
AWS Certificate Manager

用户指南

版本 1.0



AWS Certificate Manager: 用户指南

Copyright © 2018 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

什么是 AWS Certificate Manager ?	1
概念	1
ACM 证书	2
顶级域	3
非对称密钥加密	3
证书颁发机构	3
证书透明度日志	3
域名系统	4
域名	4
加密和解密	5
完全限定域名 (FQDN)	5
公有密钥基础设施	5
根证书	5
安全套接字层 (SSL)	5
安全 HTTPS	5
SSL 服务器证书	5
对称密钥加密	6
传输层安全性 (TLS)	6
信任	6
ACM 证书特征	6
支持的区域	7
集成的服务	8
网站签章和信任徽标	9
限制	9
每年的 ACM 证书的数量 (最近 365 天)	9
每个 ACM 证书的域名数量	10
私有 CA 和证书的数量	10
最佳实践	10
AWS CloudFormation	10
证书固定	11
域验证	11
添加或删除域名	11
选择退出证书透明度日志记录	11
启用 AWS CloudTrail	12
定价	12
设置	13
设置 AWS 和 IAM	13
注册 AWS	13
创建 IAM 用户	13
注册域名	14
设置站点或应用程序	14
Linux 快速入门	14
Windows 快速入门	15
(可选) 配置电子邮件	15
WHOIS 数据库	15
MX 记录	15
(可选) 配置 CAA	16
入门	18
请求公有证书	18
使用控制台请求公有证书	18
使用 CLI 请求公有证书	19
请求私有证书	20
使用控制台请求私有证书	20
使用 CLI 请求私有证书	21

导出私有证书	21
使用控制台导出私有证书	21
使用 CLI 导出私有证书	22
使用 DNS 验证	22
向您的数据库添加 CNAME	25
从您的数据库中删除 CNAME	25
使用电子邮件验证	25
列出证书	29
列出证书 (控制台)	29
列出证书 (CLI)	30
描述证书	31
描述证书 (控制台)	31
描述证书 (CLI)	31
删除证书	33
删除证书 (控制台)	33
删除证书 (CLI)	33
安装 ACM 证书	33
重新发送电子邮件 (可选)	33
重新发送电子邮件 (控制台)	33
重新发送电子邮件 (CLI)	34
托管续订	35
域验证	35
自动域验证的工作方式	35
如果自动验证失败	36
检查续订状态	37
检查状态 (控制台)	37
检查状态 (API)	38
检查状态 (CLI)	38
检查状态 (PHD)	38
请求电子邮件 (可选)	39
导入证书	41
先决条件	41
证书格式	42
导入证书	43
使用控制台导入	43
使用 AWS CLI 导入	44
重新导入证书	44
使用控制台重新导入	44
使用 AWS CLI 重新导入	45
为 ACM 证书添加标签	46
标签限制	46
管理标签	46
管理标签 (控制台)	47
管理标签 (AWS Command Line Interface)	48
管理标签 (AWS Certificate Manager API)	48
身份验证和访问控制	49
身份验证	49
访问控制	50
访问管理概述	50
ACM 资源和操作	50
了解资源所有权	51
管理对 ACM 证书的访问	51
AWS 托管策略	51
AWSCertificateManagerReadOnly	51
AWSCertificateManagerFullAccess	52
客户托管策略	52
内联策略	52

列出证书	53
检索证书	53
导入证书	53
删除证书	53
对 ACM 的只读访问	54
对 ACM 的完全访问权限	54
对所有 AWS 资源的管理员访问权限	55
ACM API 权限参考	55
使用 AWS CloudTrail	57
记录 ACM API 调用	57
添加标签	58
删除证书	59
描述证书	59
导出证书	60
导入证书	61
列出证书	62
列出标签	63
删除标签	63
请求证书	64
重新发送电子邮件	65
检索证书	65
记录 ACM 相关的 API 调用	66
创建负载均衡器	66
注册 Amazon EC2	67
加密私有密钥	68
解密私有密钥	68
使用 ACM API	70
AddTagsToCertificate	70
DeleteCertificate	72
DescribeCertificate	73
ExportCertificate	75
GetCertificate	77
ImportCertificate	79
ListCertificates	81
ListTagsForCertificate	82
RemoveTagsFromCertificate	84
RequestCertificate	85
ResendValidationEmail	87
ACM 私有密钥安全	89
故障排除	90
CAA 记录	90
电子邮件	90
未收到验证电子邮件	91
已发送到子域的电子邮件	92
隐藏的联系人信息	92
证书续订	92
WHOIS 限制	92
证书导入	92
证书固定	93
证书请求	93
证书请求超时	93
证书请求失败	93
证书续订	95
自动域验证	95
异步过程	95
证书验证	95
验证未完成	96

.IO 域	96
API 网关	96
文档历史记录	97

什么是 AWS Certificate Manager ?

欢迎使用 AWS Certificate Manager (ACM) 服务。ACM 处理在创建和管理基于 AWS 的网站和应用程序的公共 SSL/TLS 证书时的各项繁杂工作。您可以使用[由 ACM 提供的公有证书 \(p. 18\)](#) (ACM 证书) 或[您导入到 ACM 的证书 \(p. 41\)](#)。ACM 证书可以保护多个域名和域中的多个名称。您也可以使用 ACM 创建能够保护任意多个子域的通配符 SSL 证书。

ACM 与 AWS Certificate Manager Private Certificate Authority 紧密关联。您可以使用 ACM PCA 创建私有证书颁发机构 (CA)，然后使用 ACM 颁发私有证书。这些是用于在内部识别用户、计算机、应用程序、服务、服务器和其他设备的 SSL/TLS X.509 证书。私有证书不能是公开信任的。有关 ACM PCA 的更多信息，请参阅[AWS Certificate Manager Private Certificate Authority User Guide](#)。使用 ACM 颁发的私有证书非常类似于公有 ACM 证书。它们具有相似的优势和限制。这些优势包括管理与证书关联的私有密钥、续订证书以及使您能够使用控制台部署具有集成服务的私有证书。有关与使用 ACM 关联的限制的更多信息，请参阅[请求私有证书 \(p. 20\)](#)。您还可以使用 ACM 导出私有证书和加密私有密钥，以便在任何位置使用。有关更多信息，请参阅[导出私有证书 \(p. 21\)](#)。有关使用 ACM PCA 作为独立服务颁发私有证书的优势的信息，请参阅[ACM PCA 用户指南](#)中的简介。

Note

您不能直接在网站或应用程序中安装公有 ACM 证书。您必须使用与 ACM 和 ACM PCA 集成的服务之一来安装证书。有关这些服务的更多信息，请参阅[与 AWS Certificate Manager 集成的服务 \(p. 8\)](#)。

主题

- [概念 \(p. 1\)](#)
- [ACM 证书特征 \(p. 6\)](#)
- [支持的区域 \(p. 7\)](#)
- [与 AWS Certificate Manager 集成的服务 \(p. 8\)](#)
- [网站签章和信任徽标 \(p. 9\)](#)
- [限制 \(p. 9\)](#)
- [最佳实践 \(p. 10\)](#)
- [AWS Certificate Manager 定价 \(p. 12\)](#)

概念

此部分介绍与 AWS Certificate Manager (ACM) 相关的基本术语和概念。

主题

- [ACM 证书 \(p. 2\)](#)
- [顶级域 \(p. 3\)](#)
- [非对称密钥加密 \(p. 3\)](#)
- [证书颁发机构 \(p. 3\)](#)
- [证书透明度日志 \(p. 3\)](#)
- [域名系统 \(p. 4\)](#)
- [域名 \(p. 4\)](#)
- [加密和解密 \(p. 5\)](#)
- [完全限定域名 \(FQDN\) \(p. 5\)](#)
- [公有密钥基础设施 \(p. 5\)](#)
- [根证书 \(p. 5\)](#)

- [安全套接字层 \(SSL\) \(p. 5\)](#)
- [安全 HTTPS \(p. 5\)](#)
- [SSL 服务器证书 \(p. 5\)](#)
- [对称密钥加密 \(p. 6\)](#)
- [传输层安全性 \(TLS\) \(p. 6\)](#)
- [信任 \(p. 6\)](#)

ACM 证书

ACM 生成 X.509 版本 3 证书。每个有效期为 13 个月，并且包含以下扩展。

- 基本约束 - 指定主题的证书是否是证书颁发机构 (CA)
- 授权密钥标识符 - 支持识别与用于签署证书的私有密钥对应的公有密钥。
- 主题密钥标识符 - 支持识别包含特定公有密钥的证书。
- 密钥使用 - 定义在证书中嵌入的公有密钥的用途。
- 扩展密钥使用 - 指定除密钥使用扩展指定的用途外可为其使用公有密钥的一个或多个用途。
- CRL 分配点 - 指定可在其中获取 CRL 信息的位置。

```
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
```

顶级域

请参阅 [域名](#) (p. 4)。

非对称密钥加密

非对称加密不同于[对称密钥加密](#) (p. 6)，它使用不同的但在数学上相关的密钥加密和解密内容。密钥之一是有公钥，通常以 X.509 v3 证书形式提供。另一个密钥是私有密钥，以安全方式存储。X.509 证书将用户、计算机或其他资源 (证书主题) 的身份绑定到公钥。

ACM 证书是 X.509 SSL/TLS 证书，它将您网站的身份和组织的信息绑定到证书中包含的公钥。ACM 将关联的私有密钥存储在硬件安全模块 (HSM) 中。

证书颁发机构

证书颁发机构 (CA) 是一个颁发数字证书的实体。商业上，最常见的数字证书类型基于 ISO X.509 标准。CA 颁发已签名的数字证书，用于确认证书使用者的身份并将该身份绑定到证书中包含的公钥。CA 通常还会管理证书吊销。

证书透明度日志

为了防止错误地颁发或由损坏的 CA 颁发的 SSL/TLS 证书，某些浏览器要求为您的域颁发的公有证书记录在证书透明度日志中。域名将被记录。私有密钥不会被记录。未记录的证书通常会在浏览器中生成错误。

您可以监控日志，以确保只为您的域颁发您已授权的证书。您可以使用[证书搜索](#)等服务来检查日志。

在 Amazon CA 为您的域颁发公开信任的 SSL/TLS 证书之前，它会将证书提交到至少两个证书透明度日志服务器。这些服务器将证书添加到其公有数据库中，并将已签名的证书时间戳 (SCT) 返回到 Amazon CA。然后，CA 会将 SCT 嵌入到证书中，对证书进行签名，并将其颁发给您。这些时间戳包括在其他 X.509 扩展中。

```
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
```

证书透明度日志记录是在您请求或续订证书时自动进行的，除非您选择退出。有关选择退出的更多信息，请参阅[选择退出证书透明度日志记录 \(p. 11\)](#)。

域名系统

域名系统 (DNS) 是连接到 Internet 或私有网络的计算机及其他资源的分层分布式命名系统。DNS 主要用于将文本域名 (如 `aws.amazon.com`) 转换为数字 IP (Internet 协议) 地址 (形如 `111.222.333.444`)。不过，域的 DNS 数据库包含大量其他用途的记录。例如，通过 ACM，您可以使用 CNAME 记录在请求证书时验证自己拥有或可以控制某个域。有关更多信息，请参阅[使用 DNS 验证域所有权 \(p. 22\)](#)。

域名

域名是一个文本字符串 (例如 `www.example.com`)，可通过域名系统 (DNS) 转换为 IP 地址。计算机网络 (包括互联网) 使用 IP 地址而不是文本名称。域名由以句点分隔的不同标签组成：

TLD

最右边的标签称作顶级域 (TLD)。常见示例有 `.com`、`.net`、`.edu`。在某些国家或地区注册的实体的 TLD 为国家或地区名称的缩写，这称作国家/地区代码。示例包括 `.uk` (英国)、`.ru` (俄国)、`.fr` (法国)。使用国家/地区代码时，通常引入 TLD 的二级层次结构来标识注册实体的类型。例如，`.co.uk` TLD 标识英国的商业企业。

顶级域

顶级域名包括顶级域并在其上扩展。对于包含国家/地区代码的域名，顶级域包含代码和标签 (如果有)，用于标识注册实体的类型。顶级域不包含子域 (请参阅以下段落)。在 `www.example.com` 中，顶级域的名称为 `example.com`。在 `www.example.co.uk` 中，顶级域的名称为 `example.co.uk`。经常用来代替顶级 (apex) 的其他名称包括 `base`、`bare`、`root`、`root apex`、`zone apex` 等。

子域

子域名位于顶级域名之前，使用句点与顶级域名及其他域名分隔。最常见的子域名是 `www`，但允许使用任意名称。此外，子域名可以有多个级别。例如，在 `jake.dog.animals.example.com` 中，子域依次为 `jake`、`dog` 和 `animals`。

FQDN

完全限定域名 (FQDN) 是适用于已连接到网络或 Internet 的计算机、网站或其他资源的完整 DNS 名称。例如，`aws.amazon.com` 是 Amazon Web Services 的 FQDN。FQDN 包括一直到顶级域的所有域。例

如, `[subdomain1].[subdomain2]...[subdomainn].[apex domain].[top-level domain]` 代表了 FQDN 的一般格式。

PQDN

未完全限定的域名称作部分限定域名 (PQDN), 含义不明确。像 `[subdomain1.subdomain2.]` 这样的名称就是 PQDN, 这是因为无法确定根域。

注册

使用域名的权利由域名注册机构指派。注册机构通常由互联网名称和数字地址分配机构 (ICANN) 认证。此外, 还有一些称作注册管理机构的组织负责维护 TLD 数据库。当您申请域名时, 注册机构将您的信息发送给相应的 TLD 注册管理机构。注册管理机构分配域名、更新 TLD 数据库并将您的信息发布到 WHOIS。域名通常以购买方式获取。

加密和解密

加密是提供数据机密性的过程。解密将反转此过程并恢复原始数据。未加密的数据通常称为“明文”, 无论它是否为文本。加密的数据通常称为“密文”。客户端与服务器之间的消息的 HTTPS 加密使用算法和密钥。算法定义将纯文本数据转换为密文 (加密) 以及将密文转换回原始明文 (解密) 的分步过程。在加密或解密过程中, 算法将使用密钥。密钥可以是私有密钥或公有密钥。

完全限定域名 (FQDN)

请参阅 [域名 \(p. 4\)](#)。

公有密钥基础设施

公有密钥基础设施 (PKI) 由创建、颁发、管理、分发、使用、存储和撤销数字证书所需的硬件、软件、人员、策略、文档和过程组成。PKI 可推动信息在计算机网络中的安全传输。

根证书

证书颁发机构 (CA) 通常位于一个包含多个其他 CA (这些 CA 之间明确定义了父子关系) 的层次结构中。子或从属 CA 由其父 CA 认证, 这将创建证书链。位于层次结构顶部的 CA 称为“根 CA”, 而其证书称为“根证书”。此证书通常是自签名的。

安全套接字层 (SSL)

安全套接字层 (SSL) 和传输层安全性 (TLS) 是通过计算机网络提供通信安全性的加密协议。TLS 是 SSL 的后继者。二者都使用 X.509 证书对服务器进行身份验证。这两个协议都对客户端与服务器之间用于加密这两个实体之间传输的数据的对称密钥进行协商。

安全 HTTPS

HTTPS 表示 HTTP over SSL/TLS, 一个所有主要浏览器和服务器都支持的安全形式的 HTTP。所有 HTTP 请求和响应在跨网络发送之前都将进行加密。HTTPS 结合了 HTTP 协议与基于对称、非对称和 X.509 证书的加密技术。HTTPS 的工作方式是, 将加密安全层插入开放系统互连 (OSI) 模型中的 HTTP 应用程序层下方和 TCP 传输层上方。安全层使用安全套接字层 (SSL) 协议或传输层安全性 (TLS) 协议。

SSL 服务器证书

HTTPS 事务需要服务器证书来对服务器进行身份验证。服务器证书是 X.509 v3 数据结构, 用于将证书中的公有密钥绑定到证书的使用者。SSL/TLS 证书由证书颁发机构 (CA) 签署并且包含服务器的名称、有效期限、公有密钥、签名算法等。

对称密钥加密

对称密钥加密使用同一密钥来加密和解密数字数据。另请参阅 [非对称密钥加密 \(p. 3\)](#)。

传输层安全性 (TLS)

请参阅 [安全套接字层 \(SSL\) \(p. 5\)](#)。

信任

要让 Web 浏览器信任网站的身份，该浏览器必须能够验证网站的证书。不过，浏览器仅信任称为“CA 根证书”的少量证书。称为证书颁发机构 (CA) 的可信第三方将验证该网站的身份并向网站运营商颁发签名的数字证书。随后，浏览器可以检查数字签名以验证网站的身份。如果验证成功，浏览器会在地址栏中显示一个锁定图标。

ACM 证书特征

ACM 提供的证书具有这一节中所述的一些特征。

Note

这些特征仅适用于 ACM 提供的证书。它们可能不适用于 [您导入到 ACM 中的证书 \(p. 41\)](#)。

域验证 (DV)

ACM 证书是进行域验证的。也就是说，ACM 证书的主题字段仅标识域名。请求 ACM 证书时，您必须验证自己拥有或可以控制请求中指定的所有域。您可以通过使用电子邮件或 DNS 来验证所有权。有关更多信息，请参阅 [使用电子邮件验证域所有权 \(p. 25\)](#) 和 [使用 DNS 验证域所有权 \(p. 22\)](#)。

有效期

目前，ACM 证书的有效期为 13 个月。

托管续订和部署

ACM 管理续订 ACM 证书以及续订之后预置证书的过程。自动续订可以帮助您避免因证书配置错误、撤销或到期而导致的停机。有关更多信息，请参阅 [适用于 ACM 的由 Amazon 颁发的证书的托管续订 \(p. 35\)](#)。

浏览器和应用程序信任

包括 Google Chrome、Microsoft Internet Explorer 和 Microsoft Edge、Mozilla Firefox 和 Apple Safari 在内的所有主要浏览器均信任 ACM 证书。通过 SSL/TLS 连接到使用 ACM 证书的站点时，信任 ACM 证书的浏览器会在其状态栏或地址栏中显示一个挂锁图标。Java 也信任 ACM 证书。

多个域名

每个 ACM 证书都必须至少包括一个完全限定域名 (FQDN)，并且您可以在需要时添加更多名称。例如，当您为 `www.example.com` 创建 ACM 证书时，您也可以添加名称 `www.example.net`，只要客户可以使用这两个名称之一访问您的站点即可。在空域方面 (也称为顶级域或裸域)，情况同样如此。也就是说，您可以为 `www.example.com` 请求 ACM 证书并添加名称 `example.com`。有关更多信息，请参阅 [请求公有证书 \(p. 18\)](#)。

通配符名称

ACM 允许您在域名中使用星号 (*) 来创建包含通配符名称的 ACM 证书，该证书可以保护相同域中的多个站点。例如，`*.example.com` 可以保护 `www.example.com` 和 `images.example.com`。

Note

请求通配符证书时，星号 (*) 必须位于域名的最左侧位置，而且只能保护一个子域级别。例如，`*.example.com` 可保护 `login.example.com` 和 `test.example.com`，但无法保护 `test.login.example.com`。另请注意，`*.example.com` 仅保护 `example.com` 的子域，但不保护裸域或顶点域 (`example.com`)。但是，您可以通过在请求中指定多个域名来请求可保护空域或顶点域及其子域的证书。例如，您可以请求用于保护 `example.com` 和 `*.example.com` 的证书。

算法

证书必须指定算法和密钥大小。目前，ACM 支持以下公有密钥算法：

- 1024 位 RSA (RSA_1024)
- 2048 位 RSA (RSA_2048)
- 4096 位 RSA (RSA_4096)
- Elliptic Prime Curve 256 位 (EC_prime256v1)
- Elliptic Prime Curve 384 位 (EC_secp384r1)
- Elliptic Prime Curve 521 位 (EC_secp521r1)

Important

请注意，**集成服务** 仅允许将其支持的算法和密钥大小与其资源关联。此外，这种支持因证书是否导入到 IAM 或 ACM 而有所差别。有关更多信息，请参阅每个服务的文档。

- 对于 Elastic Load Balancing，请参阅 [Application Load Balancer 的 HTTPS 侦听器](#)。
- 对于 CloudFront，请参阅 [支持的 SSL/TLS 协议和密码](#)。

例外

请注意以下几点：

- ACM 不提供扩展验证 (EV) 证书或组织验证 (OV) 证书。
- ACM 不为 SSL/TLS 协议以外的任何其他协议提供证书。
- 您不能使用 ACM 证书进行电子邮件加密。
- ACM 仅允许在域名中使用 UTF-8 编码的 ASCII 字符，包括包含“xn--”的标签 (Punycode)。ACM 不接受在域名中使用 Unicode 输入 (u 型标签)。
- 对于 ACM 证书，ACM 目前不允许您退出 [托管证书续订 \(p. 35\)](#)。此外，托管续订不适用于您导入到 ACM 中的证书。
- 您无法为 Amazon 拥有的域名 (例如以 `amazonaws.com`、`cloudfront.net` 或 `elasticbeanstalk.com` 结尾的域名) 请求证书。
- 您无法为 ACM 证书下载私有密钥。
- 您不能直接在 Amazon Elastic Compute Cloud (Amazon EC2) 网站或应用程序中安装 ACM 证书。但是，您可以将自己的证书用于任何集成服务。有关更多信息，请参阅 [与 AWS Certificate Manager 集成的服务 \(p. 8\)](#)。

支持的区域

访问 AWS General Reference 中的 [AWS 区域和终端节点](#) 或 [AWS 区域表](#) 以查看 ACM 的区域可用性。

与大多数 AWS 资源相同，ACM 中的证书是区域性资源。若要在 Elastic Load Balancing 中针对多个 AWS 区域中相同的完全限定域名 (FQDN) 或 FQDN 组使用证书，您必须为每个区域请求或导入一个证书。对于 ACM 提供的证书，这意味着您必须重新验证每个区域的证书中的每个域名。您不能在各区域之间复制证书。

若要将 ACM 证书用于 Amazon CloudFront，您必须在美国东部（弗吉尼亚北部）区域中请求或导入证书。此区域中与某个 CloudFront 分配关联的 ACM 证书将会分配到为该分配配置的所有地理位置。

与 AWS Certificate Manager 集成的服务

AWS Certificate Manager 支持的 AWS 服务越来越多。您不能直接在基于 AWS 的网站或应用程序上安装 ACM 证书或私有 ACM PCA 证书。您必须使用以下服务之一。

Elastic Load Balancing

Elastic Load Balancing 在多个 Amazon EC2 实例间自动分配您的应用程序的传入流量。它会检测运行不正常的实例，并将流量重新路由到运行正常的实例，直至运行不正常的实例恢复为止。Elastic Load Balancing 自动扩展其请求处理容量以应对传入流量。有关负载均衡的更多信息，请参阅 [Elastic Load Balancing 用户指南](#)。

通常，为了通过 SSL/TLS 提供安全内容，负载均衡器会要求在负载均衡器上或后端 Amazon EC2 实例上安装 SSL/TLS 证书。ACM 将与 Elastic Load Balancing 集成，以在负载均衡器上部署 ACM 证书。有关更多信息，请参阅 [创建 Application Load Balancer](#)。

Amazon CloudFront

Amazon CloudFront 是一项 Web 服务，可通过从全球边缘站点网络传输您的内容来加快将动态和静态 Web 内容分配给最终用户的速度。最终用户请求您正在通过 CloudFront 处理的内容时，将路由到提供最低延迟的边缘站点。这样可以确保尽可能以最佳性能传输内容。如果内容目前在边缘站点上，CloudFront 将立即传送它。如果内容目前不在边缘站点上，CloudFront 将从您已确定为明确的内容源的 Amazon S3 存储桶或 Web 服务器中检索内容。有关 CloudFront 的更多信息，请参阅 [Amazon CloudFront 开发人员指南](#)。

为了通过 SSL/TLS 提供安全内容，CloudFront 会要求在 CloudFront 分发或后端内容源上安装 SSL/TLS 证书。ACM 将与 CloudFront 集成，以在 CloudFront 分发上部署 ACM 证书。有关更多信息，请参阅 [获取 SSL/TLS 证书](#)。

Note

若要将 ACM 证书与 CloudFront 搭配使用，您必须在美国东部（弗吉尼亚北部）区域中请求或导入证书。

AWS Elastic Beanstalk

Elastic Beanstalk 可帮助您在 AWS 云中部署和管理应用程序，而无需为运行这些应用程序的基础设施操心。AWS Elastic Beanstalk 可减少管理复杂性。您只需上传您的应用程序，Elastic Beanstalk 将会自动处理有关容量预置、负载均衡、扩展和运行状况监控的细节。Elastic Beanstalk 使用 Elastic Load Balancing 服务创建负载均衡器。有关 Elastic Beanstalk 的更多信息，请参阅 [AWS Elastic Beanstalk 开发人员指南](#)。

若要选择证书，您必须在 Elastic Beanstalk 控制台为您的应用程序配置负载均衡器。有关更多信息，请参阅 [配置 Elastic Beanstalk 环境的负载均衡器以终止 HTTPS](#)。

Amazon API Gateway

随着移动设备的普及和物联网 (IoT) 的发展，创建可用于访问数据并与 AWS 上的后端系统交互的 API 变得日益普遍。您可以使用 API 网关发布、维护、监控和保护您的 API。将 API 部署到 API 网关后，您可以 [设置自定义域名](#) 简化对此 API 的访问。要设置自定义域名，您必须提供 SSL/TLS 证书。您可以使用 ACM 生成或导入证书。

AWS CloudFormation

AWS CloudFormation 可帮助您对 Amazon Web Services 资源进行建模和设置。您可创建一个模板来描述您要使用的 AWS 资源，如 Elastic Load Balancing 或 API 网关。然后，AWS CloudFormation 将负责为您预配置和配置这些资源。您无需单独创建和配置 AWS 资源并了解 AWS CloudFormation 处理所有这些工作时所依赖的内容。ACM 证书作为模板资源包含在内，这意味着 AWS CloudFormation 可以请求 ACM 证书，您可以在 AWS 服务中使用这些证书来启用安全连接。有关更多信息，请参阅

[AWS::CertificateManager::Certificate](#)。此外，ACM 证书与许多您可以利用 AWS CloudFormation 设置的 AWS 资源一起包含在内。

Note

如果您使用 AWS CloudFormation 创建 ACM 证书，AWS CloudFormation 堆栈会保留在 CREATE_IN_PROGRESS 状态。任何进一步的堆栈操作将被延迟，直到您按照证书验证电子邮件中的说明操作为止。有关更多信息，请参阅[资源在创建、更新或删除堆栈操作期间无法稳定工作](#)。

网站签章和信任徽标

Amazon 不提供网站签章，也不允许其商标用作以下用途之一：

- AWS Certificate Manager (ACM) 不提供您可用于自己网站的安全网站签章。如果您希望使用网站签章，可以从第三方供应商处获得。我们建议您选择一家供应商来评估和确定您的网站或业务实践的安全性。
- Amazon 不允许将其商标或徽标用作证书徽章、网站签章或信任徽标。这种类型的网站签章和徽章会被复制到不使用 ACM 服务的网站，并且会被不正当地用于通过虚假借口建立信任。为了保护我们的客户和 Amazon 声誉，我们不允许以这种方式使用我们的商标和徽标。

限制

以下 AWS Certificate Manager (ACM) 限制适用于每个 AWS 区域和每个 AWS 账户。若要请求更高的限制，请在 [AWS Support 中心](#) 创建一个案例。新 AWS 账户在开始时的限制可能会低于此处描述的那些限制。

项目	默认限制
ACM 证书数量	100
每年的 ACM 证书的数量 (最近 365 天)	您的账户限额的两倍
已导入证书的数量	100
每年导入的证书的数量 (最近 365 天)	您的账户限额的两倍
每个 ACM 证书的域名数量	10
私有 CA 的数量	10
每个 CA 的私有证书数量	50000

主题

- [每年的 ACM 证书的数量 \(最近 365 天\)](#) (p. 9)
- [每个 ACM 证书的域名数量](#) (p. 10)
- [私有 CA 和证书的数量](#) (p. 10)

每年的 ACM 证书的数量 (最近 365 天)

您最多可以请求将您每年的 ACM 证书限制提升一倍。例如，如果您的限制为 25，则一年最多可以请求 50 份 ACM 证书。如果您请求 50 份证书，则必须在当年删除 25 份证书才能避免超出限制。如果需要的证书超过 25 份，在本示例中，您必须联系 AWS Support 中心。

Note

尽管上表指示一个账户最多可以拥有 100 份 ACM 证书，但新的 AWS 账户在一开始的限制可能会更低。

每个 ACM 证书的域名数量

每份 ACM 证书的域名数默认限额为 10 个。您的限制可能更高。您提交的第一个域名作为证书的主题公用名 (CN) 包含在内。所有名称都包含在主题替代名称扩展中。

您最多可以请求 100 个域名。要请求提高限制，请在 [AWS Support 中心](#) 创建一个案例。但在创建案例之前，请务必了解在使用电子邮件验证的情况下，添加更多域名会给您带来更多的管理工作。有关更多信息，请参阅 [域验证 \(p. 11\)](#)。

Note

每个 ACM 证书的域名数限制仅适用于 ACM 提供的证书。此限制并不适用于您导入到 ACM 中的证书。下面几节仅适用于 ACM 证书。

私有 CA 和证书的数量

ACM 与 ACM PCA 集成。您可以使用 ACM 控制台、AWS CLI 或 ACM API 从现有私有证书颁发机构 (CA) 请求私有证书。证书是在 ACM 环境中管理的，具有和 ACM 颁发的公有证书相同的限制。有关更多信息，请参阅 [请求私有证书 \(p. 20\)](#)。您还可以使用独立 ACM PCA 服务颁发私有证书。有关更多信息，请参阅 [颁发私有证书](#)。您可以为每个创建 10 个私有 CA 和 50,000 个私有证书。

最佳实践

最佳实践是一些建议，可帮助您更有效地使用 AWS Certificate Manager (AWS Certificate Manager)。以下最佳实践基于来自当前 ACM 客户的实际经验。

主题

- [AWS CloudFormation \(p. 10\)](#)
- [证书固定 \(p. 11\)](#)
- [域验证 \(p. 11\)](#)
- [添加或删除域名 \(p. 11\)](#)
- [选择退出证书透明度日志记录 \(p. 11\)](#)
- [启用 AWS CloudTrail \(p. 12\)](#)

AWS CloudFormation

利用 AWS CloudFormation，您可以创建一个模板来描述要使用的 AWS 资源。随后，AWS CloudFormation 将为您预配置和配置这些资源。AWS CloudFormation 可以为您预配置 ACM 所支持的资源，例如 Elastic Load Balancing、Amazon CloudFront 和 Amazon API Gateway。有关更多信息，请参阅 [与 AWS Certificate Manager 集成的服务 \(p. 8\)](#)。

如果您使用 AWS CloudFormation 快速创建和删除多个测试环境，建议您不要为每个环境创建单独的 ACM 证书。这样做会快速耗尽您的证书限制。有关更多信息，请参阅 [限制 \(p. 9\)](#)。相反，创建一个涵盖了您用于测试的所有域名的通配符证书。例如，如果您针对仅版本号发生变化的域名重复创建 ACM 证书 (如 `<version>.service.example.com`)，则改为针对 `<*>.service.example.com` 创建单个通配符证书。在 AWS CloudFormation 用来创建测试环境的模板中包含通配符证书。

证书固定

证书固定 (有时称作 SSL 固定) 是一个过程, 可在应用程序中使用此过程来验证远程主机, 方式是将该主机直接与其 X.509 证书或公有密钥而非证书层次结构关联。因此, 应用程序使用固定来绕过 SSL/TLS 证书链验证。典型的 SSL 验证过程将检查证书链 (从根证书颁发机构 (CA) 证书到从属 CA 证书 (如果有)) 中的签名。此外, 它还检查层次结构底部远程主机的证书。您的应用程序可改为固定到远程主机的证书以指示仅该证书 (而非根证书或链中的任何其他证书) 受信任。在应用程序开发过程中, 您可以将远程主机的证书或公有密钥添加到应用程序。或者, 应用程序也可以在首次连接到主机时添加证书或密钥。

Warning

建议您的应用程序不固定 ACM 证书。ACM 会执行[适用于 ACM 的由 Amazon 颁发的证书的托管续订 \(p. 35\)](#)以在 Amazon 颁发的 SSL/TLS 证书过期前进行续订。为了续订证书, ACM 会生成新的公有-私有密钥对。如果您的应用程序固定 ACM 证书, 并且已使用新的公有密钥成功续订证书, 则应用程序可能无法连接到您的域。

如果您决定固定证书, 则以下选项将不会阻止您的应用程序连接到您的域:

- 导入您自己的证书到 ACM, 然后将您的应用程序固定到导入的证书。ACM 不会尝试自动续订导入的证书。
- 将您的应用程序固定到 [Amazon 根证书](#)。

域验证

AWS Certificate Manager (ACM) 必须先确认您拥有或可以控制请求中指定的所有域, 然后 Amazon 证书颁发机构 (CA) 才能为网站颁发证书。您可以使用电子邮件或 DNS 执行验证。有关更多信息, 请参阅[使用电子邮件验证域所有权 \(p. 22\)](#)和[使用电子邮件验证域所有权 \(p. 22\)](#)。

添加或删除域名

您无法在现有 ACM 证书中添加或删除域名称, 而必须请求包含修订过的域名列表的新证书。例如, 如果证书有五个域名, 并且需要添加四个域名, 则必须请求包含九个域名的新证书。与任何新证书一样, 您必须对请求中的所有域名验证所有权, 包括之前为原始证书验证过的域名。

如果使用电子邮件验证, 则对于每个域, 您最多将收到 8 封验证电子邮件, 并且您必须在 72 小时之内至少根据其中 1 封邮件执行操作。例如, 如果请求包含五个域名的证书, 您最多将收到 40 封验证电子邮件, 并且您必须在 72 小时之内至少根据其中 5 封执行操作。随着证书请求中域名数量的增加, 使用电子邮件来验证域所有权所需的工作量也会增加。

如果使用 DNS 验证, 则必须为需要验证的 FQDN 向数据库写入一条新 DNS 记录。ACM 会向您发送要创建的记录, 并在稍后查询数据库以确定是否已添加该记录。添加该记录即声明您拥有或可以控制该域。在前面的示例中, 如果请求包含五个域名的证书, 则必须创建五条 DNS 记录。建议您尽量使用 DNS 验证。

选择退出证书透明度日志记录

Important

无论您采取何种操作退出证书透明度日志记录, 您的证书都可能仍被任何有权访问您将证书绑定到的公共或私有终端节点的客户端或个人所记录。不过, 证书将不会包含已签名的证书时间戳 (SCT)。只有发布证书的 CA 才能将 SCT 嵌入到证书中。

从 2018 年 4 月 30 日开始, Google Chrome 将停止信任未在证书透明度日志中记录的公有 SSL/TLS 证书。因此, 从 2018 年 4 月 24 日起, Amazon CA 将开始在至少两个公有日志中发布所有新证书和续订。证书一旦记录, 便无法删除。有关更多信息, 请参阅[证书透明度日志 \(p. 3\)](#)。

当您请求证书或续订证书时，会自动执行日志记录，但您可以选择退出。这样做的常见原因包括对安全和隐私的疑虑。例如，记录内部主机域名会向潜在的攻击者提供有关内部网络的信息，否则将不会公开。此外，日志记录还可能会泄露新的或未发布的产品和网站的名称。

要在请求证书时选择退出透明度日志记录，请使用 [request-certificate](#) AWS CLI 命令的 Options 参数或 [RequestCertificate](#) API。

如果您的证书是在 2018 年 4 月 24 日之前颁发的，并且您希望确保在续订过程中不记录它，则可以调用 `update-certificate-options` 命令或 [UpdateCertificateOptions](#) API 以选择退出。

证书一旦记录，便无法从日志中删除。在该时间点选择退出将不起作用。如果您在请求证书时选择退出日志记录，然后在稍后再选择回来，则您的证书将不会被记录，直到续订它为止。如果您希望证书被立即记录，我们建议您发布一个新的证书。

Note

目前，您不能使用控制台选择退出或加入透明度日志记录。

以下示例向您展示了在请求新的证书时如何使用 [request-certificate](#) 命令禁用证书透明度。

```
aws acm request-certificate \  
--domain-name www.example.com \  
--validation-method DNS \  
--options CertificateTransparencyLoggingPreference=DISABLED \  
--idempotency-token 184627
```

上述命令输出新证书的 ARN。

```
{  
  "CertificateArn":  
    "arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012"  
}
```

如果您已有一个证书，并且您不希望在续订它时记录它，请使用 [update-certificate-options](#) 命令。此命令不返回。值。

```
aws acm update-certificate-options \  
--certificate-arn arn:aws:acm:region:account:\  
certificate/12345678-1234-1234-1234-123456789012 \  
--options CertificateTransparencyLoggingPreference=DISABLED
```

启用 AWS CloudTrail

在开始使用 ACM 之前，请先启用 CloudTrail 日志记录。利用 CloudTrail，您可以通过检索账户的 AWS API 调用历史记录来监控您的 AWS 部署，包括通过 AWS 管理控制台、AWS 开发工具包、AWS Command Line Interface 以及更高级的 AWS 服务执行的 API 调用。您还可以确定哪些用户和账户调用了 ACM API、发出调用的源 IP 地址以及发生调用的时间。您可将 CloudTrail 集成到使用 API 的应用程序、为您的组织自动创建跟踪、检查跟踪的状态和控制管理员启用和关闭 CloudTrail 日志记录的方式。有关更多信息，请参阅[创建跟踪](#)。转到 [使用 AWS CloudTrail \(p. 57\)](#) 以查看 ACM 操作的示例跟踪。

AWS Certificate Manager 定价

AWS 不会针对您使用 AWS Certificate Manager 管理的 SSL/TLS 证书向您收取费用。您只需为您创建的用于运行网站或应用程序的 AWS 资源付费。有关最新的 ACM 定价信息，请参阅 AWS 网站上的 [AWS Certificate Manager 服务定价](#) 页面。

设置

借助 AWS Certificate Manager (ACM)，您可以为基于 AWS 的网站和应用程序预配置和管理 SSL/TLS 证书。您可以使用 ACM 创建或导入证书，然后加以管理。您必须使用其他 AWS 服务将证书部署到您的网站或应用程序。有关与 ACM 集成的服务的更多信息，请参阅[与 AWS Certificate Manager 集成的服务 \(p. 8\)](#)。以下主题将讨论在使用 ACM 之前需要执行的步骤。

Note

除了使用 ACM 提供的证书，您还可以将证书导入到 ACM 中。有关更多信息，请参阅[导入证书 \(p. 41\)](#)。

主题

- [设置 AWS 和 IAM \(p. 13\)](#)
- [注册域名 \(p. 14\)](#)
- [设置您的网站或应用程序 \(p. 14\)](#)
- [\(可选\) 为域配置电子邮件 \(p. 15\)](#)
- [\(可选\) 配置 CAA 记录 \(p. 16\)](#)

设置 AWS 和 IAM

您必须先注册 Amazon Web Services，然后才能使用 ACM。最佳实践是创建 IAM 用户以限制您的用户可执行的操作。

注册 AWS

如果您还不是 Amazon Web Services (AWS) 客户，则必须注册才能使用 ACM。您的账户会自动注册所有可用服务，但您只需为您使用的服务付费。此外，如果您是 AWS 新客户，还可以免费试用。有关更多信息，请参阅[AWS 免费套餐](#)。

如何注册 AWS 账户

1. 转至 <https://aws.amazon.com/> 并选择 Sign Up。
2. 按照屏幕上的说明进行操作。

Note

注册过程包含接收自动电话呼叫和在电话小键盘上输入提供的 PIN。您还必须提供信用卡号，即使您注册的是免费套餐。

创建 IAM 用户

所有 AWS 账户都具有根用户凭证 (即账户所有者凭证)。这些凭证允许完全访问账户中的所有资源。由于无法限制根用户凭证的权限，建议删除根用户访问密钥。然后创建 AWS Identity and Access Management (IAM) 用户凭证来执行与 AWS 的日常交互工作。有关更多信息，请参阅 IAM 用户指南中的[隐藏您的 AWS 账户 \(根\) 访问密钥](#)。

Note

您可能需要 AWS 账户根用户访问权限才能执行特定任务，如更改 AWS 支持计划或关闭账户。在这些情况下，请使用您的电子邮件地址和密码登录 AWS 管理控制台。请参阅[电子邮件和密码 \(根用户\)](#)。

有关需要 根用户 访问权限的任务的列表，请参阅[需要 AWS 账户根用户的 AWS 任务](#)。

通过 IAM，您可以安全地控制用户对 AWS 账户中 AWS 服务和资源的访问。例如，如果需要管理员级权限，您可以创建 IAM 用户，为该用户授予完全访问权限，然后使用这些凭证与 AWS 交互。如果需要修改或撤销权限，您可以删除或修改与该 IAM 用户相关联的策略。

如果多个用户需要访问您的 AWS 账户，您可以为每个用户创建唯一的凭证并定义哪些用户有权访问哪些资源。您不必共享凭证。例如，您可以创建对 AWS 账户中的资源具有只读访问权限的 IAM 用户，并将这些凭证分配给您的用户。

ACM 还提供两个 [AWS 托管策略](#) 供您使用：

- `AWSCertificateManagerFullAccess`
- `AWSCertificateManagerReadOnly`

Note

与 IAM 用户关联的任何活动或成本均将计入 AWS 账户所有者。

注册域名

完全限定域名 (FQDN) 是 Internet 上的组织或个人的唯一名称并在后面跟有一个顶级域扩展名，例如 .com 或 .org。如果您还没有注册域名，则可以通过 Amazon Route 53 或许多其他商业注册商注册一个域名。通常，您可以转到注册商的网站，请求一个域名。注册商将查询 WHOIS 以确定请求的 FQDN 是否可用。如果可用，注册商通常会列出域扩展名不同的相关名称，并向您提供获取任何可用名称的机会。注册通常会持续一个设定的时间段，例如一年或两年，在这之后必须对其进行续订。

有关使用 Amazon Route 53 注册域名的更多信息，请参阅 Amazon Route 53 开发人员指南 中的[使用 Amazon Route 53 注册域名](#)。

设置您的网站或应用程序

您可以在 Amazon EC2 Linux 或 Windows 实例上安装网站。有关 Linux Amazon EC2 实例的更多信息，请参阅[适用于 Linux 的 Amazon Elastic Compute Cloud 用户指南](#)。有关 Windows Amazon EC2 实例的更多信息，请参阅[适用于 Microsoft Windows 的 Amazon Elastic Compute Cloud 用户指南](#)。

虽然您在 Amazon EC2 实例上安装网站，但无法直接在该实例上部署 ACM 证书。您必须改用与 ACM 集成的服务之一来部署证书。有关更多信息，请参阅[与 AWS Certificate Manager 集成的服务 \(p. 8\)](#)。

若要在 Windows 或 Linux 上快速启动并运行您的网站，请参阅以下主题。

主题

- [Linux 快速入门 \(p. 14\)](#)
- [Windows 快速入门 \(p. 15\)](#)

Linux 快速入门

若要在 Linux 实例上创建网站或应用程序，您可以选择一个 Linux Amazon 系统映像 (AMI) 并在其上安装 Apache Web 服务器。有关更多信息，请参阅 Amazon EC2 用户指南（适用于 Linux 实例）中的[教程：在 Amazon Linux 上安装 LAMP Web 服务器](#)。

Windows 快速入门

若要获取可以在其上安装网站或应用程序的 Microsoft Windows Server，请选择随 Microsoft Internet Information Services (IIS) Web 服务器捆绑的 Windows Server AMI。然后使用默认网站或创建新网站。您还可以在您的 Amazon EC2 实例上安装 WIMP 服务器。有关更多信息，请参阅 Amazon EC2 用户指南（适用于 Windows 实例）中的[教程：在运行 Windows Server 的 Amazon EC2 实例上安装 WIMP 服务器](#)。

(可选) 为域配置电子邮件

Note

仅当您使用电子邮件验证来声明您拥有或可以控制证书请求中指定的 FQDN (完全限定域名) 时，才需要执行以下步骤。ACM 在颁发证书之前要求您验证所有权或控制权。您可以使用电子邮件验证或 DNS 验证。有关电子邮件验证的更多信息，请参阅[使用电子邮件验证域所有权 \(p. 25\)](#)。

如果您可以编辑 DNS 配置，建议使用 DNS 域验证而不是电子邮件验证。如果使用 DNS 验证，则无需为域名配置电子邮件。有关 DNS 验证的更多信息，请参阅[使用 DNS 验证域所有权 \(p. 22\)](#)。

使用您的注册商网站将您的联系人地址与域名关联。注册商会将联系人电子邮件地址添加到 WHOIS 数据库中，并将一个或多个邮件服务器添加到 DNS 服务器的邮件交换器 (MX) 记录中。如果选择电子邮件验证，ACM 会将电子邮件发送到联系人地址以及由 MX 记录构成的五个常用管理地址。您每次创建新证书、续订证书或请求新的验证邮件时，ACM 都最多发送八封验证电子邮件。验证电子邮件包含用于确认域所有者或指定代表批准 ACM 证书的说明。有关更多信息，请参阅[使用电子邮件验证域所有权 \(p. 25\)](#)。如果您对验证电子邮件有疑问，请参阅[排查电子邮件问题 \(p. 90\)](#)。

WHOIS 数据库

WHOIS 数据库包含您的域的联系人信息。为了验证您的身份，ACM 会发送一封电子邮件到 WHOIS 中的以下三个地址。您必须确保您的联系人信息是公开的或发送到模糊地址的电子邮件将被转发到您真实的电子邮件地址。

- 域注册者
- 技术联系人
- 管理联系人

MX 记录

当您注册域时，您的注册商会将您的邮件交换器 (MX) 记录发送到域名系统 (DNS) 服务器。MX 记录指示哪些服务器接受您的域的邮件。该记录包含完全限定域名 (FQDN)。您可以为顶点域或子域请求证书。

例如，如果您使用控制台为 `abc.xyz.example.com` 请求证书，ACM 将首先尝试查找该子域的 MX 记录。如果无法找到该记录，ACM 将针对 `xyz.example.com` 执行 MX 查找。如果无法找到该记录，ACM 将针对 `example.com` 执行 MX 查找。如果无法找到该记录，或者没有 MX 记录，ACM 将选择为其请求证书的原始域 (本例中为 `abc.xyz.example.com`)。然后，ACM 向以下五个常用系统管理地址域或子域发送电子邮件：

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

如果您使用的是 [RequestCertificate](#) API 操作或 [request-certificate](#) AWS CLI 命令，则 AWS 不会执行 MX 查找。[RequestCertificate](#) 允许您指定域名和验证域的名称。如果您指定可选的 `ValidationDomain` 参数，AWS 会将前述五封电子邮件发送到该域而不是您的域。

无论您使用的是控制台、API 还是 AWS CLI，ACM 始终将验证电子邮件发送到上面列出的五个常用地址。但是，只有在您使用控制台请求证书时，AWS 才会执行 MX 查找。

如果未收到验证电子邮件，请参阅[未收到验证电子邮件 \(p. 91\)](#)了解有关可能原因及解决方法的信息。

(可选) 配置 CAA 记录

您可以选择配置认证机构授权 (CAA) DNS 记录，以指定允许 AWS Certificate Manager (ACM) 为您的域或子域颁发证书。验证您的域之后，ACM 会检查是否存在 CAA 记录以确保它可以为您颁发证书。如果您不希望启用 CAA 检查，则可以选择不为您的域配置 CAA 记录或将记录留空。CAA 记录包含以下数据字段：

`flags`

指定 ACM 是否支持 `tag` 字段的值。将此值设置为 0。

`tags`

`tag` 字段可以为以下值之一。请注意，`iodef` 字段目前已被忽略。

`issue`

指示您在 `value` 字段中指定的 ACM CA 已被授权为您的域或子域颁发证书。

`issuewild`

指示您在 `value` 字段中指定的 ACM CA 已被授权为您的域或子域颁发通配符证书。通配符证书适用于该域或子域及其所有子域。

`value`

此字段的值取决于 `tag` 字段的值。您必须用引号 ("") 将此值括起来。

当 `tag` 为 `issue` 时

`value` 字段包含 CA 域名称。此字段可能包含 Amazon CA 以外的 CA 的名称。但是，如果您没有指定以下四个 Amazon CA 之一的 CAA 记录，ACM 将无法向您的域或子域颁发证书：

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

`value` 字段也可以包含分号 (;)，指示不应允许任何 CA 为您的域或子域颁发证书。如果您在某个时候决定您不再需要为某个特定的域颁发的证书，请使用此字段。

当 `tag` 是 `issuewild` 时

`value` 字段与 `tag` 为 `issue` 时的相同，只是它适用于通配符证书。

Example CAA 记录示例

在以下示例中，首先是您的域名，然后是记录类型 (CAA)。 `flags` 字段始终为 0。 `tags` 字段可以是 `issue` 或 `issuewild`。如果字段为 `issue` 且您在 `value` 字段中键入 CA 服务器的域名称，则 CAA 记录指示您指定的服务器已被允许颁发您请求的证书。如果您在 `value` 字段中键入分号“;”，则 CAA 记录指示不允许任何 CA 颁发证书。CAA 记录的配置因 DNS 提供商而异。

Domain	Record type	Flags	Tag	Value
example.com.	CAA	0	issue	"SomeCA.com"
example.com.	CAA	0	issue	"amazon.com"
example.com.	CAA	0	issue	"amazontrust.com"
example.com.	CAA	0	issue	"awstrust.com"
example.com.	CAA	0	issue	"amazonaws.com"
example.com	CAA	0	issue	";"

有关如何添加或修改 DNS 记录的更多信息，请与您的 DNS 提供商核实。Route 53 支持 CAA 记录。如果 Route 53 是您的 DNS 提供商，请参阅 [CAA 格式](#) 以了解有关创建记录的更多信息。

入门

登录 AWS 管理控制台，并通过以下网址打开 ACM 控制台：<https://console.aws.amazon.com/acm/home>。如果显示介绍页面，请选择 Get Started。否则，请在左侧导航窗格中选择 Certificate Manager 或 Private CAs。

ACM 支持可用于跨 Internet 或通过内部网络实现安全通信的 SSL/TLS 证书。您可以请求由 ACM 颁发的公开信任证书，也可以导入证书。导入的证书可以由第三方颁发并且是公开信任的，也可以是自签名的。您还可以使用 ACM 控制台在您的组织中请求由私有证书颁发机构 (CA) 颁发的私有证书。默认情况下，私有证书不受信任。管理员必须将其安装在客户端信任存储中。

本文档主要讨论公有 ACM 证书和第三方证书。它还讨论如何使用现有私有 CA 颁发私有证书。要了解有关创建和使用私有 CA 的更多信息，请参阅 [AWS Certificate Manager Private Certificate Authority](#)。

主题

- [请求公有证书 \(p. 18\)](#)
- [请求私有证书 \(p. 20\)](#)
- [导出私有证书 \(p. 21\)](#)
- [使用 DNS 验证域所有权 \(p. 22\)](#)
- [使用电子邮件验证域所有权 \(p. 25\)](#)
- [列出由 ACM 管理的证书 \(p. 29\)](#)
- [描述 ACM 证书 \(p. 31\)](#)
- [删除由 ACM 管理的证书 \(p. 33\)](#)
- [安装 ACM 证书 \(p. 33\)](#)
- [重新发送验证电子邮件 \(可选\) \(p. 33\)](#)

请求公有证书

以下各部分将讨论如何使用 ACM 控制台或 AWS CLI 来请求公有 ACM 证书。如果您遇到了请求证书问题，请参阅[排查证书请求问题 \(p. 93\)](#)。如果您遇到了为 .IO 域请求证书的问题，请参阅[排查 .IO 域问题 \(p. 96\)](#)。要使用私有证书颁发机构 (CA) 请求私有证书，请参阅[请求私有证书 \(p. 20\)](#)。

主题

- [使用控制台请求公有证书 \(p. 18\)](#)
- [使用 CLI 请求公有证书 \(p. 19\)](#)

使用控制台请求公有证书

请求 ACM 公有证书 (控制台)

1. 登录 AWS 管理控制台，并通过以下网址打开 ACM 控制台：<https://console.aws.amazon.com/acm/home>。
2. 在 Request a certificate 页面上，键入您的域名。您可以使用完全限定域名 (FQDN) (例如 **www.example.com**) 或裸域名或顶点域名 (例如 **example.com**)。您还可以在最左侧位置使用星号 (*) 作为通配符来保护同一域中的多个站点名称。例如，***.example.com** 可保护 **corp.example.com** 和

images.example.com。通配符名称将显示在 ACM 证书的 Subject 字段和 Subject Alternative Name 扩展中。

Note

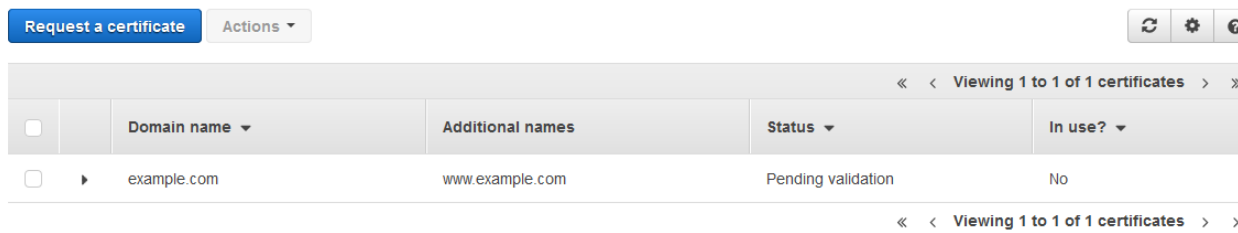
请求通配符证书时，星号 (*) 必须位于域名的最左侧位置，而且只能保护一个子域级别。例如，***.example.com** 可保护 **login.example.com** 和 **test.example.com**，但无法保护 **test.login.example.com**。另请注意，***.example.com** 仅保护 **example.com** 的子域，但不保护裸域或顶点域 (**example.com**)。要同时保护二者，请参阅下一个步骤。

3. 要将更多域名添加到 ACM 证书，请选择 Add more names，然后在打开的文本框中键入其他域名。这对于同时保护裸域或顶点域 (如 **example.com**) 及其子域 (***.example.com**) 很有用。
4. 键入有效的域名后，选择审核并请求，或者选择取消以退出。

Important

除非您选择退出，否则您的证书将自动记录在至少两个公有证书透明度数据库中。目前，您不能使用控制台来选择退出。您必须使用 AWS CLI 或 API。有关更多信息，请参阅 [选择退出证书透明度日志记录](#) (p. 11)。有关透明度日志的一般信息，请参阅 [证书透明度日志](#) (p. 3)。

5. 如果审核页面正确包含您提供的请求信息，请选择确认并请求。以下页面显示您的请求状态为“等待验证”。



在 ACM 颁发证书之前，它验证您是否拥有或可以控制证书请求中的域名。您可以使用电子邮件验证或 DNS 验证。如果选择电子邮件验证，则对于每一个域名，ACM 都会将验证电子邮件发送到在 WHOIS 数据库中注册的三个联系人地址和五个常用系统管理地址。您或授权代表必须回复其中的一封电子邮件。有关更多信息，请参阅 [使用电子邮件验证域所有权](#) (p. 25)。如果使用 DNS 验证，则只需将 ACM 提供的 CNAME 记录写入您的 DNS 配置。有关 DNS 验证的更多信息，请参阅 [使用 DNS 验证域所有权](#) (p. 22)。

Note

如果您可以编辑 DNS 配置，建议使用 DNS 域验证而不是电子邮件验证。相对于电子邮件验证，DNS 验证有多种优势。请参阅 [使用 DNS 验证域所有权](#) (p. 22)。

使用 CLI 请求公有证书

在命令行上使用 [request-certificate](#) 命令请求新的公有 ACM 证书。

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

此命令输出新私有证书的 Amazon 资源名称 (ARN)。

```
{
  "CertificateArn":
    "arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012"
```

```
}
```

请求私有证书

以下各部分讨论如何使用 ACM 控制台或 ACM PCA CLI 从现有私有证书颁发机构 (CA) 请求私有证书。有关创建私有 CA 的更多信息，请参阅[创建私有证书颁发机构](#)。

ACM 颁发的私有证书类似于 ACM 颁发的公有证书。这些证书具有以下限制：

- 您必须使用 DNS 主题名称。有关更多信息，请参阅[域名](#) (p. 4)。
- 您只能使用 2048 位 RSA 私有密钥算法。
- SHA256WithRSAEncryption 是唯一受支持的签名算法。
- 每个证书的有效期为 13 个月。
- 私有 CA 必须是 Active，CA 私有密钥类型必须是 RSA 2048 或 RSA 4096。
- ACM 会尽可能在 11 个月后自动续订证书。

ACM PCA 颁发的专用证书没有上述限制。您可以使用私有 CA 创建具有任何主题名称，使用任何受支持的私有密钥算法、任何签名算法以及任何有效期的证书。如果您必须以特定名称标识主题，或者您无法轻松轮换证书，则这是有益的。有关更多信息，请参阅[颁发私有证书](#)。

主题

- [使用控制台请求私有证书](#) (p. 20)
- [使用 CLI 请求私有证书](#) (p. 21)

使用控制台请求私有证书

1. 登录 AWS 管理控制台，并通过以下网址打开 ACM 控制台：<https://console.aws.amazon.com/acm/home>。
2. 选择 Request a private certificate，然后选择 Request a certificate。
3. 从下拉列表中选择您的私有 CA。有关 CA 的信息将填写在下面的列表中，以帮助您验证是否选择了所需的 CA。

Note

ACM 控制台对于具有 ECDSA 密钥的私有 CA 显示 Ineligible。

4. 选择 Next。
5. 在 Request a certificate 页面上，键入域名。您可以使用完全限定域名 (FQDN) (例如 **www.example.com**) 或裸域名或顶点域名 (例如 **example.com**)。您还可以在最左侧位置使用星号 (*) 作为通配符来保护同一域中的多个站点名称。例如，***.example.com** 可保护 **corp.example.com** 和 **images.example.com**。通配符名称将显示在 ACM 证书的 Subject 字段和 Subject Alternative Name 扩展中。

Note

请求通配符证书时，星号 (*) 必须位于域名的最左侧位置，而且只能保护一个子域级别。例如，***.example.com** 可保护 **login.example.com** 和 **test.example.com**，但无法保护 **test.login.example.com**。另请注意，***.example.com** 仅保护 **example.com** 的子域，但不保护裸域或顶点域 (**example.com**)。要同时保护二者，请参阅下一个步骤。

6. 要将更多域名添加到 ACM 证书，请选择 Add more names，然后在打开的文本框中键入其他域名。这对于同时保护裸域或顶点域 (如 **example.com**) 及其子域 (***.example.com**) 很有用。

- 键入有效的名称后，选择审核并请求，或者选择取消以退出。
- 检查审核页面以确保信息全部正确，然后选择 Confirm and request。

Note

您不需要验证私有证书。

使用 CLI 请求私有证书

使用 `request-certificate` 命令在 ACM 中请求私有证书。

```
aws acm request-certificate \  
--domain-name www.example.com \  
--idempotency-token 12563 \  
--options CertificateTransparencyLoggingPreference=DISABLED \  
--certificate-authority-arn arn:aws:acm-pca:region:account:\  
certificate-authority/12345678-1`234-1234-1234-123456789012
```

此命令输出新私有证书的 Amazon 资源名称 (ARN)。

```
{  
  "CertificateArn":  
    "arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012"  
}
```

导出私有证书

您可以导出私有证书以供在任何位置使用。您可以导出证书、证书链和加密的私有密钥。您必须安全地存储私有密钥。该密钥与在证书中嵌入的公有密钥相关。

私有密钥是一个 2048 位 RSA 密钥。您可以使用以下 OpenSSL 命令来解密它。在系统提示时提供密码。

```
openssl rsa -in encrypted_key.pem -out decrypted_key.pem
```

主题

- [使用控制台导出私有证书 \(p. 21\)](#)
- [使用 CLI 导出私有证书 \(p. 22\)](#)

使用控制台导出私有证书

- 登录 AWS 管理控制台，并通过以下网址打开 ACM 控制台：<https://console.aws.amazon.com/acm/home>。
- 选择 Certificate Manager。
- 选择您要导出的证书。
- 在 Actions 菜单上，选择 Export (private certificates only)。
- 输入并确认私有密钥的密码。
- 选择 Generate PEM Encoding。
- 您可以将证书、证书链和加密密钥复制到内存中，或者为每个选择 Export to a file。
- 选择完成。

使用 CLI 导出私有证书

使用 `export-certificate` 命令导出私有证书和私有密钥。为了提高安全性，请在使用此命令之前将密码安全地存储在文件中。这样可以防止密码存储在命令历史记录中，并防止在您键入密码时其他人看到密码。

```
aws acm export-certificate --certificate-arn \  
arn:aws:acm:region:account:\  
certificate/12345678-1234-1234-1234-123456789012 \  
--passphrase --file://path-to-passphrase-file
```

此命令输出 base64 编码的 PEM 格式证书、证书链和私有密钥。私有密钥以 PKCS #8 语法输出。

```
{  
  "PrivateKey":  
    "-----BEGIN ENCRYPTED PRIVATE KEY-----  
    ...PKCS8 Base64-encoded encrypted private key ...  
    -----END ENCRYPTED PRIVATE KEY-----",  
  "CertificateChain":  
    "-----BEGIN CERTIFICATE-----  
    ...Base64-encoded certificate...  
    -----END CERTIFICATE-----  
    -----BEGIN CERTIFICATE-----  
    ...Base64-encoded private key...  
    -----END CERTIFICATE-----",  
  "Certificate":  
    "-----BEGIN CERTIFICATE-----  
    ...Base64-encoded certificate...  
    -----END CERTIFICATE-----"  
}
```

要将所有内容输出到文件，请使用 `>` 重定向器，如下所示。

```
aws acm export-certificate --certificate-arn \  
arn:aws:acm:region:account:\  
certificate/12345678-1234-1234-1234-123456789012 \  
--passphrase file://path-to-passphrase-file\  
> c:\temp\export.txt
```

使用 DNS 验证域所有权

AWS Certificate Manager (ACM) 必须先确认您拥有或可以控制请求中指定的所有域名，然后 Amazon 证书颁发机构 (CA) 才能为网站颁发证书。请求证书时，您可以选择电子邮件验证或 DNS 验证。本主题讨论 DNS 验证。有关电子邮件验证的信息，请参阅[使用电子邮件验证域所有权 \(p. 25\)](#)。

Note

验证仅适用于 AWS Certificate Manager (ACM) 提供的证书。ACM 不会验证[已导入证书 \(p. 41\)](#)的域所有权。

域名系统 (DNS) 是连接到网络的资源的目录服务。在 Internet 上，DNS 服务器主要用于将域名转换为识别和定位资源 (如计算机和其他设备) 的数字 IP 地址。DNS 服务器上的数据库包含用于此种转换和启用其他功能的域记录。例如，A 记录是一种将域名映射到 IPV4 地址的 DNS 记录。MX 记录用于路由电子邮件。NS 记录列出域的所有名称服务器。

ACM 使用 CNAME (规范名称) 记录验证您是否拥有或可以控制某个域。如果选择 DNS 验证，ACM 会提供一条或多条 CNAME 记录以便插入 DNS 数据库。例如，如果为 `example.com` 域请求证书并指定

`www.example.com` 为其他名称，则 ACM 为您创建两条 CNAME 记录。每条记录都是专为您的域和账户创建的，包含名称和值。值是指向 ACM 拥有的域的别名，ACM 使用该域自动续订证书。CNAME 记录只需添加到 DNS 数据库中一次。只要证书正在使用中，并且 CNAME 记录保持不变，ACM 就会自动续订证书。此外，如果使用 Amazon Route 53 创建域，ACM 可以为您写入 CNAME 记录。

下表显示了五个域名的示例 CNAME 记录。`_x` 值是由 ACM 生成的长随机字符串。例如，`_3639ac514e785e898d2646601fa951d5.example.com` 代表生成的名称。注意，表中的前两个 `_x` 值相同。也就是说，ACM 为通配符名称 `*.example.com` 创建的随机字符串与为基本域名 `example.com` 创建的随机字符串相同。另请注意，ACM 会为 `example.com` 和 `www.example.com` 创建不同的 CNAME 记录。

域名	DNS 区域	名称	类型	值
<code>*.example.com</code>	<code>example.com</code>	<code>_x1.example.com</code>	别名记录	<code>_x2.acm-validations.aws</code>
<code>example.com</code>	<code>example.com</code>	<code>_x1.example.com</code>	别名记录	<code>_x2.acm-validations.aws</code>
<code>www.example.com</code>	<code>example.com</code>	<code>_x3.www.example.com</code>	别名记录	<code>_x4.acm-validations.aws</code>
<code>host.example.com</code>	<code>example.com</code>	<code>_x5.host.example.com</code>	别名记录	<code>_x6.acm-validations.aws</code>
<code>subdomain.example.com</code>	<code>subdomain.example.com</code>	<code>_x7.subdomain.example.com</code>	别名记录	<code>_x8.acm-validations.aws</code>
<code>host.subdomain.example.com</code>	<code>subdomain.example.com</code>	<code>_x9.host.subdomain.example.com</code>	别名记录	<code>_x10.acm-validations.aws</code>

相比电子邮件验证，DNS 验证有许多优势：

- DNS 要求您在请求 ACM 证书时为每个域名只创建一条 CNAME 记录。使用电子邮件验证时，每个域名最多需要发送八封电子邮件。
- 只要 DNS 记录保持不变，您就可以为 FQDN 请求更多的 ACM 证书，也就是说，您可以创建多个具有相同域名的证书。不必获取新的 CNAME 记录。这样做的原因有很多。例如，您可能需要覆盖不同子域的新证书。您可能需要在多个区域创建相同的证书 (此验证令牌适用于任何区域)。您可能需要替换已删除的证书。
- ACM 会自动续订您使用 DNS 验证的 ACM 证书。只要证书正在使用中，并且 DNS 记录保持不变，ACM 就会在每个证书到期前续订证书。
- 如果您使用 Route 53 管理公共 DNS 记录，ACM 可以替您添加 CNAME 记录。
- 与电子邮件验证过程相比，您可以更轻松地自动执行 DNS 验证过程。

但请注意，如果您无权修改您的域的 DNS 记录，则需要使用电子邮件验证。

使用 DNS 验证：

1. 登录 AWS 管理控制台，并通过以下网址打开 ACM 控制台：<https://console.aws.amazon.com/acm/home>。如果显示介绍页面，请选择 Get Started。否则，请选择 Request a certificate。
2. 在 Request a certificate 页面上，键入您的域名。有关键入域名的更多信息，请参阅[请求公有证书 \(p. 18\)](#)。
3. 要向 ACM 证书添加更多域名，请在您刚刚键入的名称下方打开的文本框中键入其他名称。
4. 选择 Next。

5. 选择 DNS validation。
6. 选择 Review and request。确认域名和验证方法正确无误。
7. 选择 Confirm and request。
8. 在 Validation 页面上，展开域名信息或选择 Export DNS configuration to a file。如果展开域信息，ACM 会显示您必须添加到 DNS 数据库中的 CNAME 记录的名称和值 (用以验证您控制该域)。

The screenshot shows the 'Request a certificate' page in the AWS Certificate Manager console. On the left, a sidebar lists the steps: Step 1: Add domain names, Step 2: Select validation method, Step 3: Review, and Step 4: Validation (which is highlighted). The main content area has a header 'Request a certificate' and a status box indicating 'Request in progress'. Below this, the 'Validation' section explains that a CNAME record must be added to the DNS configuration for the domain 'www.steelcity.xyz'. A table shows the required CNAME record details: Name (_ede669291a50b22db41fe58b625f521b.www.steelcity.xyz), Type (CNAME), and Value (_0da75843714b033c13ead76230a16ce1.acm-validations.aws). A note mentions that the DNS configuration can be updated by ACM for Route 53 customers. At the bottom, there is a button 'Create record in Route 53' and a link to 'Export DNS configuration to a file'.

Domain	Validation status
<input checked="" type="checkbox"/> www.steelcity.xyz	Pending validation

Name	Type	Value
_ede669291a50b22db41fe58b625f521b.www.steelcity.xyz	CNAME	_0da75843714b033c13ead76230a16ce1.acm-validations.aws

9. 如果满足以下条件，则会显示 Create record in Route 53 按钮：

- 您使用 Route 53 作为 DNS 提供商。
- 您将域托管在 Route 53 中。
- 您有权写入 Route 53 托管区域。
- 您的 FQDN 尚未经过验证。

如果您的 FQDN 已经过验证，或者您无权写入要请求的域名的 Route 53 托管区域，Create record in Route 53 按钮将显示为禁用状态。有关 Route 53 记录集的更多信息，请参阅[使用资源记录集](#)。

Note

目前，您无法以编程方式请求 ACM 在 Route 53 中自动创建您的记录。但是，您可以使用 AWS CLI 或 API 调用 Route 53 创建该记录。

10. 从控制台或导出文件将此记录添加到您的数据库。有关添加 DNS 记录的更多信息，请参阅[向您的数据库添加 CNAME \(p. 25\)](#)。您可以选择继续跳过此步骤。稍后，通过在控制台中打开证书请求可以返回此步骤。

Note

如果您以前在请求证书时验证了 FQDN，并且您正在为同一个 FQDN 请求另一个证书，则不必再添加其他 DNS 记录。

Note

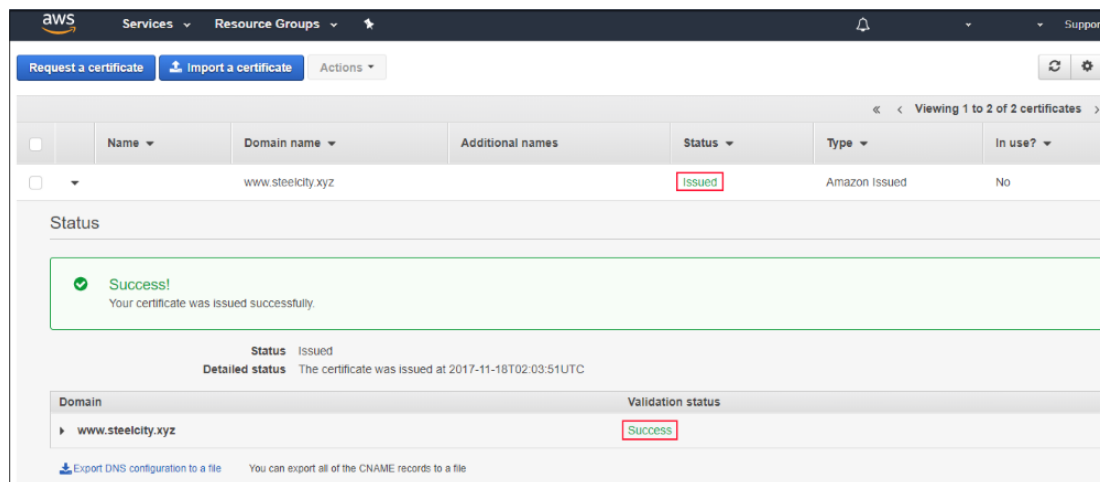
添加包含域名 (如 `.example.com`) 的 CNAME 记录可能会导致域名重复 (如 `.example.com.example.com`)。要避免重复，您可以仅手动复制所需的 CNAME 部分。其格式为 `_3639ac514e785e898d2646601fa951d5`。

11. 更新 DNS 配置后，选择继续。ACM 显示一个表格视图，其中包含您的所有证书。它显示了您请求的证书及其状态。在 DNS 提供商传播您的记录更新后，ACM 最多需要几个小时来验证域名和颁发证书。在

此期间，ACM 显示验证状态为等待验证。验证域名后，ACM 将验证状态更改为成功。AWS 颁发证书后，ACM 将证书状态更改为已颁发。

Note

如果 ACM 无法在生成 CNAME 值后的 72 小时内验证域名，ACM 会将证书状态更改为验证超时。导致此结果的最可能原因是您未使用 ACM 生成的值更新 DNS 配置。要解决此问题，您必须请求新的证书。



向您的数据库添加 CNAME

要使用 DNS 验证，您必须能够向您的域的 DNS 配置添加 CNAME 记录。如果您的 DNS 提供商不是 Route 53，请联系提供商了解如何添加记录。如果您的提供商是 Route 53，ACM 可为您创建 CNAME 记录，如前面的步骤 9 所述。如果需要自行添加此记录，请参阅 Route 53 开发人员指南 中的 [编辑资源记录集](#)。

Note

如果您无权编辑 DNS 配置，则必须使用电子邮件验证。

从您的数据库中删除 CNAME

只要证书正在使用中，并且 ACM 为您创建的 CNAME 记录仍在您的 DNS 数据库中，ACM 就会自动续订您的证书。您可以通过从证书关联的 AWS 服务删除证书或通过删除 CNAME 记录来停止自动续订。如果您的 DNS 提供商不是 Route 53，请联系提供商了解如何删除此记录。如果您的提供商是 Route 53，请参阅 Route 53 开发人员指南 中的 [删除资源记录集](#)。有关托管证书续订的更多信息，请参阅 [适用于 ACM 的由 Amazon 颁发的证书的托管续订 \(p. 35\)](#)。

使用电子邮件验证域所有权

AWS Certificate Manager (ACM) 必须先确认您拥有或可以控制请求中指定的所有域，然后 Amazon 证书颁发机构 (CA) 才能为网站颁发证书。您可以使用电子邮件或 DNS 执行验证。本主题讨论电子邮件验证。有关 DNS 验证的信息，请参阅 [使用 DNS 验证域所有权 \(p. 22\)](#)。

Note

验证仅适用于 AWS Certificate Manager (ACM) 提供的证书。ACM 不会验证 [已导入证书 \(p. 41\)](#) 的域所有权。如果在验证 ACM 证书时遇到问题，请参阅 [解决证书验证问题 \(p. 95\)](#)。如果您未收到电子邮件，请参阅 [未收到验证电子邮件 \(p. 91\)](#)。

AWS Certificate Manager (ACM) 将电子邮件发送到 WHOIS 中列出的 3 个联系人地址，以及您为每个域指定的 5 个常用系统地址。也就是说，对于您请求中包含的每个域名和主题备用名称，将发送最多 8 封电子邮件。例如，如果您仅指定 1 个域名，您最多将收到 8 封电子邮件。要进行验证，您必须在 72 小时内对这 8 封电子邮件中的 1 封电子邮件进行验证。如果您指定 3 个域名，您最多将收到 24 封电子邮件。要进行验证，您必须在 72 小时内对这些电子邮件中的至少 3 封电子邮件 (指定的每个名称对应 1 封电子邮件) 进行验证。

电子邮件将发送到 WHOIS 中以下三个已注册联系人地址：

- 域注册者
- 技术联系人
- 管理联系人

Note

某些注册商允许您在 WHOIS 列表中隐藏联系人信息，而另一些注册商允许您将真实电子邮件地址替换为私密 (或代理) 地址。为了防止在从 ACM 接收域验证电子邮件时出现问题，请确保您的联系人信息在 WHOIS 中可见。如果您的 WHOIS 列表显示私密电子邮件地址，请确保发送到该地址的电子邮件被转发到您真实的电子邮件地址，或只需改为列出您的真实电子邮件地址。

如果您使用控制台请求证书，ACM 将执行 MX 查找，以确定哪些服务器接受您的域的电子邮件，并且对于找到的第一个域，它还会将邮件发送到以下五个常见的系统地址。如果您使用 [RequestCertificate](#) API 或 [request-certificate](#) AWS CLI 命令，则 ACM 不会执行 MX 查询。相反，它会将电子邮件发送到您在 `DomainName` 参数中或在可选 `ValidationDomain` 参数中指定的域名称。有关更多信息，请参阅 [MX 记录 \(p. 15\)](#)。

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

有关 ACM 如何确定您的域的电子邮件地址的更多信息，请参阅 [\(可选\) 为域配置电子邮件 \(p. 15\)](#)。

控制台将显示针对请求中指定的第一个域名，已经将验证电子邮件发送到的地址。电子邮件的发送地址为 `no-reply@certificates.amazon.com`。

Status

Validation not complete
The status of this certificate request is "Pending validation". Further action is needed to validate and approve the certificate.

Status Pending validation

Detailed status Email to validate the request was sent at 2017-04-07T01:43:33UTC but we have not received your approval to issue the certificate for the following domains:

▼ Example.com

- postmaster@example.com
- administrator@example.com
- webmaster@example.com
- admin@example.com
- hostmaster@example.com

Details

Type	Amazon Issued	Requested at	2017-04-07
In use?	No	Public key info	RSA 2048-b
Domain name	example.com	Signature algorithm	SHA256WIT
Number of additional names	0	ARN	arn:aws:ac
Identifier	12345678-1234-1234-1234-123456789012		1234-1234-
Serial number	N/A		

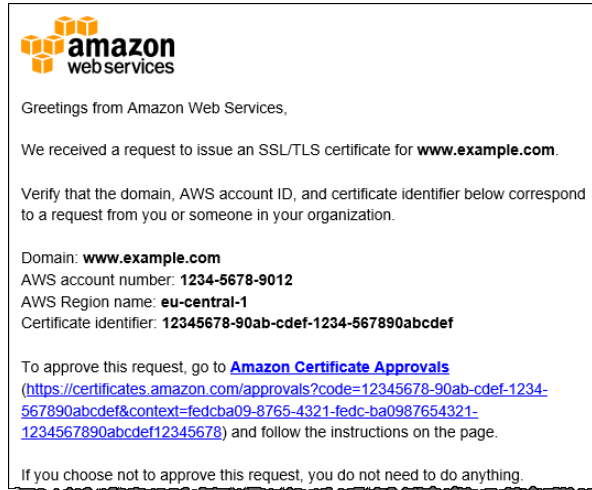
Note

上述过程有一个例外。如果您为以 **www** 或星号通配符 (*) 开头的域名请求 ACM 证书，则 ACM 将删除开头的 **www** 或星号，并将电子邮件发送到管理地址。通过在域名的剩余部分前面添加 **admin@**、**administrator@**、**hostmaster@**、**postmaster@** 和 **webmaster@** 来组成这些管理地址。例如，如果您为 **www.example.com** 请求 ACM 证书，则电子邮件将发送到 **admin@example.com** 而不是 **admin@www.example.com**。同样，如果您为 ***.test.example.com** 请求 ACM 证书，则电子邮件将发送到 **admin@test.example.com**。其余的常见管理地址的组成方式类似。

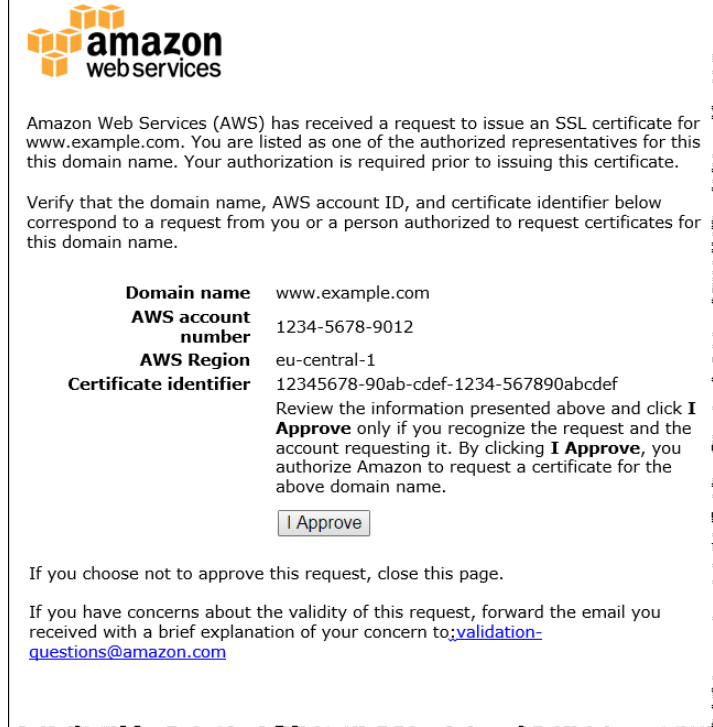
Note

确保将电子邮件发送到顶点域 (如 **example.com**) 的管理地址，而不是发送到子域 (如 **test.example.com**) 的管理地址。为此，请在 [RequestCertificate](#) API 或 [request-certificate](#) AWS CLI 命令中指定 **ValidationDomain** 选项。如果使用控制台请求证书，目前还无法使用此功能。

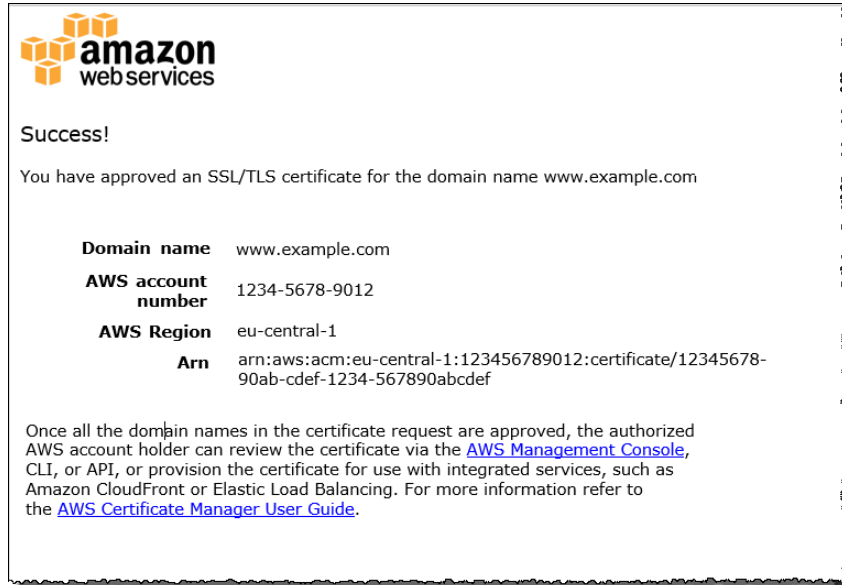
以下示例显示为证书请求中指定的每个域名发送的验证电子邮件。



选择用于转到 Amazon 证书审批网站的链接，然后选择 I Approve。



选择 I Approve 后，一个网站将打开，指示您的请求已成功。



您可以通过单击成功页面上的链接来导航回 ACM 控制台。ACM 可能需要数小时的时间验证域名和颁发证书。在此期间，ACM 显示验证状态为等待验证。验证域名后，ACM 将验证状态更改为成功。AWS 颁发证书后，ACM 将证书状态更改为已颁发。

Request a certificate Actions				
« < Viewing 1 to 1 of 1 certificates				
	Domain name	Additional names	Status	In use?
<input type="checkbox"/>	www.example.com	example.com	Issued	No
« < Viewing 1 to 1 of 1 certificates				

列出由 ACM 管理的证书

可使用 ACM 控制台或 AWS CLI 列出由 ACM 管理的证书

主题

- [列出证书 \(控制台\) \(p. 29\)](#)
- [列出证书 \(CLI\) \(p. 30\)](#)

列出证书 (控制台)


显示证书信息


每个证书在控制台中占用一行。默认情况下，系统会为每个证书显示以下列：

- Domain Name – 证书的完全限定域名。
- Additional Names – 此证书支持的其他名称。
- Status – 证书状态。它可以是以下值之一：
 - 等待验证

- 已颁发
- 非活跃
- 已过期
- 已撤销
- 已失败
- 已超时
- In Use? – ACM 证书是否主动与 Elastic Load Balancing 或 CloudFront 等 AWS 服务关联。值可以是 No 或 Yes。

自定义控制台显示

您可以通过选择控制台右上角的齿轮图标 () 来选择要显示的列。您可以从以下列中进行选择。

Show columns 

Select which columns you would like to show/hide:

- ☒ Domain name
- ☒ Additional names
- ☐ Created at
- ☒ Status
- ☐ Signature algorithm
- ☐ Key algorithm
- ☐ Not before
- ☐ Not after
- ☐ Subject
- ☐ Issuer
- ☐ Revocation reason
- ☐ Serial
- ☐ Revoked at
- ☒ In use?
- ☐ Arn

列出证书 (CLI)

可使用 `list-certificates` 命令列出由 ACM 管理的证书。

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

`list-certificates` 命令输出以下信息。

```
{
  "CertificateSummaryList": [
    {
      "CertificateArn":
"arn:aws:acm:region:account:certificate/123456789012-1234-1234-1234-12345678",
      "DomainName": "example.com"
    },
    {
      "CertificateArn":
"arn:aws:acm:region:account:certificate/123456789012-1234-1234-1234-12345678",
      "DomainName": "mydomain.com"
    }
  ]
}
```

默认情况下，仅列出受 [与 AWS Certificate Manager 集成的服务 \(p. 8\)](#) 支持的证书。也就是说，仅返回带 `keyTypes RSA_1024` 和 `RSA_2048` 的证书。要查看您拥有或控制的使用其他算法和位大小的其他证书，请使用以下示例中所示的 `--includes` 参数。利用此参数，您可以指定筛选器结构的成员。

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

描述 ACM 证书

您可以使用 ACM 控制台或 AWS CLI 列出有关证书的元数据。

主题

- [描述证书 \(控制台\) \(p. 31\)](#)
- [描述证书 \(CLI\) \(p. 31\)](#)

描述证书 (控制台)

要显示证书元数据，请选择紧邻域名左侧的箭头。控制台将显示类似于以下内容的信息。

The screenshot shows the AWS Certificate Manager console interface. At the top, there's a 'Request a certificate' button and an 'Actions' dropdown. Below this is a table with columns: Domain name, Additional names, Status, and In use?. The first row shows 'example.com' with 'www.example.com' as an additional name, status 'Issued', and 'In use?' set to 'Yes'. Below the table, the 'Status' section shows 'Status: Issued' and 'Detailed status: AWS issued the certificate at 2015-12-15T20:44:52UTC'. The 'Details' section is expanded, showing a list of metadata: In use? (Yes), Domain name (example.com), Number of additional names (1), Additional names (www.example.com), Identifier (12345678-1234-1234-1234-123456789012), Serial number (07:71:71:f4:6b:e7:bf:63:87:e6:ad:3c:b2:0f:d0:5b), Associated resources (arn:aws:cloudfront::123456789012:distribution/E12KXPQHLSYVC), Created (2015-12-15T20:43:44UTC), Not before (2015-12-15T00:00:00UTC), Validity days (397), Valid through (2017-01-15 (381 days)), Public key info (RSA 2048-bit), Signature algorithm (SHA-256 with RSA), and ARN (arn:aws:acm:us-east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012). The bottom of the console shows 'Viewing 1 to 1 of 1 certificates'.

描述证书 (CLI)

您可以使用 AWS CLI 来获取有关颁发的证书的信息，删除证书，或重新发送验证电子邮件。

检索 ACM 证书字段

您可以使用 `describe-certificate` 命令列出证书的元数据。

```
aws acm describe-certificate --certificate-arn
arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012
```

`describe-certificate` 命令输出以下信息。

```
{
```

```
"Certificate": {
  "CertificateArn":
"arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012",
  "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/E12KXPQHVL5YVC"
  ],
  "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"  
  }  
}
```

删除由 ACM 管理的证书

可以使用 ACM 控制台或 AWS CLI 删除证书。

主题

- [删除证书 \(控制台\)](#) (p. 33)
- [删除证书 \(CLI\)](#) (p. 33)

删除证书 (控制台)

在证书列表中，选中要删除的 ACM 证书对应的复选框。对于 Actions，选择 Delete。

Note

您无法删除正在由其他 AWS 服务使用的 ACM 证书。要删除正在使用的证书，您必须先删除证书关联。

删除证书 (CLI)

您可以使用 `delete-certificate` 命令列出证书的元数据。

```
aws acm delete-certificate --certificate-arn  
arn:aws:acm:region:123456789012:certificate/12345678-1234-1234-1234-123456789012
```

安装 ACM 证书

您不能使用 ACM 将 ACM 证书直接安装到基于 AWS 的网站或应用程序中。您必须使用与 ACM 集成的服务之一。有关更多信息，请参阅 [与 AWS Certificate Manager 集成的服务](#) (p. 8)。

重新发送验证电子邮件 (可选)

您可以使用电子邮件来验证自己拥有或可以控制某个域。每封电子邮件都包含一个验证令牌，您可以使用它批准证书请求。但是，由于批准过程需要的验证电子邮件可能会被垃圾邮件筛选器阻止或在传输中丢失，因此验证令牌将在 72 小时后自动过期。如果您未收到原始电子邮件或令牌已到期，可以请求重新发送电子邮件。

主题

- [重新发送电子邮件 \(控制台\)](#) (p. 33)
- [重新发送电子邮件 \(CLI\)](#) (p. 34)

重新发送电子邮件 (控制台)

选中待处理证书的复选框，选择操作，然后选择重新发送验证电子邮件。如果 72 小时的周期已过，且证书状态已更改为 Timed out，则无法重新发送验证电子邮件。

Note

前面的信息仅适用于 ACM 提供的证书，以及使用电子邮件验证的证书。[您导入到 ACM 中的证书 \(p. 41\)](#)不需要验证电子邮件。

Note

重新发送验证电子邮件仅适用于使用电子邮件验证而不是 DNS 验证的证书。有关 DNS 域验证的更多信息，请参阅[使用 DNS 验证域所有权 \(p. 22\)](#)。

重新发送电子邮件 (CLI)

您可以使用 `resend-validation-email` 命令重新发送电子邮件。

```
aws acm resend-validation-email --certificate-arn  
arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012 --validation-  
domain example.com
```

Note

`resend-validation-email` 命令仅适用于要使用电子邮件验证的 ACM 证书。对于已导入到 ACM 的证书或使用 ACM 管理的私有证书，不需要验证。

适用于 ACM 的由 Amazon 颁发的证书的托管续订

针对您的由 Amazon 颁发的 SSL/TLS 证书，ACM 提供了托管续订。这包括通过使用 ACM 颁发的公有和私有证书。ACM 会尝试在证书到期之前进行续订。如果可能，ACM 会自动续订您的证书，而无需您执行任何操作。

Note

对于 ACM 没有为其创建私有密钥和证书签名请求 (CSR) 的 ACM 私有 CA 证书，例如直接从您的 ACM 私有 CA (而不是从 ACM 证书管理) 颁发的证书，自动续订不可用。此外，对于[导入的证书 \(p. 41\)](#)，自动续订也不可用。有关更多信息，请参阅[手动域验证的工作方式](#)。

Note

当 ACM 续订证书时，证书的 Amazon 资源名称 (ARN) 保持不变。此外，ACM 证书是[区域性资源 \(p. 7\)](#)。如果您在多个 AWS 区域中具有带同一域名的证书，则 ACM 会单独续订每个证书。

Important

您的 ACM 证书必须活跃地与支持的 AWS 服务关联，然后才能自动续订。有关 ACM 支持的资源的信息，请参阅[与 AWS Certificate Manager 集成的服务 \(p. 8\)](#)。

有关托管证书续订的更多信息，请参阅以下主题。如果您遇到托管续订问题，请参阅[解决托管证书续订问题 \(p. 95\)](#)。

主题

- [域验证的工作方式 \(p. 35\)](#)
- [检查证书的续订状态 \(p. 37\)](#)
- [请求证书续订的域验证电子邮件 \(p. 39\)](#)

域验证的工作方式

续订证书之前，ACM 会尝试自动验证证书中的每个域名。有关更多信息，请参阅[自动域验证的工作方式 \(p. 35\)](#)。如果 ACM 无法自动验证域名，它会告知您需要执行操作来手动进行验证。有关更多信息，请参阅[如果自动验证失败 \(p. 36\)](#)。如果证书正在使用中 (与集成到 ACM 中的 AWS 服务关联)，并且证书中的所有域名都可以通过验证，则 ACM 会续订证书。

主题

- [自动域验证的工作方式 \(p. 35\)](#)
- [如果自动验证失败 \(p. 36\)](#)

自动域验证的工作方式

为了验证域，ACM 会向域发送自动的定期 HTTPS 请求。对于以 `www.` 开头的域，ACM 还会向父域发送 HTTPS 请求。例如，如果您的域是 `www.example.com`，则 ACM 会将周期性请求发送到

`www.example.com` 和 `example.com`。对于不是以 `www.` 开头的域，ACM 还会向 `www.domain` 发送 HTTPS 请求。ACM 处理通配符域名 (例如 `*.example.com`) 的方式与处理父域的方式相同。例如，请参阅下表。

Note

如果任何 HTTPS 连接尝试成功，ACM 会尝试自动续订证书。

ACM 用于自动验证的示例域名

证书中的域名	ACM 用于自动验证的域名
example.com	example.com www.example.com
www.example.com	www.example.com example.com
*.example.com	example.com www.example.com
subdomain.example.com	subdomain.example.com www.subdomain.example.com
www.subdomain.example.com	www.subdomain.example.com subdomain.example.com
*.subdomain.example.com	subdomain.example.com www.subdomain.example.com

如果 ACM 成功建立了 HTTPS 连接，ACM 会检查返回的证书以确保证书与 ACM 要续订的证书匹配。如果证书匹配，ACM 会考虑已验证的域名。

如果自动验证失败

如果 ACM 无法自动验证证书中的一个或多个域名，ACM 会告知您需要执行操作来手动验证域。域出于以下原因可能需要手动验证：

- ACM 无法与域建立 HTTPS 连接。
- 响应 HTTPS 请求时返回的证书与 ACM 要续订的证书不匹配。

如果您的证书距过期还有 45 天，并且证书中的一个或多个域名需要手动验证，ACM 将通过以下方式通知您：

通过电子邮件发送给域所有者 (电子邮件验证)

如果您最初请求证书时使用的是电子邮件验证，ACM 将向每个需要手动验证的域名的域所有者发送电子邮件。为确保能够收到此电子邮件，域所有者必须为每个域正确配置电子邮件。有关更多信息，请参阅 [\(可选\) 为域配置电子邮件 \(p. 15\)](#)。此电子邮件包含一个链接可引导您执行验证。此链接将在 72 小时后过期。如有必要，可以使用 AWS Certificate Manager 控制台、AWS CLI 或 API 请求 ACM 重新发送域验证电子邮件。有关更多信息，请参阅 [请求证书续订的域验证电子邮件 \(p. 39\)](#)。

通过电子邮件发送给您的 AWS 账户 (DNS 验证)

如果您在请求证书时最初使用的是 DNS 验证，ACM 将向与您的 AWS 账户关联的地址发送电子邮件。该电子邮件通知您 ACM 在尝试续订您的证书时遇到问题。最有可能的问题是，相应的位置不再有原始 CNAME 记录，或您的证书未与 ACM 集成的 AWS 服务关联。如果需要验证域和续订证书，您必须编辑 DNS 配置，以确保原始 CNAME 记录在相应位置。此外，您还必须确保自己的 ACM 证书正在使用中。有关 DNS 验证的更多信息，请参阅[使用 DNS 验证域所有权 \(p. 22\)](#)。

通过您的 AWS Personal Health Dashboard 中的通知

ACM 向您的 [AWS Personal Health Dashboard](#) 发送通知，告诉您证书中有一个或多个域名需要续订。ACM 将在您的证书还有 45 天、30 天、15 天、7 天、3 天和 1 天到期时发送这些通知。这些通知仅作参考提供信息之用。

检查证书的续订状态

可使用 AWS Certificate Manager 控制台、ACM API、AWS CLI 或 Personal Health Dashboard 检查 ACM 证书的续订状态。如果您使用控制台、AWS CLI 或 ACM API，证书续订可以具有下列四个可能的状态值之一。如果使用 Personal Health Dashboard，也会显示类似的值。

等待自动续订

ACM 正在尝试自动验证证书中的域名。有关更多信息，请参阅[域验证的工作方式 \(p. 35\)](#)。无需进一步操作。

等待验证

ACM 无法自动验证证书中的一个或多个域名。您必须执行相关操作验证这些域名，否则无法续订证书。如果您最初对证书使用的是电子邮件验证，请查找 ACM 发送的电子邮件，按照该电子邮件中的链接执行验证。如果之前使用的是 DNS 验证，请检查以确保 DNS 记录存在并且证书仍在使用中。

成功

证书中的所有域名均已验证，且 ACM 续订了证书。无需进一步操作。

已失败

证书过期之前有一个或多个域名未验证，因此 ACM 未续订证书。您可以[请求新的证书 \(p. 18\)](#)。

Note

对证书状态的更改可能需要数小时才能生效。

主题

- [检查状态 \(控制台\) \(p. 37\)](#)
- [检查状态 \(API\) \(p. 38\)](#)
- [检查状态 \(CLI\) \(p. 38\)](#)
- [检查状态 \(PHD\) \(p. 38\)](#)

检查状态 (控制台)

下面的过程介绍如何使用 ACM 控制台检查 ACM 证书的续订状态。

1. 通过 <https://console.aws.amazon.com/acm/home> 打开 AWS Certificate Manager 控制台。

2. 展开证书，查看其详细信息。
3. 在 Details 部分中查找 Renewal Status。如果没有看到状态，说明 ACM 未开始此证书的托管续订过程。

检查状态 (API)

有关介绍如何使用 [DescribeCertificate](#) 操作检查状态的 Java 示例，请参阅[描述证书](#) (p. 73)。

检查状态 (CLI)

下面的示例介绍如何使用 [AWS Command Line Interface \(AWS CLI\)](#) 检查 ACM 证书续订的状态。

```
$ aws acm describe-certificate --certificate-arn  
arn:aws:acm:region:123456789012:certificate/97b4deb6-8983-4e39-918e-ef1378924e1e
```

在响应中，请注意 RenewalStatus 字段中的值。如果没有看到 RenewalStatus 字段，说明 ACM 未开始证书托管续订过程。

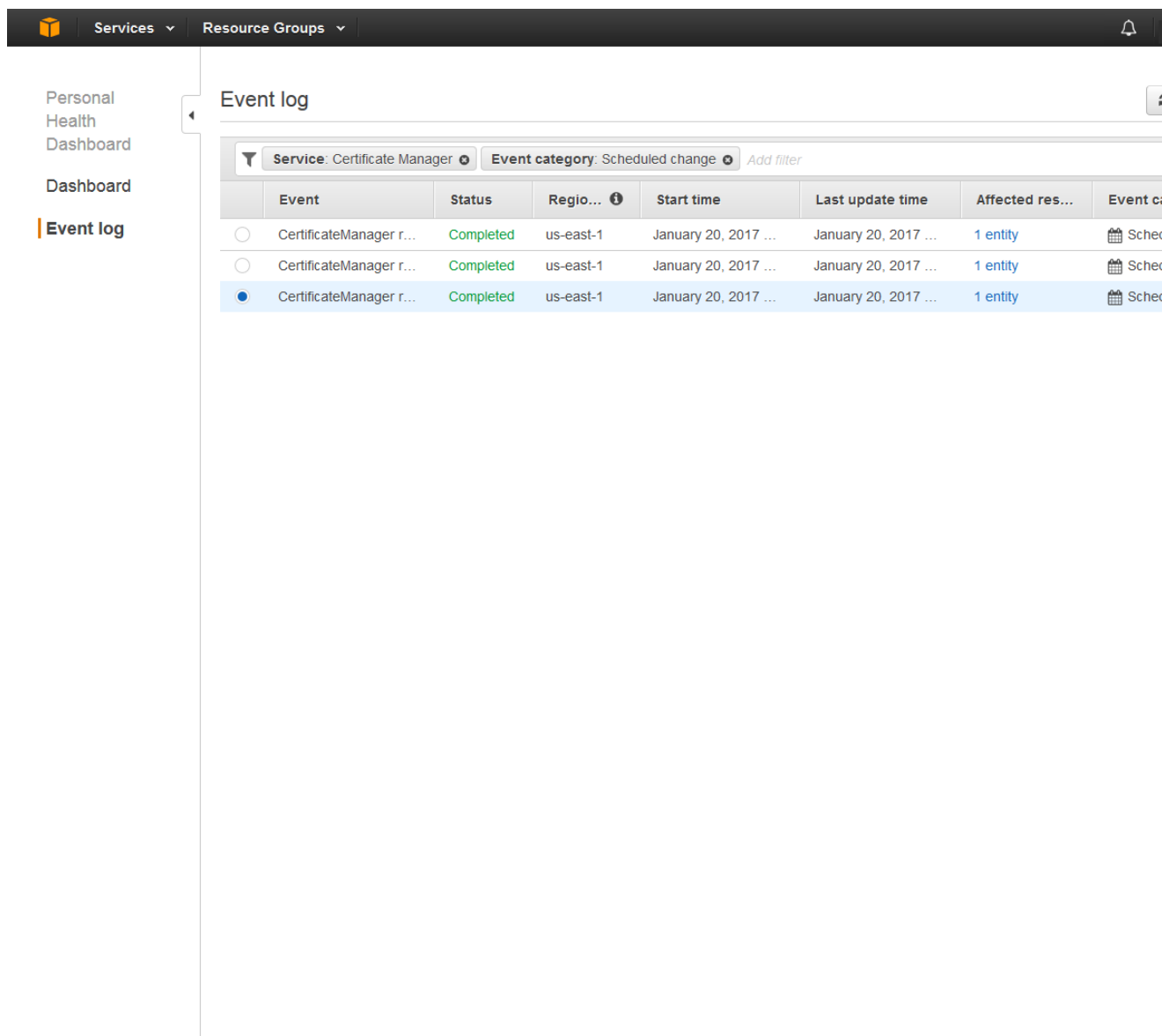
检查状态 (PHD)

ACM 在到期前 60 天会尝试自动续订您的 ACM 证书。请参阅[域验证的工作方式](#) (p. 35)。如果 ACM 无法自动续订您的证书，它会在到期前 45 天、30 天、15 天、7 天、3 天和 1 天向 Personal Health Dashboard 发送证书续订事件通知，通知您需要执行相关操作。Personal Health Dashboard 是 AWS Health 服务的一部分。它不需要设置，您的账户中通过身份验证的任何用户都可以查看。有关更多信息，请参阅[AWS Health 用户指南](#)。

使用 Personal Health Dashboard：

1. 在 <https://phd.aws.amazon.com/phd/home#/> 登录 Personal Health Dashboard。
2. 选择 Event log。
3. 对于 Filter by tags or attributes，选择 Service。
4. 选择 Certificate Manager。
5. 选择 Apply。
6. 对于 Event category，选择 Scheduled Change。
7. 选择 Apply。

如果 ACM 最近续订了某个 ACM 证书，可以看到类似于以下内容的信息。



The screenshot displays the AWS Certificate Manager console's 'Event log' section. The left-hand navigation pane includes links for 'Personal', 'Health', 'Dashboard', and 'Event log', with 'Event log' currently selected. The main content area shows a table of events filtered by 'Service: Certificate Manager' and 'Event category: Scheduled change'. The table contains three rows, all with a status of 'Completed' and occurring on 'January 20, 2017' in the 'us-east-1' region, each affecting '1 entity'. The third row is highlighted with a blue selection circle.

	Event	Status	Region	Start time	Last update time	Affected resources	Event category
<input type="radio"/>	CertificateManager r...	Completed	us-east-1	January 20, 2017 ...	January 20, 2017 ...	1 entity	Scheduled change
<input type="radio"/>	CertificateManager r...	Completed	us-east-1	January 20, 2017 ...	January 20, 2017 ...	1 entity	Scheduled change
<input checked="" type="radio"/>	CertificateManager r...	Completed	us-east-1	January 20, 2017 ...	January 20, 2017 ...	1 entity	Scheduled change

请求证书续订的域验证电子邮件

在为您的域配置联系人电子邮件地址后 (请参阅[\(可选\) 为域配置电子邮件 \(p. 15\)](#))，您可以使用 AWS Certificate Manager 控制台或 ACM API 来请求 ACM 向您发送证书续订的域验证电子邮件。您应在以下情况下执行此操作：

- 您最初请求 ACM 证书时使用的是电子邮件验证。
- 您的证书的续订状态为等待验证。有关确定证书的续订状态的信息，请参阅[检查证书的续订状态 \(p. 37\)](#)。
- 您未收到或找不到 ACM 为证书续订发送的原始域验证电子邮件。

请求 ACM 重新发送域验证电子邮件 (控制台)

1. 通过 <https://console.aws.amazon.com/acm/home> 打开 AWS Certificate Manager 控制台。
2. 选中需要手动域验证的证书旁的复选框。然后，依次选择 Actions 和 Resend validation email。

请求 ACM 重新发送域验证电子邮件 (ACM API)

在 ACM API 中使用 [ResendValidationEmail](#) 操作。在这种情况下，传递证书的 ARN、需要手动验证的域以及您要在其中接收域验证电子邮件的域。以下示例显示如何使用 AWS CLI 执行此操作。此示例包含换行符以便于阅读。

```
$ aws acm resend-validation-email --certificate-arn arn:aws:acm:us-  
east-2:111122223333:certificate/97b4deb6-8983-4e39-918e-ef1378924e1e  
--domain subdomain.example.com  
--validation-domain example.com
```

将证书导入到 AWS Certificate Manager 中

除了请求 AWS Certificate Manager (ACM) 提供的 SSL/TLS 证书，您还可以导入您在 AWS 外部获取的证书。您可能要执行此操作是因为您已从第三方发布者处获取证书，或者是因为 ACM 提供的证书不符合您的要求。

导入在 AWS 之外获取的 SSL/TLS 证书并将该证书关联到与 ACM 集成的服务后，您可以重新导入该证书，同时保留其关联。

导入证书后，您就可以将其用于与 ACM 集成的 AWS 服务 (p. 8)。您导入的证书与 ACM 提供的证书的工作方式相同，只有一个重要例外：ACM 不会为导入的证书提供托管续订 (p. 35)。

Important

您需要负责监控导入的证书的到期日期并在证书过期之前续订证书。如果导入的新证书与到期证书具有相同的 ARN，则新证书将替换旧证书。此外，ACM 会将服务和资源相同的新证书与旧证书相关联。

Important

我们建议您不要固定 ACM 证书。有关更多信息，请参阅 [证书固定 \(p. 11\)](#) 和 [排查证书固定问题 \(p. 93\)](#)。

若要续订导入的证书，您可以从证书发布者处获取新证书，然后将其导入到 ACM 中，也可以从 ACM [请求新证书 \(p. 18\)](#)。

ACM 中的所有证书都是区域性资源，包括您导入的证书。若要在不同的 AWS 区域中使用与 Elastic Load Balancing 负载均衡器相同的证书，您必须将证书导入到您要在其中使用它的每个区域。若要将证书用于 Amazon CloudFront，您必须将其导入到美国东部（弗吉尼亚北部）区域中。有关更多信息，请参阅 [支持的区域 \(p. 7\)](#)。

有关如何将证书导入到 ACM 中的信息，请参阅以下主题。如果您遇到导入证书问题，请参阅[排查证书导入问题 \(p. 92\)](#)。

主题

- [导入证书的先决条件 \(p. 41\)](#)
- [证书和密钥的导入格式 \(p. 42\)](#)
- [导入证书 \(p. 43\)](#)
- [重新导入证书 \(p. 44\)](#)

导入证书的先决条件

要将自签名 SSL/TLS 证书导入到 ACM 中，您必须提供证书及其私有密钥。要导入签名证书，还必须包含证书链。证书必须符合以下标准：

- 证书必须指定算法和密钥大小。目前，ACM 支持以下公有密钥算法：
 - 1024 位 RSA (RSA_1024)

- 2048 位 RSA (RSA_2048)
- 4096 位 RSA (RSA_4096)
- Elliptic Prime Curve 256 位 (EC_prime256v1)
- Elliptic Prime Curve 384 位 (EC_secp384r1)
- Elliptic Prime Curve 521 位 (EC_secp521r1)

Important

请注意，[集成服务](#)仅允许将其支持的算法和密钥大小与其资源关联。此外，这种支持因证书是否导入到 IAM 或 ACM 而有所差别。有关更多信息，请参阅每个服务的 [文档](#)。

- 对于 Elastic Load Balancing，请参阅 [Application Load Balancer 的 HTTPS 侦听器](#)。
- 对于 CloudFront，请参阅 [支持的 SSL/TLS 协议和密码](#)。
- 证书必须是 SSL/TLS X.509 版本 3 证书。它必须包含公有密钥、网站的完全限定域名 (FQDN) 以及有关发布者的信息。证书可以由您的私有密钥或发证 CA 的私有密钥进行自签名。如果证书由 CA 签名，则在导入证书时必须包含证书链。
- 证书在导入时必须是有有效的。在证书的有效期开始之前或结束之后，无法导入证书。NotBefore 证书字段包含有效期的开始日期，NotAfter 字段包含有效期的结束日期。
- 私有密钥必须是未加密的。您不能导入受密码或口令保护的私有密钥。
- 证书、私有密钥和证书链必须采用 PEM 编码。有关更多信息以及示例，请参阅 [证书和密钥的导入格式 \(p. 42\)](#)。

证书和密钥的导入格式

证书、私有密钥和证书链必须采用 PEM 编码。PEM 代表 Privacy Enhanced Mail。PEM 格式经常用于表示证书、证书请求、证书链和密钥。PEM 格式文件的典型扩展名是 .pem，但这并非强制要求。下面的示例介绍了这一格式。注意，如果错误地编辑 PEM 文件中的任何字符，或者向任意行的末尾添加一个或多个空格，则证书、证书链或私有密钥无效。

Example PEM 编码的证书

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

Example PEM 编码的证书链

一个证书链包含一个或多个证书。您可以使用文本编辑器、Windows 的 copy 命令或 Linux 的 cat 命令将证书文件连接到链中。证书必须按顺序连接，使得每个证书都直接认证前一个证书。最后复制根 CA 证书。以下示例包含三个证书，但您的证书链可能包含更多或更少的证书。

Important

不要将证书复制到证书链中。

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



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

Example PEM 编码的私有密钥

X.509 版本 3 证书使用公有密钥算法。在创建 X.509 证书或证书请求时，需要指定创建私有/公有密钥时必须使用的算法和密钥位大小。公有密钥放置在证书或请求中。您必须妥善保管关联的私有密钥。在导入证书时指定私有密钥。不得将密钥加密。下面的示例介绍一个 RSA 私有密钥。

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

下面的示例介绍一个 PEM 编码的椭圆曲线私有密钥。根据您的创建密钥的方式，可能不包含参数块。如果包含参数块，ACM 会在导入过程中使用此密钥前将其删除。

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

导入证书

您可以使用 AWS 管理控制台、AWS CLI 或 ACM API 将证书导入到 ACM 中。以下主题向您介绍如何使用 AWS 管理控制台和 AWS CLI。

主题

- [使用控制台导入 \(p. 43\)](#)
- [使用 AWS CLI 导入 \(p. 44\)](#)

使用控制台导入

以下示例说明如何使用 AWS 管理控制台导入证书。

1. 在 <https://console.aws.amazon.com/acm/home> 处打开 ACM 控制台。
2. 选择 Import a certificate。
3. 执行以下操作：
 - a. 对于 Certificate body，粘贴要导入的 PEM 编码证书。
 - b. 对于 Certificate private key，粘贴与证书的公有密钥匹配的 PEM 编码的未加密私有密钥。

Important

目前，与 [AWS Certificate Manager 集成的服务 \(p. 8\)](#) 仅支持 RSA_1024 和 RSA_2048 算法。

- c. (可选) 对于 Certificate chain，粘贴 PEM 编码的证书链。
4. 选择 Review and import。

5. 查看有关您的证书的信息，然后选择 Import。

使用 AWS CLI 导入

以下示例说明如何使用 [AWS Command Line Interface \(AWS CLI\)](#) 导入证书。示例假定以下各项：

- PEM 编码的证书存储在名为 `Certificate.pem` 的文件中。
- PEM 编码的证书链存储在名为 `CertificateChain.pem` 的文件中。
- PEM 编码的未加密私有密钥存储在名为 `PrivateKey.pem` 的文件中。

要使用以下示例，请将文件名替换为您自己的文件名，并在一个连续行中键入相应命令。为更便于阅读，以下示例包含了换行符和多余的空格。

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

如果 `import-certificate` 命令成功完成，则将返回导入的证书的 [Amazon 资源名称 \(ARN\)](#)。

重新导入证书

如果您已导入一个证书并将该证书与其他 AWS 服务关联，则可在该证书到期之前将其重新导入，同时保留原始证书的 AWS 服务关联。有关与 ACM 集成的 AWS 服务的更多信息，请参阅[与 AWS Certificate Manager 集成的服务 \(p. 8\)](#)。

重新导入证书时，适用以下条件：

- 您可以添加或删除域名。
- 您不能删除证书中的所有域名。
- 可以添加新密钥使用扩展，但不能删除现有扩展值。
- 您可以添加新扩展密钥使用扩展，但不能删除现有扩展值。
- 密钥类型和大小不能更改。

主题

- [使用控制台重新导入 \(p. 44\)](#)
- [使用 AWS CLI 重新导入 \(p. 45\)](#)

使用控制台重新导入

以下示例说明如何使用 AWS 管理控制台重新导入证书。

1. 在 <https://console.aws.amazon.com/acm/home> 处打开 ACM 控制台。
2. 选择或展开要重新导入的证书。
3. 打开证书的详细信息窗格并选择 Reimport certificate 按钮。如果您已通过选中证书名旁边的框来选择证书，请选择 Actions 菜单上的 Reimport certificate。
4. 对于 Certificate body，粘贴 PEM 编码的最终实体证书。
5. 对于 Certificate private key，粘贴与证书公有密钥关联的 PEM 编码的未加密私有密钥。

Important

目前，与 [AWS Certificate Manager 集成的服务 \(p. 8\)](#) 仅支持 RSA_1024 和 RSA_2048 算法。

6. (可选) 对于 Certificate chain，粘贴 PEM 编码的证书链。证书链包含所有中间发行证书机构的一个或多个证书以及根证书。如果要导入的证书是自行分配的，则不需要证书链。
7. 选择 Review and import。
8. 查看有关您的证书的信息。如果没有错误，请选择 Reimport。

使用 AWS CLI 重新导入

以下示例说明如何使用 [AWS Command Line Interface \(AWS CLI\)](#) 重新导入证书。示例假定以下各项：

- PEM 编码的证书存储在名为 Certificate.pem 的文件中。
- PEM 编码的证书链存储在名为 CertificateChain.pem 的文件中。
- PEM 编码的未加密私有密钥存储在名为 PrivateKey.pem 的文件中。
- 您具有要重新导入的证书的 ARN。

要使用以下示例，请将文件名和 ARN 替换为您自己的文件名和 ARN，并在一个连续行中键入相应命令。为更便于阅读，以下示例包含了换行符和多余的空格。

Note

要重新导入证书，您必须指定证书 ARN。

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

如果 import-certificate 命令成功完成，则将返回证书的 [Amazon 资源名称 \(ARN\)](#)。

为 AWS Certificate Manager 证书添加标签

标签是您可向 ACM 证书分配的标签。每个标签均包含一个键 和一个值。您可以使用 AWS Certificate Manager 控制台、AWS Command Line Interface (AWS CLI) 或 ACM API 来添加、查看或删除 ACM 证书的标签。您可以选择要在 ACM 控制台中显示的标签。

您可以创建满足您的需求的自定义标签。例如，您可以使用 `Environment = Prod` 或 `Environment = Beta` 标签来为多个 ACM 证书添加标签以确定每个 ACM 证书适合的环境。以下列表包含另外几个其他自定义标签的示例：

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

其他 AWS 资源也支持添加标签。因此，您可以将同一标签分配给不同的资源以指示这些资源是否相关。例如，您可以将标签 (例如 `Website = example.com`) 分配给 ACM 证书、负载均衡器以及用于 `example.com` 网站的其他资源。

主题

- [标签限制 \(p. 46\)](#)
- [管理标签 \(p. 46\)](#)

标签限制

以下是适用于 ACM 证书标签的基本限制：

- 每个 ACM 证书的最大标签数是 50。
- 标签键的最大长度是 127 个字符。
- 标签值的最大长度是 255 个字符。
- 标签键和值要区分大小写。
- 保留 `aws:` 前缀以供 AWS 使用；您无法添加、编辑或删除其键以 `aws:` 开头的标签。以 `aws:` 开头的标签不计入每个资源的标签数限制。
- 如果您计划在多个服务和资源中使用添加标签方案，请记得其他服务可能对允许使用的字符有其他限制。请参阅该服务对应的文档。
- ACM 证书标签不可在 AWS 管理控制台的 [资源组](#) 和 [标签编辑器](#) 中使用。

管理标签

您可以使用 AWS 管理控制台、AWS Command Line Interface 或 AWS Certificate Manager API 添加、编辑和删除标签。

管理标签 (控制台)

您可以使用 AWS 管理控制台添加、删除或编辑标签。您也可以在列中显示标签。

添加标签 (控制台)

使用 ACM 控制台通过以下过程添加标签。

向证书添加标签 (控制台)

1. 登录 AWS 管理控制台并通过以下网址打开 AWS Certificate Manager 控制台：<https://console.aws.amazon.com/acm/home>。
2. 选择要为其添加标签的证书旁的箭头。
3. 在详细信息窗格中，向下滚动到 Tags。
4. 选择 Edit 和 Add Tag。
5. 键入标签的键和值。
6. 选择 Save (保存)。

删除标签 (控制台)

使用 ACM 控制台通过以下过程删除标签。

删除标签 (控制台)

1. 登录 AWS 管理控制台并通过以下网址打开 AWS Certificate Manager 控制台：<https://console.aws.amazon.com/acm/home>。
2. 选择要删除其标签的证书旁的箭头。
3. 在详细信息窗格中，向下滚动到 Tags。
4. 选择 Edit。
5. 选择要删除的标签旁的 X。
6. 选择 Save (保存)。

编辑标签 (控制台)

使用 ACM 控制台通过以下过程编辑标签。


编辑标签 (控制台)

1. 登录 AWS 管理控制台并通过以下网址打开 AWS Certificate Manager 控制台：<https://console.aws.amazon.com/acm/home>。
2. 选择要编辑的证书旁的箭头。
3. 在详细信息窗格中，向下滚动到 Tags。
4. 选择 Edit。
5. 修改要更改的标签的键或值。
6. 选择 Save (保存)。

在列中显示标签 (控制台)

在 ACM 控制台中使用以下过程来在列中显示标签。

在列中显示标签 (控制台)

1. 登录 AWS 管理控制台并通过以下网址打开 AWS Certificate Manager 控制台：<https://console.aws.amazon.com/acm/home>。
2. 通过选择控制台右上角的齿轮图标  来选择要在列中显示的标签。
3. 选中要在列中显示的标签旁的复选框。

管理标签 (AWS Command Line Interface)

请参阅以下主题以了解如何使用 AWS CLI 添加、列出和删除标签。

- [add-tags-to-certificate](#)
- [list-tags-for-certificate](#)
- [remove-tags-from-certificate](#)

管理标签 (AWS Certificate Manager API)

请参阅以下主题以了解如何使用此 API 添加、列出和删除标签。

- [AddTagsToCertificate](#)
- [ListTagsForCertificate](#)
- [RemoveTagsFromCertificate](#)

身份验证和访问控制

访问 ACM 时需要有 AWS 可以用来验证您的请求的凭证。这些凭证必须有权访问 AWS 资源 (如 ACM 证书)。下面几节提供详细的信息来说明如何使用 [AWS Identity and Access Management \(IAM\)](#) 和 ACM 控制谁能访问您的资源，从而对这些资源进行保护。

主题

- [身份验证 \(p. 49\)](#)
- [访问控制 \(p. 50\)](#)

身份验证

您可以以下面任一类型的身份访问 AWS：

- **AWS 账户根用户** – 当您首次创建 AWS 账户时，最初使用的是一个对账户中所有 AWS 服务和资源有完全访问权限的单点登录身份。此身份称为 AWS 账户根用户，使用您创建账户时所用的电子邮件地址和密码登录，即可获得该身份。强烈建议您不使用根用户执行日常任务，即使是管理任务。请遵守[使用根用户的最佳实践](#)，仅将其用于创建您的首个 IAM 用户。然后请妥善保存根用户凭证，仅用它们执行少数账户和服务管理任务。
- **IAM 用户** – IAM 用户是您的 AWS 账户中的一种身份，它具有特定的自定义权限 (例如，用于在 ACM 中创建 a directory 的权限)。您可以使用 IAM 用户名和密码来登录以保护 AWS 网页，如 [AWS 管理控制台](#)、[AWS 开发论坛](#) 或 [AWS Support Center](#)。

除了用户名和密码之外，您还可以为每个用户生成[访问密钥](#)。在通过[多个软件开发工具包](#)之一或使用 [AWS Command Line Interface \(CLI\)](#) 以编程方式访问 AWS 服务时，可以使用这些密钥。SDK 和 CLI 工具使用访问密钥对您的请求进行加密签名。如果您不使用 AWS 工具，则必须自行对请求签名。ACM supports 签名版本 4，后者是一种用于对入站 API 请求进行身份验证的协议。有关验证请求的更多信息，请参阅 AWS General Reference 中的[签名版本 4 签名流程](#)。

- **IAM 角色** – IAM 角色是可在账户中创建的一种具有特定权限的 IAM 身份。它类似于 IAM 用户，但未与特定人员关联。利用 IAM 角色，您可以获得可用于访问 AWS 服务和资源的临时访问密钥。具有临时凭证的 IAM 角色在以下情况下很有用：
 - **联合身份用户访问** – 您也可以不创建 IAM 用户，而是使用来自 AWS Directory Service、您的企业用户目录或 Web 身份提供商的既有用户身份。他们被称为联合身份用户。在通过[身份提供商](#)请求访问权限时，AWS 将为联合用户分配角色。有关联合身份用户的更多信息，请参阅 IAM 用户指南中的[联合身份用户和角色](#)。
 - **AWS 服务访问** – 您可以使用您的账户中的 IAM 角色向 AWS 服务授予对您账户中资源的访问权限。例如，您可以创建一个角色，此角色允许 Amazon Redshift 代表您访问 Amazon S3 存储桶，然后将该存储桶提供的数据加载到 Amazon Redshift 群集中。有关更多信息，请参阅 IAM 用户指南中的[创建向 AWS 服务委派权限的角色](#)。

- 运行在 Amazon EC2 上的应用程序 – 对于在 EC2 实例上运行、并发出 AWS API 请求的应用程序，您可以使用 IAM 角色管理它们的临时凭证。这优先于在 EC2 实例中存储访问密钥。要将 AWS 角色分配给 EC2 实例并使其对该实例的所有应用程序可用，您可以创建一个附加到实例的实例配置文件。实例配置文件包含角色，并使 EC2 实例上运行的程序能够获得临时凭证。有关更多信息，请参阅 IAM 用户指南中的[使用 IAM 角色向在 Amazon EC2 实例上运行的应用程序授予权限](#)。

访问控制

您可以使用有效的凭证来对自己的请求进行身份验证，但您还必须拥有权限才能创建或访问 ACM 资源。例如，您必须拥有创建、导入、检索或列出证书的权限。

以下主题介绍如何管理权限。我们建议您先阅读概述。

- [管理对您的 ACM 资源的访问概述 \(p. 50\)](#)
- [AWS 托管策略 \(p. 51\)](#)
- [客户托管策略 \(p. 52\)](#)
- [内联策略 \(p. 52\)](#)
- [ACM API 权限：操作和资源参考 \(p. 55\)](#)

管理对您的 ACM 资源的访问概述

每个 AWS 资源属于一个 AWS 账户，而创建和访问资源的权限在该账户的权限策略中定义。账户管理员可以向 IAM 身份（即：用户、组和角色）挂载权限策略。一些服务（包括 ACM）还支持向资源附加权限策略。

Note

账户管理员（或管理员用户）是具有管理员权限的用户。有关更多信息，请参阅 IAM 用户指南中的[创建管理员用户和组](#)。

在管理权限时，您要决定谁可以获得权限、获得哪些资源的权限以及允许的特定操作。

主题

- [ACM 资源和操作 \(p. 50\)](#)
- [了解资源所有权 \(p. 51\)](#)
- [管理对 ACM 证书的访问 \(p. 51\)](#)

ACM 资源和操作

在 ACM 中，主要资源是证书。证书具有关联的唯一 Amazon 资源名称 (ARN)，如以下列表所示。

- ACM 证书

ARN 格式：

`arn:aws:acm:AWS region:AWS account ID:certificate/Certificate ID`

示例 ARN:

`arn:aws:acm:us-west-2:123456789012:certificate/12345678-12ab-34cd-56ef-12345678`

了解资源所有权

资源所有者 是创建了资源的 AWS 账户。也就是说，资源所有者是委托人实体的 AWS 账户，可对创建相应资源的请求进行身份验证。(委托人实体可以是 AWS 账户根用户、IAM 用户或 IAM 角色。)以下示例说明了它的工作原理。

- 如果您使用 AWS 账户根用户的凭证创建 ACM 证书，则您的 AWS 账户拥有该证书。
- 如果您在自己的 AWS 账户中创建 IAM 用户，您可以向该用户授予创建 ACM 证书的权限。但是，该用户所属的账户拥有该证书。
- 如果您在自己的 AWS 账户中创建 IAM 角色，并向该角色授予创建 ACM 证书的权限，则能够代入该角色的任何人都可以创建证书。但是，该角色所属的账户拥有该证书。

管理对 ACM 证书的访问

权限策略 规定谁可以访问哪些内容。本部分介绍创建权限策略时的可用选项。

Note

本节讨论如何在 ACM 范围内使用 IAM。它不提供有关 IAM 服务的详细信息。有关完整的 IAM 文档，请参阅 [IAM 用户指南](#)。有关 IAM 策略语法和说明的信息，请参阅 [AWS IAM 策略参考](#)。

您可以使用 IAM 创建策略，以便对 IAM 用户、组和角色应用权限。这些策略称为基于身份的策略。IAM 提供了以下类型的基于身份的策略：

- AWS 托管策略 - 由 AWS 创建和管理的策略。这些策略是可以附加到 AWS 账户中的多个用户、组和角色的独立策略。
- 客户托管策略 - 您在 AWS 账户中创建和管理的策略，您可以将它们附加到多个用户、组和角色。与使用 AWS 托管策略相比，使用客户托管策略可以更精确地进行控制。
- 内联策略 - 您创建和管理的策略，您将它们直接嵌入到单个用户、组或角色中。

其他服务 (如 Amazon S3) 还支持基于资源的权限策略。例如，您可以将策略附加到 Amazon S3 存储桶以管理对该存储桶的访问权限。ACM 不支持基于资源的策略。

AWS 托管策略

AWS 托管策略是基于身份的独立策略，可以将其附加到 AWS 账户中的多个用户、组和角色。AWS 托管策略由 AWS 创建和管理。以下 AWS 托管策略适用于 ACM。有关将托管策略附加到用户、组或角色的更多信息，请参阅 [IAM 用户指南](#) 中的 [使用托管策略](#)。

要使用 AWS 托管策略，具有管理权限的用户必须将此策略附加到用户、角色或组。有关附加 AWS 托管策略的更多信息，请参阅 [IAM 用户指南](#) 中的 [附加托管策略](#)。

主题

- [AWSCertificateManagerReadOnly](#) (p. 51)
- [AWSCertificateManagerFullAccess](#) (p. 52)

AWSCertificateManagerReadOnly

此策略提供对 ACM 证书的只读访问；它允许用户描述、列出和检索 ACM 证书。

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

要在控制台中查看此 AWS 托管策略，请转到 <https://console.aws.amazon.com/iam/home#policies/arn:aws:iam::aws:policy/AWSCertificateManagerReadOnly>。

AWSCertificateManagerFullAccess

此策略提供了对所有 ACM 操作和资源的完全访问。

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

要在控制台中查看此 AWS 托管策略，请转到 <https://console.aws.amazon.com/iam/home#policies/arn:aws:iam::aws:policy/AWSCertificateManagerFullAccess>。

客户托管策略

客户托管的策略是您可创建的基于身份的独立策略，您可以将这些策略附加到 AWS 账户中的多个用户、组或角色。您可以使用 AWS 管理控制台、AWS Command Line Interface (AWS CLI) 或 IAM API 管理和创建这些策略。有关更多信息，请参阅[客户托管策略](#)。

内联策略

内联策略是由您创建和管理的策略，它们直接嵌入在单个用户、组或角色中。以下策略示例演示如何分配权限来执行 ACM 操作。有关附加内联策略的更多信息，请参阅 [IAM 用户指南](#) 中的[使用内联策略](#)。您可以使用 AWS 管理控制台、AWS Command Line Interface (AWS CLI) 或 IAM API 创建和嵌入内联策略。

主题

- [列出证书 \(p. 53\)](#)
- [检索证书 \(p. 53\)](#)
- [导入证书 \(p. 53\)](#)
- [删除证书 \(p. 53\)](#)
- [对 ACM 的只读访问 \(p. 54\)](#)

- [对 ACM 的完全访问权限 \(p. 54\)](#)
- [对所有 AWS 资源的管理员访问权限 \(p. 55\)](#)

列出证书

以下策略允许用户列出用户账户中的所有 ACM 证书。

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

Note

ACM 证书需要此权限才能在 Elastic Load Balancing 和 CloudFront 控制台中显示。

检索证书

以下策略允许用户检索特定 ACM 证书。

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": "acm:GetCertificate",
    "Resource": "arn:aws:acm:us-east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
  }
}
```

导入证书

以下策略允许用户导入证书。

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": "acm:ImportCertificate",
    "Resource": "arn:aws:acm:ap-northeast-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
  }
}
```

删除证书

以下策略允许用户删除特定 ACM 证书。

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": "acm:DeleteCertificate",
    "Resource": "arn:aws:acm:us-east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
  }
}
```

对 ACM 的只读访问

以下策略允许用户描述和列出 ACM 证书并检索 ACM 证书和证书链。

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

Note

此策略可作为 AWS 管理控制台中的 AWS 托管策略。有关更多信息，请参阅 [AWSCertificateManagerReadOnly](#) (p. 51)。要在控制台中查看托管策略，请转到 <https://console.aws.amazon.com/iam/home#policies/arn:aws:iam::aws:policy/AWSCertificateManagerReadOnly>。

对 ACM 的完全访问权限

以下策略允许用户执行任何 ACM 操作。

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

Note

此策略可作为 AWS 管理控制台中的 AWS 托管策略。有关更多信息，请参阅 [AWSCertificateManagerFullAccess](#) (p. 52)。要在控制台中查看托管策略，请转到 <https://console.aws.amazon.com/iam/home#policies/arn:aws:iam::aws:policy/AWSCertificateManagerFullAccess>。

对所有 AWS 资源的管理员访问权限

以下策略允许用户对任何 AWS 资源执行任何操作。

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

Note

此策略可作为 AWS 管理控制台中的 AWS 托管策略。要在控制台中查看托管策略，请转到 <https://console.aws.amazon.com/iam/home#policies/arn:aws:iam::aws:policy/AdministratorAccess>。

ACM API 权限：操作和资源参考

如果您正在设置访问控制 (p. 50) 以及编写您可附加到 IAM 身份的权限策略 (基于身份的策略)，可以使用下表作为参考。表中第一列中列出每个 ACM API 操作。您可以在策略的 Action 元素中指定操作。剩余的列将提供额外的信息：

可以在您的 ACM 策略中使用 IAM 策略元素来表达条件。有关完整列表，请参阅 IAM 用户指南 中的 [可用键](#)。

Note

要指定操作，请在 API 操作名称之前使用 acm: 前缀 (例如，acm:RequestCertificate)。

ACM API 操作和权限

ACM API 操作	所需权限 (API 操作)	资源
AddTagsToCertificate	acm:AddTagsToCertificate	arn:aws:acm:AWS_region:AWS_account_ID:certificate_ID
DeleteCertificate	acm:DeleteCertificate	arn:aws:acm:AWS_region:AWS_account_ID:certificate_ID
DescribeCertificate	acm:DescribeCertificate	arn:aws:acm:AWS_region:AWS_account_ID:certificate_ID
ExportCertificate	acm:ExportCertificate	arn:aws:acm:AWS_region:AWS_account_ID:certificate_ID
GetCertificate	acm:GetCertificate	arn:aws:acm:AWS_region:AWS_account_ID:certificate_ID
ImportCertificate	acm:ImportCertificate	arn:aws:acm:AWS_region:AWS_account_ID:certificate_ID
ListCertificates	acm:ListCertificates	arn:aws:acm:AWS_region:AWS_account_ID:certificate_ID

ACM API 操作	所需权限 (API 操作)	资源
ListTagsForCertificate	acm:ListTagsForCertificate	arn:aws:acm:AWS_region:AWS_account_ID:certificate_ID
RemoveTagsFromCertificate	acm:RemoveTagsFromCertificate	arn:aws:acm:AWS_region:AWS_account_ID:certificate_ID
RequestCertificate	acm:RequestCertificate	arn:aws:acm:AWS_region:AWS_account_ID:certificate_ID
ResendValidationEmail	acm:ResendValidationEmail	arn:aws:acm:AWS_region:AWS_account_ID:certificate_ID

使用 AWS CloudTrail

您可以使用 CloudTrail 记录由 AWS Certificate Manager 和与 ACM 集成的服务进行的 API 调用，如以下主题中所讨论。

主题

- [使用 AWS CloudTrail 记录 AWS Certificate Manager API 调用 \(p. 57\)](#)
- [记录 ACM 相关的 API 调用 \(p. 66\)](#)

使用 AWS CloudTrail 记录 AWS Certificate Manager API 调用

AWS Certificate Manager (ACM) 与 AWS CloudTrail 集成在一起，后者是一项可以捕获 API 调用、将日志文件传输到您指定的 Amazon Simple Storage Service (Amazon S3) 存储桶并维护 API 调用历史记录的服务。CloudTrail 捕获从 AWS Certificate Manager 控制台、CLI 或您的代码发出的 API 调用。通过使用 CloudTrail 收集的信息，您可以确定向 ACM 发出的请求以及发出请求的 IP 地址、用户和时间等。

要了解有关 CloudTrail 的更多信息（包括如何配置和启用它），请参阅 [AWS CloudTrail 用户指南](#)。

在您的 AWS 账户中启用 CloudTrail 日志记录后，将在 CloudTrail 日志文件中跟踪对 ACM 操作执行的 API 调用。ACM 记录将与其他 AWS 服务记录一起写入。CloudTrail 基于时间段和文件大小来确定何时创建并写入新日志文件。

支持以下 ACM 操作：

- [AddTagsToCertificate](#)
- [DeleteCertificate](#)
- [DescribeCertificate](#)
- [ExportCertificate](#)
- [GetCertificate](#)
- [ImportCertificate](#)
- [ListCertificates](#)
- [ListTagsForCertificate](#)
- [RemoveTagsFromCertificate](#)
- [RequestCertificate](#)
- [ResendValidationEmail](#)

每个日志条目都包含有关生成请求的人员的信息。日志条目中的用户身份信息有助于确定请求是通过根或 IAM 用户凭证发出，或是通过某个角色或联合用户的临时安全凭证发出，还是由其他 AWS 服务发出。有关更多信息，请参阅 [CloudTrail userIdentity 元素](#)。

日志文件可以在存储桶中存储任意长时间，不过您也可以定义 Amazon S3 生命周期规则以自动存档或删除日志文件。默认情况下，将使用 Amazon S3 服务器端加密 (SSE) 对日志文件进行加密。

如果需要针对日志传输快速采取措施，可选择让 CloudTrail 在传输新日志文件时发布 Amazon SNS 通知。有关更多信息，请参阅 AWS CloudTrail 用户指南中的 [为 CloudTrail 配置 Amazon SNS 通知](#)。

您还可以将多个 AWS 区域和多个 AWS 账户中的 AWS Certificate Manager 日志文件聚合到单个 Amazon S3 存储桶中。有关更多信息，请参阅 [接收来自多个区域的 CloudTrail 日志文件](#) 和 [从多个账户中接收 CloudTrail 日志文件](#)。

CloudTrail 日志文件可包含一个或多个日志条目，每个条目列出多个 JSON 格式的事件。一个日志条目表示来自任何源的一个请求，包括有关所请求的操作、操作的日期和时间、请求参数等方面的信息。日志条目不一定具有任何特定顺序。也就是说，它们不是公用 API 调用的有序跟踪。有关组成日志条目的字段的更多信息，请参阅 [CloudTrail 事件参考](#)。

可能的 ACM CloudTrail 条目的示例，请参阅以下主题。

主题

- [向证书添加标签 \(p. 58\)](#)
- [删除证书 \(p. 59\)](#)
- [描述证书 \(p. 59\)](#)
- [导出证书 \(p. 60\)](#)
- [导入证书 \(p. 61\)](#)
- [列出证书 \(p. 62\)](#)
- [列出证书的标签 \(p. 63\)](#)
- [从证书中删除标签 \(p. 63\)](#)
- [请求证书 \(p. 64\)](#)
- [重新发送验证电子邮件 \(p. 65\)](#)
- [检索证书 \(p. 65\)](#)

向证书添加标签

以下 CloudTrail 示例显示调用 [AddTagsToCertificate](#) API 的结果。

```
{
  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/12345678-1234-1234-1234-123456789012"
    },
    responseElements: null,
    requestID: "ffd7ddb1b-fbfe-11e5-ba7b-5f4e988901f9",
    eventID: "4e7b10bb-7010-4e60-8376-0cac3bc860a5",
    eventType: "AwsApiCall",
    recipientAccountId: "123456789012"
  }]
}
```


删除证书

以下 CloudTrail 示例显示调用 [DeleteCertificate](#) API 的结果。

```
{
  "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/12345678-1234-1234-1234-123456789012"
    },
    "responseElements": null,
    "requestID": "6b0f5bb9-ec9c-11e5-a28b-51e7e3169e0f",
    "eventID": "08f18f8a-a827-4924-b864-afaf98517793",
    "eventType": "AwsApiCall",
    "recipientAccountId": "123456789012"
  }]
}
```

描述证书

以下 CloudTrail 示例显示调用 [DescribeCertificate](#) API 的结果。

Note

`DescribeCertificate` 操作的 CloudTrail 日志不会显示您指定的 ACM 证书的相关信息。您可以使用控制台、AWS Command Line Interface 或 [DescribeCertificate](#) API 查看有关证书的信息。

```
{
  "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/12345678-1234-1234-1234-123456789012"
      },
      "responseElements": null,
      "requestID": "74b91d83-ec9c-11e5-ac34-d1e4dfe1a11b",
      "eventID": "7779b6da-75c2-4994-b8c1-af3ad47b518a",
      "eventType": "AwsApiCall",
      "recipientAccountId": "123456789012"
    }
  ]
}
```

导出证书

以下 CloudTrail 示例显示调用 [ExportCertificate](#) API 的结果。

```
{
  "Records": [{
    "version": "0",
    "id": "12345678-1234-1234-1234-123456789012",
    "detail-type": "AWS API Call via CloudTrail",
    "source": "aws.acm",
    "account": "123456789012",
    "time": "2018-05-24T15:28:11Z",
    "region": "us-east-1",
    "resources": [],
    "detail": {
      "eventVersion": "1.04",
      "userIdentity": {
        "type": "Root",
        "principalId": "123456789012",
        "arn": "arn:aws:iam::123456789012:user/Alice",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "Alice"
      },
      "eventTime": "2018-05-24T15:28:11Z",
      "eventSource": "acm.amazonaws.com",
      "eventName": "ExportCertificate",
      "awsRegion": "us-east-1",
      "sourceIPAddress": "192.0.2.0",
      "userAgent": "aws-cli/1.15.4 Python/2.7.9 Windows/8 botocore/1.10.4",
      "requestParameters": {
        "passphrase": {
          "hb": [42,
            42,
            42,
            42,
            42,
            42,
            42,
            42,
            42],
          "offset": 0,
          "isReadOnly": false,
          "bigEndian": true,
          "nativeByteOrder": false,
          "mark": -1,
          "position": 0,
          "limit": 10,
          "capacity": 10,
          "address": 0
        }
      }
    }
  ]
}
```

```
    "certificateArn": "arn:aws:acm:us-east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
  },
  "responseElements": {
    "certificateChain": "-----BEGIN CERTIFICATE----- base64 certificate -----END CERTIFICATE-----\n"
    "-----BEGIN CERTIFICATE----- base64 certificate -----END CERTIFICATE-----\n",
    "privateKey": "*****",
    "certificate": "-----BEGIN CERTIFICATE----- base64 certificate -----END CERTIFICATE-----\n"
  },
  "requestID": "11802113-5f67-11e8-bc6b-d93a70b3bedf",
  "eventID": "5b66558e-27c5-43b0-9b3a-10f28c527453",
  "eventType": "AwsApiCall"
}
}]
```

导入证书

以下示例说明记录对 ACM [ImportCertificate](#) API 操作的调用的 CloudTrail 日志条目。

```
{
  "eventVersion": "1.04",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
    "arn": "arn:aws:iam::111122223333:user/Alice",
    "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "userName": "Alice"
  },
  "eventTime": "2016-10-04T16:01:30Z",
  "eventSource": "acm.amazonaws.com",
  "eventName": "ImportCertificate",
  "awsRegion": "ap-southeast-2",
  "sourceIPAddress": "54.240.193.129",
  "userAgent": "Coral/Netty",
  "requestParameters": {
    "privateKey": {
      "hb": [
        byte,
        byte,
        byte,
        ...
      ],
      "offset": 0,
      "isReadOnly": false,
      "bigEndian": true,
      "nativeByteOrder": false,
      "mark": -1,
      "position": 0,
      "limit": 1674,
      "capacity": 1674,
      "address": 0
    },
    "certificateChain": {
      "hb": [
        byte,
        byte,
        byte,
        ...
      ],
      "offset": 0,
```

```
    "isReadOnly": false,
    "bigEndian": true,
    "nativeByteOrder": false,
    "mark": -1,
    "position": 0,
    "limit": 2105,
    "capacity": 2105,
    "address": 0
  },
  "certificate": {
    "hb": [
      byte,
      byte,
      byte,
      ...
    ],
    "offset": 0,
    "isReadOnly": false,
    "bigEndian": true,
    "nativeByteOrder": false,
    "mark": -1,
    "position": 0,
    "limit": 2503,
    "capacity": 2503,
    "address": 0
  }
},
"responseElements": {
  "certificateArn": "arn:aws:acm:ap-southeast-2:111122223333:certificate/6ae06649-
ea82-4b58-90ee-dc05870d7e99"
},
"requestID": "cf1f3db7-8a4b-11e6-88c8-196af94bb7be",
"eventID": "fb443118-bfaa-4c90-95c1-beef21e07f8e",
"eventType": "AwsApiCall",
"recipientAccountId": "111122223333"
}
```

列出证书

以下 CloudTrail 示例显示调用 [ListCertificates](#) API 的结果。

Note

针对 `ListCertificates` 操作的 CloudTrail 日志不会显示您的 ACM 证书。您可以使用控制台、AWS Command Line Interface、或 [ListCertificates](#) API 查看证书列表。

```
{
  "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": "cdfel051-88aa-4aa3-8c33-a325270bff21",
    "eventType": "AwsApiCall",
    "recipientAccountId": "123456789012"
  }
}
```

列出证书的标签

以下 CloudTrail 示例显示调用 [ListTagsForCertificate](#) API 的结果。

Note

有关 [ListTagsForCertificate](#) 操作的 CloudTrail 日志不会显示您的标签。您可以使用控制台、AWS Command Line Interface 或 [ListTagsForCertificate](#) API 来查看标签列表。

```
{
  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"
  }
]}
```

从证书中删除标签

以下 CloudTrail 示例显示调用 [RemoveTagsFromCertificate](#) API 的结果。

```
{
  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: "0cf142e-ef74-4b21-9515-47197780c424",
  eventType: "AwsApiCall",
  recipientAccountId: "123456789012"
}]
}
```

请求证书

以下 CloudTrail 示例显示调用 [RequestCertificate](#) API 的结果。

```
{
  "Records": [{
    "eventVersion": "1.04",
    "userIdentity": {
      "type": "IAMUser",
      "principalId": "AIDACKCEVSQ6C2EXAMPLE",
      "arn": "arn:aws:iam::123456789012:user/Alice",
      "accountId": "123456789012",
      "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
      "userName": "Alice"
    },
    "eventTime": "2016-03-18T00:00:49Z",
    "eventSource": "acm.amazonaws.com",
    "eventName": "RequestCertificate",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "aws-cli/1.9.15",
    "requestParameters": {
      "subjectAlternativeNames": ["example.net"],
      "domainName": "example.com",
      "domainValidationOptions": [{
        "domainName": "example.com",
        "validationDomain": "example.com"
      }],
      {
        "domainName": "example.net",
        "validationDomain": "example.net"
      }
    ],
    "idempotencyToken": "8186023d89681c3ad5"
  }]
```

```
    },
    "responseElements": {
      "certificateArn": "arn:aws:acm:us-east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
    },
    "requestID": "77dacef3-ec9c-11e5-ac34-d1e4dfe1a11b",
    "eventID": "a4954cdb-8f38-44c7-8927-a38ad4be3ac8",
    "eventType": "AwsApiCall",
    "recipientAccountId": "123456789012"
  }]
}
```

重新发送验证电子邮件

以下 CloudTrail 示例显示调用 [ResendValidationEmail](#) API 的结果。

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

检索证书

以下 CloudTrail 示例显示调用 [GetCertificate](#) API 的结果。

```
{
  "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"
    }
}
```

记录 ACM 相关的 API 调用

您可以使用 CloudTrail 审核与 ACM 集成的服务发出的 API 调用。有关使用 CloudTrail 的更多信息，请参阅 [AWS CloudTrail 用户指南](#)。下面的示例显示了可生成的日志的类型，具体取决于您用于预置 ACM 证书的 AWS 资源。

主题

- [创建负载均衡器 \(p. 66\)](#)
- [使用负载均衡器注册 Amazon EC2 实例 \(p. 67\)](#)
- [加密私有密钥 \(p. 68\)](#)
- [解密私有密钥 \(p. 68\)](#)

创建负载均衡器

以下示例演示名为 Alice 的 IAM 用户对 `CreateLoadBalancer` 函数的调用。负载均衡器的名称是 `TestLinuxDefault`，而且侦听器是使用 ACM 证书创建的。

```
{
  "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"
}
```

使用负载均衡器注册 Amazon EC2 实例

当您在某个 Amazon Elastic Compute Cloud (Amazon EC2) 实例上预置您的网站或应用程序时，负载均衡器必须了解该实例。这可以通过 Elastic Load Balancing 控制台或 AWS Command Line Interface 来完成。以下示例显示了对 AWS 账户 123456789012 上名为 LinuxTest 的负载均衡器的 RegisterInstancesWithLoadBalancer 的调用。

```
{
  "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"
}
```

加密私有密钥

以下示例说明用于加密与 ACM 证书关联的私有密钥的 `Encrypt` 调用。加密是在 AWS 中执行的。

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

解密私有密钥

以下示例显示了用于对与 ACM 证书关联的私有密钥进行解密的 `Decrypt` 调用。解密将在 AWS 中进行，并且解密的密钥绝不会离开 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"
}
```

使用 ACM API

您可使用 AWS Certificate Manager API 通过发送 HTTP 请求以编程方式与该服务进行交互。有关更多信息，请参阅 [AWS Certificate Manager API 参考](#)。

除 Web API (或 HTTP API) 以外，您还可以使用 AWS 开发工具包和命令行工具来与 ACM 及其他服务进行交互。有关更多信息，请参阅[适用于 Amazon Web Services 的工具](#)。

以下主题向您说明如何使用某个 AWS 开发工具包 ([AWS SDK for Java](#)) 来在 AWS Certificate Manager API 中执行一些可用操作。

主题

- [向证书添加标签 \(p. 70\)](#)
- [删除证书 \(p. 72\)](#)
- [描述证书 \(p. 73\)](#)
- [导出证书 \(p. 75\)](#)
- [检索证书和证书链 \(p. 77\)](#)
- [导入证书 \(p. 79\)](#)
- [列出证书 \(p. 81\)](#)
- [列出证书标签 \(p. 82\)](#)
- [删除证书的标签 \(p. 84\)](#)
- [请求证书 \(p. 85\)](#)
- [重新发送验证电子邮件 \(p. 87\)](#)

向证书添加标签

以下示例说明如何使用 [AddTagsToCertificate](#) 函数。

```
package com.amazonaws.samples;

import com.amazonaws.services.certificatemanager.AWSCertificateManagerClientBuilder;
import com.amazonaws.services.certificatemanager.AWSCertificateManager;
import com.amazonaws.services.certificatemanager.model.AddTagsToCertificateRequest;
import com.amazonaws.services.certificatemanager.model.AddTagsToCertificateResult;
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.services.certificatemanager.model.TooManyTagsException;

import com.amazonaws.AmazonClientException;

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

import java.util.ArrayList;
```

```
/**
 * This sample demonstrates how to use the AddTagsToCertificate function in the AWS
 * Certificate
 * Manager service.
 *
 * Input parameters:
 *   CertificateArn - The ARN of the certificate to which to add one or more tags.
 *   Tags - An array of Tag objects to add.
 */

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 tags.
        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 and specify the ARN of the certificate.
        AddTagsToCertificateRequest req = new AddTagsToCertificateRequest();

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

        // Add tags to the specified certificate.
        AddTagsToCertificateResult result = null;
        try {
            result = client.addTagsToCertificate(req);
        }
        catch (InvalidArnException ex)
        {
            throw ex;
        }
        catch (InvalidTagException ex)
        {
            throw ex;
        }
    }
}
```

```
        catch(ResourceNotFoundException ex)
        {
            throw ex;
        }
        catch(TooManyTagsException ex)
        {
            throw ex;
        }

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

删除证书

以下示例说明如何使用 [DeleteCertificate](#) 函数。如果成功，则该函数将返回一个空集 {}。

```
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.AWSSStaticCredentialsProvider;
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);
}
}
```

描述证书

以下示例说明如何使用 [DescribeCertificate](#) 函数。

```
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，上述示例将显示类似于以下内容的信息。

```
{
  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,  
    }  
}
```

导出证书

以下示例演示如何使用 [ExportCertificate](#) 函数。该函数将导出由私有证书颁发机构 (CA) 颁发的私有证书 (PKCS #8 格式)。它还会导出证书链和私有密钥。在此示例中，密钥的密码存储在本地文件中。

```
package com.amazonaws.samples;  
  
import com.amazonaws.AmazonClientException;  
  
import com.amazonaws.auth.profile.ProfileCredentialsProvider;  
import com.amazonaws.auth.AWSSStaticCredentialsProvider;  
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();
}
}
```

检索证书和证书链

以下示例演示如何使用 [GetCertificate](#) 函数。

```
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.AWSSStaticCredentialsProvider;
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);
}
```

前面的示例将创建类似于以下内容的输出。

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

导入证书

以下示例演示如何使用 `ImportCertificate` 函数。

```
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.AWSSStaticCredentialsProvider;
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:
 * CertificateArn - 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 AWSSStaticCredentialsProvider(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);
    }
}
```

列出证书

以下示例演示如何使用 [ListCertificates](#) 函数。

```
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.AWSSessionCredentialsProvider;
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);
}
}
```

前面的示例将创建类似于以下内容的输出。

```
{
  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
  }
]
```

列出证书标签

以下示例说明如何使用 `ListTagsForCertificate` 函数。


```
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.AWSSStaticCredentialsProvider;
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 AWSSStaticCredentialsProvider(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);
    }
}
```

前面的示例将创建类似于以下内容的输出。

```
{Tags: [{Key: Purpose,Value: Test}, {Key: Short_Name,Value: My_Cert}]}
```

删除证书的标签

以下示例说明如何使用 `RemoveTagsFromCertificate` 函数。

```
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.AWSSStaticCredentialsProvider;
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);
}
```

请求证书

以下示例说明如何使用 `RequestCertificate` 函数。

```
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);
}
}
```

前面的示例将创建类似于以下内容的输出。

```
{CertificateArn:
  arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012}
```

重新发送验证电子邮件

以下示例向您说明如何使用 [ResendValidationEmail](#) 函数。

```
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());
}
}
```

前面的示例将重新发送验证电子邮件并显示一个空集。

```
{}
```

ACM 私有密钥安全

当您请求公有证书 ([p. 18](#)) 时，AWS Certificate Manager (ACM) 会生成一个公有/私有密钥对。对于导入的证书 ([p. 41](#))，您可生成密钥对。公有密钥将成为证书的一部分。ACM 将存储证书及其相应的私有密钥，并使用 AWS Key Management Service (AWS KMS) 来帮助保护私有密钥。该过程的工作方式如下所示：

1. 首次在 AWS 区域中请求或导入证书时，ACM 会在 AWS KMS 中创建一个别名为 aws/acm 的 AWS 托管客户主密钥 (CMK)。此 CMK 在每个 AWS 账户和每个 AWS 区域中都是唯一的。
2. ACM 使用此 CMK 加密证书的私有密钥。ACM 仅存储加密版的私有密钥 (ACM 不会以纯文本格式存储私有密钥)。ACM 使用同一 CMK 在特定的 AWS 账户和特定的 AWS 区域中加密所有证书的私有密钥。
3. 将证书关联到与 AWS Certificate Manager 集成的服务时，ACM 会将证书和加密的私有密钥发送给服务。您还可以在 AWS KMS 中隐式创建授权，从而允许该服务使用 AWS KMS 中的 CMK 来解密该证书的私有密钥。有关授予的更多信息，请参阅 AWS Key Management Service Developer Guide 中的 [使用授予](#)。有关 ACM 支持的服务的更多信息，请参阅 [与 AWS Certificate Manager 集成的服务](#) ([p. 8](#))。
4. 集成服务使用 AWS KMS 中的 CMK 来解密私有密钥。然后，该服务使用证书和解密的 (纯文本) 私有密钥建立与其客户端之间的安全通信通道 (SSL/TLS 会话)。
5. 当证书与集成服务取消关联时，在步骤 3 中创建的授权将被停用。这意味着，该服务不再可以使用 AWS KMS 中的 CMK 来解密证书的私有密钥。

故障排除

如果在使用 AWS Certificate Manager 时遇到问题，请参阅以下主题。

主题

- [排查认证机构授权 \(CAA\) 问题 \(p. 90\)](#)
- [排查电子邮件问题 \(p. 90\)](#)
- [排查证书导入问题 \(p. 92\)](#)
- [排查证书固定问题 \(p. 93\)](#)
- [排查证书请求问题 \(p. 93\)](#)
- [解决托管证书续订问题 \(p. 95\)](#)
- [解决证书验证问题 \(p. 95\)](#)
- [排查 .IO 域问题 \(p. 96\)](#)
- [排查 API 网关 问题 \(p. 96\)](#)

排查认证机构授权 (CAA) 问题

您可以使用 CAA DNS 记录指定 Amazon 证书颁发机构 (CA) 可以为您的域或子域颁发 ACM 证书。如果您在证书颁发期间收到一条错误，显示 One or more domain names have failed validation due to a Certification Authority Authentication (CAA) error，请检查您的 CAA DNS 记录。如果您在成功验证 ACM 证书请求后收到此错误，则必须更新您的 CAA 记录并再次请求证书。您的至少一个 CAA 记录中的值字段必须包含以下域名称之一：

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

如果您不希望 ACM 执行 CAA 检查，请勿对您的域配置 CAA 记录或将您的 CAA 记录留空。有关创建 CAA 记录的更多信息，请参阅 [\(可选\) 配置 CAA 记录 \(p. 16\)](#)。

排查电子邮件问题

如果您遇到验证电子邮件方面的问题，请参考以下主题。

主题

- [未收到验证电子邮件 \(p. 91\)](#)
- [已发送到子域的电子邮件 \(p. 92\)](#)
- [隐藏的联系人信息 \(p. 92\)](#)
- [证书续订 \(p. 92\)](#)
- [WHOIS 限制 \(p. 92\)](#)

未收到验证电子邮件

当您从 ACM 请求证书并选择电子邮件验证时，域验证电子邮件会发送到 WHOIS 中指定的三个联系人地址以及五个常用管理地址。有关更多信息，请参阅 [使用电子邮件验证域所有权 \(p. 25\)](#)。如果您在接收验证电子邮件时遇到问题，请查看以下建议。

查找电子邮件的位置

验证电子邮件将发送到 WHOIS 中列出的联系人地址以及域的常用管理地址。电子邮件不会发送到 AWS 账户所有者，除非该所有者已作为域联系人也在 WHOIS 中列出。检查在 ACM 控制台中显示 (或者从 CLI 或 API 中返回) 的电子邮件列表，以确定您应查找验证电子邮件的位置。要查看此列表，请单击标有 Validation not complete 的框中域名旁的图标。

此电子邮件已标记为垃圾邮件

请查看您的垃圾邮件文件夹中是否有验证电子邮件。

GMail 会自动对您的电子邮件进行分类

如果您使用的是 GMail，则验证电子邮件可能已被自动分类到 Updates 或 Promotions 选项卡中。

域注册者未显示联系人信息或已启用隐私保护

在某些情况下，WHOIS 中的域注册者、技术人员和管理联系人可能不会公开，因此 AWS 无法与他们联系。您可以自行决定选择将您的注册者配置为在 WHOIS 中列出您的电子邮件地址，尽管并非所有注册者都支持此选项。您可能需要在域的注册表中直接进行更改。在其他情况下，域联系人信息可能使用的是私密地址 (例如，通过 WhoisGuard 或 PrivacyGuard 提供的地址)。

对于从 Route 53 购买的域，默认情况下已启用隐私保护，而且您的电子邮件地址已映射到 whoisprivacyservice.org 或 contact.gandi.net 电子邮件地址。确保您的域注册者文件上的注册者电子邮件是最新的，以便发送到这些隐藏的电子邮件地址的电子邮件可转发到您控制的电子邮件地址。

Note

即使您选择公开您的联系人信息，您通过 Route 53 购买的一些域的隐私保护也将被启用。例如，Route 53 无法以编程方式禁用 .ca 顶级域的隐私保护。您必须联系 [AWS 支持中心](#) 并请求禁用隐私保护。

如果无法通过 WHOIS 获得您的域的电子邮件联系人信息，或者域所有者或授权代表未收到发送到联系人信息的电子邮件，建议将您的域或子域配置为接收发送到一个或多个常用管理地址 (通过在请求的域名前面加上 admin@、administrator@、hostmaster@、webmaster@ 和 postmaster@ 来构成) 的电子邮件。有关为您的域配置电子邮件的更多信息，请参阅您的电子邮件服务提供商的文档并遵循 [\(可选\) 为域配置电子邮件 \(p. 15\)](#) 处的说明。如果您使用的是 Amazon WorkMail，请参阅《Amazon WorkMail 管理员指南》中的[处理用户](#)。

在提供 AWS 将验证电子邮件发送到的八个电子邮件地址中的至少一个地址并确认您可以收到该地址的电子邮件后，您便已准备通过 ACM 请求证书。在提交证书请求后，确保预期的电子邮件地址显示在 AWS 管理控制台中的电子邮件地址列表中。在证书处于 Pending validation 状态时，您可以通过单击标有 Validation not complete 的框中域名旁的图标来展开此列表以进行查看。您还可以查看 ACM Request a Certificate 向导的 Step 3: Validate 中的列表。列出的电子邮件地址是将电子邮件发送到的地址。

MX 记录缺失或配置错误

MX 记录是域名系统 (DNS) 数据库中的资源记录，它为您的域指定一个或多个接受电子邮件的邮件服务器。如果 MX 记录缺失或配置错误，则无法将电子邮件发送到 [使用电子邮件验证域所有权 \(p. 25\)](#) 中指定的五个常用系统管理地址中的任何一个。请修复缺失或配置错误的 MX 记录，然后尝试重新发送电子邮件或请求证书。

Note

目前，建议至少等待一小时，然后再尝试重新发送电子邮件或请求您的证书。

Note

要绕过对 MX 记录的依赖，可以使用 [RequestCertificate](#) API 或 [request-certificate](#) AWS CLI 命令中的 `ValidationDomain` 选项指定 ACM 要将验证电子邮件发送到的域名。如果您使用 API 或 AWS CLI，AWS 不会执行 MX 查找。

联系支持中心

如果在查看上述准则后，您仍收不到域验证电子邮件，请访问 [AWS Support 中心](#) 并创建案例。如果您没有支持协议，请将消息发布到 [ACM 开发论坛](#)。

已发送到子域的电子邮件

如果您使用控制台并为子域名称 (如 `sub.test.example.com`) 请求证书，则 ACM 会进行检查，查看 `sub.test.example.com` 是否存在 MX 记录。如果不存在，则检查父域 `test.example.com`，依此类推，直至基础域 `example.com`。如果找到了 MX 记录，则搜索将停止，并且验证电子邮件将发送到子域的常用管理地址。例如，如果找到了 `test.example.com` 的 MX 记录，则电子邮件将发送到 `admin@test.example.com`、`administrator@test.example.com` 以及 [使用电子邮件验证域所有权 \(p. 25\)](#) 中指定的其他管理地址。如果在任何子域中均未找到 MX 记录，则电子邮件将发送到您最初为其请求证书的子域。有关如何设置电子邮件以及 ACM 如何处理 DNS 和 WHOIS 数据库的全面讨论，请参阅 [\(可选\) 为域配置电子邮件 \(p. 15\)](#)。

您可以使用 [RequestCertificate](#) API 或 [request-certificate](#) AWS CLI 命令中的 `ValidationDomain` 选项来指定 ACM 要将验证电子邮件发送到的域名，而不使用控制台。如果您使用 API 或 AWS CLI，AWS 不会执行 MX 查找。

隐藏的联系信息

当您尝试创建新证书时，会发生一个常见问题。某些注册商允许您在 WHOIS 列表中隐藏联系人信息。其他注册商允许您将真实电子邮件地址替换为私密 (或代理) 地址。这将阻止您通过注册联系人地址接收验证电子邮件。

若要接收邮件，请确保您的联系人信息在 WHOIS 中是公开的，或者，如果您的 WHOIS 列表显示私密电子邮件地址，则确保发送到该私密地址的邮件将被转发到真实的电子邮件地址。WHOIS 设置完成后，只要您的证书请求尚未超时，您就可以选择重新发送验证电子邮件。ACM 将执行新的 WHOIS/MX 查找并将验证电子邮件发送到您现在的公开的电子邮件地址。

证书续订

如果您在请求新证书时将 WHOIS 信息公开，并在这之后将您的信息模糊化，那么在您尝试续订证书时，ACM 将无法检索您的注册联系人地址。ACM 会将验证电子邮件发送到这些联系人地址并使用由您的 MX 记录构成的五个常见管理地址。若要解决此问题，请再次公开您的 WHOIS 信息并重新发送验证电子邮件。ACM 将执行新的 WHOIS/MX 查找并将验证电子邮件发送到您现在的公开的电子邮件地址。

WHOIS 限制

有时，即使您发送了多个验证电子邮件请求，ACM 也无法联系 WHOIS 服务器。此问题出在 AWS 之外。也就是说，AWS 不会控制 WHOIS 服务器，也无法阻止 WHOIS 服务器限制。如果您遇到此问题，请在 [AWS Support 中心](#) 创建一个案例以寻求解决方法。

排查证书导入问题

您可以将第三方证书导入 ACM 并将其与 [集成服务](#) 关联。如果遇到问题，请查看 [先决条件](#) 和 [证书格式](#) 主题。特别要注意以下几点：

- 您只能导入 X.509 版本 3 SSL/TLS 证书。
- 证书可以是自签名的，也可以由证书颁发机构 (CA) 签名。
- 如果证书由 CA 签名，则必须包含一直串联到颁证机构根证书的证书链。
- 不要将证书包含在证书链中。
- 链中的每个证书都必须直接认证前一个证书。
- 证书、私有密钥和证书链必须采用 PEM 编码。
- 不得加密您的私有密钥。
- 与 ACM [集成](#) 的服务仅允许将其支持的算法和密钥大小与其资源关联。支持可能会有所变化。请参阅各项服务的文档以确保证书能够正常工作。
- 集成服务对证书的支持可能因证书是否已导入 IAM 或 ACM 而有所不同。
- 导入时证书必须有效。
- 所有证书的详细信息都显示在控制台中。但是，默认情况下，如果调用 [ListCertificates](#) API 或 [list-certificates](#) AWS CLI 命令而不指定 `keyTypes` 筛选条件，则只显示 RSA_1024 或 RSA_2048 证书。

排查证书固定问题

为了续订证书，ACM 会生成新的公有-私有密钥对。如果您的应用程序使用 [证书固定](#) (p. 11)(有时称为 SSL 固定) 来固定 ACM 证书，则在 AWS 续订证书后，应用程序可能无法连接到您的域。为此，我们建议您不要固定 ACM 证书。如果您的应用程序必须固定证书，您可以执行以下操作：

- 将您的证书导入 ACM (p. 41)，然后将您的应用程序固定到导入的证书。ACM 不提供针对导入的证书的托管续订。
- 将您的应用程序固定到 [Amazon 根证书](#)。

排查证书请求问题

如果您在请求 ACM 证书时遇到问题，请参阅以下主题。

主题

- [证书请求超时](#) (p. 93)
- [证书请求失败](#) (p. 93)

证书请求超时

如果对 ACM 证书的请求在 72 小时内未进行验证，则此请求将超时。要纠正此情况，请删除您的请求并选择 [Request a certificate](#) 以重新开始。您可以使用 DNS 验证或电子邮件验证声明您拥有或可以控制请求中列出的域。建议您使用 DNS 验证。有关更多信息，请参阅[使用 DNS 验证域所有权](#) (p. 22)。

证书请求失败

对 ACM 证书的请求可能失败。如果发生这种情况，以下说明可帮助您理解请求失败的原因并建议解决此问题所需采取的步骤。

失败原因

- [无可联系人](#) (p. 94)
- [域不被允许](#) (p. 94)

- [所需的其他验证](#) (p. 94)
- [公共域无效](#) (p. 94)
- [Other](#) (p. 95)

无可用联系人

您在请求证书时选择了电子邮件验证，但 ACM 找不到用于验证证书请求中的一个或多个域名的电子邮件地址。要解决此问题，您可以执行下列操作之一：

- 确保您有一个有效的电子邮件地址，该地址已在 WHOIS 中登记且在对证书请求中的域名执行标准 WHOIS 查找时可见。通常，您可以通过域注册者执行此操作。
- 确保您的域已配置为接收电子邮件。您的域的名称服务器必须有一个邮件交换器记录 (MX 记录)，以便 ACM 的电子邮件服务器知道将[域验证电子邮件](#) (p. 25) 发送到的位置。

完成上述任务之一便足以解决此问题；您无需同时执行两项任务。在解决此问题后，请求一个新证书。您无法重新提交失败的证书请求。

有关如何确保收到来自 ACM 的域验证电子邮件的更多信息，请参阅[\(可选\) 为域配置电子邮件](#) (p. 15) 或 [未收到验证电子邮件](#) (p. 91)。如果您遵循这些步骤并继续收到 No Available Contacts 消息，请将[此情况报告给 AWS](#)，以便我们可以进行调查。

域不被允许

ACM 不允许您为指定的一个或多个域名请求证书。通常，这是因为在不安全网站的 Google Safe Browsing 列表或有效的网络钓鱼的 PhishTank 列表中发现了证书请求中的一个或多个域名。要解决此问题，您可以执行下列操作：

- 在 [Google Safe Browsing Site Status](#) 网站中搜索您的域名。如果您的域被视为不安全，请参阅 [Google Help for Hacked Websites](#) 以了解可执行的操作。如果您认为您的域是安全的，请参阅[请求检查](#)以请求来自 Google 的检查。
- 在 [PhishTank 主页](#)上搜索您的域名。如果您的域被视为网络钓鱼，请参阅 [Google Help for Hacked Websites](#) 或 [StopBadware Webmaster Help](#) 以了解可执行的操作。如果您认为您的域是安全的，请参阅 [PhishTank 常见问题](#)以获取有关如何报告误报的信息。

在解决此问题后，请求一个新证书。您无法重新提交失败的证书请求。

所需的其他验证

ACM 需要其他信息来处理此证书请求。要提供此信息，请使用[支持中心](#)来联系 AWS Support。如果您没有支持计划，请在 [AWS Certificate Manager 开发论坛](#)中发布新话题。

Note

您无法为 Amazon 拥有的域名 (例如以 amazonaws.com、cloudfront.net 或 elasticbeanstalk.com 结尾的域名) 请求证书。

公共域无效

证书请求中的一个或多个域名无效。通常，这是因为请求中的域名不是有效的顶级域。尝试再次请求证书，同时更正失败请求中的任何拼写错误或错别字，并确保请求中的所有域名适用于有效的顶级域。例如，您无法为 example.invalidpublicdomain 请求 ACM 证书，因为“invalidpublicdomain”不是有效的顶级域。如果您继续收到此失败原因，请联系[支持中心](#)。如果您没有支持计划，请在 [AWS Certificate Manager 开发论坛](#)中发布新话题。

Other

通常，此失败原因会在证书请求中的一个或多个域名出现拼写错误时出现。尝试再次请求证书，同时更正失败请求中的任何拼写错误或错别字。如果您继续收到此失败原因，请使用[支持中心](#)来联系 AWS Support。如果您没有支持计划，请在 [AWS Certificate Manager 开发论坛](#) 中发布新话题。

解决托管证书续订问题

ACM 在到期前会尝试自动续订 ACM 证书，以便您无需执行任何操作。如果对[适用于 ACM 的由 Amazon 颁发的证书的托管续订 \(p. 35\)](#)有任何疑问，请参阅以下主题。

主题

- [自动域验证 \(p. 95\)](#)
- [异步过程 \(p. 95\)](#)

自动域验证

要让 ACM 自动续订您的证书，必须满足以下条件：

- ACM 必须能够与证书中的每个域建立 HTTPS 连接。
- 对于每个连接，返回的证书必须与 ACM 要续订的证书匹配。
- 您的证书必须与 ACM 集成的某项 AWS 服务关联。
- ACM 必须能够验证证书中列出的每个域名。

要提高 ACM 可自动续订您的证书的可能性，请执行以下操作：

将证书与 AWS 资源结合使用

确保您的证书正在与支持的 AWS 资源结合使用。有关 ACM 支持的资源的信息，请参阅[与 AWS Certificate Manager 集成的服务 \(p. 8\)](#)。

将资源配置为接受来自 Internet 的 HTTPS 请求

确保将拥有您的 ACM 证书的 AWS 资源配置为接受来自 Internet 的 HTTPS 请求。

将 DNS 配置为将您的域名路由至托管您的 ACM 证书的资源

确保对您的证书中的域名发出的 HTTPS 请求已路由至具有您的证书的资源。

异步过程

[适用于 ACM 的由 Amazon 颁发的证书的托管续订 \(p. 35\)](#)是一个异步过程。这意味着这些步骤不会立即连续发生。在验证 ACM 证书中的所有域名后，在 ACM 获取新证书之前可能有一个延迟。在 ACM 获取续订的证书的时间与将证书部署到使用它的 AWS 资源的时间之间可能出现另一个延迟。因此，对证书状态的更改可能需要数小时才能显示在控制台中。

解决证书验证问题

如果您的验证似乎停滞在挂起状态，请参阅以下主题。

验证未完成

如果 ACM 证书请求状态为等待验证，说明此请求正等待您采取操作。如果您在提出请求时选择电子邮件验证，则您或授权代表必须回复验证电子邮件。这些邮件已发送到注册的 WHOIS 联系人地址以及所请求的域的其他常用电子邮件地址。有关更多信息，请参阅 [使用电子邮件验证域所有权 \(p. 25\)](#)。如果选择 DNS 验证，则必须将 ACM 为您创建的 CNAME 记录写入您的 DNS 数据库。有关更多信息，请参阅 [使用 DNS 验证域所有权 \(p. 22\)](#)。

Important

您必须验证自己拥有或可以控制证书请求中包含的所有域名。如果选择电子邮件验证，您将收到每个域的验证电子邮件。如果未收到电子邮件，请参阅[未收到验证电子邮件 \(p. 91\)](#)。如果选择 DNS 验证，则必须为每个域创建一条 CNAME 记录。

建议使用 DNS 验证而不是电子邮件验证。

排查 .IO 域问题

.IO 域已分配给英属印度洋领地。目前，域注册表不会显示 WHOIS 数据库中您的公有信息。无论您是启用还是禁用域的隐私保护，都是如此。当执行 WHOIS 查找时，仅返回模糊注册商信息。因此，ACM 无法向 WHOIS 中通常提供的以下三个注册联系人地址发送验证电子邮件。

- 域注册者
- 技术联系人
- 管理联系人

但是，ACM 确实将验证电子邮件发送到以下五个常见系统地址，其中 *your_domain* 是您最初请求证书时输入的域名，而 *.io* 是顶级域。

- administrator@*your_domain*.io
- hostmaster@*your_domain*.io
- postmaster@*your_domain*.io
- webmaster@*your_domain*.io
- admin@*your_domain*.io

要接收 .IO 域的验证邮件，请确保启用了上述五个电子邮件账户之一。如果没有启用，您不会收到验证电子邮件，并且不会向您颁发 ACM 证书。

Note

建议使用 DNS 验证而不是电子邮件验证。有关更多信息，请参阅 [使用 DNS 验证域所有权 \(p. 22\)](#)。

排查 API 网关 问题

当您部署边缘优化的 API 终端节点时，API 网关 将为您设置 CloudFront 分配。CloudFront 分配由 API 网关 而不是您的账户拥有。该分配绑定到部署 API 时所使用的 ACM 证书。要删除绑定并允许 ACM 删除您的证书，您必须删除与该证书关联的 API 网关 自定义域。

您部署区域 API 终端节点时，API 网关 将代表您创建一个应用程序负载均衡器 (ALB)。该负载均衡器由 API 网关 所有，对您不可见。该 ALB 绑定到部署 API 时所使用的 ACM 证书。要删除绑定并允许 ACM 删除您的证书，您必须删除与该证书关联的 API 网关 自定义域。

文档历史记录

下表介绍了自 2018 年起的 AWS Certificate Manager 文档发布历史记录。

update-history-change	update-history-description	update-history-date
新增内容 (p. 97)	增加了默认将 ACM 公有证书发布到证书透明度日志的功能。	April 24, 2018
新增服务扩展 (p. 97)	发布了 ACM Private Certificate Manager (CM) 和 AWS Certificate Manager 扩展，允许用户构建用于颁发和撤销私有数字证书的安全托管基础架构。有关更多信息，请参阅 AWS 私有证书颁发机构 。	April 4, 2018
新增内容 (p. 97)	在最佳实践中增加了证书透明度日志记录。	March 27, 2018

下表介绍了 2018 年以前的 AWS Certificate Manager 文档发布历史记录。

更改	描述	发行日期
新增内容	在 使用 DNS 验证域所有权 (p. 22) 中添加了 DNS 验证相关内容。	2017 年 11 月 21 日
新增内容	向 使用 ACM API (p. 70) 添加了新的 Java 代码示例。	2017 年 10 月 12 日
新增内容	向 (可选) 配置 CAA 记录 (p. 16) 添加了有关 CAA 记录的信息。	2017 年 9 月 21 日
新增内容	已将有关 .IO 域的信息添加到 故障排除 (p. 90) 中。	2017 年 07 月 7 日
新增内容	已将有关重新导入证书的信息添加到 重新导入证书 (p. 44) 中。	2017 年 07 月 7 日
新增内容	已将有关证书固定的信息添加到 最佳实践 (p. 10) 和 故障排除 (p. 90) 中。	2017 年 07 月 7 日
新增内容	已将 AWS CloudFormation 添加到 与 AWS Certificate Manager 集成的服务 (p. 8) 。	2017 年 5 月 27 日
更新	已将更多信息添加到 限制 (p. 9) 。	2017 年 5 月 27 日
新增内容	添加了有关 身份验证和访问控制 (p. 49) 的文档。	2017 年 4 月 28 日

更改	描述	发行日期
更新	添加了一个图形来显示验证电子邮件的发送地址。请参阅 使用电子邮件验证域所有权 (p. 25)。	2017 年 4 月 21 日
更新	添加了有关为您的域设置电子邮件的信息。请参阅 (可选) 为域配置电子邮件 (p. 15)。	2017 年 4 月 6 日
更新	添加了有关在控制台中检查证书续订状态的信息。请参阅 检查证书的续订状态 (p. 37)。	2017 年 3 月 28 日
更新	更新了有关使用 Elastic Load Balancing 的文档。	2017 年 3 月 21 日
新增内容	添加了对 AWS Elastic Beanstalk 和 Amazon API Gateway 的支持。请参阅 与 AWS Certificate Manager 集成的服务 (p. 8)。	2017 年 3 月 21 日
更新	更新了有关 托管续订 (p. 35)的文档。	2017 年 2 月 20 日
新增内容	添加了有关 导入证书 (p. 41)的文档。	2016 年 10 月 13 日
新增内容	添加了对 ACM 操作的 AWS CloudTrail 支持。请参阅 使用 AWS CloudTrail 记录 AWS Certificate Manager API 调用 (p. 57)。	2016 年 25 月 3 日
新指南	此版本引入了 AWS Certificate Manager。	2016 年 1 月 21 日