import java.nio.ByteBuffer;
import java.nio.charset.StandardCharsets;
import java.nio.file.Files;
import java.nio.file.Paths;
```

```
import com.amazonaws.auth.AWSStaticCredentialsProvider;
import com.amazonaws.auth.BasicAWSCredentials;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.certificatemanager.AWSCertificateManager;
import com.amazonaws.services.certificatemanager.AWSCertificateManagerClientBuilder;
import com.amazonaws.services.certificatemanager.model.ImportCertificateRequest;
import com.amazonaws.services.certificatemanager.model.ImportCertificateResult;
/**
 * This sample demonstrates how to use the ImportCertificate function in the AWS
 Certificate Manager
 * service.
 *
 * Input parameters:
 *   Accesskey - AWS access key
 *   SecretKey - AWS secret key
 *   CertificateArn - Use to reimport a certificate (not included in this example).
 *   region - AWS region
 *   Certificate - PEM file that contains the certificate to import. Ex: /data/certs/
servercert.pem
 *   CertificateChain - The certificate chain, not including the end-entity
 certificate.
 *   PrivateKey - The private key that matches the public key in the certificate.
 *
 * Output parameter:
 *   CertificcateArn - The ARN of the imported certificate.
 *
 */
public class AWSCertificateManagerSample {

    public static void main(String[] args) throws IOException {
     String accessKey = "";
     String secretKey = "";
     String certificateArn = null;
     Regions region = Regions.DEFAULT_REGION;
     String serverCertFilePath = "";
     String privateKeyFilePath = "";
     String caCertFilePath = "";

     ImportCertificateRequest req = new ImportCertificateRequest()
       .withCertificate(getCertContent(serverCertFilePath))
       .withPrivateKey(getCertContent(privateKeyFilePath))

  .withCertificateChain(getCertContent(caCertFilePath)).withCertificateArn(certificateArn);
```

```
    AWSCertificateManager client =
 AWSCertificateManagerClientBuilder.standard().withRegion(region)
       .withCredentials(new AWSStaticCredentialsProvider(new
 BasicAWSCredentials(accessKey, secretKey)))
       .build();
    ImportCertificateResult result = client.importCertificate(req);

    System.out.println(result.getCertificateArn());

    List<Tag> expectedTags =
 ImmutableList.of(Tag.builder().withKey("key").withValue("value").build());

    AddTagsToCertificateRequest addTagsToCertificateRequest =
 AddTagsToCertificateRequest.builder()
             .withCertificateArn(result.getCertificateArn())
             .withTags(tags)
             .build();

    client.addTagsToCertificate(addTagsToCertificateRequest);
   }

   private static ByteBuffer getCertContent(String filePath) throws IOException {
    String fileContent = new String(Files.readAllBytes(Paths.get(filePath)));
    return StandardCharsets.UTF_8.encode(fileContent);
   }
}
```

刪除憑證

以下範例說明如何使用 DeleteCertificate 函數。如果成功，該函數會傳回空集合 {}。

```
package com.amazonaws.samples;

import com.amazonaws.services.certificatemanager.AWSCertificateManagerClientBuilder;
import com.amazonaws.services.certificatemanager.AWSCertificateManager;
import com.amazonaws.services.certificatemanager.model.DeleteCertificateRequest;
import com.amazonaws.services.certificatemanager.model.DeleteCertificateResult;

import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.auth.AWSStaticCredentialsProvider;
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.regions.Regions;
```

```
import com.amazonaws.services.certificatemanager.model.InvalidArnException;
import com.amazonaws.services.certificatemanager.model.ResourceInUseException;
import com.amazonaws.services.certificatemanager.model.ResourceNotFoundException;
import com.amazonaws.AmazonClientException;

/**
 * This sample demonstrates how to use the DeleteCertificate function in the AWS
 Certificate
 * Manager service.
 *
 * Input parameter:
 *   CertificateArn - The ARN of the certificate to delete.
 *
 */

public class AWSCertificateManagerExample {

   public static void main(String[] args) throws Exception{

      // Retrieve your credentials from the C:\Users\name\.aws\credentials file in
 Windows
      // or the ~/.aws/credentials file in Linux.
      AWSCredentials credentials = null;
      try {
          credentials = new ProfileCredentialsProvider().getCredentials();
      }
      catch (Exception ex) {
          throw new AmazonClientException("Cannot load the credentials from file.",
 ex);
      }

      // Create a client.
      AWSCertificateManager client = AWSCertificateManagerClientBuilder.standard()
              .withRegion(Regions.US_EAST_1)
              .withCredentials(new AWSStaticCredentialsProvider(credentials))
              .build();

      // Create a request object and specify the ARN of the certificate to delete.
      DeleteCertificateRequest req = new DeleteCertificateRequest();

 req.setCertificateArn("arn:aws:acm:region:account:certificate/
12345678-1234-1234-1234-123456789012");
```

```
      // Delete the specified certificate.
      DeleteCertificateResult result = null;
      try {
         result = client.deleteCertificate(req);
      }
      catch (InvalidArnException ex)
      {
         throw ex;
      }
      catch (ResourceInUseException ex)
      {
         throw ex;
      }
      catch (ResourceNotFoundException ex)
      {
         throw ex;
      }

      // Display the result.
      System.out.println(result);

   }
}
```

描述憑證

以下範例說明如何使用 DescribeCertificate 函數。

```
package com.amazonaws.samples;

import com.amazonaws.services.certificatemanager.AWSCertificateManagerClientBuilder;
import com.amazonaws.services.certificatemanager.AWSCertificateManager;
import com.amazonaws.services.certificatemanager.model.DescribeCertificateRequest;
import com.amazonaws.services.certificatemanager.model.DescribeCertificateResult;

import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.auth.AWSStaticCredentialsProvider;
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.regions.Regions;

import com.amazonaws.services.certificatemanager.model.InvalidArnException;
import com.amazonaws.services.certificatemanager.model.ResourceNotFoundException;
import com.amazonaws.AmazonClientException;
```

```
/**
 * This sample demonstrates how to use the DescribeCertificate function in the AWS
 Certificate
 * Manager service.
 *
 * Input parameter:
 *   CertificateArn - The ARN of the certificate to be described.
 *
 * Output parameter:
 *   Certificate information
 *
 */

public class AWSCertificateManagerExample {

   public static void main(String[] args) throws Exception{

      // Retrieve your credentials from the C:\Users\name\.aws\credentials file in
 Windows
      // or the ~/.aws/credentials file in Linux.
      AWSCredentials credentials = null;
      try {
          credentials = new ProfileCredentialsProvider().getCredentials();
      }
      catch (Exception ex) {
          throw new AmazonClientException("Cannot load the credentials from file.",
 ex);
      }

      // Create a client.
      AWSCertificateManager client = AWSCertificateManagerClientBuilder.standard()
              .withRegion(Regions.US_EAST_1)
              .withCredentials(new AWSStaticCredentialsProvider(credentials))
              .build();

      // Create a request object and set the ARN of the certificate to be described.
      DescribeCertificateRequest req = new DescribeCertificateRequest();

 req.setCertificateArn("arn:aws:acm:region:account:certificate/
12345678-1234-1234-1234-123456789012");

      DescribeCertificateResult result = null;
      try{
```

```
            result = client.describeCertificate(req);
        }
        catch (InvalidArnException ex)
        {
            throw ex;
        }
        catch (ResourceNotFoundException ex)
        {
            throw ex;
        }

        // Display the certificate information.
        System.out.println(result);

    }
}
```

如果成功，上述範例會顯示類似以下內容的資訊。

```
{
    Certificate: {
        CertificateArn:
 arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012,
        DomainName: www.example.com,
        SubjectAlternativeNames: [www.example.com],
        DomainValidationOptions: [{
            DomainName: www.example.com,
        }],
        Serial: 10: 0a,
        Subject: C=US,
        ST=WA,
        L=Seattle,
        O=ExampleCompany,
        OU=sales,
        CN=www.example.com,
        Issuer: ExampleCompany,
        ImportedAt: FriOct0608: 17: 39PDT2017,
        Status: ISSUED,
        NotBefore: ThuOct0510: 14: 32PDT2017,
        NotAfter: SunOct0310: 14: 32PDT2027,
        KeyAlgorithm: RSA-2048,
        SignatureAlgorithm: SHA256WITHRSA,
        InUseBy: [],
```

```
        Type: IMPORTED,
    }
}
```

匯出憑證

以下範例說明如何使用 ExportCertificate 函數。此函數會匯出私有憑證授權機構 (CA) 發行的私有憑證 (使用 PKCS #8 格式)。(無論公有憑證是由 ACM 核發或匯出，都不可能匯出公有憑證。) 也會匯出憑證鏈和私密金鑰。在此範例中，金鑰的複雜密碼存放在本機檔案。

```
package com.amazonaws.samples;

import com.amazonaws.AmazonClientException;

import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.auth.AWSStaticCredentialsProvider;
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.regions.Regions;

import com.amazonaws.services.certificatemanager.AWSCertificateManagerClientBuilder;
import com.amazonaws.services.certificatemanager.AWSCertificateManager;


import com.amazonaws.services.certificatemanager.model.ExportCertificateRequest;
import com.amazonaws.services.certificatemanager.model.ExportCertificateResult;

import com.amazonaws.services.certificatemanager.model.InvalidArnException;
import com.amazonaws.services.certificatemanager.model.InvalidTagException;
import com.amazonaws.services.certificatemanager.model.ResourceNotFoundException;

import java.io.FileNotFoundException;
import java.io.IOException;
import java.io.RandomAccessFile;
import java.nio.ByteBuffer;
import java.nio.channels.FileChannel;

public class ExportCertificate {

   public static void main(String[] args) throws Exception {

      // Retrieve your credentials from the C:\Users\name\.aws\credentials file in
 Windows
```

```
      // or the ~/.aws/credentials in Linux.
      AWSCredentials credentials = null;
      try {
          credentials = new ProfileCredentialsProvider().getCredentials();
      }
      catch (Exception ex) {
          throw new AmazonClientException("Cannot load your credentials from file.",
ex);
      }

      // Create a client.
      AWSCertificateManager client = AWSCertificateManagerClientBuilder.standard()
              .withRegion(Regions.your_region)
              .withCredentials(new AWSStaticCredentialsProvider(credentials))
              .build();

      // Initialize a file descriptor for the passphrase file.
      RandomAccessFile file_passphrase = null;

      // Initialize a buffer for the passphrase.
      ByteBuffer buf_passphrase = null;

      // Create a file stream for reading the private key passphrase.
      try {
         file_passphrase = new RandomAccessFile("C:\\Temp\\password.txt", "r");
      }
      catch (IllegalArgumentException ex) {
         throw ex;
      }
      catch (SecurityException ex) {
         throw ex;
      }
      catch (FileNotFoundException ex) {
         throw ex;
      }

      // Create a channel to map the file.
      FileChannel channel_passphrase = file_passphrase.getChannel();

      // Map the file to the buffer.
      try {
         buf_passphrase = channel_passphrase.map(FileChannel.MapMode.READ_ONLY, 0,
channel_passphrase.size());
```

```
        // Clean up after the file is mapped.
        channel_passphrase.close();
        file_passphrase.close();
    }
    catch (IOException ex)
    {
        throw ex;
    }

    // Create a request object.
    ExportCertificateRequest req = new ExportCertificateRequest();

    // Set the certificate ARN.
    req.withCertificateArn("arn:aws:acm:region:account:"
          +"certificate/M12345678-1234-1234-1234-123456789012");

    // Set the passphrase.
    req.withPassphrase(buf_passphrase);

    // Export the certificate.
    ExportCertificateResult result = null;

    try {
        result = client.exportCertificate(req);
    }
    catch(InvalidArnException ex)
    {
        throw ex;
    }
    catch (InvalidTagException ex)
    {
        throw ex;
    }
    catch (ResourceNotFoundException ex)
    {
        throw ex;
    }

    // Clear the buffer.
    buf_passphrase.clear();

    // Display the certificate and certificate chain.
    String certificate = result.getCertificate();
    System.out.println(certificate);
```

```
        String certificate_chain = result.getCertificateChain();
        System.out.println(certificate_chain);

        // This example retrieves but does not display the private key.
        String private_key = result.getPrivateKey();
    }
}
```

擷取憑證和憑證鏈

以下範例說明如何使用 GetCertificate 函數。

```
package com.amazonaws.samples;

import com.amazonaws.regions.Regions;
import com.amazonaws.services.certificatemanager.AWSCertificateManagerClientBuilder;
import com.amazonaws.services.certificatemanager.AWSCertificateManager;
import com.amazonaws.services.certificatemanager.model.GetCertificateRequest;
import com.amazonaws.services.certificatemanager.model.GetCertificateResult;

import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.auth.AWSStaticCredentialsProvider;
import com.amazonaws.auth.AWSCredentials;

import com.amazonaws.services.certificatemanager.model.InvalidArnException;
import com.amazonaws.services.certificatemanager.model.ResourceNotFoundException;
import com.amazonaws.services.certificatemanager.model.RequestInProgressException;
import com.amazonaws.AmazonClientException;

/**
 * This sample demonstrates how to use the GetCertificate function in the AWS
 Certificate
 * Manager service.
 *
 * Input parameter:
 *   CertificateArn - The ARN of the certificate to retrieve.
 *
 * Output parameters:
 *   Certificate - A base64-encoded certificate in PEM format.
 *   CertificateChain - The base64-encoded certificate chain in PEM format.
 *
 */
```

```
public class AWSCertificateManagerExample {

   public static void main(String[] args) throws Exception{

      // Retrieve your credentials from the C:\Users\name\.aws\credentials file in
 Windows
      // or the ~/.aws/credentials file in Linux.
      AWSCredentials credentials = null;
      try {
          credentials = new ProfileCredentialsProvider().getCredentials();
      }
      catch (Exception ex) {
          throw new AmazonClientException("Cannot load the credentials from the
 credential profiles file.", ex);
      }

      // Create a client.
      AWSCertificateManager client = AWSCertificateManagerClientBuilder.standard()
              .withRegion(Regions.US_EAST_1)
              .withCredentials(new AWSStaticCredentialsProvider(credentials))
              .build();

      // Create a request object and set the ARN of the certificate to be described.
      GetCertificateRequest req = new GetCertificateRequest();

 req.setCertificateArn("arn:aws:acm:region:account:certificate/
12345678-1234-1234-1234-123456789012");

      // Retrieve the certificate and certificate chain.
      // If you recently requested the certificate, loop until it has been created.
      GetCertificateResult result = null;
      long totalTimeout = 120000l;
      long timeSlept = 0l;
      long sleepInterval = 10000l;
      while (result == null && timeSlept < totalTimeout) {
         try {
            result = client.getCertificate(req);
         }
         catch (RequestInProgressException ex) {
            Thread.sleep(sleepInterval);
         }
         catch (ResourceNotFoundException ex)
         {
```

```
            throw ex;
         }
         catch (InvalidArnException ex)
         {
            throw ex;
         }

         timeSlept += sleepInterval;
      }

      // Display the certificate information.
      System.out.println(result);
   }
}
```

上述範例會建立類似如下的輸出。

```
{Certificate: -----BEGIN CERTIFICATE-----
    base64-encoded certificate
-----END CERTIFICATE-----,
CertificateChain: -----BEGIN CERTIFICATE-----
    base64-encoded certificate chain
-----END CERTIFICATE-----
}
```

匯入憑證

以下範例說明如何使用 ImportCertificate 函數。

```
package com.amazonaws.samples;

import com.amazonaws.services.certificatemanager.AWSCertificateManagerClientBuilder;
import com.amazonaws.services.certificatemanager.AWSCertificateManager;

import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.auth.AWSStaticCredentialsProvider;
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.regions.Regions;

import com.amazonaws.services.certificatemanager.model.ImportCertificateRequest;
import com.amazonaws.services.certificatemanager.model.ImportCertificateResult;
import com.amazonaws.services.certificatemanager.model.LimitExceededException;
```

```
import com.amazonaws.services.certificatemanager.model.ResourceNotFoundException;
import com.amazonaws.AmazonClientException;
import java.io.FileNotFoundException;
import java.io.IOException;

import java.io.RandomAccessFile;
import java.nio.ByteBuffer;
import java.nio.channels.FileChannel;

/**
 * This sample demonstrates how to use the ImportCertificate function in the AWS
 Certificate Manager
 * service.
 *
 * Input parameters:
 *   Certificate - PEM file that contains the certificate to import.
 *   CertificateArn - Use to reimport a certificate (not included in this example).
 *   CertificateChain - The certificate chain, not including the end-entity
 certificate.
 *   PrivateKey - The private key that matches the public key in the certificate.
 *
 * Output parameter:
 *   CertificcateArn - The ARN of the imported certificate.
 *
 */
public class AWSCertificateManagerSample {

   public static void main(String[] args) throws Exception {

      // Retrieve your credentials from the C:\Users\name\.aws\credentials file in
 Windows
      // or the ~/.aws/credentials file in Linux.
      AWSCredentials credentials = null;
      try {
          credentials = new ProfileCredentialsProvider().getCredentials();
      }
      catch (Exception ex) {
          throw new AmazonClientException(
              "Cannot load the credentials from file.", ex);
      }

      // Create a client.
      AWSCertificateManager client = AWSCertificateManagerClientBuilder.standard()
              .withRegion(Regions.US_EAST_1)
```

```
                .withCredentials(new AWSStaticCredentialsProvider(credentials))
                .build();

        // Initialize the file descriptors.
        RandomAccessFile file_certificate = null;
        RandomAccessFile file_chain = null;
        RandomAccessFile file_key = null;

        // Initialize the buffers.
        ByteBuffer buf_certificate = null;
        ByteBuffer buf_chain = null;
        ByteBuffer buf_key = null;

        // Create the file streams for reading.
        try {
           file_certificate = new RandomAccessFile("C:\\Temp\\certificate.pem", "r");
           file_chain = new RandomAccessFile("C:\\Temp\\chain.pem", "r");
           file_key = new RandomAccessFile("C:\\Temp\\private_key.pem", "r");
        }
        catch (IllegalArgumentException ex) {
           throw ex;
        }
        catch (SecurityException ex) {
           throw ex;
        }
        catch (FileNotFoundException ex) {
           throw ex;
        }

        // Create channels for mapping the files.
        FileChannel channel_certificate = file_certificate.getChannel();
        FileChannel channel_chain = file_chain.getChannel();
        FileChannel channel_key = file_key.getChannel();

        // Map the files to buffers.
        try {
           buf_certificate = channel_certificate.map(FileChannel.MapMode.READ_ONLY, 0,
channel_certificate.size());
           buf_chain = channel_chain.map(FileChannel.MapMode.READ_ONLY, 0,
channel_chain.size());
           buf_key = channel_key.map(FileChannel.MapMode.READ_ONLY, 0,
channel_key.size());

           // The files have been mapped, so clean up.
```

```
            channel_certificate.close();
            channel_chain.close();
            channel_key.close();
            file_certificate.close();
            file_chain.close();
            file_key.close();
        }
        catch (IOException ex)
        {
            throw ex;
        }

        // Create a request object and set the parameters.
        ImportCertificateRequest req = new ImportCertificateRequest();
        req.setCertificate(buf_certificate);
        req.setCertificateChain(buf_chain);
        req.setPrivateKey(buf_key);

        // Import the certificate.
        ImportCertificateResult result = null;
        try {
            result = client.importCertificate(req);
        }
        catch(LimitExceededException ex)
        {
            throw ex;
        }
        catch (ResourceNotFoundException ex)
        {
            throw ex;
        }

        // Clear the buffers.
        buf_certificate.clear();
        buf_chain.clear();
        buf_key.clear();

        // Retrieve and display the certificate ARN.
        String arn = result.getCertificateArn();
        System.out.println(arn);
    }
}
```

列出憑證

以下範例說明如何使用 ListCertificates 函數。

```
package com.amazonaws.samples;

import com.amazonaws.services.certificatemanager.AWSCertificateManagerClientBuilder;
import com.amazonaws.services.certificatemanager.AWSCertificateManager;
import com.amazonaws.services.certificatemanager.model.ListCertificatesRequest;
import com.amazonaws.services.certificatemanager.model.ListCertificatesResult;

import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.auth.AWSStaticCredentialsProvider;
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.regions.Regions;

import com.amazonaws.AmazonClientException;

import java.util.Arrays;
import java.util.List;

/**
 * This sample demonstrates how to use the ListCertificates function in the AWS
 Certificate
 * Manager service.
 *
 * Input parameters:
 *   CertificateStatuses - An array of strings that contains the statuses to use for
 filtering.
 *   MaxItems - The maximum number of certificates to return in the response.
 *   NextToken - Use when paginating results.
 *
 * Output parameters:
 *   CertificateSummaryList - A list of certificates.
 *   NextToken - Use to show additional results when paginating a truncated list.
 *
 */

public class AWSCertificateManagerExample {

   public static void main(String[] args) throws Exception{
```

```
      // Retrieve your credentials from the C:\Users\name\.aws\credentials file in
 Windows
      // or the ~/.aws/credentials file in Linux.
      AWSCredentials credentials = null;
      try {
          credentials = new ProfileCredentialsProvider().getCredentials();
      }
      catch (Exception ex) {
          throw new AmazonClientException("Cannot load the credentials from file.",
 ex);
      }

      // Create a client.
      AWSCertificateManager client = AWSCertificateManagerClientBuilder.standard()
              .withRegion(Regions.US_EAST_1)
              .withCredentials(new AWSStaticCredentialsProvider(credentials))
              .build();

      // Create a request object and set the parameters.
      ListCertificatesRequest req = new ListCertificatesRequest();
      List<String> Statuses = Arrays.asList("ISSUED", "EXPIRED", "PENDING_VALIDATION",
 "FAILED");
      req.setCertificateStatuses(Statuses);
      req.setMaxItems(10);

      // Retrieve the list of certificates.
      ListCertificatesResult result = null;
      try {
         result = client.listCertificates(req);
      }
      catch (Exception ex)
      {
         throw ex;
      }

      // Display the certificate list.
      System.out.println(result);
   }
}
```

上述範例會建立類似如下的輸出。

```
{
```

```
    CertificateSummaryList: [{
        CertificateArn:
 arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012,
        DomainName: www.example1.com
    },
    {
        CertificateArn:
 arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012,
        DomainName: www.example2.com
    },
    {
        CertificateArn:
 arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012,
        DomainName: www.example3.com
    }]
}
```

續約憑證

以下範例說明如何使用 RenewCertificate 函數。此函數會續約由私有憑證授權機構 (CA) 發行並使用 ExportCertificate 函數匯出的私有憑證。目前，此函數只能續約匯出的私有憑證。若要使用 ACM 續約您的 AWS 私有 CA 憑證，您首先必須授與執行此程序的 ACM 服務主體許可。如需詳細資訊，請參閱指派憑證續約許可給 ACM。

```
package com.amazonaws.samples;

import com.amazonaws.AmazonClientException;

import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.auth.AWSStaticCredentialsProvider;
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.regions.Regions;

import com.amazonaws.services.certificatemanager.AWSCertificateManagerClientBuilder;
import com.amazonaws.services.certificatemanager.AWSCertificateManager;

import com.amazonaws.services.certificatemanager.model.RenewCertificateRequest;
import com.amazonaws.services.certificatemanager.model.RenewCertificateResult;


import com.amazonaws.services.certificatemanager.model.InvalidArnException;
import com.amazonaws.services.certificatemanager.model.ResourceNotFoundException;
```

```
import com.amazonaws.services.certificatemanager.model.ValidationException;

import java.io.FileNotFoundException;
import java.io.IOException;
import java.io.RandomAccessFile;
import java.nio.ByteBuffer;
import java.nio.channels.FileChannel;

public class RenewCertificate {

   public static void main(String[] args) throws Exception {

      // Retrieve your credentials from the C:\Users\name\.aws\credentials file in
 Windows
      // or the ~/.aws/credentials in Linux.
      AWSCredentials credentials = null;
      try {
          credentials = new ProfileCredentialsProvider().getCredentials();
      }
      catch (Exception ex) {
          throw new AmazonClientException("Cannot load your credentials from file.",
 ex);
      }

      // Create a client.
      AWSCertificateManager client = AWSCertificateManagerClientBuilder.standard()
              .withRegion(Regions.your_region)
              .withCredentials(new AWSStaticCredentialsProvider(credentials))
              .build();


      // Create a request object and specify the ARN of the certificate to renew.
      RenewCertificateRequest req = new RenewCertificateRequest();
      req.withCertificateArn("arn:aws:acm:region:account:"
            +"certificate/M12345678-1234-1234-1234-123456789012");


      // Renew the certificate.
      RenewCertificateResult result = null;
      try {
         result = client.renewCertificate(req);
      }
      catch(InvalidArnException ex)
      {
```

```
         throw ex;
      }
      catch (ResourceNotFoundException ex)
      {
         throw ex;
      }
      catch (ValidationException ex)
      {
         throw ex;
      }

      // Display the result.
     System.out.println(result);
   }
}
```

列出憑證標籤

以下範例說明如何使用 ListTagsForCertificate 函數。

```
package com.amazonaws.samples;

import com.amazonaws.services.certificatemanager.AWSCertificateManagerClientBuilder;
import com.amazonaws.services.certificatemanager.AWSCertificateManager;
import com.amazonaws.services.certificatemanager.model.ListTagsForCertificateRequest;
import com.amazonaws.services.certificatemanager.model.ListTagsForCertificateResult;

import com.amazonaws.services.certificatemanager.model.InvalidArnException;
import com.amazonaws.services.certificatemanager.model.ResourceNotFoundException;
import com.amazonaws.AmazonClientException;

import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.auth.AWSStaticCredentialsProvider;
import com.amazonaws.regions.Regions;


/**
 * This sample demonstrates how to use the ListTagsForCertificate function in the AWS
 Certificate
 * Manager service.
 *
 * Input parameter:
```

```
 *   CertificateArn - The ARN of the certificate whose tags you want to list.
 *
*/

public class AWSCertificateManagerExample {

   public static void main(String[] args) throws Exception{

      // Retrieve your credentials from the C:\Users\name\.aws\credentials file in
 Windows
      // or the ~/.aws/credentials file in Linux.
      AWSCredentials credentials = null;
      try {
          credentials = new ProfileCredentialsProvider().getCredentials();
      }
      catch (Exception ex) {
          throw new AmazonClientException("Cannot load your credentials from file.",
 ex);
      }

      // Create a client.
      AWSCertificateManager client = AWSCertificateManagerClientBuilder.standard()
              .withRegion(Regions.US_EAST_1)
              .withCredentials(new AWSStaticCredentialsProvider(credentials))
              .build();

      // Create a request object and specify the ARN of the certificate.
      ListTagsForCertificateRequest req = new ListTagsForCertificateRequest();

 req.setCertificateArn("arn:aws:acm:region:account:certificate/
12345678-1234-1234-1234-123456789012");

      // Create a result object.
      ListTagsForCertificateResult result = null;
      try {
         result = client.listTagsForCertificate(req);
      }
      catch(InvalidArnException ex) {
         throw ex;
      }
      catch(ResourceNotFoundException ex) {
         throw ex;
      }
```

```
      // Display the result.
      System.out.println(result);

   }
}
```

上述範例會建立類似如下的輸出。

```
{Tags: [{Key: Purpose,Value: Test}, {Key: Short_Name,Value: My_Cert}]}
```

從憑證移除標籤

以下範例說明如何使用 RemoveTagsFromCertificate 函數。

```
package com.amazonaws.samples;

import com.amazonaws.services.certificatemanager.AWSCertificateManagerClientBuilder;
import com.amazonaws.services.certificatemanager.AWSCertificateManager;
import
 com.amazonaws.services.certificatemanager.model.RemoveTagsFromCertificateRequest;
import com.amazonaws.services.certificatemanager.model.RemoveTagsFromCertificateResult;
import com.amazonaws.services.certificatemanager.model.Tag;

import com.amazonaws.services.certificatemanager.model.InvalidArnException;
import com.amazonaws.services.certificatemanager.model.InvalidTagException;
import com.amazonaws.services.certificatemanager.model.ResourceNotFoundException;
import com.amazonaws.AmazonClientException;

import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.auth.AWSStaticCredentialsProvider;
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.regions.Regions;

import java.util.ArrayList;

/**
 * This sample demonstrates how to use the RemoveTagsFromCertificate function in the
 AWS Certificate
 * Manager service.
 *
 * Input parameters:
```

```
 *   CertificateArn - The ARN of the certificate from which you want to remove one or
 more tags.
 *   Tags - A collection of key-value pairs that specify which tags to remove.
 *
*/

public class AWSCertificateManagerExample {

   public static void main(String[] args) throws Exception {

      // Retrieve your credentials from the C:\Users\name\.aws\credentials file in
 Windows
      // or the ~/.aws/credentials file in Linux.
      AWSCredentials credentials = null;
      try {
          credentials = new ProfileCredentialsProvider().getCredentials();
      }
      catch (Exception ex) {
          throw new AmazonClientException("Cannot load your credentials from file.",
 ex);
      }

      // Create a client.
      AWSCertificateManager client = AWSCertificateManagerClientBuilder.standard()
              .withRegion(Regions.US_EAST_1)
              .withCredentials(new AWSStaticCredentialsProvider(credentials))
              .build();

      // Specify the tags to remove.
      Tag tag1 = new Tag();
      tag1.setKey("Short_Name");
      tag1.setValue("My_Cert");

      Tag tag2 = new Tag()
            .withKey("Purpose")
            .withValue("Test");

      // Add the tags to a collection.
      ArrayList<Tag> tags = new ArrayList<Tag>();
      tags.add(tag1);
      tags.add(tag2);

      // Create a request object.
      RemoveTagsFromCertificateRequest req = new RemoveTagsFromCertificateRequest();
```

```
 req.setCertificateArn("arn:aws:acm:region:account:certificate/
12345678-1234-1234-1234-123456789012");
      req.setTags(tags);

      // Create a result object.
      RemoveTagsFromCertificateResult result = null;
      try {
         result = client.removeTagsFromCertificate(req);
      }
      catch(InvalidArnException ex)
      {
         throw ex;
      }
      catch(InvalidTagException ex)
      {
         throw ex;
      }
      catch(ResourceNotFoundException ex)
      {
         throw ex;
      }

      // Display the result.
      System.out.println(result);
   }
}
```

請求憑證

以下範例說明如何使用 RequestCertificate 函數。

```
package com.amazonaws.samples;

import com.amazonaws.services.certificatemanager.AWSCertificateManagerClientBuilder;
import com.amazonaws.services.certificatemanager.AWSCertificateManager;
import com.amazonaws.services.certificatemanager.model.RequestCertificateRequest;
import com.amazonaws.services.certificatemanager.model.RequestCertificateResult;

import
 com.amazonaws.services.certificatemanager.model.InvalidDomainValidationOptionsException;
import com.amazonaws.services.certificatemanager.model.LimitExceededException;
import com.amazonaws.AmazonClientException;
```

```
import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.auth.AWSStaticCredentialsProvider;
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.regions.Regions;

import java.util.ArrayList;

/**
 * This sample demonstrates how to use the RequestCertificate function in the AWS
 Certificate
 * Manager service.
 *
 * Input parameters:
 *   DomainName - FQDN of your site.
 *   DomainValidationOptions - Domain name for email validation.
 *   IdempotencyToken - Distinguishes between calls to RequestCertificate.
 *   SubjectAlternativeNames - Additional FQDNs for the subject alternative names
 extension.
 *
 * Output parameter:
 *   Certificate ARN - The Amazon Resource Name (ARN) of the certificate you requested.
 *
*/

public class AWSCertificateManagerExample {

   public static void main(String[] args) {

      // Retrieve your credentials from the C:\Users\name\.aws\credentials file in
 Windows
      // or the ~/.aws/credentials file in Linux.
      AWSCredentials credentials = null;
      try {
          credentials = new ProfileCredentialsProvider().getCredentials();
      }
      catch (Exception ex) {
          throw new AmazonClientException("Cannot load your credentials from file.",
 ex);
      }

      // Create a client.
      AWSCertificateManager client = AWSCertificateManagerClientBuilder.standard()
              .withRegion(Regions.US_EAST_1)
```

```
                .withCredentials(new AWSStaticCredentialsProvider(credentials))
                .build();

        // Specify a SAN.
        ArrayList<String> san = new ArrayList<String>();
        san.add("www.example.com");

        // Create a request object and set the input parameters.
        RequestCertificateRequest req = new RequestCertificateRequest();
        req.setDomainName("example.com");
        req.setIdempotencyToken("1Aq25pTy");
        req.setSubjectAlternativeNames(san);

        // Create a result object and display the certificate ARN.
        RequestCertificateResult result = null;
        try {
           result = client.requestCertificate(req);
        }
        catch(InvalidDomainValidationOptionsException ex)
        {
           throw ex;
        }
        catch(LimitExceededException ex)
        {
           throw ex;
        }

        // Display the ARN.
        System.out.println(result);

    }

}
```

上述範例會建立類似如下的輸出。

```
{CertificateArn:
 arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012}
```

重新傳送驗證電子郵件

以下範例顯示如何使用 ResendValidationEmail 函數。

```
package com.amazonaws.samples;

import com.amazonaws.services.certificatemanager.AWSCertificateManagerClientBuilder;
import com.amazonaws.services.certificatemanager.AWSCertificateManager;
import com.amazonaws.services.certificatemanager.model.ResendValidationEmailRequest;
import com.amazonaws.services.certificatemanager.model.ResendValidationEmailResult;

import
 com.amazonaws.services.certificatemanager.model.InvalidDomainValidationOptionsException;
import com.amazonaws.services.certificatemanager.model.ResourceNotFoundException;
import com.amazonaws.services.certificatemanager.model.InvalidStateException;
import com.amazonaws.services.certificatemanager.model.InvalidArnException;
import com.amazonaws.AmazonClientException;

import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.auth.AWSStaticCredentialsProvider;
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.regions.Regions;

/**
 * This sample demonstrates how to use the ResendValidationEmail function in the AWS
 Certificate
 * Manager service.
 *
 * Input parameters:
 *   CertificateArn - Amazon Resource Name (ARN) of the certificate request.
 *   Domain - FQDN in the certificate request.
 *   ValidationDomain - The base validation domain that is used to send email.
 *
*/

public class AWSCertificateManagerExample {

   public static void main(String[] args) {

      // Retrieve your credentials from the C:\Users\name\.aws\credentials file in
 Windows
      // or the ~/.aws/credentials file in Linux.
      AWSCredentials credentials = null;
      try {
          credentials = new ProfileCredentialsProvider().getCredentials();
      }
      catch (Exception ex) {
```

```
            throw new AmazonClientException("Cannot load your credentials from file.",
 ex);
      }

      // Create a client.
      AWSCertificateManager client = AWSCertificateManagerClientBuilder.standard()
              .withRegion(Regions.US_EAST_1)
              .withCredentials(new AWSStaticCredentialsProvider(credentials))
              .build();

      // Create a request object and set the input parameters.
      ResendValidationEmailRequest req = new ResendValidationEmailRequest();

 req.setCertificateArn("arn:aws:acm:region:account:certificate/
12345678-1234-1234-1234-123456789012");
      req.setDomain("gregpe.io");
      req.setValidationDomain("gregpe.io");

      // Create a result object.
      ResendValidationEmailResult result = null;
      try {
         result = client.resendValidationEmail(req);
      }
      catch(ResourceNotFoundException ex)
      {
         throw ex;
      }
      catch (InvalidStateException ex)
      {
         throw ex;
      }
      catch (InvalidArnException ex)
      {
         throw ex;
      }
      catch (InvalidDomainValidationOptionsException ex)
      {
         throw ex;
      }

      // Display the result.
      System.out.println(result.toString());

   }
```

```
}
```

上述範例會重新傳送您的驗證電子郵件並顯示空集合。

故障診斷

如果您在使用 AWS Certificate Manager時遇到問題，請參閱以下主題。

> **Note**
>
> 如果您在本節中沒有看到您的問題，建議您造訪 AWS 知識中心。

主題

憑證請求疑難排解

如果在要求ACM憑證時遇到問題，請參閱下列主題。

主題

憑證請求逾時

如果ACM憑證要求未在 72 小時內驗證，則憑證要求逾時。若要更正此情況，請開啟主控台，尋找憑證的記錄，按一下其核取方塊，選擇 Actions (動作)，然後選擇 Delete (刪除)。然後選擇 Actions (動作) 和 Request a certificate (請求憑證) 以重新開始。如需詳細資訊，請參閱 DNS驗證 或 電子郵件驗證 。如果可能，我們建議您使用DNS驗證。

憑證請求失敗

如果您的要求失敗，ACM而您收到下列其中一個錯誤訊息，請依照建議的步驟修正問題。您無法重新提交失敗的憑證請求 – 請在解決問題後，提交新的請求。

主題

錯誤訊息：沒有可用的聯絡人

您在要求憑證時選擇了電子郵件驗證，但ACM找不到用於驗證要求中一或多個網域名稱的電子郵件地址。若要更正此問題，可執行以下其中一項操作：

- 請確定您有註冊的有效電子郵件地址，WHOIS而且在憑證要求中執行標準WHOIS查詢網域名稱時，該地址是可見的。通常，您是透過網域註冊商執行此操作。
- 確定您的網域已設定成可接收電子郵件。您的網域名稱伺服器必須有郵件交換器記錄 (MX 記錄)，因此電ACM子郵件伺服器會知道傳送網域驗證電子郵件的位置。

只要完成前述的其中一項任務便足以更正此問題；您不需要同時執行這兩項任務。更正問題後，請求新的憑證。

如需如何確保您收到的網域驗證電子郵件的詳細資訊ACM，請參閱(選用) 為您的網域設定電子郵件或未收到驗證電子郵件。如果您遵循這些步驟執行並繼續收到 No Available Contacts (無可用聯絡人) 訊息，請將此情況回報給 AWS，以便我們可以進行調查。

錯誤訊息：需要其他驗證

ACM需要其他資訊才能處理此憑證要求。如果您的網域排名在 Alexa 前 1000 名網站，詐騙防護措施可能會發生此情況。為了提供此資訊，請使用支援中心聯絡 AWS Support。如果您沒有支援方案，請在討論區中張貼新ACM討論串。

> **Note**
>
> 您無法為 Amazon 擁有的網域名稱請求憑證，例如結尾為 amazonaws.com、cloudfront.net 或 elasticbeanstalk.com 的網域名稱。

錯誤訊息：無效的公有網域

憑證請求中的一個或多個網域名稱無效。通常，這是因為請求中的網域名稱不是有效的頂層網域。再次嘗試請求憑證，同時更正失敗請求中的任何拼字錯誤或錯別字，並確定請求中的所有網域名稱適用於有效的頂層網域。例如，您無法要求ACM憑證，例如 .invalidpublicdomain，因為「無效域名」不是有效的頂層網域。如果您繼續收到此失敗原因，請聯絡支援中心。如果您沒有支援方案，請在討論區中張貼新ACM討論串。

錯誤訊息：其他

通常，當憑證請求中的一個或多個網域名稱有輸入錯誤時，便會發生此失敗。再次嘗試請求憑證，同時更正失敗請求中的任何拼字錯誤或錯別字。如果您繼續收到此失敗訊息，請使用支援中心聯絡 AWS Support。如果您沒有支援方案，請在討論區中張貼新ACM討論串。

針對憑證驗證進行疑難排解

如果ACM憑證要求狀態為擱置驗證，表示要求正在等待您執行動作。如果您在提出申請時選擇電子郵件驗證，則您或授權代表必須回應驗證電子郵件訊息。這些郵件會傳送至所要求網域的註冊WHOIS聯絡人地址和其他一般電子郵件地址。如需詳細資訊，請參閱 電子郵件驗證。如果您選擇DNS驗證，則必須將為您ACM建立的CNAME記錄寫入資DNS料庫。如需詳細資訊，請參閱 DNS驗證。

> **Important**
>
> 您必須驗證自己擁有或控制憑證要求中包含的每個網域名稱。如果選擇電子郵件驗證，便會收到各網域的驗證電子郵件訊息。若沒有收到，請參閱「未收到驗證電子郵件」。如果您選擇DNS驗證，則必須為每個網域建立一CNAME筆記錄。

> **Note**
>
> 公有ACM憑證可以安裝在連接到硝基 Enclave 的 Amazon EC2 執行個體上，但無法安裝到其他 Amazon 執行個體。EC2如需在未連線至 Nitro Enclave 的 Amazon EC2 執行個體上設定獨立網頁伺服器的詳細資訊，請參閱教學課程：在 Amazon Linux 2 上安裝LAMP網路伺服器或教學：使用 Amazon Linux 安裝LAMP網路伺服器。AMI

我們建議您使用DNS驗證而不是電子郵件驗證。

如果您遇到驗證問題，請參閱下列主題。

主題

排解DNS驗證問題

如果您在驗證憑證時遇到問題，請參閱下列指南。DNS

DNS疑難排解的第一個步驟是使用下列工具檢查網域目前的狀態：

- dig - Linux、Windows
- nslookup - Linux、Windows
- whois - Linux、Windows

主題

提供商禁止的DNS下劃線

如果您的提DNS供商禁止CNAME值中的前導下劃線，則可以從ACM提供的值中刪除底線，並在沒有它的情況下驗證您的域。例如，`_x2.acm-validations.aws`可以將CNAME值變更`x2.acm-validations.aws`為以進行驗證。但是，CNAMEname 參數必須始終以前導下劃線開頭。

您可以使用下表右側中的任一值來驗證網域。

名稱	Type	Value
`_<random value>.example.com.`	CNAME	`_<random value>.acm-validations.aws.`

名稱	Type	Value
_<random value>.example.com.	CNAME	<random value>.acm-validations.aws.

由DNS提供者新增的預設後置週期

根據預設，某些提供DNS者會在您提供的CNAME值中新增一個後置週期。因此，自行新增句點會導致錯誤。例如，「<random_value>.acm-validations.aws.」會遭拒絕，而「<random_value>.acm-validations.aws」被接受。

DNS驗證 GoDaddy 失敗

DNS在 Godaddy 和其他註冊機構註冊的網域驗證可能會失敗，除非您修改提供的CNAMEACM值。以 example.com 作為域名，發行的CNAME記錄具有以下形式：

```
NAME: _ho9hv39800vb3examplew3vnewoib3u.example.com. VALUE:
 _cjhwou20vhu2exampleuw20vuyb2ovb9.j9s73ucn9vy.acm-validations.aws.
```

您可以在NAME欄位結尾截斷頂點網域 (包括句點) GoDaddy 來建立相容的CNAME記錄，如下所示：

```
NAME: _ho9hv39800vb3examplew3vnewoib3u VALUE:
 _cjhwou20vhu2exampleuw20vuyb2ovb9.j9s73ucn9vy.acm-validations.aws.
```

ACM控制台不顯示「在 Route 53 中創建記錄」按鈕

如果您選擇 Amazon Route 53 做為您的DNS供應商， AWS Certificate Manager 可以直接與其互動以驗證您的網域擁有權。在某些情況下，當您需要使用主控台的在 Route 53 中建立記錄按鈕時，該按鈕可能無法使用。如果發生這種情況，請檢查下列可能的原因。

- 您沒有使用 Route 53 作為您的DNS提供者。
- 您已通過不同的帳戶登錄ACM並使用 Route 53。
- 您缺乏在 Route 53 託管的區域中建立記錄的IAM權限。
- 您或別人已驗證過網域。
- 網域無法公開定址。

私有 (不信任) 網域上的 Route 53 驗證失敗

在DNS驗證期間，會CNAME在公開託管的區域中ACM搜尋。如果找不到，則系統會在 72 小時後逾時，顯示狀態為 Validation timed out (驗證逾時)。您無法使用它來託管私有網域的DNS記錄，包括 Amazon 私有託管區域中的資源、VPC私有中不受信任的網域以及自我簽署憑證。PKI

AWS 確實通過該AWS 私有 CA服務為公共不受信任的域提供支持。

驗證成功，但核發或續約失敗

如果憑證發行失敗並顯示「擱置驗證」(即使DNS是正確的)，請檢查發行是否未被憑證授權單位授權 (CAA) 記錄封鎖。如需詳細資訊，請參閱 (選擇性) 設定CAA記錄。

DNS伺服器上的驗證失敗 VPN

如果您在DNS伺服器上找到伺服器VPN，但無ACM法驗證憑證，請檢查伺服器是否可公開存取。使用ACMDNS驗證的公開憑證發行時，網域記錄必須可透過公用網際網路解析。

針對電子郵件驗證問題進行疑難排解

如果您無法使用電子郵件驗證憑證網域，請參閱下列指導方針。

主題

- 未收到驗證電子郵件
- 電子郵件傳送到子網域
- 隱藏的聯絡資訊
- 憑證續約
- WHOIS 調節
- 電子郵件驗證的用久性初始時間戳記
- 針對 .IO 頂層網域的問題進行疑難排解
- 我無法切換到DNS驗證

未收到驗證電子郵件

當您要求憑證ACM並選擇電子郵件驗證時，網域驗證電子郵件會傳送至中指定的三個連絡人地址WHOIS和五個一般管理地址。如需詳細資訊，請參閱 電子郵件驗證。如果您在接收驗證電子郵件時遇到問題，請檢閱以下建議。

尋找電子郵件的位置

驗證電子郵件會傳送至中列出的聯絡人地址，以WHOIS及網域的一般管理地址。除非擁有者在中也列為網域聯絡人，否則電子郵件不會傳送給 AWS 帳戶擁有者WHOIS。檢閱主ACM控台中顯示 (或從或傳回API) 的電子郵件地址清單，以判斷您應該在何處尋找驗證電子郵件。CLI若要查看清單，請在標記為 Validation not complete (驗證未完成) 的方塊中按一下網域名稱旁的圖示。

電子郵件標記為垃圾郵件

檢查您的垃圾郵件資料夾中是否有驗證電子郵件。

GMail自動排序電子郵件

如果您正在使用GMail，驗證電子郵件可能已自動排序到「更新」或「促銷」標籤中。

網域註冊商未顯示聯絡資訊或已啟用隱私權保護

在某些情況下，中的網域註冊人、技術人員和管理聯絡人WHOIS可能無法公開使用， AWS 因此無法與這些聯絡人聯絡。儘管並非所有註冊商都支持此選項，但您可以自行決定配置註冊商以列出您的電子郵件地址。WHOIS您可能需要在您網域的登錄中直接進行變更。在其他情況下，網域聯絡資訊可能會使用隱私權地址，例如透過 WhoisGuard 或提供的隱私權地址 PrivacyGuard。

向 Route 53 購買的網域已預設啟用隱私保護機制，而且您的電子郵件地址已映射至 `whoisprivacyservice.org`、`contact.gandi.net` 或 `identity-protect.org` 電子郵件地址。確定您的網域註冊商檔案上的註冊者電子郵件地址是最新的，以便傳送到這些隱蔽電子郵件地址的電子郵件可以轉送到您控制的電子郵件地址。

> **Note**
>
> 即使您選擇公開您的聯絡資訊，您透過 Route 53 購買的某些網域的隱私保護機制仍會啟用。例如，.ca 頂層網域的隱私保護機制無法由 Route 53 透過編寫程式的方式停用。您必須聯絡 AWS Support 中心並請求停用隱私權保護。

如果您的網域無法透過電子郵件聯絡資訊取得WHOIS，或是傳送給聯絡人資訊的電子郵件沒有聯絡到網域擁有者或授權代表，我們建議您將您的網域或子網域設定為接收寄至一或多個由預先管理員 @、管理員 @、webmaster@ 和 postmaster @ 所形成的一般管理位址的電子郵件。如需為網域設定電子郵件的詳細資訊，請參閱您電子郵件服務供應商的文件，並遵循 (選用) 為您的網域設定電子郵件的指示操作。如果您使用的是 Amazon WorkMail，請參閱 Amazon WorkMail 管理員指南中的與使用者合作。

提供 AWS 傳送驗證電子郵件的八個電子郵件地址中的至少一個，並確認您可以接收該地址的電子郵件後，您就可以透過以下方式要求憑證ACM。提出憑證請求後，請確定預期的電子郵件地址有顯示在 AWS Management Console中的電子郵件地址清單中。在憑證處於 Pending validation (待定驗證) 狀態時，您可以在標記為 Validation not complete (驗證未完成) 的方塊中按一下網域名稱旁的圖示，以展開清單進行檢視。您也可以在步驟 3：驗證ACM要求憑證精靈中檢視清單。列出的電子郵件地址

MX 記錄遺失或設定不正確

MX 記錄是網域名稱系統 (DNS) 資料庫中的資源記錄，可指定一或多個接受您網域之電子郵件訊息的郵件伺服器。如果您的 MX 記錄遺失或設定錯誤，電子郵件便無法傳送到電子郵件驗證中指定的五個常用系統管理地址中的任何一個。請修正 MX 記錄遺失或設定錯誤的問題，然後嘗試重新傳送電子郵件或重新請求憑證。

> **Note**
>
> 目前，我們建議您至少等待一小時，然後再嘗試重新傳送電子郵件或請求憑證。

> **Note**
>
> 若要略過要求 MX 記錄，您可以使用RequestCertificateAPI或要求憑證 AWS CLI 命令中的ValidationDomain選項來指定ACM傳送驗證電子郵件的網域名稱。如果您使用API或 AWS CLI，則 AWS 不會執行 MX 查詢。

聯絡支援中心

如果在檢閱前述指導方針後，您仍沒有收到網域驗證電子郵件，請造訪 AWS Support 中心並建立案例。如果您沒有支援合約，請在ACM討論區張貼訊息。

電子郵件傳送到子網域

如果您使用主控台並要求子網域名稱的憑證`sub.test.example.com`，例如，請ACM檢查是否有 MX 記錄。`sub.test.example.com`如果沒有，則檢查父系網域 `test.example.com`，依此類推，直到基礎網域 `example.com`。如果找到了 MX 記錄，搜尋便停止，而且驗證電子郵件會傳送到子網域的常用管理地址。例如，如果找到了 `test.example.com` 的 MX 記錄，電子郵件便會傳送到 admin@test.example.com、administrator@test.example.com 和電子郵件驗證中指定的其他管理地址。如果在任何子網域中都找不到 MX 記錄，電子郵件便會傳送到您最初為其請求憑證的子網域。有

關如何設置電子郵件以及如何ACM使DNS用和WHOIS數據庫的詳細討論，請參閱(選用) 為您的網域設定電子郵件。

您可以使用RequestCertificateAPI或要求憑證 AWS CLI 命令中的ValidationDomain選項來指定ACM傳送驗證電子郵件的網域名稱，而不是使用主控台。如果您使用API或 AWS CLI，則 AWS 不會執行 MX 查詢。

隱藏的聯絡資訊

當您嘗試建立新憑證時，會發生一個常見問題。部分註冊商允許您在WHOIS刊登物品中隱藏您的聯絡資料。另一些註冊商允許您使用隱私 (或代理) 地址取代您的真實電子郵件地址。這會使您無法在註冊的聯絡地址接收驗證電子郵件。

要接收郵件，請確保您的聯繫信息公開WHOIS，或者如果您的WHOIS列表顯示了隱私電子郵件地址，請確保將發送到隱私地址的電子郵件轉發到您的真實電子郵件地址。WHOIS設定完成後，只要您的憑證要求未逾時，您就可以選擇重新傳送驗證電子郵件。ACM執行新的 WHOIS /MX 查詢，並將驗證電子郵件傳送至您現在的公開連絡人地址。

憑證續約

如果您在申請新憑證時公開您的WHOIS資訊，然後再將您的資訊混淆，則當您嘗試續訂憑證時，將ACM無法擷取您註冊的聯絡地址。ACM將驗證電子郵件傳送至這些連絡人地址，以及使用 MX 記錄所形成的五個常見管理地址。若要解決此問題，請再次公開您的WHOIS資訊，然後重新傳送驗證電子郵件。ACM執行新的 WHOIS /MX 查詢，並將驗證電子郵件傳送至您現在的公開連絡人地址。

WHOIS 調節

即使您發送了多個驗證電子郵件請求，有時ACM也無法與WHOIS服務器聯繫。這個問題是外在於AWS. 也就是說， AWS 不會控制WHOIS伺服器，也無法防止WHOIS伺服器節流。如果您遇到這個問題，請在 AWS Support 中心建立案例以尋求解決方法。

電子郵件驗證的用久性初始時間戳記

憑證的第一個電子郵件驗證請求時間戳記會永久存在於之後的驗證續約請求中。這不是ACM操作錯誤的證據。

針對 .IO 頂層網域的問題進行疑難排解

.IO 頂層網域指派至英屬印度洋領土。目前，網域註冊處不會顯示資WHOIS料庫中的公開資訊。無論您啟用或停用網域的隱私權保護，都是如此。如果禁用隱私保護，註冊商可能會在自己的WHOIS輸出中

顯示此信息，但這種做法因註冊商而異。ACM如果無法從中的註冊商取得，則無法將驗證電子郵件傳送至下列三個已註冊的聯絡地址WHOIS。

- 網域註冊者
- 技術聯絡人
- 管理聯絡人

ACM但是，會將驗證電子郵件發送到以下五個常見的系統地址，其中 *your_domain* 是您最初要求憑證時輸入的網域名稱，而且.io是頂層網域。

- 管理員 @*your_domain*.io
- 主持人 @*your_domain*.io
- 郵政管理員 @*your_domain*.io
- 網站管理員 @*your_domain*.io
- 行政人員 @*your_domain*.io

若要收到 .IO 網域的驗證郵件，請確保您啟用上述其中一個電子郵件帳戶。如果不這樣做，您將不會收到驗證電子郵件，也不會收到ACM憑證。

> **Note**
>
> 我們建議您使用DNS驗證而不是電子郵件驗證。如需詳細資訊，請參閱 DNS驗證。

我無法切換到DNS驗證

建立具有電子郵件驗證的憑證之後，您無法切換為使DNS用驗證憑證。若要使用DNS驗證，請刪除憑證，然後建立使用驗證的新憑DNS證。

針對受管憑證續約進行疑難排解

ACM嘗試在憑證到期前自動更新ACM憑證，這樣您就不需要採取任何動作。如果您有 ACM憑證的受管理續約 的相關問題，請參閱下列主題。

準備自動網域驗證

下列條件必須成立，才能自動續訂憑證：ACM

- 您的憑證必須與整合的 AWS 服務相關聯ACM。若要取得有關ACM支援的資源的資訊，請參閱服務整合 AWS Certificate Manager。
- 對於經過電子郵件驗證的憑證，ACM必須能夠透過系統管理員電子郵件地址與您聯絡，以取得憑證中列出的每個網域。系統將嘗試的電子郵件地址會列於 電子郵件驗證 中。
- 對於DNS已驗證的憑證，請確定您的DNS組態包含正確的CNAME記錄，如中DNS驗證所述。

受管憑證續約處理失敗

由於憑證即將到期 (60 天DNS，私人為 45 天)EMAIL，如果憑證符合資格條件，則ACM嘗試更新憑證。您可能必須採取行動才能成功續約。如需詳細資訊，請參閱 ACM憑證的受管理續約。

經電子郵件驗證之憑證的受管憑證續約

ACM憑證的有效期為 13 個月 (395 天)。續訂憑證需要網域擁有者採取行動。ACM會在到期前 45 天開始傳送續約通知至與網域相關聯的電子郵件地址。通知包含網域擁有者可以按一下進行續約的連結。驗證所有列出的網域後，會以相同的方式ACM發行續約憑證ARN。

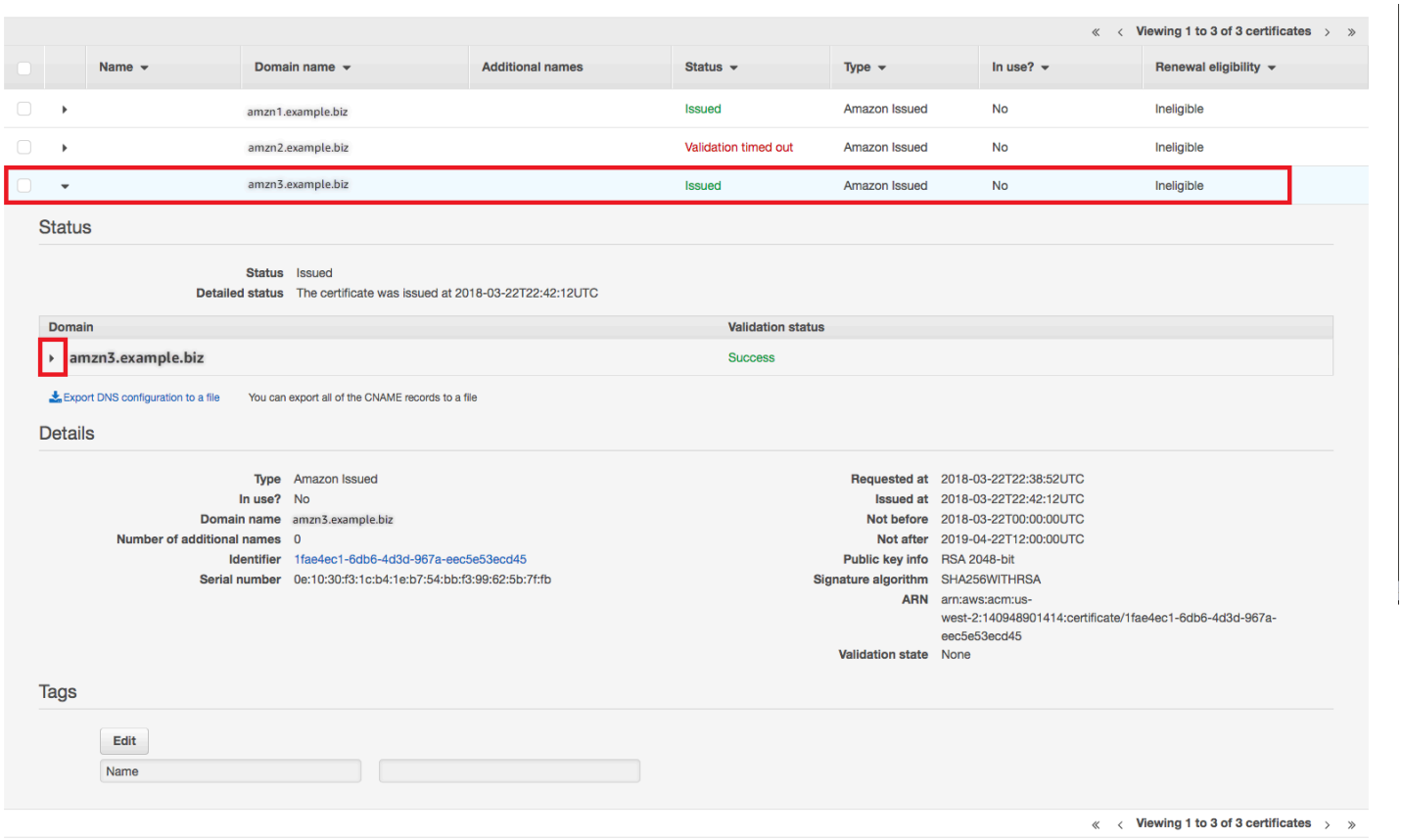
請參閱使用電子郵件驗證取得指引，了解如何識別哪些網域處於 `PENDING_VALIDATION` 狀態，並針對這些網域重複執行驗證程序。

DNS已驗證憑證的受管憑證續約

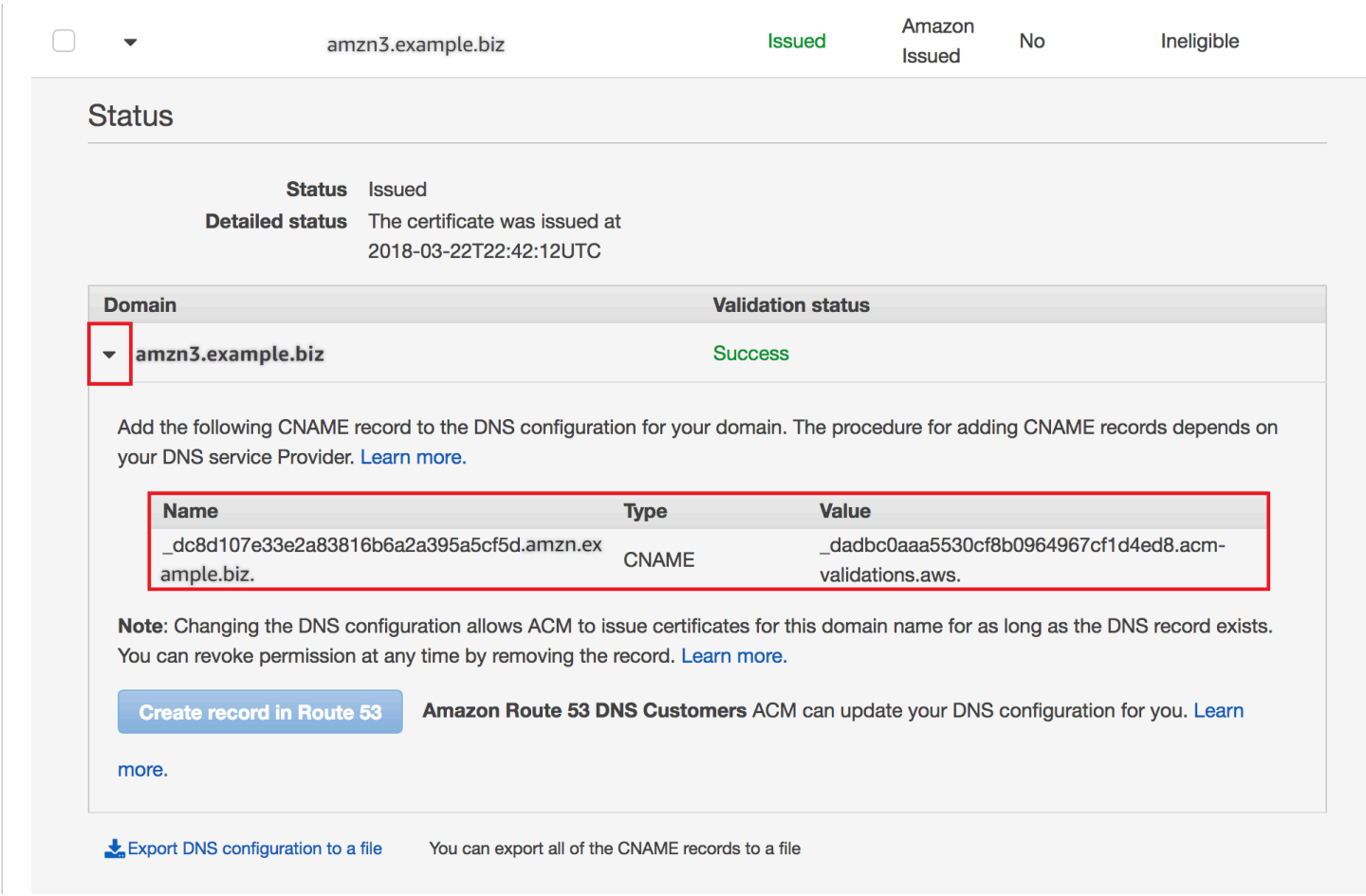
ACM不會嘗試TLS驗證已驗證DNS的憑證。如果ACM無法更新您通過驗證DNS驗證的憑證，很可能是因為DNS組態中的CNAME記錄遺失或不正確。如果發生這種情況，會ACM通知您無法自動更新憑證。

> **Important**
>
> 您必須將正確的CNAME記錄插入到DNS數據庫中。操作方式請洽詢您的網域註冊商。

您可以在主控台中展開憑證及其網域項目，以尋找網域的CNAME記ACM錄。如需詳細資訊，請參閱下圖。您也可以使用ACMAPI或中的描述憑證命令中的DescribeCertificate作業來擷取CNAME記錄。ACM CLI如需詳細資訊，請參閱 DNS驗證。



請從主控台選擇目標憑證。



展開憑證視窗以尋找憑證的CNAME資訊。

如果問題仍存在，請聯絡支援中心。

了解續約時機

ACM憑證的受管理續約 是非同步的程序。這表示步驟不會緊接著連續發生。驗ACM證憑證中的所有網域名稱後，在ACM取得新憑證之前可能會有一段延遲。在ACM取得更新憑證到將憑證部署至使用憑證的 AWS 資源之間，可能會發生額外的延遲。因此，憑證狀態的變更可能需要數小時才會在主控台顯示。

針對其他問題進行疑難排解

本節包含與簽發或驗ACM證憑證無關的問題的指引。

主題

- 憑證授權單位授權 (CAA) 問題

- 憑證匯入問題
- 憑證關聯定問題
- API閘道問題
- 工作憑證未預期失敗時該如何處理
- ACM服務連結角色的問題 () SLR

憑證授權單位授權 (CAA) 問題

您可以使用CAADNS記錄指定 Amazon 憑證授權單位 (CA) 可以為您的網域或子網域核發ACM憑證。如果您在憑證發行期間收到錯誤訊息，指出一個或多個網域名稱因憑證授權單位授權 (CAA) 錯誤而驗證失敗，請檢查您的CAADNS記錄。如果您在成功驗ACM證憑證要求後收到此錯誤，您必須更新CAA記錄並再次要求憑證。CAA記錄中的值欄位必須包含下列其中一個網域名稱：

- amazon.com
- amazontrust.com
- awstrust.com
- amazonaws.com

如需建立CAA記錄的詳細資訊，請參閱(選擇性) 設定CAA記錄。

> **Note**
>
> 如果您不想啟用CAA檢查功能，可以選擇不設定網域的CAA記錄。

憑證匯入問題

您可以匯入協力廠商憑證，ACM並將其與整合式服務產生關聯。如果您遇到問題，請檢閱先決條件和憑證格式主題。特別要注意下列事項：

- 您只能匯入 X.509 第 3 版SSL/TLS憑證。
- 您的憑證可以自我簽署，也可以由憑證授權機構 (CA) 簽署。
- 如果您的憑證是由 CA 簽署，則必須包含提供授權根路徑的中繼憑證鏈。
- 如果您的憑證為自我簽署，則必須納入純文字形式的私有金鑰。

- 鏈中的每個憑證皆必須直接認證上一個憑證。
- 請不要將您的最終實體憑證包含在中繼憑證鏈中。
- 您的憑證、憑證鏈結和私密金鑰 (如果有的話) 必須是 PEM —coded。一般而言，PEM編碼由 Base64 編碼的文字區塊組成，這些ASCII文字以純文字表頭和頁尾行開頭和結尾。複製或上傳檔案時，不得新增行或空格，或對PEM檔案進行任何其他變更。您可以使用開啟驗證公用程式來SSL驗證憑證鏈結。
- 您的私有金鑰 (如果有) 不能加密。(提示：如果設有密碼短語便會加密。)
- 與整合的服務ACM必須使用ACM支援的演算法和金鑰大小。請參閱 AWS Certificate Manager 使用者指南和每項服務的說明文件，以確保您的憑證可正常運作。
- 整合式服務的憑證支援可能會因憑證匯入IAM或匯入憑證而有所不同ACM。
- 匯入時，憑證必須有效。
- 所有憑證的詳細資訊都會顯示在主控台中。但是，依預設，如果您呼叫ListCertificatesAPI或 list 憑證 AWS CLI 命令而未指定`keyTypes`篩選器，則只會顯示`RSA_1024`或`RSA_2048`憑證。

憑證關聯定問題

若要更新憑證，請ACM產生新的公開-私密 key pair。如果您的應用程式使用 憑證關聯 (有時稱為SSL釘選) 來釘選ACM憑證，則應用程式在續訂憑證後 AWS 可能無法連線到您的網域。因此，我們建議您不要釘選ACM憑證。如果您的應用程式必須關聯憑證，您可以執行以下操作：

- 將您自己的憑證匯入，ACM然後將您的應用程式釘選至匯入的憑證。ACM不為匯入的憑證提供受管理的續約。
- 如果您使用的是公有憑證，請將應用程式釘選到所有可用的 Amazon 根憑證。如果您使用的是私有憑證，請將您的應用程式釘選到 CA 根憑證。

API閘道問題

當您部署邊緣最佳化API端點時，APIGateway 會為您設定 CloudFront 分發。該 CloudFront 分配是由 API Gateway 擁有，而不是由您的帳戶所擁有。發行版本繫ACM結至您在部署API. 若要移除繫結並ACM允許刪除憑證，您必須移除與憑證相關聯的API閘道自訂網域。

當您部署地區API端點時，API閘道會代表您建立應用程式負載平衡器 (ALB)。負載平衡器由API閘道擁有，您看不到。系統ALB會繫ACM結至您在部署API. 若要移除繫結並ACM允許刪除憑證，您必須移除與憑證相關聯的API閘道自訂網域。

工作憑證未預期失敗時該如何處理

如果您已成功將ACM憑證與整合式服務產生關聯，但憑證停止運作，而且整合式服務開始傳回錯誤，原因可能是服務使用ACM憑證所需的權限變更。

例如，Elastic Load Balancing (ELB) 需要權限才 AWS KMS key 能解密憑證的私密金鑰。此權限由以資源為基礎的策略授與，該策略會在您將憑證與ELB憑證產生關聯時ACM套用。如果ELB遺失該權限的授與，下次嘗試解密憑證金鑰時將失敗。

若要調查問題，請使用的 AWS KMS 主控台檢查授權的狀態https://console.aws.amazon.com/kms。然後執行下列其中一個動作：

- 如果您認為授與整合服務的許可已被撤銷，請造訪整合服務的主控台，取消憑證與該服務的關聯，然後重新建立關聯。這麼做會重新套用資源型政策，並以新的授權取代。
- 如果您認為授予的權限ACM已被撤銷，請通過 https://console.aws.amazon.com/support/ home#/ 聯繫 AWS Support 。

ACM服務連結角色的問題 () SLR

當您發行私有 CA 所簽署且已由其他帳戶與您共用的憑證時，會ACM嘗試首次使用設定服務連結角色 (SLR)，以作為主體與 AWS 私有 CA 資源型存取原則互動。如果您從共用 CA 核發私人憑證，但該SLR憑證尚未到位，ACM將無法自動為您續約該憑證。

ACM可能會提醒您，它無法確定您的帳戶中是否SLR存在。如果您的帳戶已授與所需的`iam:GetRole`ACMSLR權限，則在建立之後不會再次發生警示。SLR如果重複發生，則您或您的帳戶管理員可能需要授與`iam:GetRole`權限ACM，或將您的帳戶與 ACM-managed 策略建立關聯。`AWSCertificateManagerFullAccess`

如需詳細資訊，請參閱IAM使用指南中的服務連結角色權限。

處理例外狀況

AWS Certificate Manager 命令失敗可能有幾個原因。如需每個例外狀況的資訊，請參閱下表。

私有憑證例外狀況處理

當您嘗試更新由發行的私人PKI憑證時，可能會發生下列例外狀況 AWS 私有 CA。

> **Note**
>
> AWS 私有 CA 中國（北京）地區和中國（寧夏）地區不支援。

ACM 失敗代碼	註解
PCA_ACCESS_DENIED	私有 CA 尚未授與ACM權限。這會觸發 AWS 私有 CA AccessDeniedException 失敗代碼。 若要修正問題，請使用 AWS 私有 CA CreatePermission作業將必要的權限授與ACM服務主體。
PCA_INVALID_DURATION	所要求憑證的有效期間超過發行私有憑證授權機構的有效期間。這會觸發 AWS 私有 CA ValidationException 失敗代碼。 若要修正此問題，請安裝新的憑證授權機構憑證(需具有適當的有效期間)。
PCA_INVALID_STATE	被調用的私有 CA 不是在正確的狀態來執行請求的ACM操作。這會觸發 AWS 私有 CA InvalidStateException 失敗代碼。 解決此問題的方法如下所示： • 如果 CA 的狀態為 CREATING，請等待建立完成，然後安裝憑證授權機構憑證。 • 如果 CA 的狀態為 PENDING_CERTIFICATE ，請安裝憑證授權機構憑證。 • 如果 CA 的狀態為 DISABLED，請將其更新為 ACTIVE 狀態。 • 如果 CA 的狀態為 DELETED，請將其還原。 • 如果 CA 的狀態為 EXPIRED，請安裝新的憑證

ACM 失敗代碼	註解
	• 如果 CA 的狀態為 FAILED，而且您無法解決問題，請連絡 AWS Support。
PCA_LIMIT_EXCEEDED	私有 CA 已達到發行配額。這會觸發 AWS 私有 CA LimitExceededException 失敗代碼。在繼續使用此說明之前，請嘗試重複提出您的請求。 如果錯誤仍存在，請連絡 AWS Support 以請求增加配額。
PCA_REQUEST_FAILED	發生網路或系統錯誤。這會觸發 AWS 私有 CA RequestFailedException 失敗代碼。在繼續使用此說明之前，請嘗試重複提出您的請求。 如果錯誤仍存在，請聯絡 AWS Support。
PCA_RESOURCE_NOT_FOUND	私有 CA 已被永久刪除。這會觸發 AWS 私有 CA ResourceNotFoundException 失敗代碼。請確認您使用的是正確的ARN。如果失敗，您將無法使用此 CA。 若要修正此問題，請建立新的 CA。
SLR_NOT_FOUND	若要更新由位於其他帳戶的私有 CA 所簽署的憑證，ACM需要憑證所在的帳戶上具有服務連結角色 (SLR)。如果您需要重新建立已刪除的項目 SLR，請參閱為 ACM 建立 SLR。

概念

本節提供 AWS Certificate Manager 所使用概念的定義。

主題

ACM 憑證

ACM 會產生 X.509 第 3 版憑證。每個的有效期限為 13 個月 (395 天)，並包含下列延伸項目。

- 基本限制 - 指定憑證主體是否是認證機構 (CA)。
- 授權機構金鑰識別符 - 支援識別與用於簽署憑證的私有金鑰對應的公有金鑰。
- 主體金鑰識別符 - 支援識別包含特定公有金鑰的憑證。

- 金鑰使用 - 定義內嵌於憑證的公有金鑰的用途。
- 擴充金鑰使用 - 除了金鑰使用延伸所指定的用途外，為公有金鑰指定的一個或多個用途。
- CRL 分佈點 - 指定可取得 CRL 資訊的位置。

ACM 所發行憑證的純文字類似於以下範例：

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            f2:16:ad:85:d8:42:d1:8a:3f:33:fa:cc:c8:50:a8:9e
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: O=Example CA
        Validity
            Not Before: Jan 30 18:46:53 2018 GMT
            Not After : Jan 31 19:46:53 2018 GMT
        Subject: C=US, ST=VA, L=Herndon, O=Amazon, OU=AWS, CN=example.com
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:ba:a6:8a:aa:91:0b:63:e8:08:de:ca:e7:59:a4:
                    69:4c:e9:ea:26:04:d5:31:54:f5:ec:cb:4e:af:27:
                    e3:94:0f:a6:85:41:6b:8e:a3:c1:c8:c0:3f:1c:ac:
                    a2:ca:0a:b2:dd:7f:c0:57:53:0b:9f:b4:70:78:d5:
                    43:20:ef:2c:07:5a:e4:1f:d1:25:24:4a:81:ab:d5:
                    08:26:73:f8:a6:d7:22:c2:4f:4f:86:72:0e:11:95:
                    03:96:6d:d5:3f:ff:18:a6:0b:36:c5:4f:78:bc:51:
                    b5:b6:36:86:7c:36:65:6f:2e:82:73:1f:c7:95:85:
                    a4:77:96:3f:c0:96:e2:02:94:64:f0:3a:df:e0:76:
                    05:c4:56:a2:44:72:6f:8a:8a:a1:f3:ee:34:47:14:
                    bc:32:f7:50:6a:e9:42:f5:f4:1c:9a:7a:74:1d:e5:
                    68:09:75:19:4b:ac:c6:33:90:97:8c:0d:d1:eb:8a:
                    02:f3:3e:01:83:8d:16:f6:40:39:21:be:1a:72:d8:
                    5a:15:68:75:42:3e:f0:0d:54:16:ed:9a:8f:94:ec:
                    59:25:e0:37:8e:af:6a:6d:99:0a:8d:7d:78:0f:ea:
                    40:6d:3a:55:36:8e:60:5b:d6:0d:b4:06:a3:ac:ab:
                    e2:bf:c9:b7:fe:22:9e:2a:f6:f3:42:bb:94:3e:b7:
                    08:73
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
```

```
                CA:FALSE
            X509v3 Authority Key Identifier:
                keyid:84:8C:AC:03:A2:38:D9:B6:81:7C:DF:F1:95:C3:28:31:D5:F7:88:42
            X509v3 Subject Key Identifier:
                97:06:15:F1:EA:EC:07:83:4C:19:A9:2F:AF:BA:BB:FC:B2:3B:55:D8
            X509v3 Key Usage: critical
                Digital Signature, Key Encipherment
            X509v3 Extended Key Usage:
                TLS Web Server Authentication, TLS Web Client Authentication
            X509v3 CRL Distribution Points:
                Full Name:
                  URI:http://example.com/crl

    Signature Algorithm: sha256WithRSAEncryption
         69:03:15:0c:fb:a9:39:a3:30:63:b2:d4:fb:cc:8f:48:a3:46:
         69:60:a7:33:4a:f4:74:88:c6:b6:b6:b8:ab:32:c2:a0:98:c6:
         8d:f0:8f:b5:df:78:a1:5b:02:18:72:65:bb:53:af:2f:3a:43:
         76:3c:9d:d4:35:a2:e2:1f:29:11:67:80:29:b9:fe:c9:42:52:
         cb:6d:cd:d0:e2:2f:16:26:19:cd:f7:26:c5:dc:81:40:3b:e3:
         d1:b0:7e:ba:80:99:9a:5f:dd:92:b0:bb:0c:32:dd:68:69:08:
         e9:3c:41:2f:15:a7:53:78:4d:33:45:17:3e:f2:f1:45:6b:e7:
         17:d4:80:41:15:75:ed:c3:d4:b5:e3:48:8d:b5:0d:86:d4:7d:
         94:27:62:84:d8:98:6f:90:1e:9c:e0:0b:fa:94:cc:9c:ee:3a:
         8a:6e:6a:9d:ad:b8:76:7b:9a:5f:d1:a5:4f:d0:b7:07:f8:1c:
         03:e5:3a:90:8c:bc:76:c9:96:f0:4a:31:65:60:d8:10:fc:36:
         44:8a:c1:fb:9c:33:75:fe:a6:08:d3:89:81:b0:6f:c3:04:0b:
         a3:04:a1:d1:1c:46:57:41:08:40:b1:38:f9:57:62:97:10:42:
         8e:f3:a7:a8:77:26:71:74:c2:0a:5b:9e:cc:d5:2c:c5:27:c3:
         12:b9:35:d5
```

ACM 根 CA

ACM 發行的公有最終實體憑證會從下列 Amazon 根 CA 衍生其信任：

辨別名稱	加密演算法
CN=Amazon 根 CA 1、O=Amazon、C=美國	2048 位元 RSA (RSA_2048)
CN=Amazon 根 CA 2、O=Amazon、C=美國	4096 位元 RSA (RSA_4096)

辨別名稱	加密演算法
CN=Amazon 根 CA 3、O=Amazon、C=美國	橢圓主要曲線 256 位元 (EC_prime256v1)
CN=Amazon 根 CA 4、O=Amazon、C=美國	橢圓主要曲線 384 位元 (EC_secp384r1)

ACM 發行憑證的預設信任根是 CN=Amazon 根 CA 1、O=Amazon、C=US，這可提供 2048 位元 RSA 安全性。其他根保留供日後使用。所有根都是由 Starfield Services Root Certificate Authority 憑證交叉簽署。

如需詳細資訊，請參閱 Amazon Trust Services。

Apex 網域

請參閱 網域名稱。

非對稱金鑰加密法

與對稱金鑰加密法不同，非對稱加密法使用不同但屬於數學算法的金鑰來加密和解密內容。其中一個金鑰為公有，且通常包含於 X.509 v3 憑證。另一個金鑰為私有，且存放在安全的位置。X.509 憑證會將使用者、電腦或其他資源 (憑證主體) 的身分繫結至公有金鑰。

ACM 憑證是 X.509 SSL/TLS 憑證，它會將您網站的身分和組織的詳細資訊繫結至憑證中包含的公有金鑰。ACM 會使用您的 AWS KMS key 加密私有金鑰。如需詳細資訊，請參閱 憑證私有金鑰的安全性。

憑證授權單位

憑證授權機構 (CA) 是發行數位憑證的實體。商業上，最常見的數位憑證類型是根據 ISO X.509 標準。CA 發行已簽署的數位憑證，以確認憑證主體的身分並將該身分繫結至憑證中包含的公有金鑰。CA 通常還會管理憑證撤銷。

憑證透明度記錄

為了防備因失誤而發行或由遭入侵的 CA 發行的 SSL/TLS 憑證，某些瀏覽器要求為您網域發行的公有憑證必須記錄在憑證透明度日誌中。網域名稱會被記錄。私有金鑰不會被記錄。未記錄的憑證通常會在瀏覽器中產生錯誤。

您可以監控日誌，以確保只為您的網域發行已獲得您授權的憑證。您可以使用 Certificate Search 等服務來檢查日誌。

在 Amazon CA 為您的網域發行公開信任的 SSL/TLS 憑證前，它會將憑證提交到至少三個憑證透明度日誌伺服器。這些伺服器會將憑證加入其公有資料庫，並將已簽署的憑證時間戳記 (SCT) 傳回到 Amazon CA。然後，CA 會將 SCT 嵌入憑證中，簽署憑證，並發行給您。時間戳記會隨附於其他 X.509 延伸。

```
X509v3 extensions:

  CT Precertificate SCTs:
    Signed Certificate Timestamp:
      Version   : v1(0)
        Log ID    : BB:D9:DF:...8E:1E:D1:85
        Timestamp : Apr 24 23:43:15.598 2018 GMT
        Extensions: none
        Signature : ecdsa-with-SHA256
                    30:45:02:...18:CB:79:2F
    Signed Certificate Timestamp:
      Version   : v1(0)
        Log ID    : 87:75:BF:...A0:83:0F
        Timestamp : Apr 24 23:43:15.565 2018 GMT
        Extensions: none
        Signature : ecdsa-with-SHA256
                    30:45:02:...29:8F:6C
```

憑證透明度記錄會在您請求或續約憑證時自動執行，除非您選擇退出。如需選擇退出的詳細資訊，請參閱取消使用憑證透明度記錄功能。

網域名稱系統

網域名稱系統 (DNS) 是階層分散式命名系統，適用於連接到網際網路或私有網路的電腦和其他資源。DNS 主要用於將文字網域名稱 (例如 aws.amazon.com) 轉換成形式為 111.122.133.144 的數字 IP (網際網路通訊協定) 地址。不過，您網域的 DNS 資料庫包含一些其他用途的記錄。例如，當您透過 ACM 請求憑證時，可以使用 CNAME 記錄來驗證您擁有或控制某個網域。如需詳細資訊，請參閱 DNS驗證。

網域名稱

網域名稱為可由網域名稱系統 (DNS) 轉換為 IP 地址的文字字串，例如 www.example.com。電腦網路 (包括網際網路) 使用 IP 地址，而不是文字名稱。網域名稱包含多個不同標籤，並以句點區隔：

TLD

最右邊的標籤稱為頂層網域 (TLD)。常見的範例包括 .com、.net 和 .edu。此外，註冊在某些國家/地區的實體 TLD 為該國家/地區名稱的縮寫，稱為國碼 (地區碼)。例如：.uk 代表英國、.ru 代表俄羅斯，而 .fr 代表法國。使用國碼 (地區碼) 時，TLD 的第二層通常用於識別註冊實體的類型。例如，.co.uk TLD 代表英國的商業企業。

Apex 網域

Apex 網域名稱包含及擴展於頂層網域。針對包含國碼 (地區碼) 的網域名稱，Apex 網域包含用來識別註冊實體類型的代碼和標籤 (如果有)。Apex 網域不包含子網域 (請參閱以下段落)。在 www.example.com 中，Apex 網域的名稱為 example.com。在 www.example.co.uk 中，Apex 網域的名稱為 example.co.uk。其他經常使用的非 Apex 名稱包括 base、bare、root、root apex 或 zone apex。

子網域

子網域名稱在 Apex 網域名稱前面，並由句號隔開。最常見的子網域名稱為 www，但也可以是任何名稱。子網域名稱也可以具有多個層級。例如，在 jake.dog.animals.example.com 中，子網域依序為 jake、dog 和 animals。

超级域名

子網域所屬的網域。

FQDN

完整網域名稱 (FQDN) 是電腦、網站或其他連接到網路或網際網路的資源的完整 DNS 名稱。例如 aws.amazon.com 是 Amazon Web Services 的 FQDN。FQDN 包含所有網域，上至頂層網域。例如，[subdomain$_1$].[subdomain$_2$]...[subdomain$_n$].[apex domain].[top-level domain] 代表 FQDN 的一般格式。

PQDN

不完整的網域名稱稱為部分網域名稱 (PQDN) 且不明確。[subdomain$_1$.subdomain$_2$.] 這樣的名稱屬於 PQDN，因為無法判斷根網域。

註冊

使用網域名稱的權限是由網域名稱註冊商委派。註冊商通常由網際網路名稱和數字指派公司 (ICANN) 認可。此外，其他稱為註冊機構的組織負責維護 TLD 資料庫。申請網域名稱時，註冊商會將您的資訊傳送到適當的 TLD 註冊機構。註冊機構會指派網域名稱、更新 TLD 資料庫，並將您的資訊發佈至 WHOIS。通常您必須購買網域名稱。

加密和解密

加密是提供資料機密性的程序。解密會反轉此程序並恢復原始資料。未加密的資料通常稱為純文字，無論它是否為文字。加密的資料通常稱為加密文字。用戶端和伺服器之間的訊息 HTTPS 加密會使用演算法和金鑰。演算 step-by-step法定義明文資料轉換成密文 (加密) 和密文轉換回原始明文 (解密) 的程序。在加密或解密程序中，演算法會使用金鑰。金鑰可以是私有或公有。

完整網域名稱 (FQDN)

請參閱 網域名稱。

公有金鑰基礎設施

公有金鑰基礎設施 (PKI) 包含建立、發行、管理、散佈、使用、存放和撤銷數位憑證所需的硬體、軟體、人員、政策、文件和程序。PKI 可促進資訊在電腦網路中的安全傳輸。

根憑證

憑證授權機構 (CA) 通常位於包含多個其他 CA 且清楚定義父子關係的階層結構內。下層 CA 或次級 CA 是由上層 CA 認證，並形成憑證鏈。階層頂端的 CA 稱為根 CA，其憑證稱為根憑證。此憑證通常為自我簽署。

Secure Sockets Layer (SSL)

Secure Sockets Layer (SSL) 和 Transport Layer Security (TLS) 是透過電腦網路提供通訊安全的加密通訊協定。TLS 的前身是 SSL。兩者皆使用 X.509 憑證對伺服器進行身分驗證。這兩個通訊協定會在用戶端和伺服器之間交涉對稱金鑰，該金鑰則用來對兩個實體之間流動的資料進行加密。

安全 HTTPS

HTTPS 代表 HTTP over SSL/TLS，其為所有主要瀏覽器和伺服器支援的 HTTP 安全形式。所有 HTTP 請求和回應皆會先加密，再傳送到網路。HTTPS 結合 HTTP 通訊協定與對稱、非對稱和 X.509 憑證型加密技術。HTTPS 的運作方式是插入「開放系統互相連線 (OSI) 模型」中低於 HTTP 應用程式層且高於 TCP 傳輸層的加密安全層。安全層使用 Secure Sockets Layer (SSL) 通訊協定或 Transport Layer Security (TLS) 通訊協定。

SSL 伺服器憑證

HTTPS 交易需要伺服器憑證，才能對伺服器進行身分驗證。伺服器憑證是 X.509 v3 資料結構，將憑證中的公有金鑰繫結至憑證主體。SSL/TLS 憑證是由憑證授權機構 (CA) 簽署，包含伺服器名稱、有效期間、公有金鑰、簽章演算法等。

對稱金鑰加密法

對稱金鑰加密法使用相同的金鑰來加密和解密數位資料。另請參閱 非對稱金鑰加密法。

Transport Layer Security (TLS)

請參閱 Secure Sockets Layer (SSL)。

信任

為了讓 Web 瀏覽器信任網站的身分，瀏覽器必須能驗證網站的憑證。不過，瀏覽器只信任少數稱為 CA 根憑證的憑證。稱為憑證授權機構 (CA) 的信任第三方會驗證網站的身分，並將已簽署的數位憑證發給網站的營運商。然後，瀏覽器便可檢查數位簽章，以驗證網站的身分。如果驗證成功，瀏覽器會在網址列中顯示鎖定圖示。

文件歷史紀錄

下表說明 2018 年 AWS Certificate Manager 開始的文件發行歷程記錄。

變更	描述	日期
更新了電子郵件驗證	主ACM控台不再支援使用的電子郵件驗證WHOIS。如果您想要透過電子郵件驗證您的網域，請使用主控台、APISDK、或來設定網域驗證CLI。	2024年7月11日
棄用郵件交換器 (MX) 電子郵件驗證	主ACM控台不再支援郵件交換程式 (MX)。	2024年7月11日
新增帳戶層級分隔的最佳做法	盡可能在您的政策中使用帳戶層級分隔。如果不可能，您可以在帳戶層級或策略中透過加密內容條件金鑰來限制權限。	2024年6月11日
即將淘汰的WHOIS電子郵件驗證	新增自 2024 年 6 月起停用WHOIS電子郵件驗證的相關附註。	2024年2月5 日
新增條件索引鍵支援	新增要求ACM憑證時對IAM條件金鑰的支援。如需支援條件清單，請參閱 https://docs.aws.amazon.com/acm/latest/userguide/acm-conditions.html#acm-conditions-supported。	2023 年 8 月 24 日
ECDSA支持添加	新增要求公用ACM憑證時對橢圓曲線數位簽章演算法 (ECDSA) 的支援。如需支援金鑰演算法的清單，請參閱 https://docs.aws.amazon.com/	2022 年 11 月 8 日

	acm/latest/userguide/acm-certificate.html#algorithms。	
最新 CloudWatch 活動	新增「ACM憑證過期」、「可用ACM憑證」和「需要ACM憑證續訂動作」事件。如需支援的 CloudWatch 事件清單，請參閱https://docs.aws.amazon.com/acm/latest/userguide/cloudwatch-events.html。	2022 年 10 月 27 日
更新用於匯入的金鑰演算法類型	匯入的憑證現在ACM可能具有含有其他RSA和橢圓曲線演算法的金鑰。如需目前所支援金鑰演算法的清單，請參閱「https://docs.aws.amazon.com/acm/latest/userguide/import-certificate-prerequisites.html」。	2021 年 7 月 14 日
將「監控和記錄」作為單獨的章節加以宣導	將監控和記錄的說明文件移至各自專屬的章節。此變更涵蓋 CloudWatch量度、CloudWatch 事件/EventBridge和。 CloudTrail如需詳細資訊，請參閱 https://docs.aws.amazon.com/acm/latest/userguide/monitoring-and-logging.html。	2021 年 3 月 23 日

新增 CloudWatch 指標和事件支援	添加了 DaysToExpiry 指標和事件和支持APIs。如需更多詳細資訊，請參閱「https://docs.aws.amazon.com/acm/latest/userguide/cloudwatch-metrics.html」及「https://docs.aws.amazon.com/acm/latest/userguide/cloudwatch-events.html」。	2021 年 3 月 3 日
新增跨帳戶支援	添加了跨帳戶支持使用私有CAs來源 AWS 私有 CA。如需詳細資訊，請參閱 https://docs.aws.amazon.com/acm/latest/userguide/ca-access.html。	2020 年 8 月 17 日
已新增的區域支援	增加了對 AWS 中國（北京和寧夏）地區的區域支持。如需支援區域的完整清單，請參閱 https://docs.aws.amazon.com/general/latest/gr/rande.html#acm-pca_region。	2020 年 3 月 4 日
新增續約工作流程測試	客戶現在可以手動測試其ACM受管續約工作流程的組態。如需詳細資訊，請參閱測試ACM的受管續訂組態。	2019 年 3 月 14 日
憑證透明度記錄成為預設功能	新增預設將ACM公用憑證發佈至憑證透明度記錄的功能。	2018 年 4 月 24 日

啟動 AWS 私有 CA	推出ACM私有 Certificate Manager（CM），擴展允許用戶建立一個安全的託管基礎設施，用於發行和撤銷私有數碼證書。 AWS Certificate Manager 如需詳細資訊，請參閱 AWS Private Certificate Authority。	2018 年 4 月 4 日
憑證透明度記錄	新增憑證透明度記錄到最佳實務。	2018 年 3 月 27 日

下表說明 2018 之前的文件發 AWS Certificate Manager 行歷史記錄。

變更	描述	版本日期
新內容	添加DNS驗證到DNS驗證.	2017 年 11 月 21 日
新內容	已新增新的 Java 程式碼範例到使用 API (Java 範例)。	2017 年 10 月 12 日
新內容	已新增關於CAA記錄的資訊(選擇性) 設定CAA記錄。	2017 年 9 月 21 日
新內容	已新增 .IO 網域的相關資訊到故障診斷。	2017 年 7 月 07 日
新內容	已新增重新匯入憑證的相關資訊到 重新匯入憑證。	2017 年 7 月 07 日
新內容	已新增憑證關聯的相關資訊到最佳實務 和 故障診斷。	2017 年 7 月 07 日
新內容	已新增 AWS CloudFormation 至服務整合 AWS Certificate Manager.	2017 年 5 月 27 日
更新	已新增詳細資訊到 配額。	2017 年 5 月 27 日

變更	描述	版本日期
新內容	已新增 的 Identity and Access Management AWS Certificate Manager 的相關文件。	2017 年 4 月 28 日
更新	已新增圖形，顯示傳送驗證電子郵件的位置。請參閱電子郵件驗證。	2017 年 4 月 21 日
更新	已新增為您的網域設定電子郵件的相關資訊。請參閱(選用) 為您的網域設定電子郵件。	2017 年 4 月 6 日
更新	已新增在主控台中檢查憑證續約狀態的相關資訊。請參閱檢查憑證的續約狀態。	2017 年 3 月 28 日
更新	更新 Elastic Load Balancing 的使用說明文件。	2017 年 3 月 21 日
新內容	增加了對 AWS Elastic Beanstalk Amazon API 網關的支持。請參閱服務整合 AWS Certificate Manager。	2017 年 3 月 21 日
更新	已更新 受管續約 的相關文件。	2017 年 2 月 20 日
新內容	已新增 匯入憑證 的相關文件。	2016 年 10 月 13 日
新內容	增加了對ACM操作的 AWS CloudTrail 支持。請參閱 CloudTrail 搭配使用 AWS Certificate Manager。	2016 年 3 月 25 日
新指南	此版本推出 AWS Certificate Manager。	2016 年 1 月 21 日

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。