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、儲存和續約公有和私有 SSL/TLS X.509 憑證和金鑰的複雜性，以保護 AWS 網站和應用程式。您可以藉由直接透過 ACM 發行憑證，或將第三方憑證匯入 ACM 管理系統中，為整合式 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、萬用字元網域或這些網域的組合。ACM 萬用字元憑證可以保護不限數量的子網域。您也可以匯出由 簽署的 ACM 憑證 AWS 私有 CA ，以便在內部 PKI 的任何位置使用。

> **Note**
>
> ACM 不適用於獨立 Web 伺服器。如果您想要在 Amazon EC2 執行個體上設置獨立的安全伺服器，請參閱以下教學中的指示：在 Amazon Linux 2023上設定 SSL/TLS。

主題

支援地區

ACM 在公有端點上支援 IPv4 和 IPv6。請造訪 AWS 一般參考 中的 AWS 區域與端點或 AWS 區域表來了解 ACM 的可用區域。

ACM 中的憑證為區域性資源。若要對多個 AWS 區域中的相同完整網域名稱 (FQDN) 或一組 FQDNs 使用具有 Elastic Load Balancing 的憑證，您必須請求或匯入每個區域的憑證。若使用 ACM 提供的憑證，表示您必須為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。

若要搭配 Amazon CloudFront 使用 ACM 憑證，您必須在美國東部 (維吉尼亞北部) 區域請求或匯入憑證。此區域中與 CloudFront 分佈相關聯的 ACM 憑證，會分佈至所有為該分佈設定的地理位置。

的定價 AWS Certificate Manager

如果您使用 AWS Certificate Manager管理 SSL/TLS 憑證，則無需支付額外費用。您只需為您為執行網站或應用程式所建立 AWS 的資源付費。如需最新的 ACM 定價資訊，請參閱 AWS 網站上的AWS Certificate Manager 服務定價頁面。

AWS Certificate Manager 概念

本節提供 使用的概念定義 AWS Certificate Manager。

主題

- ACM 憑證
- ACM 根 CA
- Apex 網域
- 非對稱金鑰加密法
- 憑證授權單位
- 憑證透明度記錄
- 網域名稱系統
- 網域名稱
- 加密和解密
- 完整網域名稱 (FQDN)
- 超文字傳輸通訊協定 (HTTP)
- 公有金鑰基礎設施 (PKI)
- 根憑證
- Secure Sockets Layer (SSL)
- 安全 HTTPS
- SSL 伺服器憑證
- 對稱金鑰加密法
- Transport Layer Security (TLS)
- 信任

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)
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 AWS KMS key 來加密私有金鑰。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AWS Certificate Manager 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`。

超級網域

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，$[subdomain_1].[subdomain_2]...[subdomain_n].[apex\ domain].[top\text{-}level\ domain]$ 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。

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。演算法定義將純文字資料轉換為加密文字 (加密) 以及將加密文字轉換回原始純文字 (解密) 的逐步程序。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。

超文字傳輸通訊協定 (HTTP)

超文字傳輸通訊協定 (HTTP) 是全球資訊網資料通訊的基礎。它是一種應用程式層通訊協定，可交換各種內容類型。HTTP 在用戶端-伺服器模型上運作，其中 Web 瀏覽器通常充當從 Web 伺服器請求資源的用戶端。作為無狀態通訊協定，HTTP 會獨立處理每個請求，而不會保留先前請求的資訊。

在 ACM 的內容中，HTTP 可在發行 SSL/TLS 憑證時用於網域驗證。此程序涉及 ACM 傳送特定 HTTP 請求來驗證網域擁有權。伺服器正確回應這些請求的能力顯示了對網域的控制。

與電子郵件或 DNS 驗證的憑證不同，ACM 客戶無法直接從 ACM 發行 HTTP 驗證的憑證。反之，這些憑證會在 CloudFront 佈建程序中自動發行和管理。客戶可以使用 ACM 來檢視、監控和管理這些憑證，但初始發行是由 ACM 和 CloudFront 之間的整合處理。

雖然 HTTP 廣泛使用，但請務必注意，它會以純文字傳輸資料。為了安全通訊，會使用 HTTPS (HTTP Secure)，這會使用 SSL/TLS 通訊協定來加密資料。如需安全通訊的詳細資訊，請參閱 安全 HTTPS。

公有金鑰基礎設施 (PKI)

公有金鑰基礎設施 (PKI) 是一種程序、技術和政策的系統，可透過公有網路進行安全通訊。在 ACM 環境中，PKI 在數位憑證的發行、管理和驗證中扮演重要角色。PKI 使用一對密碼編譯金鑰：自由分佈的公有金鑰，以及由擁有者保密的私有金鑰。此系統允許安全資料傳輸、數位簽章和數位實體的身分驗證。

ACM 實作 PKI 的數個關鍵元件。它充當憑證授權機構 (CA)，這是可信任的第三方，可發出數位憑證，將公有金鑰繫結至網域或組織等實體。ACM 發行 X.509 憑證，其中包含實體、其公有金鑰和憑證有效

期間的相關資訊。它還處理憑證的完整生命週期，包括發行、續約和撤銷。為了確保憑證請求的合法性，ACM 支援各種方法來驗證網域擁有權，例如 DNS 驗證和 HTTP 驗證。

透過利用 PKI，ACM 可為 AWS 資源和應用程式啟用安全的 HTTPS 連線、數位簽章和加密通訊。此基礎設施對於維護透過網際網路傳輸之資料的機密性、完整性和真實性至關重要。如需 ACM 如何實作 PKI 的詳細資訊，請參閱 AWS Certificate Manager 憑證。

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。

什麼是符合我需求的 AWS 憑證服務？

AWS 提供兩種選項給部署受管 X.509 憑證的客戶。選擇最適合您需求的選項。

1. AWS Certificate Manager (ACM) — 此服務適用於需要使用 TLS 的安全 Web 存在的企業客戶。ACM 憑證是透過 Elastic Load Balancing、Amazon CloudFront、Amazon API Gateway 和其他整合 AWS 服務部署。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 也會自動續約即將過期的憑證，藉此簡化安全管理作業。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 雲端內建置公有金鑰基礎設施 (PKI) 的企業客戶，並供組織內私有使用。使用 AWS 私有 CA，您可以建立自己的憑證授權機構 (CA) 階層，並發行憑證來驗證使用者、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」。

AWS Certificate Manager 憑證

ACM 會管理公有、私有和匯入的憑證。憑證用於建立跨網際網路或內部網路的安全通訊。您可以直接從 ACM (「ACM 憑證」) 請求公開信任的憑證，匯入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，您可以發行由私有憑證授權機構 (CA) 簽署並由 AWS 私有 CA 管理的 ACM 憑證。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 Nitro Enclave 的 Amazon EC2 執行個體上。您也可以匯出公有憑證，以便在任何 Amazon EC2 執行個體上使用。如需了解如何在未連接至 Nitro Enclave 的 Amazon EC2 執行個體上設定獨立 Web 伺服器，請參閱教學課程：在 Amazon Linux 2 上安裝 LAMP Web 伺服器或教學課程：使用 Amazon Linux AMI 安裝 LAMP Web 伺服器。

> **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，並在 https：//https://console.aws.amazon.com/acm/home 開啟 ACM 主控台。出現簡介頁面時，請選擇 Get Started (開始使用)。否則，請選擇左側導覽窗格中的 Certificate Manager 或 Private CAs (私有 CA)。

主題

設定 以使用 AWS Certificate Manager

透過 AWS Certificate Manager (ACM)，您可以為 AWS 以 為基礎的網站和應用程式佈建和管理 SSL/TLS 憑證。您可以使用 ACM 建立或匯入憑證，然後加以管理。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。如需深入了解與 ACM 整合的服務，請參閱 與 ACM 整合的服務。以下區段討論使用 ACM 之前需要執行的步驟。

主題

註冊 AWS 帳戶

如果您沒有 AWS 帳戶，請完成下列步驟來建立一個。

註冊 AWS 帳戶

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

   註冊程序的一部分包括接聽電話或文字訊息，並在電話鍵盤上輸入驗證碼。

   當您註冊 時 AWS 帳戶，AWS 帳戶根使用者會建立 。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 會在註冊程序完成後傳送確認電子郵件給您。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 AWS 帳戶，請保護 AWS 帳戶根使用者、啟用 AWS IAM Identity Center和建立管理使用者，以免將根使用者用於日常任務。

保護您的 AWS 帳戶根使用者

1. 選擇根使用者並輸入 AWS 帳戶 您的電子郵件地址，以帳戶擁有者AWS Management Console身分登入 。在下一頁中，輸入您的密碼。

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

2. 若要在您的根使用者帳戶上啟用多重要素驗證 (MFA)。

   如需說明，請參閱《IAM 使用者指南》中的為您的 AWS 帳戶 根使用者 （主控台） 啟用虛擬 MFA 裝置。

建立具有管理存取權的使用者

1. 啟用 IAM Identity Center。

   如需指示，請參閱《AWS IAM Identity Center 使用者指南》中的啟用 AWS IAM Identity Center。

2. 在 IAM Identity Center 中，將管理存取權授予使用者。

   如需使用 IAM Identity Center 目錄 做為身分來源的教學課程，請參閱AWS IAM Identity Center《 使用者指南》中的使用預設值設定使用者存取 IAM Identity Center 目錄。

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

  如需使用 IAM Identity Center 使用者登入的說明，請參閱AWS 登入 《 使用者指南》中的登入 AWS 存取入口網站。

指派存取權給其他使用者

1. 在 IAM Identity Center 中，建立一個許可集來遵循套用最低權限的最佳實務。

   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 ACM 的網域名稱

完整網域名稱 (FQDN) 是網際網路上組織或個人的唯一名稱，後面接著頂層網域延伸，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。網域名稱註冊通常持續一定的時間，例如必須續約一年或兩年。

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) 設定 CAA 記錄

CAA 記錄會指定允許哪種憑證授權單位 (CA) 為網域或子網域發出憑證。建立與 ACM 搭配使用的 CAA 記錄有助於防止錯誤的 CAs為您的網域發行憑證。CAA 記錄不能替代由您的憑證授權單位指定的安全要求，例如驗證您是網域擁有者的要求。

在 ACM 在憑證請求程序中驗證您的網域之後，它會檢查是否存在 CAA 記錄，以確保它可以為您發行憑證。設定 CAA 記錄是選用的。

當您設定 CAA 記錄時，請使用下列值：

flags

指定 ACM 是否支援 tag 欄位的值。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 欄位目前被忽略。

issue

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

issuewild

表示您在 value 欄位中指定的 ACM CA 已獲授權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。請注意，如果您計劃使用 HTTP 驗證，此設定將不適用，因為 HTTP 驗證不支援萬用字元憑證。對於萬用字元憑證，請改用 DNS 或電子郵件驗證。

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。不過，如果您沒有指定以下四個 Amazon CA 其中一個的 CAA 記錄，ACM 就無法為您的網域或子網域發行憑證：

- 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。

若存在不含 ACM CA 值的 issuewild CAA 記錄，ACM 就無法發行任何萬用字元。如果 issuewild 不存在，但有一筆 ACM 的 issue CAA 記錄，那麼 ACM 也許可發行萬用字元。

Example CAA 記錄範例

在以下範例中，首先是您的網域名稱，然後是記錄類型 (CAA)。flags 欄位一律為 0。tags 欄位可以是 issue 或 issuewild。如果欄位為 issue 且您在 value 欄位中輸入 CA 伺服器的網域名稱，則 CAA 記錄表示您指定的伺服器已獲許可發行您請求的憑證。如果您在 value 欄位中輸入分號 ";"，則 CAA 記錄表示不允許任何 CA 發行憑證。CAA 記錄的組態因 DNS 供應商而異。

> **Important**
>
> 如果您打算搭配 CloudFront 使用 HTTP 驗證，則不需要設定 issuewild 記錄，因為 HTTP 驗證不支援萬用字元憑證。對於萬用字元憑證，請改用 DNS 或電子郵件驗證。

```
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      ";"
```

如需如何新增或修改 DNS 記錄的詳細資訊，請聯絡您的 DNS 供應商。Route 53 支援 CAA 記錄。如果您的 DNS 供應商是 Route 53，請參閱 CAA 格式以了解有關建立記錄的詳細資訊。

AWS Certificate Manager 公有憑證

請求公有憑證之後，您必須驗證網域擁有權，如中所述驗證 AWS Certificate Manager 公有憑證的網域擁有權。

公有 ACM 憑證遵循 X.509 標準，且受制於下列各項限制：

- 名稱:您必須使用符合 DNS 規範的主旨名稱。如需詳細資訊，請參閱網域名稱。
- 演算法：針對加密，憑證私有金鑰演算法必須是 2048 位元 RSA、256 位元的 ECDSA 或 384 位元 ECDSA。
- 有效期限：每個憑證的有效期限皆為 13 個月 (395 天)。
- 續約：ACM 會在 11 個月後自動嘗試續約私有憑證。

管理員可以使用 ACM 條件索引鍵政策，控制最終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。如果您在要求憑證時遇到問題，請參閱對憑證請求進行故障診斷。

若要使用 請求私有 PKI 的憑證 AWS 私有 CA，請參閱 在 中請求私有憑證 AWS Certificate Manager。

AWS Certificate Manager 公有憑證特性和限制

ACM 提供的公有憑證具有下列特性和限制。這些僅適用於 ACM 提供的憑證。它們可能不適用於匯入的憑證。

瀏覽器和應用程式信任

ACM 憑證受 Google Chrome、Microsoft Edge、Mozilla Firefox 和 Apple Safari 等所有主要瀏覽器信任。透過 TLS 使用 ACM 憑證連線至網站時，瀏覽器會顯示鎖定圖示。Java 也會信任 ACM 憑證。

憑證授權單位和階層

透過 ACM 請求的公有憑證來自 Amazon Trust Services，Amazon 受管的公有憑證授權機構 (CA)。Amazon 根 CAs1 到 4 由 Starfield G2 根憑證授權機構 – G2 交叉簽署。Starfield 根在 Android （較舊的 Gingerbread 版本） 和 iOS (4.1 版以上） 上受信任。Amazon 根受 iOS 11+ 信任。包括 Amazon 或 Starfield 根的瀏覽器、應用程式或OSes將信任 ACM 公有憑證。

ACM 透過中繼 CAs 向客戶發行分葉或終端實體憑證，並根據憑證類型 (RSA 或 ECDSA) 隨機指派。由於此隨機選取，ACM 不提供中繼 CA 資訊。

網域驗證 (DV)

ACM 憑證經過網域驗證，僅識別網域名稱。請求 ACM 憑證時，您必須證明所有指定網域的擁有權或控制權。您可以使用電子郵件或 DNS 驗證所有權。如需詳細資訊，請參閱AWS Certificate Manager 電子郵件驗證及AWS Certificate Manager DNS 驗證。

HTTP 驗證

ACM 在發行公有 TLS 憑證以搭配 CloudFront 使用時，支援網域擁有權驗證的 HTTP 驗證。此方法使用 HTTP 重新導向來證明網域擁有權，並提供類似於 DNS 驗證的自動續約。HTTP 驗證目前只能透過 CloudFront 分佈租用戶功能使用。

HTTP 重新導向

對於 HTTP 驗證，ACM 會提供 `RedirectFrom` URL 和 `RedirectTo` URL。您必須設定從 `RedirectFrom` 到 的重新導向`RedirectTo`，以示範網域控制。`RedirectFrom` URL 包含已驗證的網域，同時`RedirectTo`指向 CloudFront 基礎設施中包含唯一驗證字符的 ACM 控制位置。

管理者

由其他服務管理的 ACM 中的憑證會在 `ManagedBy` 欄位中顯示該服務的身分。對於搭配 CloudFront 使用 HTTP 驗證的憑證，此欄位會顯示 "CLOUDFRONT"。這些憑證只能透過

CloudFront 使用。ManagedBy 欄位會出現在 DescribeCertificate和 ListCertificates APIs 中，以及 ACM 主控台中的憑證庫存和詳細資訊頁面上。

ManagedBy 欄位與「可與」屬性互斥。對於 CloudFront 受管憑證，您無法透過其他服務新增新的用量 AWS 。您只能透過 CloudFront API 將這些憑證與更多資源搭配使用。

中繼和根 CA 輪換

Amazon 可能會終止中繼 CA，恕不另行通知以維護彈性憑證基礎設施。這些變更不會影響客戶。如需詳細資訊，請參閱「Amazon 引進動態中繼憑證授權機構」。

如果 Amazon 終止根 CA，變更將視需要快速發生。Amazon 將使用所有可用的方法來通知 AWS 客戶，包括 AWS Health Dashboard、電子郵件以及與技術客戶經理的聯絡。

用於撤銷的防火牆存取

撤銷的最終實體憑證使用 OCSP 和 CRLs 來驗證和發佈撤銷資訊。有些客戶防火牆可能需要額外的規則，才能允許這些機制。

使用這些 URL 萬用字元模式來識別撤銷流量：

- OCSP

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

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

- CRL

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

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

星號 (*) 代表一或多個英數字元，問號 (?) 代表單一英數字元，雜湊符號 (#) 代表數字。

金鑰演算法

憑證必須指定演算法和金鑰大小。ACM 支援這些 RSA 和 ECDSA 公有金鑰演算法：

- RSA 1024 位元 (RSA_1024)
- RSA 2048 位元 (RSA_2048)*
- RSA 3072 位元 (RSA_3072)
- RSA 4096 位元 (RSA_4096)

- ECDSA 256 位元 (EC_prime256v1)*
- ECDSA 384 位元 (EC_secp384r1)*
- ECDSA 521 位元 (EC_secp521r1)

ACM 可以使用標記星號 (*) 的演算法來請求新憑證。其他演算法僅適用於匯入的憑證。

> **Note**
>
> 對於由 AWS Private CA CA 簽署的私有 PKI 憑證，簽署演算法系列 (RSA 或 ECDSA) 必須符合 CA 的私密金鑰演算法系列。

ECDSA 金鑰比具有相當安全性的 RSA 金鑰更小且運算更有效率，但並非所有網路用戶端都支援 ECDSA。此資料表改編自 NIST，比較 RSA 和 ECDSA 金鑰大小（以位元為單位）的同等安全強度：

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	521

安全強度為 2，與中斷加密所需的猜測次數相關。例如，3072 位元的 RSA 金鑰和 256 位元的 ECDSA 金鑰皆可在不超過 2^{128} 次猜測的情況下擷取到。

如需選擇演算法的說明，請參閱 AWS 部落格文章如何評估和使用 中的 ECDSA 憑證 AWS Certificate Manager。

> **Important**
>
> 整合式服務僅允許其資源支援的演算法和金鑰大小。支援會根據憑證是否匯入 IAM 或 ACM 而有所不同。如需詳細資訊，請參閱每個服務的文件：
>
> - 針對 Elastic Load Balancing，請參閱 Application Load Balancer 的 HTTPS 接聽程式)。
> - 針對 CloudFront，請參閱受支援的 SSL/TLS 協定和密碼。

受管續約和部署

ACM 會管理 ACM 憑證的續約和佈建。自動續約有助於防止停機時間設定錯誤、撤銷或過期的憑證。如需詳細資訊，請參閱中的受管憑證續約 AWS Certificate Manager。

多個網域名稱

每個 ACM 憑證必須至少包含一個完整網域名稱 (FQDN)，並且可以包含其他名稱。例如， 的憑證`www.example.com`也可以包含 `www.example.net`。這也適用於裸機網域 （區域頂點或裸機網域）。您可以為 www.example.com 請求憑證，並包含 example.com。如需詳細資訊，請參閱AWS Certificate Manager 公有憑證。

Punycode

必須符合下列國際化網域名稱的 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
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

有效期間

ACM 憑證的有效期限為 13 個月 (395 天)。

萬用字元名稱

ACM 允許網域名稱中的星號 (*) 建立萬用字元憑證，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。

在萬用字元憑證中，星號 (*) 必須保留在網域名稱中，並僅保護一個子網域層級。例如，*.example.com會保護 login.example.com和 test.example.com，但不會保護 test.login.example.com。此外， 僅*.example.com保護子網域，而不是裸機或頂點網域 (example.com)。您可以指定多個網域名稱，例如 example.com和 ，同時為裸機網域及其子網域請求憑證*.example.com。

> **Important**
>
> 如果您使用 CloudFront，請注意 HTTP 驗證不支援萬用字元憑證。對於萬用字元憑證，您必須使用 DNS 驗證或電子郵件驗證。我們建議您進行 DNS 驗證，因為它支援自動憑證續約。

在 中請求公有憑證 AWS Certificate Manager

您可以從 ACM 主控台 AWS CLI或 API 請求 AWS Certificate Manager 公有憑證。您可以將這些憑證與整合的 搭配使用， AWS 服務 或將其匯出以供 外部使用 AWS 雲端。

下列清單說明公有憑證和可匯出公有憑證之間的差異。

公用憑證

使用 ACM 公有憑證搭配整合 AWS 服務 的 Elastic Load Balancing、Amazon CloudFront 和 Amazon API Gateway。如需詳細資訊，請參閱與 ACM 整合的服務。

> **Note**
>
> 2025 年 6 月 17 日之前建立的 ACM 公有憑證無法匯出。

可匯出的公有憑證

可匯出的公有憑證可與整合式 搭配使用 AWS 服務 ，也可以在外部使用 AWS 雲端。如需詳細資訊，請參閱AWS Certificate Manager 可匯出的公有憑證及與 ACM 整合的服務。您必須建立新的 ACM 公有憑證，並啟用可匯出，才能匯出公有憑證。

下列各節討論如何請求、匯出和撤銷公有 ACM 憑證。

主題

- 使用主控台請求公有憑證
- 使用 CLI 請求公有憑證
- AWS Certificate Manager 可匯出的公有憑證
- 匯出 AWS Certificate Manager 公有憑證
- 撤銷 AWS Certificate Manager 公有憑證
- 設定自動續約事件
- 強制憑證續約

使用主控台請求公有憑證

請求 ACM 公有憑證 (主控台)

1. 登入 AWS 管理主控台，並在 https：//https://console.aws.amazon.com/acm/home 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 bare 或 apex 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**。萬用字元名稱會顯示在 ACM 憑證的主旨欄位和主旨別名延伸中。

   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**)。若要保護兩者，請參閱下一個步驟。

> **Note**
>
> 為遵循 RFC 5280，您在此步驟中輸入的網域名稱 (技術上來說為「通用名稱」) 長不得超過 64 個八位元組 (字元)，包括句點在內。但您之後提供的每個主體別名 (SAN) 長度最高可達 253 個八位元，如同下個步驟所示。

- 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. 如果您想要建立 ACM 匯出公有憑證，請選取啟用匯出選項。您將能夠存取憑證的私有金鑰，並在外部使用 AWS 雲端。如需詳細資訊，請參閱AWS Certificate Manager 可匯出的公有憑證。
4. 根據您的需求，在 Validation method (驗證方法) 區段，選擇 DNS validation – recommended (DNS 驗證 – 建議) 或 Email validation (電子郵件驗證)。

> **Note**
>
> 如果您能編輯 DNS 組態，我們建議您使用 DNS 網域驗證，而不使用電子郵件驗證。與電子郵件驗證相比，DNS 驗證有多個優點。請參閱 AWS Certificate Manager DNS 驗證。

ACM 發行憑證前，會先驗證您是否擁有或控制憑證請求中的網域名稱。您可以使用電子郵件驗證或 DNS 驗證。

a. 如果您選擇電子郵件驗證，ACM 會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，ACM 會改為將電子郵件傳送至該驗證網域。如需電子郵件驗證的詳細資訊，請參閱「AWS Certificate Manager 電子郵件驗證」。
b. 如果您使用 DNS 驗證，則只需將 ACM 提供的 CNAME 記錄新增至您的 DNS 組態。如需 DNS 驗證的詳細資訊，請參閱 AWS Certificate Manager DNS 驗證。

5. 在金鑰演算法區段中，選擇演算法。
6. 在 Tags (標籤) 頁面上，您可以選擇標記您的憑證。標籤是鍵值對，可做為識別和組織 AWS 資源的中繼資料。如需 ACM 標籤參數清單以及如何在建立後將標籤新增至憑證的相關說明，請參閱「標籤 AWS Certificate Manager 資源」。

   完成新增標籤後，請選擇 Request (請求)。
7. 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 會重複嘗試驗憑證 72 小時，然後逾時。如果憑證顯示失敗或者驗證逾時狀態，請刪除請求，修正 DNS 驗證或者電子郵件驗證問題，然後重試。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。

使用 CLI 請求公有憑證

在命令列中使用 request-certificate 命令來請求新的公有 ACM 憑證。驗證方法的可選值是 DNS 和 EMAIL (電子郵件)。金鑰演算法的選用值為 RSA_2048 (若未明確提供參數，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 Export=ENABLED
```

此命令會輸出新公有憑證的 Amazon Resource Name (ARN)。

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

AWS Certificate Manager 可匯出的公有憑證

AWS Certificate Manager 可匯出的公有憑證可讓您佈建、管理和部署 SSL/TLS 憑證，包括 Amazon EC2 執行個體、容器和內部部署主機。此功能會將 ACM 發行的公有憑證延伸到整合之外 AWS 服務，讓您集中控制整個基礎設施的憑證。

優勢

以下概述 ACM 匯出公有憑證的優點：

- 簡化憑證管理：使用 ACM 集中管理所有 資源的憑證。
- 更快速的憑證發行：在更短的時間內存取和使用憑證。
- 自動化續約：ACM 會自動處理憑證續約，並在新憑證準備好進行部署時通知您。如需詳細資訊，請參閱ACM 的 Amazon EventBridge 支援。
- 成本效益：只需為您建立的可匯出公有憑證付費。
- 彈性部署：搭配支援標準 SSL/TLS 憑證的任何伺服器或應用程式使用憑證。

ACM 匯出公有憑證的運作方式

以下概述 ACM 匯出公有憑證的運作方式：

1. 透過 ACM 為您的網域請求可匯出的憑證。
2. 使用 DNS 或電子郵件驗證來驗證網域擁有權。
3. 匯出憑證、私有金鑰和憑證鏈。
4. 將憑證部署到您的伺服器或應用程式。
5. ACM 會管理續約，並在有新憑證可用時傳送通知。

安全考量

以下是使用 ACM 匯出公有憑證時的安全考量。如需詳細資訊，請參閱中的資料保護 AWS Certificate Manager。

- 使用安全儲存和存取控制保護匯出的私有金鑰。
- 如果您懷疑金鑰遭到入侵，請使用 ACM 的撤銷功能。
- 部署續約的憑證時，請實作適當的金鑰輪換程序。

限制

以下是一些 ACM 憑證限制：

- 憑證的有效期間為 13 個月 (395 天）。
- ACM 會在 11 個月後續約憑證。ACM 會將設定為過期日期前 60 天的憑證續約。
- 您必須管理匯出憑證的部署程序。

定價

您需要為使用 建立的可匯出公有 SSL/TLS 憑證支付額外費用 AWS Certificate Manager。如需最新的 ACM 定價資訊，請參閱 AWS 網站上的AWS Certificate Manager 服務定價頁面。

最佳實務

以下是使用 ACM 憑證時的一些最佳實務：

- 憑證續約後，您應該立即開始使用。
- 測試和實作更新憑證的自動化部署程序。
- 使用 Amazon EventBridge 指標和警示監控憑證部署。

匯出 AWS Certificate Manager 公有憑證

下列程序會逐步解說如何在 ACM 主控台中匯出 ACM 公有憑證。或者，您可以使用 `export-certificate` AWS CLI 或 ExportCertificate API 動作。

> **Note**
>
> 2025 年 6 月 17 日之前建立的 ACM 公有憑證無法匯出。

匯出公有憑證（主控台）

1. 登入 AWS Management Console 並開啟位於 https：//https://console.aws.amazon.com/acm/ 的 ACM 主控台。
2. 選擇列出憑證，然後選取您要匯出之憑證的核取方塊。
   - 或者，您可以選取憑證。在憑證詳細資訊頁面中，選取匯出。
3. 選擇更多動作，然後選擇匯出。
4. 輸入並確認私有金鑰的密碼短語。
5. 您可以下載或複製憑證檔案。

> **Note**
>
> 在 ACM 主控台中，您可以匯出 .pem 憑證檔案。您可以將 .pem 檔案轉換為其他檔案格式，例如 .ppk。如需詳細資訊，請參閱此 re：Post 文章。

匯出公有憑證 (AWS CLI)

使用 export-certificate AWS CLI 命令或 ExportCertificate API 動作匯出公有憑證和私有金鑰。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。

```
[Windows/Linux]$ aws acm export-certificate \
    --certificate-arn arn:aws:acm:us-east-1:111122223333: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-----
```

若要將所有內容輸出到檔案，請將>重新導向附加到上一個範例，產生下列命令：

```
$ aws acm export-certificate \
     --certificate-arn arn:aws:acm:us-east-1:111122223333:certificate/certificate_ID \
     --passphrase fileb://path-to-passphrase-file \
     | jq -r '"\(.Certificate)\(.CertificateChain)\(.PrivateKey)"' \
     > /tmp/export.txt
```

撤銷 AWS Certificate Manager 公有憑證

您可以使用 ACM 主控台 AWS CLI或 API 動作撤銷可 AWS Certificate Manager 匯出的公有憑證。

您可能需要撤銷憑證，以符合組織的政策或緩解金鑰洩露。撤銷憑證時需要一個原因。可以使用下列原因：

- 未指定
- 關聯已變更
- 已取代
- 停止操作

若要進一步了解，請參閱 Amazon Trust Services 憑證訂閱者協議和 Amazon Trust Service。

AWS 提供兩種檢查憑證撤銷的服務：線上憑證狀態通訊協定 (OCSP) 和憑證撤銷清單。使用 OCSP，用戶端會查詢授權撤銷資料庫，以即時傳回狀態。OCSP 取決於內嵌在憑證中的驗證資訊。

考量事項

以下是撤銷憑證前的考量事項：

- 您只能撤銷先前匯出的憑證。
- 您無法撤銷不可匯出的公有憑證。如果您不再需要這些憑證，您應該改為將其刪除。
- 如果您不再需要憑證，您應該刪除憑證，而不是撤銷憑證。
- 憑證撤銷程序是全域的。您選擇撤銷的所有有效憑證都會與其相關聯的 ARNs一併撤銷。
- 憑證撤銷是永久的。您無法擷取撤銷的憑證以重複使用。
- 憑證撤銷最多可能需要 24 小時才會生效。

撤銷憑證（主控台）

下列程序會逐步解說如何撤銷 ACM 公有或私有憑證。

1. 登入 AWS Management Console 並開啟位於 https：//https://console.aws.amazon.com/acm/ 的 ACM 主控台。
2. 選擇列出憑證，然後選取您要撤銷之憑證的核取方塊。
   - 或者，您可以選取憑證。在憑證詳細資訊頁面中，選取撤銷。
3. 選擇更多動作，然後選擇撤銷。
4. 出現對話方塊，您必須提供撤銷原因，輸入 **`revoke`**，然後選擇撤銷。

> **Warning**
>
> 一旦憑證遭到撤銷，您就無法重複使用憑證。撤銷憑證是永久的。

撤銷憑證 (AWS CLI)

使用 `revoke-certificate` AWS CLI 命令或 `RevokeCertificate` API 動作來撤銷 ACM 公有或私有憑證。您可以呼叫 `list-certificates`命令來擷取憑證的 ARN。

```
$ aws acm revoke-certificate \
    --certificate-arn arn:aws:acm:us-
east-1:111122223333:certificate/12345678-1234-1234-1234 \
    --revocation-reason "UNSPECIFIED"
```

> **Warning**
>
> 一旦憑證遭到撤銷，您就無法重複使用憑證。撤銷憑證是永久的。

以下是 `revoke-certificate`命令的輸出。

```
arn:aws:acm:us-east-1:111122223333:certificate/12345678-1234-1234-1234
```

設定自動續約事件

透過可 AWS Certificate Manager 匯出的公有憑證和 Amazon EventBridge，您可以設定自動憑證續約事件。

1. 設定 Amazon EventBridge 事件以監控憑證續約。如需詳細資訊，請參閱 ACM 的 Amazon EventBridge 支援。
2. 建立自動化以在續約時處理憑證部署。如需詳細資訊，請參閱在 ACM 中使用 Amazon EventBridge 啟動動作。
3. 設定 EventBridge 事件，以提醒您任何續約或部署失敗。

強制憑證續約

您可以使用 ACM 主控台、 renew-certificate AWS CLI或 `RenewCertificate` API 動作來續約 ACM 公有和私有憑證。您只能續約先前匯出的憑證。

> **Important**
>
> 當您續約 ACM 匯出公有憑證時，需支付額外費用。如需最新的 ACM 定價資訊，請參閱 AWS 網站上的AWS Certificate Manager 服務定價頁面。

續約憑證（主控台）

下列程序會逐步解說如何強制續約 ACM 公有或私有憑證。

1. 登入 AWS Management Console 並開啟位於 https：//https://console.aws.amazon.com/acm/ 的 ACM 主控台。
2. 選擇列出憑證，然後選取您要續約之憑證的核取方塊。

- 或者，您可以選取憑證。在憑證詳細資訊頁面中，選取續約。

3. 選擇更多動作，然後選擇續約。
4. 出現對話方塊，您必須在其中輸入 **renew**，然後選擇續約。

續約憑證 (AWS CLI)

使用 renew-certificate AWS CLI 命令或 RenewCertificate API 動作來續約 ACM 公有或私有憑證。您可以呼叫 list-certificates命令來擷取憑證的 ARN。renew-certificate 命令不會傳回回應。

```
$ aws acm renew-certificate \
    --certificate-arn arn:aws:acm:us-
east-1:111122223333:certificate/12345678-1234-1234-1234-123456789012
```

驗證 AWS Certificate Manager 公有憑證的網域擁有權

在 Amazon 憑證授權機構 (CA) 可以為您的網站發出憑證之前， 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。當您請求憑證時，您可以選擇使用網域名稱系統 (DNS) 驗證、電子郵件驗證或 HTTP 驗證來證明擁有權。

> **Note**
>
> 驗證僅適用於 ACM 發行的公開信任憑證。ACM 不會為匯入的憑證或由私有 CA 簽署的憑證驗證網域所有權。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，請參閱對憑證驗證進行故障診斷。

我們建議您在電子郵件驗證時使用 DNS 驗證，原因如下：

- 如果您使用 Amazon Route 53 管理您的公有 DNS 記錄，可以直接透過 ACM 更新您的記錄。
- 只要憑證使用中，而且保有 CNAME 記錄，ACM 便會自動續約經 DNS 驗證的憑證。
- 電子郵件驗證憑證需要網域擁有者續約 動作。ACM 開始在過期前 45 天傳送續約通知。這些通知會前往網域的五個常見管理員地址中的一個或多個。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，ACM 會發行具有相同 ARN 的續約憑證。

如果您無法編輯網域的 DNS 資料庫，則必須改用電子郵件驗證。

HTTP 驗證適用於與 CloudFront 搭配使用的憑證。此方法使用 HTTP 重新導向來證明網域擁有權，並提供類似於 DNS 驗證的自動續約。

> **Note**
>
> 使用電子郵件驗證建立憑證之後，您無法切換為使用 DNS 進行驗證。若要使用 DNS 驗證，請刪除憑證，然後建立使用 DNS 驗證的新憑證。

主題

AWS Certificate Manager DNS 驗證

網域名稱系統 (DNS) 是一個目錄服務，適用於連接到網路的資源。您的 DNS 供應商會維護一個資料庫，其中包含定義您網域的記錄。當您選擇 DNS 驗證，ACM 會提供一或多個 CNAME 記錄，這些記錄必須新增至此資料庫中。這些記錄包含唯一的鍵值組，可作為您控制網域的證明。

> **Note**
>
> 使用電子郵件驗證建立憑證之後，您無法切換為使用 DNS 進行驗證。若要使用 DNS 驗證，請刪除憑證，然後建立使用 DNS 驗證的新憑證。

例如，如果您為 `example.com` 網域請求憑證，並以 `www.example.com` 做為額外名稱，ACM 會為您建立兩筆 CNAME 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。此值是指向 ACM 用來自動續約憑證的 AWS 網域的別名。必須將 CNAME 記錄新增到 DNS 資料庫一次。只要憑證使用中，而且保有 CNAME 記錄，ACM 便會自動續約憑證。

> **Important**
>
> 如果您不是使用 Amazon Route 53 來管理公有 DNS 記錄，請聯絡您的 DNS 供應商以了解如何新增記錄。如果您沒有編輯網域 DNS 資料庫的授權，則必須改用電子郵件驗證。

無需重複驗證，只要保有 CNAME 記錄，您就可以為完整網域名稱 (FQDN) 請求額外的 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 驗證字符適用於任何 AWS 區域，因此您可以在多個區域中重新建立相同的憑證。您也可以取代已刪除的憑證。

您可以從 AWS 服務移除相關聯的憑證或刪除 CNAME 記錄，以停止自動續約。如果您的 DNS 供應商不是 Route 53，請聯絡您的供應商，了解如何刪除記錄。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「中的受管憑證續約 AWS Certificate Manager」。

> **Note**
>
> 如果您的 DNS 組態中有超過五個 CNAME 鏈結在一起，CNAME 解析將會失敗。如果您需要更長的鏈結，建議使用電子郵件驗證。

ACM 的 CNAME 記錄運作方式

> **Note**
>
> 本節適用於採用 Route 53 做為 DNS 供應商的客戶。

如果您不是使用 Route 53 做為 DNS 供應商，則需要手動將 ACM 提供的 CNAME 記錄輸入供應商的資料庫 (通常是透過網站操作)。CNAME 記錄用於許多用途，包括重新引導機制以及供應商專屬中繼資料的容器。對 ACM 來說，有了這些記錄，才能進行初始網域所有權驗證和持續自動化憑證續約。

下表顯示六個網域名稱的 CNAME 記錄範例。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`，ACM 建立的字串與其基本網域 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。

CNAME 記錄範例

網域名稱	記錄名稱	記錄值	註解
*.example.com	*_x1*.example.com。	*_x2*.acm-validations.aws。	Identical (相同)

網域名稱	記錄名稱	記錄值	註解
example。com	*_x1*.example.com。	*_x2*.acm-validations.aws。	
www.example.com	*_x3*.www.example.com。	*_x4*.acm-validations.aws。	唯一
host.example.com	*_x5*.host.example.com。	*_x6*.acm-validations.aws。	唯一
subdomain.example.com	*_x7*.subdomain.example.com。	*_x8*.acm-validations.aws。	唯一
host.subdomain.example.com	*_x9*.host.subdomain.example.com。	*_x10*.acm-validations.aws。	唯一

底線 (_) 後的 *xN* 值為 ACM 產生的長字串。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.
```

針對相同的 DNS 記錄。

> **Note**
>
> 如果您的 DNS 供應商不支援包含前置底線的 CNAME 值，請參閱針對 DNS 驗證問題進行疑難排解。

當您請求憑證並指定 DNS 驗證時，ACM 會以下列格式提供 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。

網域名稱是憑證的相關聯 FQDN。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 (Domain Name (網域名稱)、Record Name (記錄名稱) 和 Record Value (記錄值)) 都必須輸入 DNS 供應商 Web 介面上用於新增 DNS 記錄的適用欄位中。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有憑證以使用 DNS 驗證。

在主控台上設定 DNS 驗證

> **Note**
>
> 此程序假設您已建立至少一個憑證，且您正在建立憑證的 AWS 區域中工作。如果您嘗試開啟主控台並改為看到第一次使用畫面，或者您成功開啟主控台，但未在清單中看到您的憑證，請確認您已指定正確的區域。

1. 前往 https://console.aws.amazon.com/acm/ 開啟 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 做為 DNS 供應商。
      - 您擁有寫入 Route 53 託管區域的許可。
      - 您的 FQDN 未經驗證。

      > **Note**
      >
      > 如果您實際上使用 Route 53，但 Route 53 中的建立記錄遺失或停用，請參閱 ACM 主控台未顯示「在 Route 53 中建立記錄」按鈕。

      選擇在 Route 53 中建立記錄，然後選擇建立記錄。所以此 Certificate status（憑證狀態）頁面應該開啟並顯示狀態橫幅報告 Successfully created DNS records（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**
      >
      > 您無法透過編寫程式的方式請求 ACM 自動在 Route 53 中建立記錄。不過，您可以對 Route 53 進行 AWS CLI 或 API 呼叫，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. (選用) 如果您不是使用 Route 53 做為 DNS 供應商，您必須擷取 CNAME 資訊並新增至 DNS 資料庫。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：

      - 複製顯示在 Domains（網域）部分的 CNAME 元件。這些資訊需手動新增至 DNS 資料庫。
      - 或者，選擇 Export to CSV（匯出至 CSV）。結果檔案中的資訊需手動新增至 DNS 資料庫。

> **Important**
>
> 為避免驗證問題，請先檢閱「ACM 的 CNAME 記錄運作方式」，再將資訊新增至您 DNS 供應商的資料庫。如果您遇到問題，請參閱「針對 DNS 驗證問題進行疑難排解」。

如果 ACM 無法在為您產生 CNAME 值後的 72 小時內驗證網域名稱，ACM 會將憑證狀態變更為 Validation timed out (驗證逾時)。此結果最有可能的原因是您未使用 ACM 產生的值成功更新 DNS 組態。若要修正此問題，您必須在檢閱 CNAME 指示之後請求新憑證。

AWS Certificate Manager 電子郵件驗證

在 Amazon 憑證授權機構 (CA) 可以為您的網站發出憑證之前， AWS Certificate Manager (ACM) 必須驗證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。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。

電子郵件驗證的運作方式

ACM 會為每個網域傳送驗證電子郵件訊息到以下五個常見的系統電子郵件。或者，如果您想要改為在該網域接收這些電子郵件，您可以將超級網域指定為驗證網域。最小網站地址之前的任何子網域都是有效的，並在 之後用作電子郵件地址的網域。 @例如，如果您將 example.com 指定為 的驗證網域，則可能會收到 admin@example.com 的電子郵件 subdomain.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。當憑證過期 45 天後，ACM 也會傳送驗證電子郵件到這些相同的地址來續約憑證。

使用 ACM API 或 CLI 對多網域憑證請求進行電子郵件驗證，會導致每個請求的網域傳送電子郵件訊息，即使請求包含請求中其他網域的子網域。網域擁有者必須先驗證每個網域的電子郵件訊息，ACM 才能發行憑證。

此程序的例外情況

如果您為開頭為 **www**或萬用字元星號 (*) 的網域名稱請求 ACM 憑證，ACM 會移除前置**www**或星號，並傳送電子郵件到管理地址。這些地址的形成方式是在網域名稱的剩餘部分加上 admin@、admin@、hostmaster@、postmaster@ 和 webmaster@。例如，如果您為 www.example.com 請求 ACM 憑證，則電子郵件會傳送到 admin@example.com，而非 admin@www.example.com。同樣地，如果您為 *.test.example.com 請求 ACM 憑證，則電子郵件會傳送到 admin@test.example.com。其餘的常用管理地址也是以類似方式形成。

> **Important**
>
> ACM 不再支援新憑證或續約的 WHOIS 電子郵件驗證。常用系統地址仍受支援。如需詳細資訊，請參閱部落格文章。

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 ACM 發行的公開信任憑證。ACM 不會為匯入的憑證或由私有 CA 簽署的憑證驗證網域所有權。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，請參閱對憑證驗證進行故障診斷。
- 使用電子郵件驗證建立憑證之後，您無法切換為使用 DNS 進行驗證。若要使用 DNS 驗證，請刪除憑證，然後建立使用 DNS 驗證的新憑證。

憑證過期日期和續約

ACM 憑證的有效期限為 13 個月 (395 天)。續約憑證需要網域擁有者執行動作。ACM 會在過期前 45 天開始將續約通知傳送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以進行續約的連結。驗證所有列出的網域後，ACM 會發行具有相同 ARN 的續約憑證。

（選用）重新傳送驗證電子郵件

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，請參閱對 的問題進行故障診斷 AWS Certificate Manager中的針對電子郵件驗證問題進行疑難排解一節。

自動化 AWS Certificate Manager 電子郵件驗證

透過電子郵件驗證的 ACM 憑證通常需要網域擁有者手動操作。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
```

一旦您收到來自 的新驗證訊息 AWS，我們建議您將其用作剖析器up-to-date和授權範本。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 現在呈現了整個憑證 ARN 而不是縮短的形式，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 URL 所開啟的核准表單現在包含核准按鈕。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請求時，它會傳送驗證電子郵件訊息到您請求公有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，ACM 才能發行憑證。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 或 CLI 對多網域憑證請求進行電子郵件驗證，會導致每個請求的網域傳送電子郵件訊息，即使請求包含請求中其他網域的子網域。網域擁有者必須先驗證每個網域的電子郵件訊息，ACM 才能發行憑證。

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請求中指定的驗證網域，如果未指定驗證網域，則會傳送至確切的網域。若要在不同的網域接收驗證電子郵件，您可以請求新的憑證，指定要用於驗證的驗證網域。或者，您可以使用 API、 SDK 或 CLI 呼叫 ResendValidationEmail 搭配 `ValidationDomain` 參數。不過，`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 Resource Name (ARN) 以供未來驗證電子郵件使用。每次您想要以原始憑證請求中未指定的網域名稱接收驗證電子郵件`ResendValidationEmail`時，都必須呼叫 。

> **Note**
>
> 在 2020 年 11 月之前，客戶只需要驗證頂層網域，ACM 就會發行同樣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 ACM API 或 CLI 時，您可以強制將多網域憑證請求的所有驗證電子郵件訊息傳送至頂層網域。在 API 中，使用 RequestCertificate 動作的 `DomainValidationOptions` 參數來指定 `ValidationDomain` 的值，其為 DomainValidationOption 類型的成員。在 CLI 中，使用 request-certificate 命令的 --domain-validation-options 參數來指定 `ValidationDomain` 的值。

AWS Certificate Manager HTTP 驗證

超文字傳輸通訊協定 (HTTP) 是全球資訊網上資料通訊的基礎通訊協定。當您為與 CloudFront 搭配使用的憑證選擇 HTTP 驗證時，ACM 會利用此通訊協定來驗證您的網域擁有權。ACM 可與 CloudFront 搭配使用，為您提供特定的 URL 和唯一的權杖，該權杖必須在網域上的該 URL 上進行存取。此字符可做為您控制網域的證明。透過設定從網域重新導向至 CloudFront 基礎設施內的 ACM 控制位置，您可以示範修改網域上內容的能力，藉此驗證您的擁有權。ACM 和 CloudFront 之間的無縫整合可簡化憑證發行程序，尤其是 CloudFront 分發。

> **Important**
>
> HTTP 驗證不支援萬用字元網域憑證 （ 例如 *.example.com)。對於萬用字元憑證，您必須改用 DNS 驗證或電子郵件驗證。

例如，如果您使用 CloudFront 為 `example.com` 網域請求憑證`www.example.com`作為額外的名稱，ACM 會為您提供兩組 URLs以進行 HTTP 驗證。每個集合都包含 `redirectFrom` URL 和 `redirectTo` URL，專為您的網域和 AWS 帳戶建立。`redirectFrom` URL 是您網域上需要設定的路徑 （ 例如 `http://example.com/.well-known/pki-validation/`

example.txt)。redirectTo URL 指向 CloudFront 基礎設施中存放唯一驗證字符的 ACM 控制位置。您只需要設定這些重新導向一次。當憑證授權機構嘗試驗證您的網域擁有權時，它會從 redirectFrom URL 請求檔案，CloudFront 會將該 URL 重新導向至 redirectTo URL，以允許存取驗證字符。只要憑證與 CloudFront 搭配使用，且您的重新導向仍然存在，ACM 就會自動續約您的憑證。

使用 CloudFront 設定完整網域名稱 (FQDN) 的 HTTP 驗證後，您可以為該 FQDN 請求額外的 ACM 憑證，而無需重複驗證程序，只要 HTTP 重新導向仍然存在。這表示您可以使用相同的網域名稱或涵蓋不同子網域的憑證來建立替代憑證。由於 HTTP 驗證字符適用於任何可使用 CloudFront AWS 的區域，因此您可以在多個區域中重新建立相同的憑證。您也可以取代已刪除的憑證，而不需再次進行驗證程序，前提是重新導向仍處於作用中狀態。

若要停止 HTTP 驗證憑證的自動續約，您有兩個選項。您可以從與其相關聯的 CloudFront 分佈中移除憑證，也可以刪除您設定用於驗證的 HTTP 重新導向。如果您使用 CloudFront 以外的內容交付網路 (CDN) 或 Web 伺服器來管理重新導向，請參閱其文件以了解如何移除重新導向。如果您使用 CloudFront 來管理重新導向，您可以透過更新分佈的組態來移除重新導向。如需受管的憑證續約的詳細資訊，請參閱「中的受管憑證續約 AWS Certificate Manager」。請記住，停止自動續約可能會導致憑證過期，這可能會中斷您的 HTTPS 流量。

ACM 的 HTTP 重新導向如何運作

> **Note**
>
> 本節適用於使用 CloudFront 進行內容交付和 ACM for SSL/TLS 憑證管理的客戶。

搭配 ACM 和 CloudFront 使用 HTTP 驗證時，您需要設定 HTTP 重新導向。這些重新導向可讓 ACM 驗證您的網域擁有權，以進行初始憑證發行和持續自動續約。重新導向機制的運作方式是將網域上的特定 URL 指向 CloudFront 基礎設施中存放唯一驗證字符的 ACM 控制位置。

下表顯示網域名稱的重新導向組態範例。請注意，HTTP 驗證不支援萬用字元網域（例如 *.example.com)。每個組態的重新導向自重新導向至 配對都用於驗證網域名稱擁有權。

HTTP 重新導向組態範例

網域名稱	從 重新導向	重新導向至	註解
example。com		https://validation. *region*.acm-	唯一

網域名稱	從 重新導向	重新導向至	註解
	http://example.com/.well-known/pki-validation/ *x2*.txt	validations.aws/ *y2*/.well-known/pki-validation/ *x2*.txt	
www.example.com	http://www.example.com/.well-known/pki-validation/ *x3*.txt	https://validation. *region*.acm-validations.aws/ *y3*/.well-known/pki-validation/ *x3*.txt	唯一
host.example.com	http://host.example.com/.well-known/pki-validation/ *x4*.txt	https://validation. *region*.acm-validations.aws/ *y4*/.well-known/pki-validation/ *x4*.txt	唯一
subdomain.example.com	http://subdomain.example.com/.well-known/pki-validation/ *x5*.txt	https://validation. *region*.acm-validations.aws/ *y5*/.well-known/pki-validation/ *x5*.txt	唯一
host.subdomain.example.com	http://host.subdomain.example.com/.well-known/pki-validation/ *x6*.txt	https://validation. *region*.acm-validations.aws/ *y6*/.well-known/pki-validation/ *x6*.txt	唯一

檔案名稱中的 *xN* 值和 ACM 控制網域中的 *yN* 值是 ACM 產生的唯一識別符。例如

```
http://example.com/.well-known/pki-validation/3639ac514e785e898d2646601fa951d5.txt
```

代表產生的重新導向來源 URL。關聯的重新導向至 URL 可能是

```
https://validation.region.acm-validations.aws/98d2646601fa/.well-known/pki-
validation/3639ac514e785e898d2646601fa951d5.txt
```

相同的驗證記錄。

> **Note**
>
> 如果您的 Web 伺服器或內容交付網路不支援在指定路徑設定重新導向，請參閱對 HTTP 驗證問題進行故障診斷。

當您請求憑證並指定 HTTP 驗證時，ACM 會以下列格式提供重新導向資訊：

網域名稱	重新導向至
example。com	https://validation.*region*.acm-validations.aws/*a424c7224e9b* /.well-known/pki-validation/*a79865eb4cd1a6ab990a45779b4e0b96* .txt

網域名稱	重新導向至

網域名稱是憑證的相關聯 FQDN。Redirect From 是您網域上的 URL，ACM 會在其中尋找驗證檔案。重新導向至 是託管實際驗證檔案的 ACM 控制 URL。

您需要設定 Web 伺服器或 CloudFront 分佈，將請求從重新導向自 URL 重新導向至重新導向至 URL。設定此重新導向的確切方法取決於您的 Web 伺服器軟體或 CloudFront 組態。確保正確設定重新導向，以允許 ACM 驗證您的網域擁有權，並發行或續約您的憑證。

設定 HTTP 驗證

ACM 在發行公有 SSL/TLS 憑證以搭配 CloudFront 使用時，會使用 HTTP 驗證來驗證您的網域擁有權。本節說明如何設定公有憑證以使用 HTTP 驗證。

在主控台中設定 HTTP 驗證

> **Note**
>
> 此程序假設您已透過 CloudFront 請求憑證，且您正在建立憑證的 AWS 區域中工作。HTTP 驗證只能透過 CloudFront 分佈租用戶功能使用。

1. 前往 https://console.aws.amazon.com/acm/ 開啟 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. 在網域區段中，您可以查看憑證請求中每個網域的重新導向來源和重新導向至值。
4. 對於每個網域，設定從 URL 重新導向至 URL 的 HTTP 重新導向。您可以透過 CloudFront 分佈組態執行此操作。
5. 設定您的 CloudFront 分佈，將請求從重新導向自 URL 重新導向至重新導向至 URL。設定此重新導向的方法取決於您的 CloudFront 組態。

6. 設定重新導向之後，ACM 會自動嘗試驗證您的網域擁有權。此程序最多需要 30 分鐘的時間。

如果 ACM 無法在為您產生重新導向值的 72 小時內驗證網域名稱，ACM 會將憑證狀態變更為驗證逾時。此結果最可能的原因是您未成功設定 HTTP 重新導向。若要修正此問題，您必須在檢閱重新導向指示後請求新的憑證。

> **Important**
>
> 為了避免驗證問題，請確定重新導向來源位置的內容與重新導向至位置的內容相符。如果您遇到問題，請參閱故障診斷 HTTP 驗證問題。

> **Note**
>
> 與 DNS 驗證不同，您無法以程式設計方式請求 ACM 自動建立 HTTP 重新導向。您必須透過 CloudFront 分佈設定來設定這些重新導向。

如需 HTTP 驗證如何運作的詳細資訊，請參閱 ACM 的 HTTP 重新導向如何運作。

中的私有憑證 AWS Certificate Manager

如果您有權存取 建立的現有私有 CA AWS 私有 CA， AWS Certificate Manager (ACM) 可以請求適用於私有金鑰基礎設施 (PKI) 的憑證。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。

預設情況下，私有 CA 簽署的憑證不受信任，且 ACM 不為此支援任何形式的驗證。因此，管理員必須採取行動，將它們安裝在組織的用戶端信任存放區中。

私有 ACM 憑證遵循 X.509 標準，且受制於下列各項限制：

- 名稱:您必須使用符合 DNS 規範的主旨名稱。如需詳細資訊，請參閱網域名稱。
- 演算法：針對加密，憑證私有金鑰演算法必須是 2048 位元 RSA、256 位元的 ECDSA 或 384 位元 ECDSA。

  > **Note**
  >
  > 指定簽署演算法系列 (RSA 或 ECDSA) 必須符合 CA 私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 會在 11 個月後自動嘗試續約私有憑證。

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 私有金鑰演算法必須是 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。

使用 AWS Private CA 簽署 ACM 私有憑證的條件

您可以在兩種情況下使用 AWS 私有 CA 簽署 ACM 憑證：

- 單一帳戶：簽署 CA 和發行的 AWS Certificate Manager (ACM) 憑證位於同一個 AWS 帳戶中。

  若要啟用單一帳戶發行和續約功能， AWS 私有 CA 管理員必須授與許可給 ACM 服務主體，才能建立、擷取和列出憑證。這是使用 API AWS 私有 CA 動作 CreatePermission 或 AWS CLI 命令 create-permission 來完成。帳戶擁有者會將這些許可指派給負責發行憑證的 IAM 使用者、群組或角色。
- 跨帳戶：簽署 CA 和發行的 ACM 憑證位於不同的 AWS 帳戶中，並且已將 CA 的存取權授予憑證所在的帳戶。

  若要啟用跨帳戶發行和續約，管理員必須使用 AWS 私有 CA API AWS 私有 CA 動作 PutPolicy 或 AWS CLI 命令 put-policy，將資源型政策連接至 CA。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 Private CA 使用資源型政策。

  跨帳戶案例也需要 ACM 設定服務連結角色 (SLR)，才能以委託人身分與 PCA 政策互動。ACM 會在發行第一個憑證時自動建立 SLR。

  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，或為您的帳戶與由 ACM 管理的政策 `AWSCertificateManagerFullAccess` 建立關聯。

如需詳細資訊，請參閱搭配 ACM 使用服務連結角色。

> **Important**
>
> 您的 ACM 憑證必須主動與支援的 AWS 服務建立關聯，才能自動續約。如需有關 ACM 支援的資源的資訊，請參閱 與 ACM 整合的服務。

在 中請求私有憑證 AWS Certificate Manager

請求私有憑證 （主控台）

1. 登入 AWS 管理主控台，並在 https：//https://console.aws.amazon.com/acm/home 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. 在憑證授權機構詳細資訊區段中，選取憑證授權機構選單，然後選擇其中一個可用的私有 CAs。如果 CA 是從另一個帳戶共用，ARN 前面會加上所有權資訊。

   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 bare 或 apex 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**。萬用字元名稱會顯示在 ACM 憑證的主旨欄位和主旨別名延伸中。

> **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**)。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，選擇演算法。

   如需協助您選擇演算法的資訊，請參閱 AWS 部落格文章如何評估和使用 中的 ECDSA 憑證 AWS Certificate Manager。

6. 在 Tags（標籤）部分，您可以選擇標記您的憑證。標籤是鍵值組，可做為識別和組織 AWS 資源的中繼資料。如需 ACM 標籤參數清單以及如何在建立後將標籤新增至憑證的相關說明，請參閱「標籤 AWS Certificate Manager 資源」。

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 ACM 使用服務連結角色。

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。

請求私有憑證 (CLI)

使用 request-certificate 命令在 ACM 中請求私有憑證。

> **Note**
>
> 當您請求由 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 私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 Resource Name (ARN)。

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

在大多數情況下，ACM 會在您第一次使用共用 CA 時，自動將服務連結角色 (SLR) 連接至您的帳戶。SLR 會為您發行的終端實體憑證啟用自動續約功能。若要檢查 SLR 是否存在，您可以使用以下命令查詢 IAM：

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

如果 SLR 存在，命令輸出應類似以下內容：

```
{
   "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"
      }
   }
}
```

如果缺少 SLR，請參閱搭配 ACM 使用服務連結角色。

匯出 AWS Certificate Manager 私有憑證

您可以匯出 發行的憑證 AWS 私有 CA ，以便在私有 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。如需 的詳細資訊 AWS 私有 CA，請參閱 AWS Private Certificate Authority 使用者指南。

> **Note**
>
> 您無法匯出公開信任的憑證或其私有金鑰，無論其是由 ACM 發行或匯入。

主題

- 匯出私有憑證（主控台）
- 匯出私有憑證 (CLI)

匯出私有憑證（主控台）

1. 登入 AWS 管理主控台，並在 https：//https://console.aws.amazon.com/acm/home 開啟 ACM 主控台。
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。

```
[Windows/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)"'
```

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-----
```

若要將所有內容輸出到檔案，請將>重新導向附加到上一個範例，產生以下內容。

```
$ 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
```

將憑證匯入至 AWS Certificate Manager

除了請求 AWS Certificate Manager (ACM) 提供的 SSL/TLS 憑證之外，您還可以匯入您在 外部取得的憑證 AWS。這樣做的原因是您已經擁有第三方憑證授權機構 (CA) 提供的憑證，或 ACM 發行的憑證不符合應用程式特有的需求。

您可以將匯入的憑證與任何AWS 與 ACM 整合的服務一起使用。您匯入的憑證運作方式與 ACM 提供的憑證相同，但有個重要例外：ACM 不為匯入的憑證提供受管續約。

若要續約匯入的憑證，您可以向憑證發行者索取新憑證，然後手動匯入到 ACM。此動作將保留憑證的關聯及其 Amazon Resource Name (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 CloudWatch Events 在匯入的憑證即將過期時傳送通知，以簡化此任務。如需詳細資訊，請參閱使用 Amazon EventBridge。

ACM 中的所有憑證皆為區域性資源，包括您匯入的憑證在內。若要在不同 AWS 區域搭配 Elastic Load Balancing 負載平衡器使用同一個憑證，您必須將憑證匯入到每個需要使用該憑證的區域。若要搭配 Amazon CloudFront 使用憑證，您必須在美國東部 (維吉尼亞北部) 區域請求或匯入憑證。如需詳細資訊，請參閱支援地區。

如需有關如何將憑證匯入 ACM 的資訊，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 ACM 憑證的先決條件

若要將自我簽署的 SSL/TLS 憑證匯入 ACM，您必須提供憑證及其私有金鑰兩者皆需要。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 支援下列演算法 (括號中為 API 名稱)：

- RSA 1024 位元 (`RSA_1024`)
- RSA 2048 位元 (`RSA_2048`)
- RSA 3072 位元 (`RSA_3072`)
- RSA 4096 位元 (`RSA_4096`)
- ECDSA 256 位元 (`EC_prime256v1`)
- ECDSA 384 位元 (`EC_secp384r1`)
- ECDSA 521 位元 (`EC_secp521r1`)

另請注意以下額外要求：

- 請注意，ACM 整合服務僅允許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 RSA、2048 位元 RSA、3072 位元 RSA、4096 位元 RSA 和 Elliptic Prime Curve 256 位元金鑰，而 Application Load Balancer 支援 ACM 提供的所有演算法。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/TLS X.509 版本 3 憑證。憑證必須包含公開金鑰、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。
- 若憑證是由 CA 簽署，且您選擇提供憑證鏈，則憑證鏈必須採用 PEM 編碼。
- 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。
- 所有要求的憑證材料（憑證、私有金鑰和憑證鏈）都必須採用 PEM 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 會要求您分別匯入憑證、憑證鏈和私有金鑰 (如有)，並以 PEM 格式編碼每個元件。PEM 代表隱私權增強式郵件。PEM 格式通常用於代表憑證、憑證要求、憑證鏈和金鑰。一般 PEM 格式檔案的副檔名為 `.pem`，但這不是必要的副檔名。

> **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 (例如轉換檔案格式或擷取金鑰)，請使用免費的開源工具，例如現成可用的 OpenSSL。

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，如果您在 PEM 檔案中不正確地編輯任何字元，或者，如果您新增一或多個空格到任一行的尾端，憑證、憑證鏈或私有金鑰會無效。

Example 1. PEM 編碼的憑證

```
-----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。以下範例顯示 RSA 私有金鑰。

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

下一個範例顯示以 PEM 編碼的橢圓曲線私有金鑰。視您建立金鑰的方式而定，可能不會包含參數區塊。如果包含參數區塊，ACM 會在匯入程序期間使用該金鑰之前移除參數區塊。

```
-----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 、 AWS Management Console AWS CLI或 ACM API，將外部取得的憑證 （即第三方信任服務提供者提供的憑證） 匯入 ACM。下列主題說明如何使用 AWS Management Console 和 AWS CLI。從非AWS 發行者取得憑證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 ACM 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. 前往 https://console.aws.amazon.com/acm/home 開啟 ACM 主控台。如果這是您第一次使用 ACM，請尋找 AWS Certificate Manager 標題並選擇標題下方的開始使用按鈕。
2. 選擇 Import a certificate (匯入憑證)。
3. 請執行下列操作：
   a. 對於 Certificate body (憑證內文)，貼上要匯入的 PEM 編碼憑證。開頭應為 `-----BEGIN CERTIFICATE-----` 而結尾是 `-----END CERTIFICATE-----`。
   b. 針對 Certificate private key (憑證私有金鑰)，請貼上憑證以 PEM 編碼的未加密私有金鑰。開頭應為 `-----BEGIN PRIVATE KEY-----` 而結尾是 `-----END PRIVATE KEY-----`。
   c. (選用) 對於 Certificate chain (憑證鏈)，貼上 PEM 編碼的憑證鏈。
4. （選用）若要將標籤新增至匯入的憑證，請選擇標籤。標籤是您指派給 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。您可以使用標籤來組織資源或追蹤 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 編碼的憑證存放在名為 `Certificate.pem` 的檔案中。
- PEM 編碼的憑證鏈存放在名為 `CertificateChain.pem` 的檔案中。
- PEM 編碼的未加密私有金鑰存放在名為 `PrivateKey.pem` 的檔案中。

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 `import-certificate` 命令成功，它會傳回匯入的憑證的 Amazon Resource Name (ARN)。

重新匯入憑證

如果您匯入憑證並將其與其他 AWS 服務相關聯，您可以在憑證過期之前重新匯入該憑證，同時保留原始憑證 AWS 的服務關聯。如需與 ACM 整合 AWS 之服務的詳細資訊，請參閱 與 ACM 整合的服務。

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. 前往 https://console.aws.amazon.com/acm/home 開啟 ACM 主控台。
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 Certificate body (憑證內文)，貼上 PEM 編碼的最終實體憑證。
5. 對於 Certificate private key (憑證私有金鑰)，貼上與憑證公有金鑰關聯的未加密 PEM 編碼私有金鑰。
6. (選用) 對於 Certificate chain (憑證鏈)，貼上 PEM 編碼的憑證鏈。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 編碼的憑證存放在名為 `Certificate.pem` 的檔案中。
- PEM 編碼的憑證鏈存放在名為 `CertificateChain.pem` 的檔案中。
- (僅限私有憑證) PEM 編碼的未加密私有金鑰會存放在名為 `PrivateKey.pem` 的檔案中。
- 您具有要重新匯入的憑證的 ARN。

若要使用以下範例，請將檔案名稱和 ARN 取代為您自己的檔案名稱和 ARN，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。

> **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 `import-certificate` 命令成功，它會傳回憑證的 Amazon Resource Name (ARN)。

列出由 管理的憑證 AWS Certificate Manager

您可以使用 ACM 主控台或 AWS CLI 列出由 ACM 管理的憑證。主控台可以在一個頁面中列出最多 500 個憑證，而 CLI 則可列出最多 1000 個憑證。

使用主控台列出憑證

1. 前往 https://console.aws.amazon.com/acm/ 開啟 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：

- Domain name (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 憑證是否主動與 Elastic Load Balancing 或 CloudFront 等 AWS 服務相關聯。此值可以是 No (否) 或 Yes (是)。
- Renewal eligibility (續約資格) – 憑證即將到期時，ACM 是否可以自動更新憑證。可能值為：Eligible (符合資格) | Ineligible (不符合資格)。如需資格規則，請參閱 中的受管憑證續約 AWS Certificate Manager。

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量、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)。
- Requested at (請求時間) – ACM 請求憑證的時間。
- 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 檢視 AWS Certificate Manager 憑證詳細資訊。

使用 列出您的憑證 AWS CLI

使用 list-certificates 命令列出由 ACM 管理的憑證，如以下範例所示：

```
$ 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"
        },...
    ]
}
```

根據預設，系統只會傳回具有 keyTypes RSA_1024 或 RSA_2048，以及至少具有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
```

檢視 AWS Certificate Manager 憑證詳細資訊

您可以使用 ACM 主控台或 AWS CLI 來列出憑證的詳細中繼資料。

在主控台中檢視憑證詳細資訊

1. 前往 https://console.aws.amazon.com/acm/ 開啟 ACM 主控台顯示您的憑證。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 - 格式為 `arn:aws:acm:Region:444455556666:certificate/certificate_ID` 的 Amazon Resource Name (ARN)
     - Type (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「AWS Certificate Manager 公有憑證」、「在 中請求私有憑證 AWS Certificate Manager」或「將憑證匯入至 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
     - Domain（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)
     - Domain name（網域名稱) - 憑證的第一個完整網域名稱 (FQDN)。
     - 管理者：識別 AWS 使用 ACM 管理憑證的服務。

- 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。
- Can be used with (可搭配使用) – 支援具有這些參數的憑證之 ACM 整合服務清單
- 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 中的受管憑證續約 AWS Certificate Manager。
- 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 CA 的 ARN

- 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。
  - None (無) - 憑證適用於私有 PKI 或自我簽署，不需要驗證。

使用 檢視憑證詳細資訊 AWS CLI

使用 中的 describe-certificate AWS CLI 來顯示憑證詳細資訊，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 AWS Certificate Manager

您可以使用 ACM 主控台或 AWS CLI 來刪除憑證。刪除票證最終會保持一致。在刪除憑證之後，憑證可能會出現在清單中一小段時間。

> **Important**
>
> - 您無法刪除正在由其他 AWS 服務使用的 ACM 憑證。若要刪除使用中的憑證，您必須先移除憑證關聯。此操作需使用相關聯服務的主控台或 CLI 來完成。

- 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. 前往 https://console.aws.amazon.com/acm/ 開啟 ACM 主控台。
2. 在憑證清單中，選取 ACM 憑證的核取方塊，然後選擇 Delete（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。

使用 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
```

中的受管憑證續約 AWS Certificate Manager

ACM 為 Amazon 發行的 SSL/TLS 憑證提供受管續約服務。這表示 ACM 會自動續約您的憑證 (如果您使用 DNS 驗證)，或在即將過期時傳送電子郵件通知給您。這些服務可供公有和私有 ACM 憑證使用。

根據下列考量，憑證符合自動續約的資格：

- 如果與其他服務相關聯 AWS ，例如 Elastic Load Balancing 或 CloudFront，則符合資格。
- 符合資格 (如果在發行或上次續約之後匯出)。
- 如果是呼叫 ACM RequestCertificate API 發行的私有憑證，然後匯出或與其他 AWS 服務相關聯，則符合資格。
- 如果是透過管理主控台發行，且已匯出或與另一個 AWS 服務相關聯的私有憑證，則符合資格。
- 如果它是透過呼叫 IssueCertificate API 發行的私有憑證，則 AWS 私有 CA 不符合資格。
- 不符合資格 (如果是匯入)。
- 如果已過期，則不符合資格。

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
xn--xyz.com	是	是	✓	有效的國際化網域名稱 (解析為簡.com)

網域名稱	滿足 #1	滿足 #2	允許	注意
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

ACM 續約憑證時，憑證的 Amazon Resource Name (ARN) 保持不變。另外，ACM 憑證為區域性資源。如果您在多個區域中擁有相同網域名稱的憑證 AWS ，則必須個別續約這些憑證。

主題

- 續約 ACM 公有憑證
- 中的私有憑證續約 AWS Certificate Manager
- 檢查憑證的續約狀態

續約 ACM 公有憑證

發行受管、公開信任的憑證時， AWS Certificate Manager 會要求您證明您是網域擁有者。透過 DNS 驗證或電子郵件驗證時，就會發生這種情況。當憑證需要續約時，ACM 會使用您之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

- 續約透過 DNS 驗證的網域
- 電子郵件驗證網域的續約
- HTTP 驗證的網域續約

續約透過 DNS 驗證的網域

原本使用 DNS 驗證的 ACM 憑證之受管續約作業完全自動化，。

ACM 會在過期前 60 天檢查以下續約條件：

- AWS 服務目前正在使用憑證。

- 所有由 ACM 提供的必要 DNS CNAME 記錄 (每個唯一的主題備用名稱一個) 皆存在且可透過公有 DNS 存取。

如果這些條件都符合，ACM 會將網域名稱視為已驗證並續約憑證。

如果在續約期間無法自動驗證網域，ACM 會傳送 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ACM 的 Amazon EventBridge 支援。

電子郵件驗證網域的續約

ACM 憑證的有效期限為 13 個月 (395 天)。續約憑證需要網域擁有者執行動作。ACM 會在過期前 45 天開始將續約通知傳送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以進行續約的連結。驗證所有列出的網域後，ACM 會發行具有相同 ARN 的續約憑證。

如果在續約期間無法自動驗證網域，ACM 會傳送 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ACM 的 Amazon EventBridge 支援。

如需有關驗證電子郵件的詳細資訊，請參閱「AWS Certificate Manager 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 AWS Certificate Manager 電子郵件驗證。

重新傳送驗證電子郵件

在請求憑證時為網域設定電子郵件驗證後 （請參閱AWS Certificate Manager 電子郵件驗證)，您可以使用 AWS Certificate Manager API 請求 ACM 傳送網域驗證電子郵件給您以進行憑證續約。您應在以下情況執行此動作：

- 您在一開始請求 ACM 憑證時使用電子郵件驗證。
- 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。
- 您無法接收或找不到 ACM 針對憑證續約傳送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憑證請求中原始設定不同的網域，您可以在 ACM API AWS CLI或 AWS SDKs 中使用 ResendValidationEmail 操作。ACM 會將電子郵件傳送至指定的驗證網域。您可以在支援的 AWS CLI 區域中使用 在瀏覽器 AWS CloudShell 中存取 。

請求 ACM 重新傳送網域驗證電子郵件訊息 (主控台)

1. 在 https：//https://console.aws.amazon.com/acm/home 開啟 AWS Certificate Manager 主控台。
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。

請求 ACM 重新傳送網域驗證電子郵件 (ACM API)

在 ACM API 中使用 ResendValidationEmail 作業。透過此動作來傳遞憑證 ARN、需要手動驗證的網域和您要接收網域驗證電子郵件的網域。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
```

HTTP 驗證的網域續約

ACM 為最初透過 CloudFront 使用 HTTP 驗證發行的憑證提供自動受管續約。

ACM 會在過期前 60 天檢查以下續約條件：

- CloudFront 目前正在使用憑證。
- 您可以存取所有必要的 HTTP 驗證記錄，並包含預期的內容。

如果這些條件都符合，ACM 會將網域名稱視為已驗證並續約憑證。

如果在續約期間無法自動驗證網域，ACM 會傳送 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ACM 的 Amazon EventBridge 支援。

為了確保續約成功，請確定 `RedirectFrom` 位置的內容與憑證中每個網域`RedirectTo`的位置內容相符。

中的私有憑證續約 AWS Certificate Manager

由私有 CA 從 簽署的 ACM 憑證 AWS 私有 CA 符合受管續約的資格。與公開信任的 ACM 憑證不同，私有 PKI 的憑證不需要驗證。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 ACM API 的 RequestCertificate 動作取得的憑證才符合受管續約的資格。 AWS 私有 CA 使用 AWS 私有 CA API 的 IssueCertificate 動作直接從 發出的憑證不會由 ACM 管理。

在受管憑證過期前 60 天，ACM 會自動嘗試每小時續約一次。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 ACM API 的 RenewCertificate 動作強制續約。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：

- 如果憑證有與 ACM 整合服務相關聯，則新憑證會取代舊憑證，而不需要客戶採取額外動作。
- 如果憑證沒有與 ACM 整合服務相關聯，則需要客戶採取動作，才能匯出並安裝續約的憑證。您可以手動執行這些動作，或在 AWS Health、Amazon EventBridge 以及 AWS Lambda 的協助下執行，如下所示：如需詳細資訊，請參閱自動化匯出續約的憑證

自動化匯出續約的憑證

下列程序提供了一個範例解決方案，可在 ACM 續約憑證時自動化私有 PKI 憑證的匯出作業。此範例僅會從 ACM 匯出憑證及其私有金鑰；匯出後，憑證仍然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，建立和設定呼叫 ACM 匯出 API 的 Lambda 函數。
   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。此政策可授與函數中程式碼的許可，以擷取已續約的憑證和私有金鑰，方法是呼叫 ACM API 的 ExportCertificate 動作。

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

2. 在 Amazon EventBridge 中建立規則以接聽 ACM 運作狀態事件，並在檢測到 ACM 運作狀態事件時呼叫您的 Lambda 函數。ACM 每次嘗試續約憑證時都會寫入 AWS Health 事件。如需這些通知的詳細資訊，請參閱「使用 Personal Health Dashboard (PHD) 檢查狀態」。

   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 憑證的受管續約

您可以使用 ACM API 或 AWS CLI 手動測試 ACM 受管續約工作流程的組態。這樣做可以確認您的憑證在過期時會由 ACM 自動續約。

> **Note**
>
> 您只能測試 發行和匯出的憑證續約 AWS 私有 CA。

使用下述 API 動作或 CLI 命令時，ACM 會嘗試續約憑證。如果續約成功，ACM 會更新管理主控台或 API 輸出中顯示的憑證中繼資料。如果憑證與 ACM 整合服務相關聯，則會部署新憑證，並在 Amazon CloudWatch Events 中產生續約事件。如果續約失敗，ACM 會傳回錯誤並建議補救動作。(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**
>
> 若要使用 ACM 續約您的 AWS 私有 CA 憑證，您必須先授予 ACM 服務委託人執行此作業的許可。如需詳細資訊，請參閱指派憑證續約許可給 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
```

手動測試憑證續約 (ACM API)

- 傳送 RenewCertificate 請求，指定私有憑證的 ARN 以便續約。然後，使用 DescribeCertificate 操作來確認憑證的續約詳細資訊已更新。

檢查憑證的續約狀態

在您嘗試續約憑證時，ACM 會在憑證詳細資訊中提供 Renewal status (續約狀態) 資訊欄位。您可以使用 AWS Certificate Manager 主控台、ACM API 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 ACM API，續約狀態可以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，請參閱續約透過 DNS 驗證的網域。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。如果您原本使用電子郵件驗證憑證，請尋找來自 ACM 的電子郵件，然後點選該電子郵件中的連結來執行驗證。如果您使用 DNS 驗證、檢查，請檢查以確定 DNS 記錄存在，且憑證仍使用中。

成功

憑證中的所有網域名稱都經過驗證，而且 ACM 已續約憑證。無需採取進一步動作。

失敗

一或多個網域名稱未在憑證過期之前驗證，而且 ACM 未續約憑證。您可以要求新的憑證。

如果憑證與 Elastic Load Balancing 或 CloudFront AWS 等其他服務相關聯，或者憑證自發行或上次續約以來已匯出，則有資格續約。

> **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，請參閱對憑證請求進行故障診斷。

主題

- 使用 Personal Health Dashboard (PHD) 檢查狀態

檢查狀態 (主控台)

下列程序討論如何使用 ACM 主控台檢查 ACM 憑證的續約狀態。

1. 在 https：//https://console.aws.amazon.com/acm/home 開啟 AWS Certificate Manager 主控台。
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 尚未開始此憑證的受管續約程序。

檢查狀態 (API)

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

檢查狀態 (CLI)

以下範例說明如何使用 AWS Command Line Interface (AWS CLI) 檢查 ACM 憑證續約的狀態。

```
$ 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 `RenewalStatus` 欄位，表示 ACM 尚未開始憑證的受管續約程序。

使用 Personal Health Dashboard (PHD) 檢查狀態

ACM 會在過期前 60 天嘗試自動續約 ACM 憑證。如果 ACM 無法自動續約憑證，它會 AWS Health Dashboard 在過期後每隔 45 天、30 天、15 天、7 天、3 天和 1 天將憑證續約事件通知傳送給 ，以通知您需要採取動作。 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 會將連續的續約事件通知寫入 PhD 時間線中的單一事件。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 AWS Health Dashboard https：//https://phd.aws.amazon.com/phd/home#/。
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。

標籤 AWS Certificate Manager 資源

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。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。
- 字`aws:`首保留供 AWS 使用；您無法新增、編輯或刪除以 開頭的索引鍵標籤`aws:`。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 憑證標籤不適用於 AWS Management Console的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、 或 AWS Certificate Manager API 來新增 AWS Command Line Interface、編輯和刪除標籤。

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. 登入 AWS Management Console ，並在 https：//https://console.aws.amazon.com/acm/home 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. 登入 AWS Management Console ，並在 https：//https://console.aws.amazon.com/acm/home 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. 登入 AWS Management Console ，並在 https：//https://console.aws.amazon.com/acm/home 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. 登入 AWS Management Console ，並在 https：//https://console.aws.amazon.com/acm/home 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

與 ACM 整合的服務

AWS Certificate Manager 支援越來越多 AWS 的服務。您無法直接在以 AWS 為基礎的網站或應用程式上安裝 ACM 憑證或私有 AWS 私有 CA 憑證。

> **Note**
>
> 公有 ACM 憑證可以安裝在連接到 Nitro Enclave 的 Amazon EC2 執行個體上。您也可以匯出公有憑證，以便在任何 Amazon EC2 執行個體上使用。如需了解如何在未連接至 Nitro Enclave 的 Amazon EC2 執行個體上設定獨立 Web 伺服器，請參閱教學課程：在 Amazon Linux 2 上安裝 LAMP Web 伺服器或教學課程：使用 Amazon Linux AMI 安裝 LAMP Web 伺服器。

下列服務支援 ACM 憑證：

Elastic Load Balancing

Elastic Load Balancing 會自動將您的傳入應用程式流量分散到多個 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/TLS 提供安全的內容，負載平衡器會要求在負載平衡器或後端 Amazon EC2 執行個體上安裝 SSL/TLS 憑證。ACM 已和 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 是一項 Web 服務，可透過從全球節點網路交付您的內容，以加速將動態和靜態 Web 內容分佈給最終使用者。最終使用者請求您透過 CloudFront 提供的內容時，系統會自動將該使用者路由到提供最低延遲的節點。這可確保盡可能以最佳效能交付內容。如果內容目前位於節點，CloudFront 會立即交付該內容。如果內容目前不在節點，CloudFront 會從您指定為最終內容來源的 Amazon S3 儲存貯體或 Web 伺服器擷取該內容。如需 CloudFront 的詳細資訊，請參閱 Amazon CloudFront 開發人員指南。

為了透過 SSL/TLS 提供安全的內容，CloudFront 會要求在 CloudFront 分佈或後端內容來源上安裝 SSL/TLS 憑證。ACM 已和 CloudFront 整合，可在 CloudFront 分佈上部署 ACM 憑證。如需詳細資訊，請參閱取得 SSL/TLS 憑證。

> **Note**
>
> 若要搭配 CloudFront 使用 ACM 憑證，您必須在美國東部 (維吉尼亞北部) 區域請求或匯入憑證。

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。使用者可以直接使用您的 AWS 帳戶 登入資料或透過第三方登入，例如 Facebook、Amazon、Google 或 Apple。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 憑證來保護自訂網域。在這種情況下，請注意，您必須先移除憑證與 CloudFront 的關聯，才能刪除憑證。

AWS Elastic Beanstalk

Elastic Beanstalk 可協助您在 AWS 雲端中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設施。 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，請參閱設定您 Elastic Beanstalk 環境的負載平衡器來終止 HTTPS。

AWS App Runner

App Runner 是一項 AWS 服務，提供快速、簡單且符合成本效益的方式，將原始碼或容器映像直接部署到 AWS 雲端中可擴展且安全的 Web 應用程式。您不需要學習新技術、決定要使用的運算服務，或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。這些憑證會儲存在 ACM 中。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 Gateway

隨著行動裝置的普及和物聯網 (IoT) 的成長，建立可用來存取資料並與 AWS上的後端系統互動的 API 變得越來越普遍。您可以使用 API Gateway 發佈、維護、監控和保護您的 API。將 API 部署到 API Gateway 後，您可以設定自訂網域名稱以簡化存取該 API 的作業。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 產生或匯入憑證。如需有關 Amazon API Gateway 的詳細資訊，請參閱《Amazon API Gateway 開發人員指南》。

AWS Nitro Enclaves

AWS Nitro Enclaves 是一種 Amazon EC2 功能，可讓您從 Amazon EC2 執行個體建立稱為 enclaves 的隔離執行環境。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。使用者無法透過 SSH 進入隔離區，且上層執行個體的處理序、應用程式或使用者 (包括根或管理員) 都無法存取隔離區內的資料和應用程式。

連接到 Nitro Enclaves 的 EC2 執行個體支援 ACM 憑證。如需詳細資訊，請參閱適用於 Nitro Enclaves 的AWS Certificate Manager。

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

AWS CloudFormation

AWS CloudFormation 可協助您建立模型和設定 Amazon Web Services 資源。您可以建立範本來描述您想要使用 AWS 的資源，例如 Elastic Load Balancing 或 API Gateway。然後， AWS CloudFormation 會負責佈建和設定這些資源。您不需要個別建立和設定 AWS 資源，並了解什麼依賴； AWS CloudFormation 處理所有這些。ACM 憑證包含為範本資源，這表示 AWS CloudFormation 可以請求可與 服務搭配使用 AWS 的 ACM 憑證，以啟用安全連線。此外，ACM 憑證包含許多您可以設定 AWS 的資源 AWS CloudFormation。

如需有關 CloudFormation 的一般資訊，請參閱《AWS CloudFormation 使用者指南》。如需有關 CloudFormation 支援之 ACM 資源的詳細資源，請參閱 AWS::CertificateManager::Certificate。

透過 提供的強大自動化 AWS CloudFormation，您可以輕鬆超過憑證配額，尤其是新 AWS 帳戶。我們建議您遵循 的 ACM 最佳實務 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 是一組專門建置的工具和功能，可讓前端 Web 和行動開發人員快速輕鬆地在其上建置完整的堆疊應用程式 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 建置具有即時可用 UI 元件的前端 UI、建立應用程式後端，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，則 Amplify 主控台會發行 ACM 憑證來保護應用程式。

Amazon OpenSearch Service

Amazon OpenSearch Service 是一個搜尋和分析引擎，適用於日誌分析、即時應用程式監控及點擊流分析等使用案例。如需詳細資訊，請參閱《Amazon OpenSearch Service 開發人員指南》。

當您建立包含了自訂網域和端點的 OpenSearch Service 叢集時，您可以使用 ACM 來佈建具有憑證之關聯的 Application Load Balancer。

AWS Network Firewall

AWS Network Firewall 是一種受管服務，可讓您輕鬆為所有 Amazon Virtual Private Clouds (VPCs) 部署基本網路保護。如需詳細資訊，請參閱 AWS Network Firewall 開發人員指南。

Network Firewall 防火牆與 ACM 整合，可進行 TLS 檢查。如果您在 Network Firewall 中使用 TLS 檢查，則必須設定 ACM 憑證，以解密和重新加密通過防火牆的 SSL/TLS 流量。如需有關 Network Firewall 如何搭配 ACM 使用進行 TLS 檢查的詳細資訊，請參閱 AWS Network Firewall 開發人員指南中的將 SSL/TLS 憑證與 TLS 檢查組態搭配使用的要求。

中的安全性 AWS Certificate Manager

的雲端安全 AWS 是最高優先順序。身為 AWS 客戶，您可以受益於資料中心和網路架構，這些架構專為符合最安全敏感組織的需求而建置。

安全性是 AWS 與您之間的共同責任。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 Cloud AWS 中執行 AWS 服務的基礎設施。 AWS 也為您提供可安全使用的服務。作為AWS 合規計劃的一部分，第三方稽核人員會定期測試和驗證我們安全的有效性。若要了解適用的合規計劃 AWS Certificate Manager，請參閱合規AWS 計劃的 服務範圍合規。
- 雲端的安全性 – 您的責任取決於您使用 AWS 的服務。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。

本文件可協助您了解如何在使用 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。您也會了解如何使用其他 AWS 服務來協助您監控和保護 ACM 資源。

主題

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

中的資料保護 AWS Certificate Manager

AWS 共同責任模型適用於 中的資料保護 AWS Certificate Manager。如此模型所述， AWS 負責保護執行所有 的 全球基礎設施 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，我們建議您保護 AWS 帳戶 登入資料，並使用 AWS IAM Identity Center 或 AWS Identity and Access Management (IAM) 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。如需有關使用 CloudTrail 追蹤擷取 AWS 活動的資訊，請參閱AWS CloudTrail 《 使用者指南》中的使用 CloudTrail 追蹤。
- 使用 AWS 加密解決方案，以及其中的所有預設安全控制 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 AWS 透過命令列界面或 API 存取 時需要 FIPS 140-3 驗證的密碼編譯模組，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) 140-3。

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 ACM 或使用 AWS 服務 主控台、API AWS CLI或其他 AWS SDKs 時。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

當您請求公有憑證時， AWS Certificate Manager (ACM) 會產生公有/私有金鑰對。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 會存放憑證及其對應的私有金鑰，並使用 AWS Key Management Service (AWS KMS) 協助保護私有金鑰。運作程序如下：

1. 當您第一次在 AWS 區域中請求或匯入憑證時，ACM AWS KMS key 會使用別名 aws/acm 建立受管 。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有金鑰。
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。在 中也會建立授予 AWS KMS ，允許服務使用 KMS 金鑰來解密憑證的私有金鑰。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 與 ACM 整合的服務。

> **Note**
>
> 您可以控制自動建立的 AWS KMS 授予。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) 是一種 AWS 服務 ，可協助管理員安全地控制對 AWS 資源的存取。IAM 管理員可以控制誰能完成身分驗證 (登入) 和獲得授權 (取得許可)，而得以使用 ACM 資源。IAM 是 AWS 服務 您可以免費使用的 。

主題

目標對象

使用方式 AWS Identity and Access Management (IAM) 會有所不同，取決於您在 ACM 中執行的工作。

服務使用者：如果使用 ACM 服務執行工作，管理員會為您提供所需的憑證和許可。隨著您為了執行工作而使用越來越多 ACM 功能，您可能會需要額外的許可。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。若您無法存取 ACM 中的某項功能，請參閱 對 AWS Certificate Manager 身分和存取進行故障診斷。

服務管理員：如果您負責管理公司內的 ACM 資源，您可能具備 ACM 的完整存取權限。您的工作是判斷服務使用者應存取的 ACM 功能及資源。接著，您必須將請求提交給您的 IAM 管理員，來變更您服

務使用者的許可。檢閱此頁面上的資訊，了解 IAM 的基本概念。若要進一步了解貴公司可搭配 ACM 使用 IAM 的方式，請參閱 AWS Certificate Manager 如何使用 IAM。

IAM 管理員：如果您是 IAM 管理員，建議您掌握如何撰寫政策以管理 ACM 存取權的詳細資訊。若要檢視您可以在 IAM 中使用的範例 ACM 身分型政策，請參閱 的身分型政策範例 AWS Certificate Manager。

使用身分驗證

身分驗證是您 AWS 使用身分憑證登入 的方式。您必須以 AWS 帳戶根使用者身分、IAM 使用者身分或擔任 IAM 角色來驗證 （ 登入 AWS)。

您可以使用透過身分來源提供的登入資料，以聯合身分 AWS 身分身分登入 。 AWS IAM Identity Center (IAM Identity Center) 使用者、您公司的單一登入身分驗證，以及您的 Google 或 Facebook 登入資料，都是聯合身分的範例。您以聯合身分登入時，您的管理員先前已設定使用 IAM 角色的聯合身分。當您使用聯合 AWS 身分存取 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。如需登入的詳細資訊 AWS，請參閱AWS 登入 《 使用者指南》中的如何登入您的 AWS 帳戶 。

如果您以 AWS 程式設計方式存取 ， AWS 會提供軟體開發套件 (SDK) 和命令列界面 (CLI)，以使用您的 憑證以密碼編譯方式簽署您的請求。如果您不使用 AWS 工具，則必須自行簽署請求。如需使用建議的方法自行簽署請求的詳細資訊，請參閱《IAM 使用者指南》中的適用於 API 請求的AWS Signature 第 4 版。

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 建議您使用多重驗證 (MFA) 來提高帳戶的安全性。如需更多資訊，請參閱《AWS IAM Identity Center 使用者指南》中的多重要素驗證和《IAM 使用者指南》中的 IAM 中的AWS 多重要素驗證。

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，該身分可完整存取帳戶中的所有 AWS 服務 和資源。此身分稱為 AWS 帳戶 Theroot 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。如需這些任務的完整清單，了解需以根使用者登入的任務，請參閱 IAM 使用者指南中的需要根使用者憑證的任務。

聯合身分

根據最佳實務， 要求人類使用者，包括需要管理員存取權的使用者，使用 聯合身分提供者 AWS 服務來使用臨時憑證來存取 。

聯合身分是您企業使用者目錄、Web 身分提供者、 AWS Directory Service、Identity Center 目錄或任何使用透過身分來源提供的登入資料 AWS 服務 存取的使用者。當聯合身分存取時 AWS 帳戶，它們會擔任 角色，而角色會提供臨時登入資料。

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，也可以連接並同步到您自己的身分來源中的一組使用者 AWS 帳戶 和群組，以便在所有 和應用程式中使用。如需 IAM Identity Center 的詳細資訊，請參閱 AWS IAM Identity Center 使用者指南中的什麼是 IAM Identity Center？。

IAM 使用者和群組

IAM 使用者是 中的身分 AWS 帳戶 ，具有單一人員或應用程式的特定許可。建議您盡可能依賴臨時憑證，而不是擁有建立長期憑證 (例如密碼和存取金鑰) 的 IAM 使用者。但是如果特定使用案例需要擁有長期憑證的 IAM 使用者，建議您輪換存取金鑰。如需更多資訊，請參閱 IAM 使用者指南中的為需要長期憑證的使用案例定期輪換存取金鑰。

IAM 群組是一種指定 IAM 使用者集合的身分。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 IAMAdmins 的群組，並給予該群組管理 IAM 資源的許可。

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，但角色僅提供臨時憑證。如需更多資訊，請參閱《IAM 使用者指南》中的 IAM 使用者的使用案例。

IAM 角色

IAM 角色是 中具有特定許可 AWS 帳戶 的身分。它類似 IAM 使用者，但不與特定的人員相關聯。若要暫時在 中擔任 IAM 角色 AWS Management Console，您可以從使用者切換至 IAM 角色（主控台）。您可以透過呼叫 AWS CLI 或 AWS API 操作或使用自訂 URL 來擔任角色。如需使用角色的方法詳細資訊，請參閱《IAM 使用者指南》中的擔任角色的方法。

使用臨時憑證的 IAM 角色在下列情況中非常有用：

- 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。如需有關聯合角色的相關資訊，請參閱《IAM 使用者指南》中的為第三方身分提供者 (聯合) 建立角色。如果您使用 IAM Identity Center，則需要設定許可集。為控制身分驗證後可以存取的內容，IAM Identity Center 將許可集與 IAM 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 使用者許可 – IAM 使用者或角色可以擔任 IAM 角色來暫時針對特定任務採用不同的許可。

- 跨帳戶存取權：您可以使用 IAM 角色，允許不同帳戶中的某人 (信任的主體) 存取您帳戶的資源。角色是授予跨帳戶存取權的主要方式。不過，在某些 中 AWS 服務，您可以將政策直接連接到資源（而不是使用角色做為代理）。如需了解使用角色和資源型政策進行跨帳戶存取之間的差異，請參閱《IAM 使用者指南》中的 IAM 中的跨帳戶資源存取。
- 跨服務存取 – 有些 AWS 服務 使用其他 中的功能 AWS 服務。例如，當您在服務中進行呼叫時，該服務通常會在 Amazon EC2 中執行應用程式或將物件儲存在 Amazon Simple Storage Service (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。
  - 轉送存取工作階段 (FAS) – 當您使用 IAM 使用者或角色在其中執行動作時 AWS，您會被視為委託人。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 使用呼叫 的委託人許可 AWS 服務，結合 AWS 服務 請求向下游服務提出請求。只有當服務收到需要與其他 AWS 服務 或 資源互動才能完成的請求時，才會提出 FAS 請求。在此情況下，您必須具有執行這兩個動作的許可。如需提出 FAS 請求時的政策詳細資訊，請參閱《轉發存取工作階段》。
  - 服務角色 – 服務角色是服務擔任的 IAM 角色，可代表您執行動作。IAM 管理員可以從 IAM 內建立、修改和刪除服務角色。如需詳細資訊，請參閱《IAM 使用者指南》中的建立角色以委派許可權給 AWS 服務。
  - 服務連結角色 – 服務連結角色是一種連結至 的服務角色類型 AWS 服務。服務可以擔任代表您執行動作的角色。服務連結角色會出現在您的 中 AWS 帳戶 ，並由服務擁有。IAM 管理員可以檢視，但不能編輯服務連結角色的許可。
- 在 Amazon EC2 上執行的應用程式 – 您可以使用 IAM 角色來管理在 EC2 執行個體上執行之應用程式的臨時登入資料，以及提出 AWS CLI 或 AWS API 請求。這是在 EC2 執行個體內儲存存取金鑰的較好方式。若要將 AWS 角色指派給 EC2 執行個體並將其提供給其所有應用程式，您可以建立連接至執行個體的執行個體描述檔。執行個體設定檔包含該角色，並且可讓 EC2 執行個體上執行的程式取得臨時憑證。如需詳細資訊，請參閱《IAM 使用者指南》中的使用 IAM 角色來授予許可權給 Amazon EC2 執行個體上執行的應用程式。

使用政策管理存取權

您可以透過建立政策並將其連接到身分或資源 AWS 來控制 AWS 中的存取。政策是 中的物件，當與身分或資源建立關聯時， AWS 會定義其許可。當委託人（使用者、根使用者或角色工作階段）發出請求時， 會 AWS 評估這些政策。政策中的許可決定是否允許或拒絕請求。大多數政策會以 JSON 文件 AWS 的形式存放在 中。如需 JSON 政策文件結構和內容的詳細資訊，請參閱 IAM 使用者指南中的 JSON 政策概觀。

管理員可以使用 AWS JSON 政策來指定誰可以存取內容。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。若要授予使用者對其所需資源執行動作的許可，IAM 管理員可以建立 IAM 政策。然後，管理員可以將 IAM 政策新增至角色，使用者便能擔任這些角色。

IAM 政策定義該動作的許可，無論您使用何種方法來執行操作。例如，假設您有一個允許 `iam:GetRole` 動作的政策。具有該政策的使用者可以從 AWS Management Console AWS CLI、 或 AWS API 取得角色資訊。

身分型政策

身分型政策是可以附加到身分 (例如 IAM 使用者、使用者群組或角色) 的 JSON 許可政策文件。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。如需了解如何建立身分型政策，請參閱《IAM 使用者指南》中的透過客戶管理政策定義自訂 IAM 許可。

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。受管政策是獨立的政策，您可以連接到 中的多個使用者、群組和角色 AWS 帳戶。受管政策包括 AWS 受管政策和客戶受管政策。如需了解如何在受管政策及內嵌政策之間選擇，請參閱《IAM 使用者指南》中的在受管政策和內嵌政策間選擇。

資源型政策

資源型政策是連接到資源的 JSON 政策文件。資源型政策的最常見範例是 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。委託人可以包含帳戶、使用者、角色、聯合身分使用者或 AWS 服務。

資源型政策是位於該服務中的內嵌政策。您無法在資源型政策中使用來自 IAM 的 AWS 受管政策。

存取控制清單 (ACL)

存取控制清單 (ACL) 可控制哪些主體 (帳戶成員、使用者或角色) 擁有存取某資源的許可。ACL 類似於資源型政策，但它們不使用 JSON 政策文件格式。

Amazon S3 AWS WAF和 Amazon VPC 是支援 ACLs的服務範例。如需進一步了解 ACL，請參閱 Amazon Simple Storage Service 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政策類型。這些政策類型可設定較常見政策類型授予您的最大許可。

- 許可界限 – 許可範圍是一種進階功能，可供您設定身分型政策能授予 IAM 實體 (IAM 使用者或角色) 的最大許可。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。如需許可界限的詳細資訊，請參閱 IAM 使用者指南中的 IAM 實體許可界限。

- 服務控制政策 SCPs) – SCPs是 JSON 政策，可指定 中組織或組織單位 (OU) 的最大許可 AWS Organizations。 AWS Organizations 是一種用於分組和集中管理您企業擁有 AWS 帳戶 的多個 的服務。若您啟用組織中的所有功能，您可以將服務控制政策 (SCP) 套用到任何或所有帳戶。SCP 會限制成員帳戶中實體的許可，包括每個實體 AWS 帳戶根使用者。如需 Organizations 和 SCP 的詳細資訊，請參閱《AWS Organizations 使用者指南》中的服務控制政策。
- 資源控制政策 (RCP) - RCP 是 JSON 政策，可用來設定您帳戶中資源的可用許可上限，採取這種方式就不需要更新附加至您所擁有的每個資源的 IAM 政策。RCP 會限制成員帳戶中資源的許可，並可能影響身分的有效許可，包括 AWS 帳戶根使用者，無論它們是否屬於您的組織。如需 Organizations 和 RCPs的詳細資訊，包括支援 RCPs AWS 服務 的 清單，請參閱AWS Organizations 《 使用者指南》中的資源控制政策 (RCPs)。
- 工作階段政策 – 工作階段政策是一種進階政策，您可以在透過撰寫程式的方式建立角色或聯合使用者的暫時工作階段時，做為參數傳遞。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 使用者指南中的工作階段政策。

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。若要了解如何 AWS 在涉及多個政策類型時決定是否允許請求，請參閱《IAM 使用者指南》中的政策評估邏輯。

AWS Certificate Manager 如何使用 IAM

使用 IAM 管理 ACM 的存取權之前，請先了解您可以搭配 ACM 使用哪些 IAM 功能。

您可以搭配 使用的 IAM 功能 AWS Certificate Manager

IAM 功能	ACM 支援
身分型政策	是
資源型政策	否
政策動作	是

IAM 功能	ACM 支援
政策資源	是
政策條件索引鍵 (服務特定)	是
ACL	否
ABAC(政策中的標籤)	部分
臨時憑證	是
主體許可	是
服務角色	否
服務連結角色	是

若要全面了解 ACM 和其他 AWS 服務如何與大多數 IAM 功能搭配使用，請參閱《IAM 使用者指南》中的與 IAM AWS 搭配使用的 服務。

適用於 ACM 的身分型政策

支援身分型政策：是

身分型政策是可以附加到身分 (例如 IAM 使用者、使用者群組或角色) 的 JSON 許可政策文件。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。如需了解如何建立身分型政策，請參閱《IAM 使用者指南》中的透過客戶管理政策定義自訂 IAM 許可。

使用 IAM 身分型政策，您可以指定允許或拒絕的動作和資源，以及在何種條件下允許或拒絕動作。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。如要了解您在 JSON 政策中使用的所有元素，請參閱《IAM 使用者指南》中的 IAM JSON 政策元素參考。

適用於 ACM 的身分型政策範例

若要檢視 ACM 身分型政策範例，請參閱 的身分型政策範例 AWS Certificate Manager。

ACM 內的資源型政策

支援資源型政策：否

資源型政策是附加到資源的 JSON 政策文件。資源型政策的最常見範例是 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。委託人可以包含帳戶、使用者、角色、聯合身分使用者或 AWS 服務。

如需啟用跨帳戶存取權，您可以指定在其他帳戶內的所有帳戶或 IAM 實體，做為資源型政策的主體。新增跨帳戶主體至資源型政策，只是建立信任關係的一半。當委託人和資源位於不同位置時 AWS 帳戶，信任帳戶中的 IAM 管理員也必須授予委託人實體（使用者或角色）存取資源的許可。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 使用者指南》中的 IAM 中的快帳戶資源存取。

適用於 ACM 的政策動作

支援政策動作：是

管理員可以使用 AWS JSON 政策來指定誰可以存取內容。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 政策的 `Action` 元素描述您可以用來允許或拒絕政策中存取的動作。政策動作通常具有與相關聯 AWS API 操作相同的名稱。有一些例外狀況，例如沒有相符的 API 操作的僅限許可動作。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。

如要查看 ACM 動作的清單，請參閱《服務授權參考》中的 AWS Certificate Manager定義的動作。

ACM 中的政策動作會在動作之前使用以下字首：

```
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 ACM 身分型政策範例，請參閱 的身分型政策範例 AWS Certificate Manager。

ACM 的政策資源

支援政策資源：是

管理員可以使用 AWS JSON 政策來指定誰可以存取內容。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。

Resource JSON 政策元素可指定要套用動作的物件。陳述式必須包含 Resource 或 NotResource 元素。最佳實務是使用其 Amazon Resource Name (ARN) 來指定資源。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": "*"
```

如要查看 ACM 資源類型及其 ARN 的清單，請參閱《服務授權參考》中的 AWS Certificate Manager定義的資源。若要了解您可以使用哪些動作指定每個資源的 ARN，請參閱 AWS Certificate Manager定義的動作。

若要檢視 ACM 身分型政策範例，請參閱 的身分型政策範例 AWS Certificate Manager。

適用於 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 政策來指定誰可以存取內容。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， 會使用邏輯OR操作 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，您可以只在使用者使用其 IAM 使用者名稱標記時，將存取資源的許可授予該 IAM 使用者。如需更多資訊，請參閱 IAM 使用者指南中的 IAM 政策元素：變數和標籤。

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

如要查看 ACM 條件索引鍵的清單，請參閱《服務授權參考》中的適用於 AWS Certificate Manager 的條件索引鍵。若要了解您可以使用條件金鑰的動作和資源，請參閱 定義的動作 AWS Certificate Manager。

若要檢視 ACM 身分型政策範例，請參閱 的身分型政策範例 AWS Certificate Manager。

在 ACM 中使用 ACL

支援 ACL：否

存取控制清單 (ACL) 可控制哪些主體 (帳戶成員、使用者或角色) 擁有存取某資源的許可。ACL 類似於資源型政策，但它們不使用 JSON 政策文件格式。

搭配使用 ACM 和 ABAC

支援 ABAC (政策中的標籤)：部分

屬性型存取控制 (ABAC) 是一種授權策略，可根據屬性來定義許可。在 中 AWS，這些屬性稱為標籤。您可以將標籤連接到 IAM 實體 （ 使用者或角色 ） 和許多 AWS 資源。為實體和資源加上標籤是 ABAC 的第一步。您接著要設計 ABAC 政策，允許在主體的標籤與其嘗試存取的資源標籤相符時操作。

ABAC 在成長快速的環境中相當有幫助，並能在政策管理變得繁瑣時提供協助。

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 ABAC 的詳細資訊，請參閱《IAM 使用者指南》中的使用 ABAC 授權定義許可。如要查看含有設定 ABAC 步驟的教學課程，請參閱 IAM 使用者指南中的使用屬性型存取控制 (ABAC)。

將臨時憑證與 ACM 搭配使用

支援臨時憑證：是

當您使用臨時登入資料登入時，有些 AWS 服務 無法運作。如需詳細資訊，包括哪些 AWS 服務 使用臨時登入資料，請參閱《AWS 服務 IAM 使用者指南》中的使用 IAM 的 。

如果您 AWS Management Console 使用使用者名稱和密碼以外的任何方法登入 ，則會使用臨時登入資料。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登入資料。

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。如需切換角色的詳細資訊，請參閱《IAM 使用者指南》中的從使用者切換至 IAM 角色 (主控台)。

您可以使用 AWS CLI 或 AWS API 手動建立臨時登入資料。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 中的暫時性安全憑證。

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 IAM 使用者或角色在 中執行動作時 AWS，您會被視為委託人。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 使用呼叫 的委託人許可 AWS 服務，結合 AWS 服務 請求向下游服務提出請求。只有當服務收到需要與其他 AWS 服務 或 資源互動才能完成的請求時，才會提出 FAS 請求。在此情況下，您必須具有執行這兩個動作的許可。如需提出 FAS 請求時的政策詳細資訊，請參閱轉發存取工作階段。

ACM 的服務角色

支援服務角色：否

服務角色是服務擔任的 IAM 角色，可代您執行動作。IAM 管理員可以從 IAM 內建立、修改和刪除服務角色。如需詳細資訊，請參閱《IAM 使用者指南》中的建立角色以委派許可權給 AWS 服務。

> **Warning**
>
> 變更服務角色的許可有可能使 ACM 功能發生故障。只有 ACM 提供指引時，才能編輯服務角色。

ACM 的服務連結角色

支援服務連結角色：是

服務連結角色是連結至 的一種服務角色 AWS 服務。服務可以擔任代表您執行動作的角色。服務連結角色會出現在您的 中 AWS 帳戶 ，並由服務擁有。IAM 管理員可以檢視，但不能編輯服務連結角色的許可。

如需建立或管理服務連結角色的詳細資訊，請參閱可搭配 IAM 運作的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。

的身分型政策範例 AWS Certificate Manager

根據預設，使用者和角色不具備建立或修改 ACM 資源的許可。他們也無法使用 AWS Management Console、 AWS Command Line Interface (AWS CLI) 或 AWS API 來執行任務。若要授予使用者對其所需資源執行動作的許可，IAM 管理員可以建立 IAM 政策。然後，管理員可以將 IAM 政策新增至角色，使用者便能擔任這些角色。

如需了解如何使用這些範例 JSON 政策文件建立 IAM 身分型政策，請參閱《IAM 使用者指南》中的建立 IAM 政策 (主控台)。

如需 ACM 所定義之動作和資源類型的詳細資訊，包括每種資源類型的 ARN 格式，請參閱《服務授權參考》中的適用於 AWS Certificate Manager的動作、資源和條件索引鍵。

主題

政策最佳實務

身分型政策會判斷您帳戶中的某個人員是否可以建立、存取或刪除 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 受管政策並邁向最低權限許可 – 若要開始將許可授予使用者和工作負載，請使用將許可授予許多常見使用案例的 AWS 受管政策。它們可在您的 中使用 AWS 帳戶。我們建議您定義特定於使用案例 AWS 的客戶受管政策，以進一步減少許可。如需更多資訊，請參閱 IAM 使用者指南中的 AWS 受管政策或任務職能的AWS 受管政策。
- 套用最低權限許可 – 設定 IAM 政策的許可時，請僅授予執行任務所需的許可。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。如需使用 IAM 套用許可的更多相關資訊，請參閱 IAM 使用者指南中的 IAM 中的政策和許可。

- 使用 IAM 政策中的條件進一步限制存取權 – 您可以將條件新增至政策，以限制動作和資源的存取。例如，您可以撰寫政策條件，指定必須使用 SSL 傳送所有請求。如果透過特定 例如 使用服務動作 AWS 服務，您也可以使用條件來授予其存取權 AWS CloudFormation。如需詳細資訊，請參閱 IAM 使用者指南中的 IAM JSON 政策元素：條件。
- 使用 IAM Access Analyzer 驗證 IAM 政策，確保許可安全且可正常運作 – IAM Access Analyzer 驗證新政策和現有政策，確保這些政策遵從 IAM 政策語言 (JSON) 和 IAM 最佳實務。IAM Access Analyzer 提供 100 多項政策檢查及切實可行的建議，可協助您撰寫安全且實用的政策。如需詳細資訊，請參閱《IAM 使用者指南》中的使用 IAM Access Analyzer 驗證政策。
- 需要多重要素驗證 (MFA) – 如果您的案例需要 IAM 使用者或 中的根使用者 AWS 帳戶，請開啟 MFA 以提高安全性。如需在呼叫 API 操作時請求 MFA，請將 MFA 條件新增至您的政策。如需詳細資訊，請參閱《IAM 使用者指南》https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_mfa_configure-api-require.html中的透過 MFA 的安全 API 存取。

如需 IAM 中最佳實務的相關資訊，請參閱 IAM 使用者指南中的 IAM 安全最佳實務。

使用 ACM 主控台

若要存取 AWS Certificate Manager 主控台，您必須擁有一組最低的許可。這些許可必須允許您列出和檢視 中 ACM 資源的詳細資訊 AWS 帳戶。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。

對於僅呼叫 AWS CLI 或 AWS API 的使用者，您不需要允許最低主控台許可。反之，只需允許存取符合他們嘗試執行之 API 操作的動作就可以了。

為了確保使用者和角色仍然可以使用 ACM 主控台，也請將 ACM *AWSCertificateManagerReadOnly* AWS 受管政策連接到實體。如需詳細資訊，請參閱《IAM 使用者指南》中的新增許可到使用者。

允許使用者檢視他們自己的許可

此範例會示範如何建立政策，允許 IAM 使用者檢視附加到他們使用者身分的內嵌及受管政策。此政策包含在主控台或使用 或 AWS CLI AWS API 以程式設計方式完成此動作的許可。

```
{
    "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

以下政策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**
>
> 需要此許可才能讓 ACM 憑證出現在 Elastic Load Balancing 和 CloudFront 主控台中。

請求憑證

下列政策拒絕使用者請求 ACM 匯出公有憑證。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyACMCertificateRequest",
      "Effect": "Deny",
      "Action": [
        "acm:RequestCertificate"
      ],
      "Resource": [
        "*"
      ],
      "Condition": {
        "StringEquals": {
          "acm:Export": "ENABLED"
        }
      }
    }
  ]
}
```

擷取憑證

以下政策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

以下政策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	*
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*

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

AWS 的 受管政策 AWS Certificate Manager

AWS 受管政策是由 AWS AWS 受管政策建立和管理的獨立政策旨在為許多常用案例提供許可，以便您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予特定使用案例的最低權限許可，因為這些許可可供所有 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。

您無法變更 AWS 受管政策中定義的許可。如果 AWS 更新受 AWS 管政策中定義的許可，則更新會影響政策連接的所有主體身分（使用者、群組和角色）。當新的 AWS 服務 啟動或新的 API 操作可用於現有服務時， AWS 最有可能更新 AWS 受管政策。

如需詳細資訊，請參閱《IAM 使用者指南》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。

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

如需政策詳細資訊的 JSON 清單，請參閱 AWSCertificateManagerReadOnly。

AWSCertificateManagerFullAccess

此政策提供所有 ACM 動作和資源的完整存取權限。

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

如需政策詳細資訊的 JSON 清單，請參閱 AWSCertificateManagerFullAccess。

AWS 受管政策的 ACM 更新

檢視自此服務開始追蹤這些變更以來，ACM AWS 受管政策更新的詳細資訊。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 會開始追蹤 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) 條件金鑰來限制對憑證請求的存取。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**
>
> 結合 ACM 條件金鑰與 AWS 全域條件金鑰，例如 `aws:PrincipalArn`，以進一步限制特定使用者或角色的動作。

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	Type	描述
`acm:ValidationMethod`	RequestCertificate	字串 (`DNS`、`EMAIL`、`HTTP`)	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，為共用的另一個帳戶啟用從私有 CA 發出的私有憑證自動續約 AWS Resource Access Manager。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 信任下列服務可擔任該角色：

- `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、 AWS CLI或 AWS API 發行 ACM 憑證時，ACM 會在您第一次為 共用的另一個帳戶建立私有 CA AWS RAM 來簽署憑證時，為您建立 SLR。

如果您遇到訊息，指出 ACM 無法判斷您的帳戶上是否存在 SLR，這可能表示您的帳戶未授予 AWS 私有 CA 所需的讀取許可。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 (SLR) 的問題。

> **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，2017 年 1 月 1 日才開始支援 SLR，若您在這之前就有使用 ACM，則 ACM 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 Role (角色)、Create role (建立角色)、Certificate Manager，以便透過 CertificateManagerServiceRolePolicy 使用案例來建立新的角色。
- 使用 IAM API CreateServiceLinkedRole 或對應的 AWS CLI 命令 create-service-linked-role，以`acm.amazonaws.com`服務名稱建立 SLR。

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

您通常不需要手動刪除 AWSServiceRoleForCertificateManager 角色。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 支援在所有 AWS 私有 CA 可使用 ACM 和 的區域中使用 SLRs。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
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

區域名稱	區域身分	ACM 中的支援
AWS GovCloud （美國東部） 東部	us-gov-east-1	是

對 AWS Certificate Manager 身分和存取進行故障診斷

請參考以下資訊，診斷及修正使用 ACM 和 IAM 時可能發生的常見問題。

主題

我未獲授權，不得在 ACM 中執行動作

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。

下列範例錯誤會在mateojackson IAM 使用者嘗試使用主控台檢視一個虛構 *my-example-widget* 資源的詳細資訊，但卻無虛構 acm:*GetWidget* 許可時發生。

```
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 管理員。您的管理員提供您的簽署憑證。

我未獲授權，無法在 ACM 中請求取得憑證

如果發生此錯誤，表示您的 ACM 或 PKI 管理員已設定規則，防止您在憑證處於目前的狀態時請求取得。

如果 IAM 使用者嘗試透過主控台，使用組織管理員以 DENY 設定的選項來請求取得憑證，會發生下列範例所示的錯誤。

```
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。

我未獲得執行 iam:PassRole 的授權

如果錯誤訊息告知您未獲得授權，無法執行 `iam:PassRole` 動作，您的政策就必須更新，允許您將角色傳遞給 ACM。

有些 AWS 服務 可讓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。

名為 `marymajor` 的 IAM 使用者嘗試使用主控台在 ACM 中執行動作時，會發生下列範例所示的錯誤。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 管理員。您的管理員提供您的簽署憑證。

我想要允許 以外的人員 AWS 帳戶 存取我的 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。針對支援基於資源的政策或存取控制清單 (ACL) 的服務，您可以使用那些政策來授予人員存取您的資源的許可。

如需進一步了解，請參閱以下內容：

- 若要了解 ACM 是否支援這些功能，請參閱 AWS Certificate Manager 如何使用 IAM。
- 若要了解如何 AWS 帳戶 在您擁有的 資源之間提供存取權，請參閱《IAM 使用者指南》中的在您擁有 AWS 帳戶 的另一個 中提供存取權給 IAM 使用者。
- 若要了解如何將資源的存取權提供給第三方 AWS 帳戶，請參閱《IAM 使用者指南》中的將存取權提供給第三方 AWS 帳戶 擁有。
- 如需了解如何透過聯合身分提供存取權，請參閱 IAM 使用者指南中的將存取權提供給在外部進行身分驗證的使用者 (聯合身分)。

- 如需了解使用角色和資源型政策進行跨帳戶存取之間的差異，請參閱《IAM 使用者指南》中的 IAM 中的跨帳戶資源存取。

中的彈性 AWS Certificate Manager

AWS 全球基礎設施是以 AWS 區域 和 可用區域為基礎建置。 AWS 區域 提供多個實體隔離和隔離的可用區域，這些可用區域與低延遲、高輸送量和高備援聯網連接。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 AWS 區域和可用區域的詳細資訊，請參閱 AWS 全球基礎設施。

AWS Certificate Manager中的基礎設施安全

作為受管服務， AWS Certificate Manager 受到 AWS 全球網路安全的保護。如需 AWS 安全服務以及如何 AWS 保護基礎設施的資訊，請參閱AWS 雲端安全。若要使用基礎設施安全的最佳實務來設計您的 AWS 環境，請參閱安全支柱 AWS Well-Architected Framework 中的基礎設施保護。

您可以使用 AWS 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 外部互動，則需要程式設計存取 AWS Management Console。授予程式設計存取權的方式取決於正在存取的使用者類型 AWS。

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	根據
人力資源身分 (IAM Identity Center 中管理的使用者)	使用暫時登入資料簽署對 AWS CLI、 AWS SDKs程式設計請求。 AWS APIs	請依照您要使用的介面所提供的指示操作。 • 如需 AWS CLI，請參閱AWS Command Line Interface 《 使用者指南》中的設定 AWS CLI 要使用 AWS IAM Identity Center的 。 • AWS SDKs、工具和 AWS APIs，請參閱 AWS SDKs 和工具參考指南中的 IAM Identity Center 身分驗證。
IAM	使用暫時登入資料簽署對 AWS CLI、 AWS SDKs程式設計請求。 AWS APIs	遵循《IAM 使用者指南》中將臨時登入資料與 AWS 資源搭配使用中的指示。
IAM	(不建議使用) 使用長期登入資料來簽署對 AWS CLI、 AWS SDKs程式設計請求。 AWS APIs	請依照您要使用的介面所提供的指示操作。 • 如需 AWS CLI，請參閱AWS Command Line Interface 《 使用者指南》中的使用 IAM 使用者憑證進行身分驗證。 • AWS SDKs和工具，請參閱 AWS SDKs和工具參考指南中的使用長期憑證進行身分驗證。 • 對於 AWS APIs，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

最佳實務是可協助您更有效地使用 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

帳戶層級分離

在您的政策中使用帳戶層級區隔來控制誰可以在帳戶層級存取憑證。將生產憑證保留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，您可以透過拒絕政策中的`kms:CreateGrant`動作來限制對特定角色的存取。這會限制帳戶中哪些角色可以高階簽署憑證。如需授予的相關資訊，包括授予術語，請參閱《 AWS Key Management Service 開發人員指南》中的 中的授予 AWS KMS。

如果您想要比限制`kms:CreateGrant`帳戶使用 更精細的控制，您可以使用 kms:EncryptionContext 條件金鑰限制`kms:CreateGrant`為特定憑證。指定 `arn:aws:acm`做為金鑰，以及要限制的 ARN 值。下列範例政策會防止使用特定憑證，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

透過 AWS CloudFormation ，您可以建立範本來描述您想要使用 AWS 的資源。 AWS CloudFormation 然後 會為您佈建和設定這些資源。 AWS CloudFormation 可以佈建 ACM 支援的資源，例如 Elastic Load Balancing、Amazon CloudFront 和 Amazon API Gateway。如需詳細資訊，請參閱與 ACM 整合的服務。

如果您使用 AWS CloudFormation 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單獨的 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。

> **Important**
>
> 如果您使用的是 Amazon CloudFront 分佈，請注意 HTTP 驗證不支援萬用字元憑證。在 AWS CloudFormation 範本中包含萬用字元憑證以搭配 Amazon CloudFront 使用時，您必須使用 DNS 驗證或電子郵件驗證。我們建議對自動續約功能進行 DNS 驗證。

在 AWS CloudFormation 用於建立測試環境的 範本中包含萬用字元憑證。

自訂信任存放區

為了確保連線至受 ACM 憑證保護的端點，建議您的自訂信任存放區中包含 Amazon 根目錄。Amazon 根憑證授權單位可以代表不同的金鑰類型和演算法。Starfield Services 根憑證授權機構 - G2 是較舊的根目錄，與其他較舊的信任存放區和用戶端相容，無法更新。透過包含所有根 CAs，您將能夠確保應用程式的最大相容性。

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 中的受管憑證續約 AWS Certificate Manager 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

在 Amazon 憑證授權機構 (CA) 可以為您的網站發出憑證之前， AWS Certificate Manager (ACM) 必須驗證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「AWS Certificate Manager DNS 驗證」及「AWS Certificate Manager 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。

若要在請求憑證時選擇退出透明度記錄，請使用 request-certificate AWS CLI 命令的 `options` 參數或 RequestCertificate API 操作。如果您的憑證是在 2018 年 4 月 24 日之前發行，而且您想要確保憑證不會在續約期間記錄，您可以呼叫 update-certificate-options 命令或 UpdateCertificateOptions API 作業來取消使用此功能。

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"
}
```

如果您已有憑證，而且不希望在續約憑證時記錄憑證，請使用 update-certificate-options 命令。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。CloudTrail 可讓您透過擷取帳戶的 AWS API 呼叫歷史記錄來監控 AWS 部署，包括透過 AWS 管理主控台、 AWS SDKs AWS Command Line Interface、和更高層級的 Amazon Web Services 進行的 API 呼叫。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 將 CloudTrail 整合至應用程式，以自動建立組織的追蹤記錄、查看追蹤記錄的狀態，並控制管理員開啟和關閉 CloudTrail 記錄功能的方式。如需詳細資訊，請參閱建立追蹤記錄。前往 搭配 使用 CloudTrail AWS Certificate Manager 查看 ACM 動作的追蹤記錄範例。

監控和記錄 AWS Certificate Manager

監控是維護 和 AWS 解決方案的可靠性、可用性 AWS Certificate Manager 和效能的重要部分。您應該從 AWS 解決方案的所有部分收集監控資料，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
- 搭配 使用 CloudTrail AWS Certificate Manager
- 支援的 CloudWatch 指標

使用 Amazon EventBridge

您可以使用 Amazon EventBridge （先前稱為 CloudWatch Events) 來自動化您的 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 在內的 AWS 服務事件會以近乎即時的方式交付至 Amazon EventBridge。您可以使用事件來觸發目標，包括 AWS Lambda 函數、 AWS Batch 工作、Amazon SNS 主題等。如需詳細資訊，請參閱什麼是 Amazon EventBridge？

主題

- ACM 的 Amazon EventBridge 支援
- 在 ACM 中使用 Amazon EventBridge 啟動動作

ACM 的 Amazon EventBridge 支援

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。此時機可以使用 ACM API 的 PutAccountConfiguration 動作來變更。

ACM 會自動啟動其發行之合格憑證的續約，但匯入的憑證需要在過期之前重新發行和重新匯入，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 在 ACM 中使用 Amazon EventBridge 啟動動作。

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**
>
> 憑證續約動作 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
   }
}
```

ACM 憑證已撤銷事件

如果帳戶中的 ACM 發行公有或私有憑證遭到撤銷，客戶可以接聽此事件來提醒他們。

> **Note**
>
> 匯入的憑證無法透過撤銷憑證撤銷。

ACM Certificate Revoked 事件具有下列結構。

```
{
  "version": "0",
  "id": "id",
  "detail-type": "ACM Certificate Revoked",
  "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",
    "CertificateExpirationDate" : "2019-12-22T18:43:48Z",
    "Exportable": TRUE | FALSE
  }
}
```

ACM 憑證更新事件

客戶可以接聽此事件，以便在帳戶中的 ACM 發行公有或私有憑證更新時提醒他們。

ACM 憑證 更新的事件具有下列結構。

```
{
  "version": "0",
  "id": "id",
  "detail-type": "ACM Certificate Revoked",
  "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" : "2018-12-22T18:43:48Z",
    "CertificateExpirationDate" : "2019-12-22T18:43:48Z",
    "DaysToExpiry" : 395,
    "InUse" : TRUE | FALSE,
    "Exported" : TRUE | FALSE
    "Exportable" : TRUE | FALSE
  }
}
```

AWS 運作狀態事件

AWS 運作狀態事件會針對符合續約資格的 ACM 憑證產生。如需有關續約資格的資訊，請參閱 中的受管憑證續約 AWS Certificate Manager。

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"
      ]
   }
}
```

在 ACM 中使用 Amazon EventBridge 啟動動作

您可以根據這些事件建立 Amazon EventBridge 規則，並使用 Amazon EventBridge 主控台來設定偵測到事件時所執行的動作。本節提供了設定 Amazon EventBridge 規則和產生動作的範例程序。

主題

- 使用 Amazon SNS 回應事件
- 使用 Lambda 函數回應事件

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。

建立 Amazon EventBridge 規則並觸發動作

1. 建立 Amazon EventBridge 規則。如需詳細資訊，請參閱建立回應事件的 Amazon EventBridge 規則。

   a. 前往 https://console.aws.amazon.com/events/ 進入 Amazon EventBridge 主控台中，導覽至 Events (事件) > Rules (規則) 頁面，然後選擇 Create rule (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 運作狀態事件 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

此程序示範如何使用 在 Amazon EventBridge 上 AWS Lambda 接聽、使用 Amazon Simple Notification Service (SNS) 建立通知，以及將調查結果發佈至 AWS Security Hub，為管理員和安全團隊提供可見性。

設定 Lambda 函數和 IAM 角色

1. 首先設定 AWS Identity and Access Management (IAM) 角色，並定義 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。不建議直接在使用者帳戶下執行大多數 AWS 操作，尤其是在管理員帳戶下。

   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 帳戶詳細資訊。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。如需有關建立 IAM 角色和連接政策的資訊，請參閱為 AWS 服務建立角色（主控台）。
4. 在 https：//https://console.aws.amazon.com/lambda/ 開啟 AWS Lambda 主控台。
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 Function name (函數名稱) 欄位中指定名稱，例如「handle-expiring-certificates」。
   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 Create function (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

  指定不同 AWS Security Hub 區域中的 。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

在本範例中，我們在透過 Amazon EventBridge 引發事件時，會為每個即將過期的憑證提供一封單一電子郵件。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。（此期間可以使用 ACM API 的 PutAccountConfiguration 操作進行自訂。）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. 建立 Amazon EventBridge 規則來觸發 Lambda 函數。如需詳細資訊，請參閱建立回應事件的 Amazon EventBridge 規則。

   前往 https://console.aws.amazon.com/events/ 進入 Amazon EventBridge 主控台中，導覽至 Events (事件) > Rules (規則) 頁面，然後選擇 Create rule (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 Events 規則
- 與 Lambda 相關聯的 CloudWatch Logs
- SNS 主題

搭配 使用 CloudTrail AWS Certificate Manager

AWS Certificate Manager 已與 服務整合 AWS CloudTrail，此服務提供由使用者、角色或 ACM 中的 AWS 服務所採取之動作的記錄。根據預設， AWS 帳戶會啟用 CloudTrail。CloudTrail 會將 ACM 的所有 API 呼叫擷取為事件，包括來自 ACM 主控台的呼叫以及對 ACM API 作業發出的程式碼呼叫。如果

設定了追蹤，就可以將 CloudTrail 事件持續遞送到 Amazon S3 儲存貯體，包括 ACM 的事件。即使您未設定追蹤，依然可以透過 CloudTrail 主控台中的事件歷史記錄檢視最新事件。

您可以利用 CloudTrail 所收集的資訊來判斷向 ACM 發出的請求，以及發出請求的 IP 地址、人員、時間和其他詳細資訊。如需詳細資訊，請參閱《使用 CloudTrail 事件歷史記錄檢視事件》https://docs.aws.amazon.com/awscloudtrail/latest/userguide/view-cloudtrail-events.html。當 ACM 中發生支援的事件活動時，該活動會與事件歷史記錄中的其他 AWS 服務事件一起記錄在 CloudTrail 事件中。您可以檢視、搜尋和下載 AWS 帳戶的最新事件。

此外，您可以設定其他 AWS 服務，以進一步分析和處理 CloudTrail 日誌中收集的事件資料。

如需 CloudTrail 的詳細資訊，請參閱下列文件：

- AWS CloudTrail 使用者指南。
- 建立追蹤的概觀
- CloudTrail 支援的服務和整合
- 設定 CloudTrail 的 Amazon SNS 通知
- 從多個區域接收 CloudTrail 日誌檔案，以及從多個帳戶接收 CloudTrail 日誌檔案

主題

CloudTrail 記錄中支援的 ACM API 動作

ACM 支援將下列 API 動作記錄為 CloudTrail 日誌檔案中的事件：

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：

- 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請求。
- 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 userIdentity 元素。

下列各節提供所支援之 API 操作的範例日誌。

- 新增標籤到憑證 (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)

新增標籤到憑證 (AddTagsToCertificate)

以下 CloudTrail 範例顯示呼叫 AddTagsToCertificate API 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)

以下 CloudTrail 範例顯示呼叫 DeleteCertificate API 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)

以下 CloudTrail 範例顯示呼叫 DescribeCertificate API 的結果。

> **Note**
>
> `DescribeCertificate` 作業的 CloudTrail 日誌不會顯示您指定的 ACM 憑證相關資訊。您可以使用 主控台 AWS Command Line Interface、 或 DescribeCertificate API 來檢視憑證的相關資訊。

```
{
   "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)

以下 CloudTrail 範例顯示呼叫 ExportCertificate API 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":{
```

```
               "certificateArn":"arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012",
                "passphrase": "HIDDEN_DUE_TO_SECURITY_REASONS"
            },
            "responseElements":{
                "certificateChain":
                "-----BEGIN CERTIFICATE-----
                base64 certificate
                -----END CERTIFICATE-----
                -----BEGIN CERTIFICATE-----
                base64 certificate
                -----END CERTIFICATE-----",
                "privateKey":"**********",
                "certificate":
                "-----BEGIN CERTIFICATE-----
                base64 certificate
                -----END CERTIFICATE-----",
                "privateKey": "HIDDEN_DUE_TO_SECURITY_REASONS"
            },
            "requestID":"01234567-89ab-cdef-0123-456789abcdef",
            "eventID":"fedcba98-7654-3210-fedc-ba9876543210",
            "readOnly": false,
            "eventType":"AwsApiCall"
                "managementEvent": true,
                "recipientAccountId": "123456789012",
                "eventCategory": "Management",
                "tlsDetails": {
                     "tlsVersion": "TLSv1.3",
                     "cipherSuite": "TLS_AES_128_GCM_SHA256",
                     "clientProvidedHostHeader": "acm.us-east-1.amazonaws.com"
                 },
                 "sessionCredentialFromConsole": "true"
}
```

匯入憑證 (ImportCertificate)

以下範例顯示的 CloudTrail 日誌項目記錄了對 ACM ImportCertificate API 作業的呼叫。

```
{
   "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)

以下 CloudTrail 範例顯示呼叫 ListCertificates API 的結果。

> **Note**
>
> `ListCertificates` 作業的 CloudTrail 日誌不會顯示 ACM 憑證。您可以使用 主控台 AWS Command Line Interface、 或 ListCertificates API 檢視憑證清單。

```
{
```

```
   "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)

以下 CloudTrail 範例顯示呼叫 ListTagsForCertificate API 的結果。

> **Note**
>
> `ListTagsForCertificate` 作業的 CloudTrail 日誌不會顯示標籤。您可以使用主控台、AWS Command Line Interface或 ListTagsForCertificate API 檢視標籤清單。

```
{
   "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)

以下 CloudTrail 範例顯示呼叫 RemoveTagsFromCertificate API 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)

以下 CloudTrail 範例顯示呼叫 RequestCertificate API 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":{
            "domainName":"example.com",
            "validationMethod": "DNS",
            "idempotencyToken":"8186023d89681c3ad5",
            "options": {
            "export": "ENABLED"
        },
        "keyAlgorithm": "RSA_2048"
         },
         "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",
         "tlsDetails": {
           "tlsVersion": "TLSv1.3",
           "cipherSuite": "TLS_AES_128_GCM_SHA256",
           "clientProvidedHostHeader": "acm.us-east-1.amazonaws.com"
          },
         "recipientAccountId":"123456789012"
      }
   ]
}
```

撤銷憑證 (RevokeCertificate)

下列 CloudTrail 範例顯示呼叫 RevokeCertificate API 的結果。

```
{
    "eventVersion": "1.11",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE:Role-Session-Name",
```

```
        "arn": arn:aws:sts::111122223333:assumed-role/Role-Name/Role-Session-Name",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/Admin",
                "accountId": "123456789012",
                "userName": "Admin"
            },
            "attributes": {
                "creationDate": "2016-01-01T19:35:52Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime":"2016-01-01T21:11:45Z",
    "eventSource": "acm.amazonaws.com",
    "eventName": "RevokeCertificate",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:128.0) Gecko/20100101
 Firefox/128.0",
    "requestParameters": {
        "certificateArn": "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012",
        "revocationReason": "UNSPECIFIED"
    },
    "responseElements": {
        "certificateArn": "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
    },
    "requestID": "01234567-89ab-cdef-0123-456789abcdef",
    "eventID": "01234567-89ab-cdef-0123-456789abcdef",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management",
    "tlsDetails": {
        "tlsVersion": "TLSv1.3",
        "cipherSuite": "TLS_AES_128_GCM_SHA256",
        "clientProvidedHostHeader": "acm.us-east-1.amazonaws.com"
```

```
    },
    "sessionCredentialFromConsole": "true"
}
```

重新傳送驗證電子郵件 (ResendValidationEmail)

以下 CloudTrail 範例顯示呼叫 ResendValidationEmail API 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)

以下 CloudTrail 範例顯示呼叫 GetCertificate API 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

您可以使用 CloudTrail 稽核整合 ACM 的服務所發出的 API 呼叫。如需使用 CloudTrail 的詳細資訊，請參閱 AWS CloudTrail 使用者指南。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

您可以使用 CloudTrail 稽核整合 ACM 的服務所發出的 API 呼叫。如需使用 CloudTrail 的詳細資訊，請參閱 AWS CloudTrail 使用者指南。下列範例顯示根據您佈建 ACM 憑證 AWS 的資源，可以產生的日誌類型。

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。下列範例顯示針對 AWS 帳戶 123456789012 上名為 LinuxTest `RegisterInstancesWithLoadBalancer` 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。解密會在 內執行 AWS，解密的金鑰永遠不會離開 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。您可以使用 CloudWatch 來收集和追蹤指標、設定警示，並自動回應 AWS 資源的變更。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 指標的詳細資訊，請參閱下列主題：

- 使用 Amazon CloudWatch 指標
- 建立 Amazon CloudWatch 警示

AWS Certificate Manager 搭配適用於 Java 的 SDK 使用

您可以使用 AWS Certificate Manager API 透過傳送 HTTP 請求，以程式設計方式與服務互動。如需詳細資訊，請參閱 AWS Certificate Manager API 參考。

除了 Web API （或 HTTP API)，您可以使用 AWS SDKs和命令列工具與 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 SDKs，適用於 Java 的 AWS SDK在 AWS Certificate Manager API 中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，您必須先授予 ACM 服務主體許可才能執行此操作。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。

對 的問題進行故障診斷 AWS Certificate Manager

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

對憑證請求進行故障診斷

如果您在請求 ACM 憑證時遇到問題，請參閱下列主題。

主題

憑證請求逾時

對 ACM 憑證的請求如果沒有在 72 小時內通過驗證，該請求便會逾時。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 AWS Certificate Manager DNS 驗證 或 AWS Certificate Manager 電子郵件驗證 。我們建議您盡可能使用 DNS 驗證。

憑證請求失敗

如果您的 ACM 請求失敗，而且收到下列其中一個錯誤訊息，請依照建議的步驟來修正問題。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

您在請求憑證時選擇了電子郵件驗證，但是 ACM 找不到用來驗證請求中的一個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：

- 確定您的網域已設定成可接收電子郵件。您網域的名稱伺服器必須有郵件交換程式記錄 (MX 記錄)，ACM 的電子郵件伺服器才知道要將網域驗證電子郵件傳送到何處。

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。

如需如何確保收到來自 ACM 的網域驗證電子郵件的詳細資訊，請參閱「AWS Certificate Manager 電子郵件驗證」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 需要其他資訊來處理此憑證請求。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 支援。如果您沒有支援方案，請在 ACM 開發論壇中張貼新的討論主題。

> **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，您無法為 請求 ACM 憑證，`example.invalidpublicdomain`因為「invalidpublicdomain」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 ACM 開發論壇中張貼新的討論主題。

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 支援。如果您沒有支援方案，請在 ACM 開發論壇中張貼新的討論主題。

對憑證驗證進行故障診斷

如果 ACM 憑證請求狀態為 Pending validation (待定驗證)，表示請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。這些訊息已傳送至所請求網域的常見電子郵件地址。如需詳細資訊，請參閱AWS Certificate Manager 電子郵件驗證。如果您選擇 DNS 驗證，則必須將 ACM 為您建立的 CNAME 記錄寫入 DNS 資料庫。如需詳細資訊，請參閱AWS Certificate Manager 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」。如果選擇 DNS 驗證，則必須為每個網域建立一個 CNAME 記錄。

> **Note**
>
> 公有 ACM 憑證可以安裝在連接到 Nitro Enclave 的 Amazon EC2 執行個體上。您也可以匯出公有憑證，以便在任何 Amazon EC2 執行個體上使用。如需了解如何在未連接至 Nitro Enclave 的 Amazon EC2 執行個體上設定獨立 Web 伺服器，請參閱教學課程：在 Amazon Linux 2 上安裝 LAMP Web 伺服器或教學課程：使用 Amazon Linux AMI 安裝 LAMP Web 伺服器。

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

- 針對 DNS 驗證問題進行疑難排解
- 針對電子郵件驗證問題進行疑難排解
- 故障診斷 HTTP 驗證問題

針對 DNS 驗證問題進行疑難排解

如果您無法使用 DNS 驗證憑證，請參閱下列指導方針。

DNS 疑難排解的第一步是使用如下工具來檢查網域的目前狀態：

- dig - Linux、Windows
- nslookup - Linux、Windows

主題

- DNS 供應商禁止使用底線
- DNS 供應商新增的預設結尾句點
- GoDaddy 上的 DNS 驗證失敗
- ACM 主控台未顯示「在 Route 53 中建立記錄」按鈕
- 私有 (不信任) 網域上的 Route 53 驗證失敗
- 驗證成功，但發行或續約失敗
- VPN 上的 DNS 伺服器驗證失敗

DNS 供應商禁止使用底線

如果您的 DNS 供應商禁止在 CNAME 值中使用前置底線，您可以從 ACM 提供的值中移除底線，並使用沒有底線的值來驗證網域。例如，CNAME 值 `_x2.acm-validations.aws` 可以變更為 `x2.acm-validations.aws` 以用於驗證用途。不過，CNAME 名稱參數必須一律有前置底線。

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.
_<random value>.example.com.	CNAME	<random value>.acm-validations.aws.

DNS 供應商新增的預設結尾句點

某些 DNS 供應商預設會為您所提供的 CNAME 值新增結尾句點。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。

GoDaddy 上的 DNS 驗證失敗

除非您修改 ACM 提供的 CNAME 值，否則使用 Godaddy 和其他登錄檔註冊之網域的 DNS 驗證可能會失敗。若以 example.com 做為網域名稱，發出的 CNAME 記錄則會具有下列形式：

```
NAME: _ho9hv39800vb3examplew3vnewoib3u.example.com. VALUE:
 _cjhwou20vhu2exampleuw20vuyb2ovb9.j9s73ucn9vy.acm-validations.aws.
```

您可以截斷「名稱」欄位結尾處的 Apex 網域 (包括句號)，藉此建立與 GoDaddy 相容的 CNAME 記錄，如下所示：

```
NAME: _ho9hv39800vb3examplew3vnewoib3u VALUE:
 _cjhwou20vhu2exampleuw20vuyb2ovb9.j9s73ucn9vy.acm-validations.aws.
```

ACM 主控台未顯示「在 Route 53 中建立記錄」按鈕

如果您選取 Amazon Route 53 做為 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。

- 您不是使用 Route 53 做為您的 DNS 供應商。
- 您用不同的帳戶登入 ACM 和 Route 53。
- 您不具備在 Route 53 託管區域中建立記錄的 IAM 許可。
- 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，ACM 會在公開託管區域中搜尋 CNAME。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，包含 Amazon VPC 私有託管區域中的資源、私有 PKI 中的不受信任網域，以及自我簽署憑證。

AWS 會透過 AWS 私有 CA服務提供公開不受信任網域的支援。

驗證成功，但發行或續約失敗

如果憑證發行失敗並顯示「待定驗證」(即使 DNS 是正確的)，請檢查發行未被憑證授權機構授權 (CAA) 記錄封鎖。如需詳細資訊，請參閱(選用) 設定 CAA 記錄。

VPN 上的 DNS 伺服器驗證失敗

如果您在 VPN 上找到 DNS 伺服器，且 ACM 無法驗證憑證，請檢查該伺服器是否可公開存取。使用 ACM DNS 驗證的公有憑證發行需透過公有網際網路解析網域記錄。

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
- 電子郵件驗證的用久性初始時間戳記
- 我無法切換為 DNS 驗證

未收到驗證電子郵件

當您向 ACM 請求憑證並選擇電子郵件驗證時，網域驗證電子郵件會傳送到五個常見的管理地址。如需詳細資訊，請參閱AWS Certificate Manager 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

ACM 會將驗證電子郵件訊息傳送至您請求的網域名稱。如果您想要改為在該網域接收這些電子郵件，也可以指定超級網域做為驗證網域。最小網站地址之前的任何子網域都是有效的，並在 @ 之後用作電子郵件地址的網域。例如，如果您將 example.com 指定為 的驗證網域，則可能會收到

admin@example.com 的電子郵件 subdomain.example.com 。檢閱 ACM 主控台中顯示 (或從 CLI 或 API 傳回) 的電子郵件地址清單，判斷您應該在何處尋找驗證電子郵件。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 自動分類您的電子郵件

如果您使用的是 GMail，驗證電子郵件可能已被自動分類到最新快訊或促銷內容標籤中。

網域註冊商未顯示聯絡資訊或已啟用隱私權保護

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。

在提供 傳送驗證電子郵件的 AWS 五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 ACM 請求憑證。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 ACM Request a Certificate (請求憑證) 精靈的 Step 3: Validate (步驟 3：驗證) 中檢視清單。列出的電子郵件地址

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 支援 中心並建立案例。如果您沒有支援協議，請在 ACM 開發論壇中張貼訊息。

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。這並非 ACM 作業中發生錯誤的證據。

我無法切換為 DNS 驗證

使用電子郵件驗證建立憑證之後，您無法切換為使用 DNS 進行驗證。若要使用 DNS 驗證，請刪除憑證，然後建立使用 DNS 驗證的新憑證。

故障診斷 HTTP 驗證問題

如果您在使用 HTTP 驗證憑證時遇到問題，請參閱下列指引。

HTTP 故障診斷的第一步是使用下列工具檢查網域的目前狀態：

- curl — Linux 和 Windows
- wget — Linux 和 Windows

主題

RedirectFrom 和 RedirectTo 位置之間的內容不相符

如果`RedirectFrom`位置的內容與`RedirectTo`位置的內容不相符，則驗證將會失敗。確保憑證中每個網域的內容都相同。

CloudFront 組態不正確

請確定您的 CloudFront 分佈已正確設定為提供驗證內容。檢查原始伺服器和行為設定是否正確，以及分佈是否已部署。

HTTP 重新導向問題

如果您使用重新導向而非直接提供內容，請依照下列步驟驗證您的組態。

驗證重新導向組態

1. 複製 `RedirectFrom` URL 並將其貼到瀏覽器的地址列。

2. 在新的瀏覽器索引標籤中，貼上 RedirectTo URL。
3. 比較兩個 URLs的內容，以確保它們完全相符。
4. 確認重新導向傳回 302 狀態碼。

驗證逾時

如果內容在預期的時間範圍內無法使用，HTTP 驗證可能會逾時。若要疑難排解驗證問題，請依照下列步驟進行。

疑難排解驗證逾時

1. 執行下列其中一項來檢查哪些網域正在等待驗證：
   a. 開啟 ACM 主控台並檢視憑證詳細資訊頁面。尋找標記為待定驗證的網域。
   b. 呼叫 `DescribeCertificate` API 操作以檢視每個網域的驗證狀態。
2. 對於每個待定網域，請確認驗證內容可從網際網路存取。

對受管憑證續約進行故障診斷

ACM 會在 ACM 憑證過期之前嘗試自動續約，您無須執行任何動作。如果您有 中的受管憑證續約 AWS Certificate Manager 的相關問題，請參閱下列主題。

準備自動網域驗證

必須符合下列條件，ACM 才可以自動續約您的憑證：

- 您的憑證必須與與 ACM 整合的 AWS 服務相關聯。如需有關 ACM 支援的資源的資訊，請參閱 與 ACM 整合的服務。
- 對於電子郵件驗證的憑證，ACM 必須能透過憑證中列出之每個網域的管理員電子郵件地址來聯繫您。系統將嘗試的電子郵件地址會列於 AWS Certificate Manager 電子郵件驗證 中。
- 對於 DNS 驗證的憑證，請確定您的 DNS 組態包含正確的 CNAME 記錄，如 AWS Certificate Manager DNS 驗證 中所述。
- 對於 HTTP 驗證的憑證，請確定您的重新導向已如中所述進行設定AWS Certificate Manager HTTP 驗證。

受管憑證續約處理失敗

憑證即將到期 (DNS 為 60 天、電子郵件為 45 天，私人憑證為 60 天)，如果憑證符合資格條件，ACM 會嘗試更新憑證。您可能必須採取行動才能成功續約。如需詳細資訊，請參閱中的受管憑證續約 AWS Certificate Manager。

經電子郵件驗證之憑證的受管憑證續約

ACM 憑證的有效期限為 13 個月 (395 天)。續約憑證需要網域擁有者執行動作。ACM 會在過期前 45 天開始將續約通知傳送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以進行續約的連結。驗證所有列出的網域後，ACM 會發行具有相同 ARN 的續約憑證。

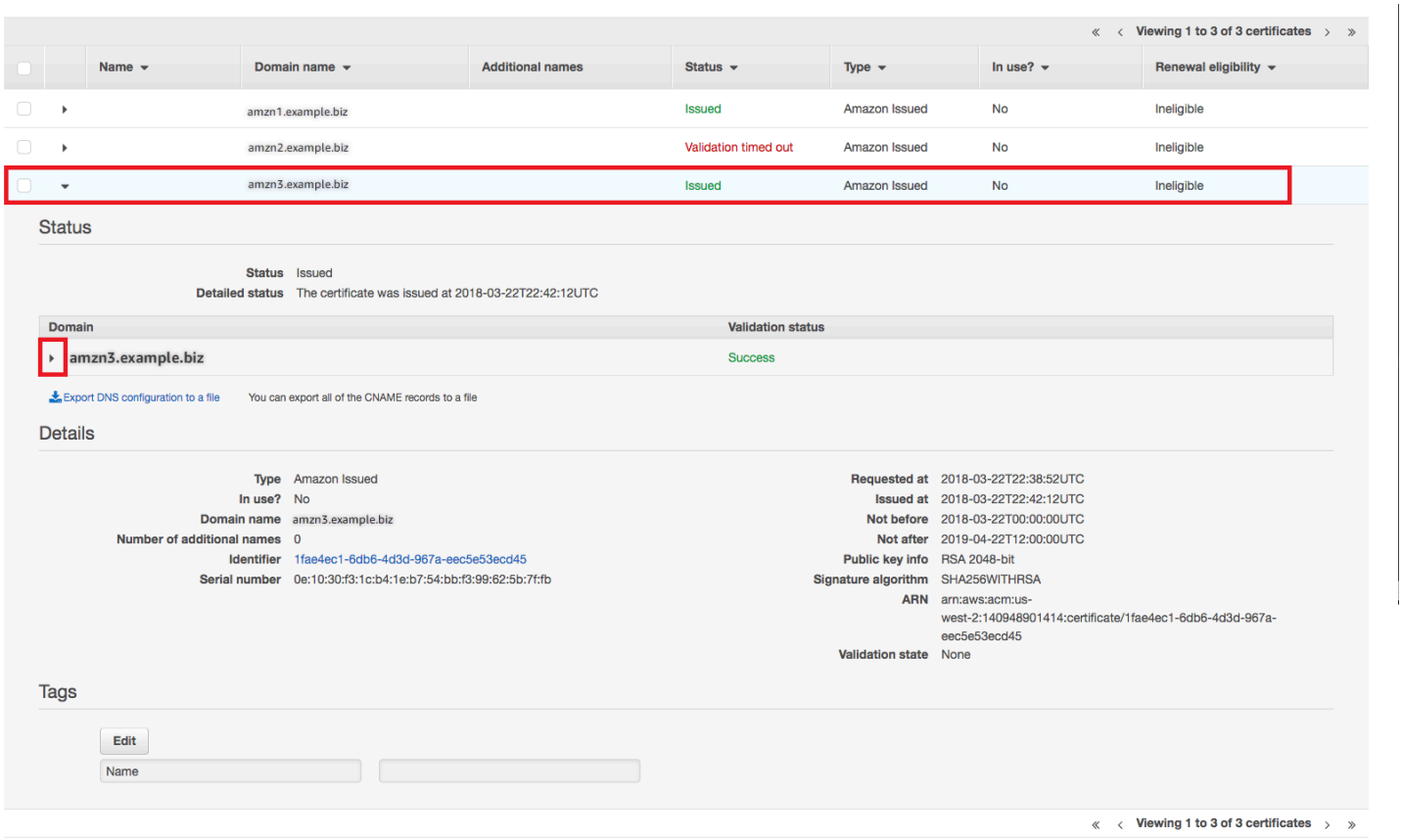
請參閱使用電子郵件驗證取得指引，了解如何識別哪些網域處於 `PENDING_VALIDATION` 狀態，並針對這些網域重複執行驗證程序。

經 DNS 驗證之憑證的受管憑證續約

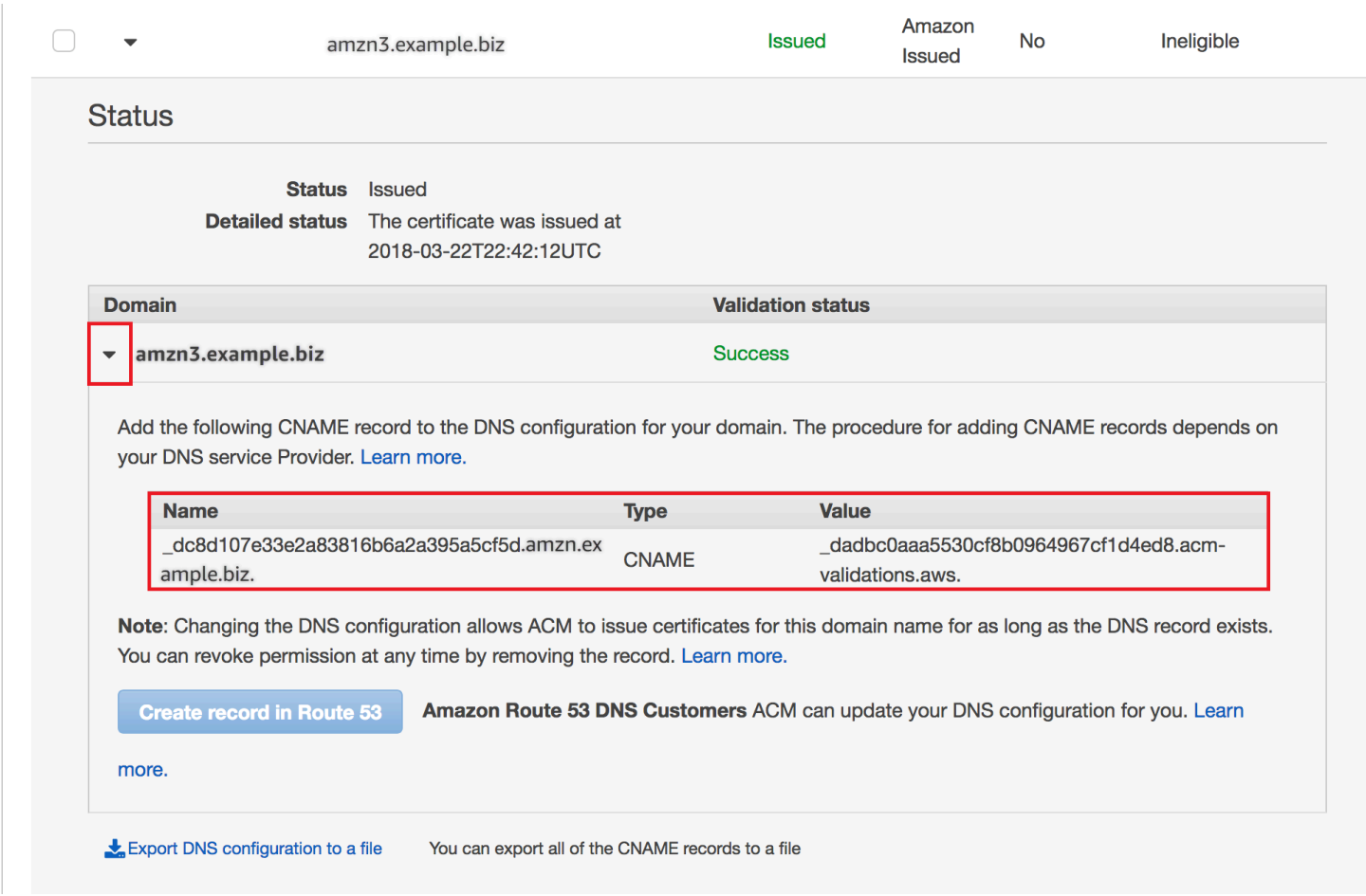
ACM 不會嘗試對經 DNS 驗證的憑證進行 TLS 驗證。如果 ACM 無法將您之前透過 DNS 驗證的憑證續約，最有可能的原因是 DNS 組態中的 CNAME 記錄遺失或不準確。如果發生這種情況，ACM 會通知您可能無法自動續約憑證。

> **Important**
>
> 您必須將正確的 CNAME 記錄插入您的 DNS 資料庫。操作方式請洽詢您的網域註冊商。

您可以在 ACM 主控台中展開您的憑證及其網域項目，尋找您網域的 CNAME 記錄。如需詳細資訊，請參閱下圖。若要擷取 CNAME 記錄，您也可以在 ACM API 中使用 DescribeCertificate 作業，或在 ACM CLI 中使用 describe-certificate 命令。如需詳細資訊，請參閱AWS Certificate Manager 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。

HTTP 驗證憑證的受管憑證續約

ACM 會嘗試自動續約 HTTP 驗證的憑證。如果續約失敗，可能是因為 HTTP 驗證記錄發生問題。在這種情況下，ACM 會通知您憑證無法自動續約。

> **Important**
>
> 您必須確保`RedirectFrom`位置的內容與憑證中每個網域`RedirectTo`位置的內容相符。

您可以在 ACM 主控台中擴展憑證及其網域項目，以尋找網域的 HTTP 驗證資訊。您也可以使用 ACM API 中的 DescribeCertificate 操作或 ACM CLI 中的 describe-certificate 命令來擷取此資訊。如需詳細資訊，請參閱AWS Certificate Manager HTTP 驗證。

如果問題仍存在，請聯絡支援中心。

了解續約時機

中的受管憑證續約 AWS Certificate Manager 是非同步的程序。這表示步驟不會緊接著連續發生。驗證 ACM 憑證中的所有網域名稱後，ACM 取得新憑證的過程可能會發生延遲。ACM 取得續約後的憑證到將憑證部署至使用該憑證的 AWS 資源期間，可能還會發生延遲。因此，憑證狀態的變更可能需要數小時才會在主控台顯示。

故障診斷其他問題

本節包含解決與發行或驗證 ACM 憑證無關之問題的指引。

主題

- 憑證授權機構授權 (CAA) 的問題
- 憑證匯入問題
- 憑證關聯定問題
- API Gateway 問題
- 工作憑證未預期失敗時該如何處理
- ACM 服務連結角色 (SLR) 的問題

憑證授權機構授權 (CAA) 的問題

您可以使用 CAA DNS 記錄指定 Amazon 憑證授權機構 (CA) 可為您的網域或子網域發行 ACM 憑證。如果您在憑證發行期間收到顯示由於憑證授權機構授權 (CAA) 錯誤，一或多個網域名稱的驗證失敗的錯誤，請檢查您的 CAA DNS 記錄。如果您在成功驗證 ACM 憑證請求後收到此錯誤，則必須更新 CAA 記錄，然後重新請求憑證。您 CAA 記錄中的數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 (選用) 設定 CAA 記錄。

> **Note**
>
> 如果您不想要啟用 CAA 檢查，可以選擇不為您的網域設定 CAA 記錄。

憑證匯入問題

您可以將第三方憑證匯入 ACM 並將它們與整合服務建立關聯。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 版本 3 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、憑證鏈和私有金鑰 (若有) 都必須以 PEM 編碼。一般而言，PEM 編碼是由以 Base64 編碼的 ASCII 文字區塊組成，這些區塊的開頭和結尾是全文字標頭和註腳行。您不得在複製或上傳 PEM 檔案時新增行或空格，或對 PEM 檔案進行任何其他變更。您可以使用 OpenSSL 驗證公有程式來驗證憑證鏈。
- 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 ACM 整合的服務必須使用 ACM 支援的演算法和金鑰大小。請參閱 AWS Certificate Manager 使用者指南和每個服務的文件，以確保您的憑證可以正常運作。
- 整合服務對憑證的支援可能因憑證是匯入 IAM 還是 ACM 而有所不同。
- 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。不過，如果您呼叫 ListCertificates API 或 list-certificates AWS CLI 命令而未指定`keyTypes`篩選條件，則只會顯示 `RSA_1024`或 `RSA_2048`憑證。

憑證關聯定問題

為了續約憑證，ACM 會產生新的公私有金鑰對。如果您的應用程式使用 憑證關聯，有時稱為 SSL 鎖定，以鎖定 ACM 憑證，則應用程式在 AWS 續約憑證後可能無法連線到您的網域。因此，建議您不要關聯 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 不會為匯入的憑證提供受管續約。
- 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 Gateway 問題

部署邊緣最佳化 API 端點時，API Gateway 會為您設定 CloudFront 分佈。CloudFront 分佈的擁有者是 API Gateway，而不是您的帳戶。分佈會繫結至您在部署 API 時使用的 ACM 憑證。若要移除繫結並允許 ACM 刪除您的憑證，您必須移除與憑證相關聯的 API Gateway 自訂網域。

當您部署區域 API 端點時，API Gateway 會代表您建立 Application Load Balancer (ALB)。負載平衡器的擁有者是 API Gateway，而且不會向您顯示。ALB 會繫結至您在部署 API 時使用的 ACM 憑證。若要移除繫結並允許 ACM 刪除您的憑證，您必須移除與憑證相關聯的 API Gateway 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 憑證與整合服務建立關聯，但憑證停止運作，而整合服務開始傳回錯誤，可能是因為服務使用 ACM 憑證所需的許可有所改變。

例如，Elastic Load Balancing (ELB) 需要解密 的許可， AWS KMS key 進而解密憑證的私有金鑰。此許可是由資源型政策所授權，當您為憑證與 ELB 建立關聯時，ACM 會套用此政策。如果 ELB 失去該許可的授權，則會在下次嘗試解密憑證金鑰時失敗。

若要調查問題，請使用 AWS KMS 主控台檢查授予的狀態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 ACM 的許可已撤銷，請聯絡 支援 https：//https://console.aws.amazon.com/support/home#/。

ACM 服務連結角色 (SLR) 的問題

當您發行由另一個帳戶與您共用的私有 CA 簽署的憑證時，ACM 會先嘗試設定服務連結角色 (SLR)，以委託人身分與 AWS 私有 CA 以資源為基礎的存取政策互動。如果您從共用 CA 發行私有憑證，但 SLR 尚未就緒，ACM 將無法自動為您續約該憑證。

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。

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。

處理例外狀況

AWS Certificate Manager 命令可能會因為多種原因而失敗。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

當您嘗試續約 發行的私有 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）區域和中國（寧夏）區域不支援 。

ACM 失敗代碼	註解
`PCA_ACCESS_DENIED`	私有憑證授權機構尚未獲得 ACM 許可。這會觸發 AWS 私有 CA `AccessDeniedException` 失敗代碼。 若要修正此問題，請使用 AWS 私有 CA CreatePermission 操作將必要的許可授予 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)。

ACM 失敗代碼	註解
PCA_INVALID_STATE	呼叫的私有憑證授權機構並未處於執行所請求 ACM 作業的正確狀態。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 • 如果 CA 的狀態為 FAILED，而且您無法解決問題，請連絡 支援。
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 支援 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 支援。

ACM 失敗代碼	註解
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。確認您使用了正確的 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	為了續約位於另一個帳戶的私有憑證授權機構所簽署的憑證，ACM 需要憑證所在帳戶上的服務連結角色 (SLR)。如果您需要重新建立已刪除的 SLR，請參閱「為 ACM 建立 SLR」。

配額

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

若要查看哪些配額可以調整，請參閱《AWS 一般參考指南》中的 ACM 配額資料表。如需申請提高配額，請在 支援 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。 CA 從 簽署的憑證 AWS 私有 CA 不會計入此總計。	2500
每年 (過去 365 天) 的 ACM 憑證數量 每個區域的每個帳戶每年最多可以請求 ACM 憑證配額的兩倍數量。例如，如果您的配額是 2,500 個，則每年可以在每個指定的區域和帳戶中請求最多 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。如果您在某段時間內需要超過 2,500 個憑證，您必須聯絡 支援 Center。 CA 從 簽署的憑證 AWS 私有 CA 不會計入此總計。	5,000
匯入憑證的數量	2,500
每年 (過去 365 天) 匯入的憑證數量	5,000
每個 ACM 憑證的網域名稱數量	10

項目	預設配額
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。若要請求提高配額，請在Service Quotas主控台中為 ACM 服務建立請求。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 ACM 憑證網域名稱的數量配額僅適用於 ACM 提供的憑證。此配額不適用於匯入 ACM 的憑證。以下章節僅適用於 ACM 憑證。	
私有 CA 的數量 ACM 已與 AWS Private Certificate Authority () 整合AWS 私有 CA。您可以使用 ACM 主控台 AWS CLI或 ACM API，從託管的現有私有憑證授權機構 (CA) 請求私有憑證 AWS 私有 CA。這些憑證是在 ACM 環境內管理，且其限制與 ACM 發行的公有憑證相同。如需詳細資訊，請參閱在 中請求私有憑證 AWS Certificate Manager。您也可以使用獨立 AWS 私有 CA 服務發行私有憑證。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 速率配額

下列配額適用於每個區域和帳戶的 ACM API。ACM 會根據 API 作業以不同速率調節 API 請求。調節表示 ACM 會因為請求超過作業對每秒請求數的配額，而拒絕本應有效的請求。當請求受到調節，ACM 會傳回 `ThrottlingException` 錯誤。下表列出每個 API 作業以及 ACM 會為該作業調節請求的配額。

> **Note**
>
> 除了下表中列出的 API 動作外，ACM 也可以從 AWS 私有 CA呼叫外部 `IssueCertificate` 動作。如需 `IssueCertificate` 即時費率配額的相關資訊，請參閱適用於 AWS 私有 CA 的端點和配額。

每個 ACM API 作業的每秒請求數配額

API 呼叫	每秒請求數
`AddTagsToCertificate`	5
`DeleteCertificate`	10
`DescribeCertificate`	10
`ExportCertificate`	10
`GetAccountConfiguration`	1
`GetCertificate`	10
`ImportCertificate`	1
`ListCertificates`	8
`ListTagsForCertificate`	10
`PutAccountConfiguration`	1
`RemoveTagsFromCertificate`	5

API 呼叫	每秒請求數
RenewCertificate	5
RequestCertificate	5
ResendValidationEmail	1
UpdateCertificateOptions	5

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

文件歷史紀錄

下表說明 2018 年 AWS Certificate Manager 開始的文件發行歷史記錄。

變更	描述	日期
AWS Certificate Manager 可匯出的公有憑證	您可以匯出 ACM 公有憑證。	2025 年 6 月 17 日
ACM 支援使用 CloudFront 進行 HTTP 驗證	ACM 現在支援在為 CloudFront 分佈發行憑證時進行網域擁有權驗證的 HTTP 驗證。	2025 年 4 月 24 日
取代郵件交換程式 (MX) 電子郵件驗證	ACM 主控台不再支援郵件交換程式 (MX)。	2024 年 7 月 11 日
新增帳戶層級區隔的最佳實務	盡可能在政策中使用帳戶層級區隔。如果不可行，您可以在帳戶層級或透過政策中的加密內容條件金鑰來限制許可。	2024 年 6 月 11 日
WHOIS 電子郵件驗證即將棄用	已新增自 2024 年 6 月起棄用 WHOIS 電子郵件驗證的注意事項。	2024 年 2 月 5 日
新增條件索引鍵支援	新增請求取得 ACM 憑證時，對 IAM 條件索引鍵的支援。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
已新增 ECDSA 支援	新增在要求公有 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	2022 年 11 月 8 日

	https://docs.aws.amazon.com/acm/latest/userguide/acm-certificate.html#algorithms。	
新 CloudWatch 事件	新增 ACM Certificate Expired (ACM 憑證已過期)、ACM Certificate Available (ACM 憑證可用)，以及 ACM Certificate Renewal Action Required (需要 ACM 憑證更新動作) 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 ACM 的憑證現在可具有使用其他 RSA 和橢圓曲線演算法的索引鍵。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。這項變更的影響範圍涵蓋 CloudWatch 指標、CloudWatch Events/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	新增 DaysToExpiry 指標和事件以及支援的 API。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	新增使用私有 CAs的跨帳戶支援 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	新增對 AWS 中國（北京和寧夏）區域的區域支援。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 ACM 受管續約工作流程的組態。如需更多資訊，請參閱測試 ACM 受管續約的組態。	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 公有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	已啟動 ACM Private Certificate Manager (CM) AWS Certificate Manager ，以及可讓使用者建立安全受管基礎設施以發行和撤銷私有數位憑證的 延伸。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 年 AWS Certificate Manager 之前的文件發行歷史記錄。

變更	描述	版本日期
新內容	已新增 DNS 驗證到 AWS Certificate Manager DNS 驗證。	2017 年 11 月 21 日
新內容	已新增新的 Java 程式碼範例到 AWS Certificate Manager 搭配適用於 Java 的 SDK 使用。	2017 年 10 月 12 日
新內容	已新增 CAA 記錄的相關資訊到 (選用) 設定 CAA 記錄。	2017 年 9 月 21 日
新內容	已新增 .IO 網域的相關資訊到對 的問題進行故障診斷 AWS Certificate Manager。	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 對 的問題進行故障診斷 AWS Certificate Manager。	2017 年 7 月 07 日

變更	描述	版本日期
新內容	已 AWS CloudFormation 新增至 與 ACM 整合的服務。	2017 年 5 月 27 日
更新	已新增詳細資訊到 配額。	2017 年 5 月 27 日
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 AWS Certificate Manager 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 AWS Certificate Manager 電子郵件驗證。	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	新增對 AWS Elastic Beanstalk 和 Amazon API Gateway 的支援。請參閱 與 ACM 整合的服務。	2017 年 3 月 21 日
更新	已更新 受管憑證續約 的相關文件。	2017 年 2 月 20 日
新內容	已新增 匯入的憑證 的相關文件。	2016 年 10 月 13 日

變更	描述	版本日期
新內容	新增對 ACM 動作的 AWS CloudTrail 支援。請參閱 搭配使用 CloudTrail 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。