

---

# Amazon Certificate Manager

用户指南

版本 1.0

**亚马逊云科技**  


---

## Amazon Certificate Manager: 用户指南

Copyright © 2022 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon 的商标和商业外观不得用于任何非 Amazon 的商品或服务，也不得以任何可能引起客户混淆、贬低或诋毁 Amazon 的方式使用。所有非 Amazon 拥有的其它商标均为各自所有者的财产，这些所有者可能附属于 Amazon、与 Amazon 有关联或由 Amazon 赞助，也可能不是如此。

Amazon Web Services 文档中描述的 Amazon Web Services 服务或功能可能因区域而异。要查看适用于中国区域的差异，请参阅[中国的 Amazon Web Services 服务入门](#)。

## Table of Contents

什么是 Amazon Certificate Manager ? .....	1
ACM 服务是否适合我? .....	1
ACM 证书特征 .....	1
支持的区域 .....	3
集成服务 .....	3
网站签章和信任徽标 .....	6
配额 .....	6
常规配额 .....	6
API 速率配额 .....	7
定价 .....	8
安全性 .....	9
数据保护 .....	9
ACM 私有密钥安全性 .....	10
身份和访问管理 .....	10
身份验证 .....	10
访问控制 .....	11
有关管理访问的概述 .....	11
托管式策略 .....	13
客户管理型策略 .....	14
内联策略 .....	14
服务相关角色 .....	17
ACM API 权限参考 .....	19
故障恢复能力 .....	20
基础设施安全性 .....	20
最佳实践 .....	21
Amazon CloudFormation .....	21
证书固定 .....	21
域验证 .....	22
添加或删除域名 .....	22
选择退出证书透明度日志记录 .....	22
启用 Amazon CloudTrail。 .....	23
设置 .....	24
设置 Amazon 和 IAM .....	24
注册 Amazon .....	24
创建 IAM 用户 .....	24
注册域名 .....	25
(可选) 配置电子邮件 .....	25
WHOIS 数据库 .....	25
MX 记录 .....	26
(可选) 配置 CAA .....	26
颁发和管理证书 .....	28
请求公有证书 .....	28
使用控制台请求公有证书 .....	29
使用 CLI 请求公有证书 .....	30
验证域所有权 .....	30
DNS 验证 .....	30
电子邮件验证 .....	33
列出证书 .....	35
描述证书 .....	36
删除证书 .....	39
安装 ACM 证书 .....	39
托管续订 .....	40
公开信任的证书 .....	40
DNS 验证 .....	41

电子邮件验证 .....	41
私有 PKI 证书 .....	43
自动导出已续订的证书 .....	43
测试托管式续订 .....	45
检查续订状态 .....	45
检查状态 (控制台) .....	46
检查状态 (API) .....	46
检查状态 (CLI) .....	46
使用 Personal Health Dashboard (PHD) 检查状态 .....	47
导入证书 .....	48
先决条件 .....	48
凭证格式 .....	49
导入证书 .....	50
导入 (控制台) .....	50
导入 (Amazon CLI) .....	51
重新导入证书 .....	51
重新导入 (控制台) .....	52
重新导入 (Amazon CLI) .....	52
导出证书 .....	53
导出私有证书 (控制台) .....	53
导出私有证书 (CLI) .....	53
标记 ACM 证书 .....	55
标签限制 .....	55
管理标签 .....	55
管理标签 (控制台) .....	56
管理标签 (CLI) .....	57
管理标签 .....	57
监控和日志记录 .....	58
CloudWatch 指标 .....	58
使用 CloudWatch Events .....	58
支持的事件 .....	59
操作示例 .....	60
使用 CloudTrail .....	66
支持的 API 操作 .....	67
集成服务的 API 调用 .....	76
使用 API (Java 示例) .....	80
AddTagsToCertificate .....	80
DeleteCertificate .....	81
DescribeCertificate .....	83
ExportCertificate .....	84
GetCertificate .....	86
ImportCertificate .....	88
ListCertificates .....	90
RenewCertificate .....	92
ListTagsForCertificate .....	93
RemoveTagsFromCertificate .....	95
RequestCertificate .....	96
ResendValidationEmail .....	98
问题排查 .....	100
证书请求 .....	100
请求超时 .....	100
请求失败 .....	100
证书验证 .....	101
DNS 验证 .....	102
电子邮件验证 .....	104
证书续订 .....	106
准备进行自动域验证 .....	107

---

处理托管证书续订失败 .....	107
其他问题 .....	111
CAA 记录 .....	111
证书导入 .....	111
证书固定 .....	112
API Gateway .....	112
意外故障 .....	112
ACM 服务关联角色 (SLR) 问题 .....	113
处理异常 .....	3
私有证书异常处理 .....	113
Concepts .....	115
ACM 证书 .....	115
ACM 根 CA .....	116
顶级域 .....	117
非对称密钥加密 .....	117
证书颁发机构 .....	117
证书透明度日志 .....	117
域名系统 .....	118
域名 .....	118
加密和解密 .....	119
完全限定域名 (FQDN) .....	119
公有密钥基础设施 .....	119
根证书 .....	119
安全套接字层 (SSL) .....	119
安全 HTTPS .....	119
SSL 服务器证书 .....	120
对称密钥加密 .....	120
传输层安全性 (TLS) .....	120
Trust .....	120
文档历史记录 .....	121

# 什么是 Amazon Certificate Manager？

Amazon Certificate Manager (ACM) 处理创建、存储和续订公有及私有 SSL/TLS X.509 证书和密钥的复杂操作，这些证书和密钥可保护您的 Amazon 网站和应用程序。您可以通过 ACM 签发证书，或者通过将第三方证书 [导入 \(p. 48\)](#) ACM 管理系统中，为 [集成 Amazon 服务 \(p. 3\)](#) 提供证书。ACM 证书可以保护单一域名、多个特定域名、通配符域或这些域的组合。ACM 通配符证书可以保护无限数量的子域。您还可以 [导出 \(p. 53\)](#) 由 ACM Private CA 签名的 ACM 证书，以便在内部 PKI 中的任何位置使用。

## ACM 服务是否适合我？

Amazon 为部署托管 X.509 证书的客户提供了两个选项。选择最能满足您需求的服务。

1. Amazon Certificate Manager (ACM) – 此服务面向需要使用 TLS 确保 Web 存在安全性的企业客户。ACM 证书通过 Elastic Load Balancing、Amazon CloudFront、Amazon API Gateway 和其他 [集成 Amazon 服务 \(p. 3\)](#) 进行部署。此类最常见的应用是一个需要大量流量的安全公共网站。ACM 还通过自动续订过期证书简化了安全管理。此服务正好适用于您。

[Concepts \(p. 115\)](#)

[ACM 证书特征 \(p. 1\)](#)

[支持的区域 \(p. 3\)](#)

[与 Amazon Certificate Manager 集成的服务 \(p. 3\)](#)

[网站签章和信任徽标 \(p. 6\)](#)

[API 速率配额 \(p. 7\)](#)

[最佳实践 \(p. 21\)](#)

[Amazon Certificate Manager 定价 \(p. 8\)](#)

## ACM 证书特征

ACM 提供的公有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 \(p. 48\)](#)。

### 域验证 (DV)

ACM 证书需要经过域验证。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 [电子邮件验证 \(p. 33\)](#) 和 [DNS 验证 \(p. 30\)](#)。

## 有效期

目前，ACM 证书的有效期为 13 个月（395 天）。

## 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、撤销或过期而导致的停机。有关更多信息，请参阅[ACM 证书的托管续订 \(p. 40\)](#)。

## 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其它域名。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 证书并添加域名 `example.com`。有关更多信息，请参阅[请求公有证书 \(p. 28\)](#)。

## 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，该证书可以保护同一个域中的多个站点。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` 可以保护 `login.example.com` 和 `test.example.com`，但不能保护 `test.login.example.com`。另请注意，`*.example.com` 仅保护 `example.com` 的子域，而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（括号中为 API 名称）：

- 1024 位 RSA (RSA\_1024)
- 2048 位 RSA (RSA\_2048)
- 3072 位 RSA (RSA\_3072)
- 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，这种支持因证书是导入到 IAM 还是 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[受支持的 SSL/TLS 协议和密码](#)。

## Punycode

必须满足以下与[国际化域名](#)有关的 [Punycode](#) 要求：

1. 以“<character><character>--”模式开头的域名必须与“xn--”一致。
2. 以“xn--”开头的域名也必须是有效的国际化域名。

#### Punycode 示例

域名	满足条件 1	满足条件 2	已允许	注意
example.com	不适用	不适用	✓	不是以“<character><character>--”开头
a--example.com	不适用	不适用	✓	不是以“<character><character>--”开头
abc--example.com	不适用	不适用	✓	不是以“<character><character>--”开头
xn--xyz.com	是	是	✓	有效的国际化域名 ( 解析为简.com )
xn--example.com	是	否	✗	不是有效的国际化域名
ab--example.com	否	否	✗	必须以“xn--”开头

#### 异常

请注意以下几点：

- ACM 不提供扩展验证 (EV) 证书或企业验证 (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。
- 对于 ACM 证书，ACM 目前不允许您退出[托管证书续订 \(p. 40\)](#)。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。
- 您不能在 Amazon Elastic Compute Cloud (Amazon EC2) 网站或应用程序上直接安装 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 Amazon Certificate Manager 集成的服务 \(p. 3\)](#)。

## 支持的区域

请访问 Amazon 一般参考中的[Amazon 区域和终端节点](#)，或查看[Amazon 区域列表](#)，了解 ACM 的区域可用性。

ACM 中的证书属于区域性资源。若要为多个 Amazon 区域中相同的完全限定域名 (FQDN) 或 FQDN 组将证书用于 Elastic Load Balancing，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。

要在 Amazon CloudFront 中使用 ACM 证书，请确保您在美国东部 (弗吉尼亚北部) 区域中请求 (或导入) 该证书。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 Amazon Certificate Manager 集成的服务

Amazon Certificate Manager 支持的 Amazon 服务越来越多。您无法直接在基于 Amazon 的网站或应用程序上安装 ACM 证书或私有 ACM Private CA 证书。

## Note

公有 ACM 证书可以安装在连接到 [Nitro Enclave \(p. 5\)](#) 的 Amazon EC2 实例上，但不能安装到其他 Amazon EC2 实例上。有关在未连接到 Nitro Enclave 的 Amazon EC2 实例上设置独立 Web 服务器的信息，请参阅教程：[在 Amazon Linux 2 上安装 LAMP Web 服务器](#) 或者教程：[使用 Amazon Linux AMI 安装 LAMP Web 服务器](#)。

ACM 证书受以下服务支持：

### Elastic Load Balancing

Elastic Load Balancing 将传入的应用程序流量自动分配到多个 Amazon EC2 实例。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 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，可从全球边缘站点网络传输内容，从而加快将动态和静态 Web 内容分配给终端用户的速度。终端用户请求您正在通过 CloudFront 提供的内容时，该用户将路由至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 将立即传送内容。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 分配或后端内容源上安装 SSL/TLS 证书。ACM 与 CloudFront 集成以在 CloudFront 分配上部署 ACM 证书。有关更多信息，请参阅 [获取 SSL/TLS 证书](#)。

## Note

要在 CloudFront 中使用 ACM 证书，请确保您在美国东部（弗吉尼亚北部）区域中请求或导入该证书。

### Amazon Cognito

Amazon Cognito 为您的 Web 和移动应用程序提供身份验证、授权和用户管理。用户可使用用户名和密码直接登录，也可以通过第三方（例如 Facebook、Amazon、Google 或 Apple）登录。有关 Amazon Cognito 的更多信息，请参阅 [《Amazon Cognito 开发人员指南》](#)。

当您为 Cognito 用户池配置为使用 Amazon CloudFront 代理时，CloudFront 可能会使用 ACM 证书来保护自定义域的安全。在这种情况下，请注意，必须先删除证书与 CloudFront 的关联，然后才能将其删除。

### Amazon Elastic Beanstalk

Elastic Beanstalk 可帮助您在 Amazon 云中部署和管理应用程序，而无需担心运行这些应用程序的基础设施。Amazon Elastic Beanstalk 可降低管理复杂性。您只需上传应用程序，Elastic Beanstalk 将自动处理有关容量预置、负载均衡、扩展和运行状况监控的部署详细信息。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 [Amazon Elastic Beanstalk 开发人员指南](#)。

若要选择证书，您必须在 Elastic Beanstalk 控制台中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 App Runner

App Runner 是一项 Amazon 服务，它提供了一种快速、简单且经济高效的方式，从源代码或容器镜像直接部署到 Amazon 云中可扩展且安全的 Web 应用程序。您不需要学习新技术、决定要使用的计算服

务，也不需要了解如何预置和配置 Amazon 资源。有关 App Runner 的更多信息，请参阅 [Amazon App Runner 开发人员指南](#)。

当您自定义域名与 App Runner 服务关联时，App Runner 会在内部创建用于跟踪域有效性的证书。它们都存储在 ACM 中。在域与您的服务取消关联或服务被删除后七天内，App Runner 不会删除这些证书。整个过程自动执行，您无需自行添加或管理任何证书。有关更多信息，请参阅 Amazon App Runner 开发人员指南中的 [管理 App Runner 服务的自定义域名](#)。

#### 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 Amazon 上的后端系统交互的 API 变得日益普遍。您可以使用 API Gateway 发布、维护、监控和保护您的 API。将 API 部署到 API Gateway 后，您可以 [设置自定义域名](#) 以简化对它的访问。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。有关 Amazon API Gateway 的更多信息，请参阅 [《Amazon API Gateway 开发人员指南》](#)。

#### Amazon Nitro Enclaves

Amazon Nitro Enclaves 是一项 Amazon EC2 功能，它允许您从 Amazon EC2 实例创建隔离的执行环境，称为 Enclave。Enclave 是独立的、强化的和高度受限的虚拟机。它们仅提供与父实例的安全本地套接字连接。它们没有持久性存储、交互式访问或外部联网。用户无法通过 SSH 进入 Enclave，并且父实例的进程、应用程序或用户（包括根用户或管理员）无法访问该 Enclave 内部的数据和应用程序。

连接到 Nitro Enclaves 的 EC2 实例支持 ACM 证书。有关更多信息，请参阅 [用于 Nitro Enclaves 的 Amazon Certificate Manager](#)。

#### Note

您不能将 ACM 证书与未连接到 Nitro Enclave 的 EC2 实例相关联。

#### Amazon CloudFormation

Amazon CloudFormation 可帮助您对 Amazon Web Services 资源进行建模和设置。您创建一个模板来描述希望使用的 Amazon 资源，如 Elastic Load Balancing 或 API Gateway。然后，Amazon CloudFormation 将负责为您预置和配置这些资源。您无需单独创建和配置 Amazon 资源并了解 what; Amazon CloudFormation 句柄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 Amazon CloudFormation 可以请求 ACM 证书，您可以在 Amazon 服务中使用这些证书来启用安全连接。此外，ACM 证书与许多您可以利用 Amazon CloudFormation 设置的 Amazon 资源一起包含在内。

有关 CloudFormation 的一般信息，请参阅 [《Amazon CloudFormation 用户指南》](#)。要了解 CloudFormation 支持的 ACM 资源，请参阅 [AWS::CertificateManager::Certificate](#)。

由于 Amazon CloudFormation 提供强大的自动化功能，导致很容易超出 [证书配额](#)，特别是在使用新 Amazon 账户的情况下。我们建议您遵循适用于 Amazon CloudFormation 的 ACM [最佳实践](#)。

#### Note

如果您使用 Amazon CloudFormation 创建 ACM 证书，Amazon 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](#)。

#### Amazon Amplify

Amplify 是一组专门构建的工具和功能，使前端 Web 和移动开发人员能够快速轻松地在 Amazon 上构建全栈应用程序。Amplify 提供两项服务：Amplify Hosting 和 Amplify Studio。Amplify Hosting 提供了基于 git 的工作流，用于托管持续部署的全栈无服务器 Web 应用程序。Amplify Studio 是一个直观的开发环境，可简化可扩展的全栈 Web 和移动应用程序的创建。使用 Studio 使用一组随时可用的 UI 组件构建前端 UI，创建应用程序后端，然后将两者连接在一起。有关 Amplify 的更多信息，请参阅 [《Amazon Amplify 用户指南》](#)。

如果您将自定义域连接到应用程序，Amplify 控制台将颁发一个 ACM 证书来保护该域。

## Amazon OpenSearch Service

Amazon OpenSearch Service 是一种搜索和分析引擎，使用案例包括日志分析、实时应用程序监控、点击流分析等。有关更多信息，请参阅《[Amazon OpenSearch Service 开发人员指南](#)》。

创建包含[自定义域和终端节点](#)的 OpenSearch Service 集群时，您可使用 ACM 和一个证书来预置关联的 Application Load Balancer。

# 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：

- Amazon 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可能会被复制到不使用 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。

# 配额

以下 Amazon Certificate Manager (ACM) Service Quotas 适用于每个 Amazon 区域的每个 Amazon 账户。

要了解可以调整哪些配额，请参阅 Amazon 一般参考指南中的 [ACM 配额表](#)。要请求增加配额，请在 [Amazon Web Services Support 中心](#) 创建案例。

## 常规配额

项目	默认配额
ACM 证书数量 过期和吊销的证书将继续计入此总数。 由 ACM Private CA 中的 CA 签名的证书不计入此总数。	2500
每年的 ACM 证书的数量 (最近 365 天) 您按年、按区域和按账户可以申请的证书数量最多是 ACM 证书配额的两倍。例如，如果您的配额为 2500，则您每年可以在给定区域和账户中申请最多 5000 份 ACM 证书。在任何给定时间只能有 2500 份证书。若在一年内请求 5000 份证书，则必须在当年删除 2500 份证书才能避免超出配额。如果在任何给定的时间所需的证书超出 2500 份，您必须联系 <a href="#">Amazon Web Services Support 中心</a> 。 由 ACM Private CA 中的 CA 签名的证书不计入此总数。	您的账户配额的两倍
已导入证书的数量	2,500
每年导入的证书的数量 (最近 365 天)	您的账户配额的两倍

项目	默认配额
<p>每份 ACM 证书的域名数量</p> <p>每份 ACM 证书的默认配额为 10 个域名。您的配额可以提升。</p> <p>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。</p> <p>您最多可以请求 100 个域名。要请求提高配额，请在 <a href="#">Amazon Web Services Support 中心</a> 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 <a href="#">域验证 (p. 22)</a>。</p> <p>每个 ACM 证书的域名数配额仅适用于 ACM 提供的证书。此配额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。</p>	10

## API 速率配额

以下配额适用于每个区域和账户的 ACM API。ACM 将以不同的配额限制 API 请求，具体取决于 API 操作。节流意味着由于请求超出了每秒请求数的操作配额，ACM 会拒绝原本有效的请求。如果请求受到限制，ACM 将返回 `ThrottlingException` 错误。下表列出了每个 API 操作及 ACM 限制该操作请求的配额。

### Note

除了下表中列出的 API 操作外，ACM 还可以调用来自 ACM Private CA 的外部 `IssueCertificate` 操作。有关 `IssueCertificate` 的最新速率配额信息，请参阅适用于 ACM Private CA 的 [终端节点和配额](#)。

每个 ACM API 操作的每秒请求数配额

API 调用	每秒请求数
<code>AddTagsToCertificate</code>	5
<code>DeleteCertificate</code>	10
<code>DescribeCertificate</code>	10
<code>ExportCertificate</code>	5
<code>GetAccountConfiguration</code>	1
<code>GetCertificate</code>	10
<code>ImportCertificate</code>	1
<code>ListCertificates</code>	8
<code>ListTagsForCertificate</code>	10
<code>PutAccountConfiguration</code>	1

API 调用	每秒请求数
RemoveTagsFromCertificate	5
RenewCertificate	5
RequestCertificate	5
ResendValidationEmail	1
UpdateCertificateOptions	5

有关更多信息，请参阅 [Amazon Certificate Manager API 参考](#)。

## Amazon Certificate Manager 定价

对于使用 Amazon Certificate Manager 管理的 SSL/TLS 证书，您不需要支付额外费用。您只需为您创建用于运行网站或应用程序的 Amazon 资源付费。有关最新的 ACM 定价信息，请参阅 Amazon 网站上的 [Amazon Certificate Manager 服务定价](#) 页面。

# Amazon Certificate Manager 中的安全性

Amazon 十分重视云安全性。为了满足对安全性最敏感的组织的需求，我们打造了具有超高安全性的数据中心和网络架构。作为 Amazon 客户，您也将从这些数据中心和网络架构受益。

安全性是 Amazon 和您的共同责任。[责任共担模型](#)将其描述为云的安全性和云中的安全性：

- 云的安全性 – Amazon 负责保护在 Amazon 云中运行 Amazon 服务的基础设施。Amazon 还向您提供可安全使用的服务。第三方审核员定期测试和验证我们的安全性的有效性，作为 [Amazon Compliance Programs](#) 的一部分。要了解适用于 Amazon Certificate Manager 的合规性计划，请参阅[合规性计划范围内的 Amazon 服务](#)。
- 云中的安全性 - 您的责任由您使用的 Amazon 服务决定。您还需要对其它因素负责，包括您的数据的敏感性、您公司的要求以及适用的法律法规。

此文档将帮助您了解如何在使用 Amazon Certificate Manager (ACM) 时应用责任共担模式。以下主题说明如何配置 ACM 以实现您的安全性和合规性目标。您还会了解如何使用其他 Amazon 服务以帮助您监控和保护 ACM 资源。

## 主题

- [Amazon Certificate Manager 中的数据保护 \(p. 9\)](#)
- [Amazon Certificate Manager 的身份和访问管理 \(p. 10\)](#)
- [Amazon Certificate Manager 中的故障恢复能力 \(p. 20\)](#)
- [Amazon Certificate Manager 中的基础设施安全性 \(p. 20\)](#)
- [最佳实践 \(p. 21\)](#)

## Amazon Certificate Manager 中的数据保护

Amazon [责任共担模式](#) 适用于 Amazon Certificate Manager 中的数据保护。如该模式中所述，Amazon 负责保护运行所有 Amazon Web Services 云的全球基础设施。您负责维护对托管在此基础设施上的内容的控制。此内容包括您所使用的 Amazon Web Services 的安全配置和管理任务。有关数据隐私的更多信息，请参阅[数据隐私常见问题](#)。

出于数据保护目的，我们建议您保护 Amazon Web Services 账户凭证并使用 Amazon Identity and Access Management (IAM) 设置单独的用户账户。这仅向每个用户授予履行其工作职责所需的权限。我们还建议您通过以下方式保护您的数据：

- 对每个账户使用 Multi-Factor Authentication (MFA)。
- 使用 SSL/TLS 与 Amazon 资源进行通信。建议使用 TLS 1.2 或更高版本。
- 使用 Amazon CloudTrail 设置 API 和用户活动日志记录。
- 使用 Amazon 加密解决方案以及 Amazon 服务中的所有默认安全控制。
- 使用高级托管安全服务（例如 Amazon Macie），它有助于发现和保护存储在 Simple Storage Service (Amazon S3) 中的个人数据。
- 如果在通过命令行界面或 API 访问 Amazon 时需要经过 FIPS 140-2 验证的加密模块，请使用 FIPS 端点。有关可用的 FIPS 端点的更多信息，请参阅 [《美国联邦信息处理标准 \(FIPS\) 第 140-2 版》](#)。

我们强烈建议您切勿将机密信息或敏感信息（例如您客户的电子邮件地址）放入标签或自由格式字段（例如名称字段）。这包括使用控制台、API、Amazon CLI 或 Amazon 开发工具包处理 ACM 或其他 Amazon 服务时。您在用于名称的标签或自由格式字段中输入的任何数据都可能会用于计费或诊断日志。如果您向外部服务器提供 URL，我们强烈建议您不要在 URL 中包含凭证信息来验证您对该服务器的请求。

## ACM 私有密钥安全性

当您请求公有证书 (p. 28) 时，Amazon Certificate Manager (ACM) 会生成一个公有/私有密钥对。对于导入的证书 (p. 48)，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 Amazon Key Management Service (Amazon 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 Amazon 区域中请求或导入证书时，ACM 会创建一个别名为 aws/acm 的托管 Amazon KMS key。此 KMS 密钥在每个 Amazon 账户和每个 Amazon 区域中都是唯一的。
2. ACM 使用此 KMS 加密证书的私有密钥。ACM 仅存储加密版的私有密钥，而不会以纯文本格式存储私有密钥。ACM 使用同一 KMS 密钥在特定的 Amazon 账户和特定的 Amazon 区域中加密所有证书的私有密钥。
3. 将证书关联到与 Amazon 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。还会在 Amazon KMS 中创建授权，从而允许该服务使用 KMS 密钥来解密该证书的私有密钥。有关授权的更多信息，请参阅 Amazon Key Management Service 开发人员指南中的 [使用授权](#)。有关 ACM 支持的服务的更多信息，请参阅 [与 Amazon Certificate Manager 集成的服务 \(p. 3\)](#)。

### Note

您可以控制自动创建的 Amazon KMS 授权。如果由于任何原因删除此授权，则会失去集成服务的 ACM 功能。

4. 集成服务使用 KMS 密钥来解密私有密钥。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 KMS 密钥来解密证书的私有密钥。

## Amazon Certificate Manager 的身份和访问管理

对 Amazon Certificate Manager (ACM) 的访问要求提供 Amazon 可以用来验证请求的凭据。以下主题提供详细信息来说明如何使用 [Amazon Identity and Access Management \(IAM\)](#) 控制谁能访问您的 Private Certificate Authority (CA)，从而帮助保护它们。

### 身份验证

您可以以下面任一类型的身份访问 Amazon：

- Amazon Web Services 账户 根用户

当您创建 Amazon Web Services 账户时，最初使用的是一个对账户中所有 Amazon Web Services 和资源拥有完全访问权限的登录身份。此身份称为 Amazon Web Services 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。强烈建议您不要使用根用户执行日常任务。保护好根用户凭证，并使用这些凭证来执行仅根用户可以执行的任务。有关要求您以根用户身份登录的任务的完整列表，请参阅《Amazon 一般参考》中的 [需要根用户凭证的任务](#)。

- IAM 用户和组

[IAM 用户](#) 是 Amazon Web Services 账户内对某个人员或应用程序具有特定权限的一个身份。在可能的情况下，我们建议依赖于临时凭证，而不是创建具有长期凭证（如密码和访问密钥）的 IAM 用户。但是，如果您有一些特定的使用场景需要长期凭证以及 IAM 用户，我们建议您轮换访问密钥。有关更多信息，请参阅《IAM 用户指南》中的 [对于需要长期凭证的使用场景定期轮换访问密钥](#)。

**IAM 组**是一个指定一组 IAM 用户的身份。您不能使用组的身份登录。您可以使用组来一次性为多个用户指定权限。如果有大量用户，使用组可以更轻松地管理用户权限。例如，您可能具有一个名为 IAMAdmins 的组，并为该组授予权限以管理 IAM 资源。

用户与角色不同。用户唯一地与某个人员或应用程序关联，而角色旨在让需要它的任何人代入。用户具有永久的长期凭证，而角色提供临时凭证。要了解更多信息，请参阅 IAM 用户指南中的[何时创建 IAM 用户（而不是角色）](#)。

- 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。IAM 角色类似于 IAM 用户，因为它是一个 Amazon 身份，具有确定其在 Amazon 中可执行和不可执行的操作的权限策略。但是，角色旨在让需要它的任何人代入，而不是唯一地与某个人员关联。此外，角色没有关联的标准长期凭证（如密码或访问密钥）。相反，当您代入角色时，它会为您提供角色会话的临时安全凭证。具有临时凭证的 IAM 角色在以下情况下很有用：

- **Federated user access（联合用户访问）** – 要向联合身份分配权限，请创建角色并为角色定义权限。当联合身份进行身份验证时，该身份将与角色相关联并被授予由此角色定义的权限。有关联合身份验证的角色的信息，请参阅《IAM 用户指南》中的[为第三方身份提供商创建角色](#)。
- **Amazon Web Service 访问 – 服务角色**是一个 **IAM 角色**，服务代入该角色以代表您执行操作。IAM 管理员可以在 IAM 中创建、修改和删除服务角色。有关更多信息，请参阅 IAM 用户指南中的[创建向 Amazon Web Service 委派权限的角色](#)。
- **在 Amazon EC2 上运行的应用程序** – 您可以使用 IAM 角色管理在 EC2 实例上运行并发出 Amazon CLI 或 Amazon API 请求的应用程序的临时凭证。这优先于在 EC2 实例中存储访问密钥。要将 Amazon 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 角色为 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 Amazon Certificate Manager 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 \(p. 11\)](#)
- [适用于 Amazon Certificate Manager 的 Amazon 托管策略 \(p. 13\)](#)
- [客户管理型策略 \(p. 14\)](#)
- [内联策略 \(p. 14\)](#)
- [ACM API 权限：操作和资源参考 \(p. 19\)](#)

## 管理对您的 ACM 资源的访问概述

每个 Amazon Certificate Manager (ACM) 资源属于一个 Amazon 账户，而创建或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）附加权限策略。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 \(p. 12\)](#)
- [了解资源所有权 \(p. 12\)](#)
- [管理对 ACM 证书的访问 \(p. 12\)](#)

## 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:region:account: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

资源所有者是创建资源的 Amazon 账户。也就是说，资源所有者是 Amazon 委托人实体的账户，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 Amazon 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 Amazon 账户根用户的凭证创建 ACM 证书，则您的 Amazon 账户拥有该证书。
- 如果您在自己的 Amazon 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 Amazon 账户中创建 IAM 角色，并向该角色授予创建 ACM 证书的权限，则任何能够担任该角色的人员均可创建证书。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 上下文中使用 IAM。这里不提供有关 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 policy 语法和说明的信息，请参阅 [IAM Policy 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：

- Amazon 托管策略 - 由 Amazon 创建和管理的策略。这些策略是可以附加到 Amazon 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 Amazon 账户中创建和管理的策略，可以将这些策略附加到多个用户、组和角色。与使用 Amazon 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。

## 适用于 Amazon Certificate Manager 的 Amazon 托管策略

要向用户、组和角色添加权限，与自己编写策略相比，使用 Amazon 托管策略更简单。创建仅为团队提供所需权限的 **IAM 客户托管策略** 需要时间和专业知识。要快速入门，您可以使用我们的 Amazon 托管策略。这些策略涵盖常见使用案例，可在您的 Amazon 账户中使用。有关 Amazon 托管策略的更多信息，请参阅 IAM 用户指南中的 **Amazon 托管策略**。

Amazon 服务负责维护和更新 Amazon 托管策略。您无法更改 Amazon 托管策略中的权限。服务偶尔会向 Amazon 托管策略添加额外权限以支持新功能。此类更新会影响附加策略的所有身份（用户、组和角色）。当启动新功能或新操作可用时，服务最有可能更新 Amazon 托管策略。服务不会从 Amazon 托管策略中删除权限，因此策略更新不会破坏您的现有权限。

此外，Amazon 还支持跨多种服务的工作职能的托管策略。例如，ReadOnlyAccess Amazon 托管策略提供对所有 Amazon 服务和资源的只读访问权限。当服务启动新功能时，Amazon 会为新操作和资源添加只读权限。有关工作职能策略的列表和说明，请参阅 IAM 用户指南中的 **适用于工作职能的 Amazon 托管策略**。

### 主题

- [AWSCertificateManagerReadOnly](#) (p. 13)
- [AWSCertificateManagerFullAccess](#) (p. 13)
- [ACM 对 Amazon 托管策略的更新](#) (p. 14)

## 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",
      "acm:GetAccountConfiguration"
    ],
    "Resource": "*"
  }
}
```

要在控制台中查看此 Amazon 托管策略，请转至 <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": "*"
  },
  {
    "Effect": "Allow",
    "Action": "iam:CreateServiceLinkedRole",
    "Resource": "arn:aws:iam::*:role/aws-service-role/acm.amazonaws.com/AWSServiceRoleForCertificateManager*",
    "Condition": {
      "StringEquals": {
        "iam:AWSServiceName": "acm.amazonaws.com"
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
      "iam:DeleteServiceLinkedRole",
      "iam:GetServiceLinkedRoleDeletionStatus",
      "iam:GetRole"
    ],
    "Resource": "arn:aws:iam::*:role/aws-service-role/acm.amazonaws.com/AWSServiceRoleForCertificateManager*"
  }
]
}

```

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

## ACM 对 Amazon 托管策略的更新

查看有关 ACM 的 Amazon 托管策略更新的详细信息（从该服务开始跟踪这些更改开始）。要获得有关此页面更改的自动提示，请订阅 ACM [文档历史记录](#) (p. 121) 页面上的 RSS 源。

更改	说明	日期
增加了对 <a href="#">AWSCertificateManagerReadOnly</a> 策略的 <code>GetAccountConfiguration</code> 支持。	AWSCertificateManagerReadOnly 策略现在包含调用 <code>GetAccountConfiguration</code> API 操作的权限。	2021 年 3 月 3 日
ACM 开始跟踪更改	ACM 开始跟踪 Amazon 托管策略的更改。	2021 年 3 月 3 日

## 客户管理型策略

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 Amazon 账户中的多个用户、组或角色。您可以使用 Amazon Web Services Management Console、Amazon Command Line Interface (Amazon CLI) 或 IAM API 管理和创建策略。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 Amazon Web Services Management Console、Amazon Command Line Interface (Amazon CLI) 或 IAM API 来创建和嵌入内联策略。

#### 主题

- [列出证书 \(p. 15\)](#)
- [检索证书 \(p. 15\)](#)
- [导入证书 \(p. 15\)](#)
- [删除证书 \(p. 16\)](#)
- [对 ACM 的只读访问 \(p. 16\)](#)
- [对 ACM 的完全访问 \(p. 16\)](#)
- [对所有 Amazon 资源的管理员访问权限 \(p. 17\)](#)

## 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:region:account:certificate/certificate_ID"
  }
}
```

## 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:region:account:certificate/certificate_ID"
  }
}
```

## 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:region:account:certificate/certificate_ID"
  }
}
```

## 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 Amazon Web Services Management Console 中的 Amazon 托管策略使用。有关更多信息，请参阅 [AWSCertificateManagerReadOnly](#) (p. 13)。要在控制台中查看托管策略，请转至 <https://console.aws.amazon.com/iam/home#policies/arn:aws:iam::aws:policy/AWSCertificateManagerReadOnly>。

## 对 ACM 的完全访问

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 Amazon Web Services Management Console 中的 Amazon 托管策略使用。有关更多信息，请参阅 [AWSCertificateManagerFullAccess](#) (p. 13)。要在控制台中查看托管

策略，请转至 <https://console.aws.amazon.com/iam/home#policies/arn:aws:iam::aws:policy/AWSCertificateManagerFullAccess>。

## 对所有 Amazon 资源的管理员访问权限

以下策略允许用户对任何 Amazon 资源执行任何操作。

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

### Note

此策略可作为 Amazon Web Services Management Console 中的 Amazon 托管策略使用。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>。

## 将服务相关角色 (SLR) 用于 ACM

Amazon Certificate Manager 使用 Amazon Identity and Access Management (IAM) [服务相关角色](#) 以启用托管 ACM 证书的自动续订。服务相关角色是一种与 ACM 服务直接相关的 IAM 角色。SLR 由 ACM 预定义，并包含该服务代表您调用其他 Amazon 服务所需的一切权限。

SLR 可让您更轻松地进行设置，因为您无需手动添加必要的权限即可进行无人参与的证书签名。ACM 定义其 SLR 的权限，除非另外定义，否则只有 ACM 可以担任该角色。定义的权限包括信任策略和权限策略，以及不能附加到任何其它 IAM 实体的权限策略。

有关支持 SLR 的其他服务的信息，请参阅 [可与 IAM 配合使用的 Amazon 服务](#)，并查找 Service-Linked Role (服务相关角色) 列中为 Yes (是) 的服务。选择带有链接的 Yes 可以查看该服务的 SLR 文档。

## ACM 的 SLR 权限

ACM 使用名为 Amazon Certificate Manager 服务角色策略的 SLR。

AWSServiceRoleForCertificateManager SLR 信任以下服务以担任该角色：

- `acm.amazonaws.com`

角色权限策略允许 ACM 对指定资源完成以下操作：

- 操作：对 "\*" 执行 `acm-pca:IssueCertificate`、`acm-pca:GetCertificate`

您必须配置权限以允许 IAM 实体（例如，用户、组或角色）创建、编辑或删除 SLR。有关更多信息，请参阅 IAM 用户指南中的 [服务相关角色权限](#)。

### Important

ACM 可能会提示您，它无法确定您的账户中是否存在 SLR。如果所需的 `iam:GetRole` 权限已被授予您账户的 ACM SLR，则在创建 SLR 后不会再发出提示。如果提示再次发生，那么您或您

的账户管理员可能需要授予 `iam:GetRole` 访问 ACM 的权限，或者将您的账户与 ACM 托管策略 `AWSCertificateManagerFullAccess` 关联。

## 为 ACM 创建 SLR

无需手动创建 ACM 使用的 SLR。当您使用 Amazon Web Services Management Console、Amazon CLI 或 Amazon API 颁发 ACM 证书时，ACM 会在您首次选择私有 CA 对证书签名时为您创建 SLR。

如果您收到消息指示 ACM 无法确定您的账户上是否存在 SLR，则可能意味着您的账户尚未授予 ACM Private CA 所需的读取权限。这不会阻止安装 SLR，您仍然可以颁发证书，但 ACM 将无法自动续订证书，直到您解决该问题。有关更多信息，请参阅[ACM 服务关联角色 \(SLR\) 问题 \(p. 113\)](#)。

### Important

如果您在使用此角色支持的功能的其他服务中完成某个操作，此 SLR 可以出现在您的账户中。此外，如果您在 2017 年 1 月 1 日（从此时开始支持 SLR）以前在使用 ACM 服务，则 ACM 已在您的账户中创建 `AWSServiceRoleForCertificateManager` 角色。要了解更多信息，请参阅[我的 IAM 账户中出现新角色](#)。

如果您删除此 SLR，然后需要再次创建它，则可以使用以下任一方法：

- 在 IAM 控制台中，选择 Role（角色）、Create role（创建角色）、Certificate Manager，以创建一个用于 `CertificateManagerServiceRolePolicy` 使用案例的新角色。
- 使用 IAM API `CreateServiceLinkedRole` 或相应的 Amazon CLI 命令 `create-service-linked-role`，创建一个具有 `acm.amazonaws.com` 服务名称的 SLR。

有关更多信息，请参阅《IAM 用户指南》中的[创建服务相关角色](#)。

## 编辑 ACM 的 SLR

ACM 不允许您编辑 `AWSServiceRoleForCertificateManager` 服务相关角色。在创建 SLR 以后，您无法更改该角色的名称，因为可能有不同的实体引用该角色。但是可以使用 IAM 编辑角色说明。有关更多信息，请参见 IAM 用户指南中的[编辑服务相关角色](#)。

## 删除 ACM 的 SLR

您一般不需要删除 `AWSServiceRoleForCertificateManager` SLR。但是，您可以使用 IAM 控制台、Amazon CLI 或 Amazon API 手动删除该角色。有关更多信息，请参见 IAM 用户指南中的[删除服务相关角色](#)。

## ACM SLR 的支持区域

ACM 支持在同时提供 ACM 和 ACM Private CA 的所有区域中使用 SLR。有关更多信息，请参阅[Amazon 区域和终端节点](#)。

区域名称	区域标识	ACM 中支持
美国东部（弗吉尼亚州北部）	us-east-1	是
美国东部（俄亥俄州）	us-east-2	是
美国西部（北加利福尼亚）	us-west-1	是
美国西部（俄勒冈州）	us-west-2	是
亚太地区（孟买）	ap-south-1	是

区域名称	区域标识	ACM 中支持
亚太地区 ( 大阪 )	ap-northeast-3	是
亚太地区 ( 首尔 )	ap-northeast-2	是
亚太地区 ( 新加坡 )	ap-southeast-1	是
Asia Pacific (Sydney)	ap-southeast-2	是
亚太地区 ( 东京 )	ap-northeast-1	是
Canada (Central)	ca-central-1	是
欧洲 ( 法兰克福 )	eu-central-1	是
Europe (Ireland)	eu-west-1	是
欧洲 ( 伦敦 )	eu-west-2	是
欧洲 ( 巴黎 )	eu-west-3	是
South America (São Paulo)	sa-east-1	是
中国 ( 北京 )	cn-north-1	否
中国 ( 宁夏 )	cn-northwest-1	否
Amazon GovCloud ( 美国西部 )	us-gov-west-1	是
Amazon GovCloud (US-East) East	us-gov-east-1	是

## ACM API 权限：操作和资源参考

在设置访问控制 (p. 11) 和编写可附加到 IAM 身份的权限策略 (基于身份的策略) 时, 您可以将下表作为参考。该表的第一列中列出了每个 Amazon Certificate Manager 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 policy 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	必需的权限 ( API 操作 )	资源
<a href="#">AddTagsToCertificate</a>	acm:AddTagsToCertificate	arn:aws:acm: <i>region</i> : <i>account</i> :certificate: <i>certificate_ID</i>
<a href="#">DeleteCertificate</a>	acm>DeleteCertificate	arn:aws:acm: <i>region</i> : <i>account</i> :certificate: <i>certificate_ID</i>
<a href="#">DescribeCertificate</a>	acm:DescribeCertificate	arn:aws:acm: <i>region</i> : <i>account</i> :certificate: <i>certificate_ID</i>
<a href="#">ExportCertificate</a>	acm:ExportCertificate	arn:aws:acm: <i>region</i> : <i>account</i> :certificate: <i>certificate_ID</i>

ACM API 操作	必需的权限 ( API 操作 )	资源
<a href="#">GetAccountConfiguration</a>	acm:GetAccountConfiguration*	
<a href="#">GetCertificate</a>	acm:GetCertificate	arn:aws:acm:region:account:certificate:certificate_ID
<a href="#">ImportCertificate</a>	acm:ImportCertificate	arn:aws:acm:region:account:certificate:* 或 *
<a href="#">ListCertificates</a>	acm:ListCertificates	*
<a href="#">ListTagsForCertificate</a>	acm:ListTagsForCertificate	arn:aws:acm:region:account:certificate:certificate_ID
<a href="#">PutAccountConfiguration</a>	acm:PutAccountConfiguration*	
<a href="#">RemoveTagsFromCertificate</a>	acm:RemoveTagsFromCertificate	arn:aws:acm:region:account:certificate:certificate_ID
<a href="#">RequestCertificate</a>	acm:RequestCertificate	arn:aws:acm:region:account:certificate:* 或 *
<a href="#">ResendValidationEmail</a>	acm:ResendValidationEmail	arn:aws:acm:region:account:certificate:certificate_ID
<a href="#">UpdateCertificateOptions</a>	acm:UpdateCertificateOptions	arn:aws:acm:region:account:certificate:certificate_ID

## Amazon Certificate Manager 中的故障恢复能力

Amazon 全球基础设施围绕 Amazon 区域和可用区构建。Amazon 区域提供多个在物理上独立且隔离的可用区，这些可用区通过延迟低、吞吐量高且冗余性高的网络连接在一起。利用可用区，您可以设计和操作在可用区之间无中断地自动实现故障转移的应用程序和数据库。与传统的单个或多个数据中心基础设施相比，可用区具有更高的可用性、容错性和可扩展性。

有关 Amazon 区域和可用区的更多信息，请参阅 [Amazon 全球基础设施](#)。

## Amazon Certificate Manager 中的基础设施安全性

作为一项托管服务，Amazon Certificate Manager 由 [Amazon Web Services : 安全流程概述](#) 白皮书中所述的 Amazon 全球网络安全程序提供保护。

您可以使用 Amazon 发布的 API 调用通过网络访问 ACM。客户端应支持传输层安全性 (TLS) 1.0 或更高版本。建议使用 TLS 1.2 或更高版本。客户端还必须支持具有完全向前保密 (PFS) 的密码套件，例如 Ephemeral Diffie-Hellman (DHE) 或 Elliptic Curve Ephemeral Diffie-Hellman (ECDHE)。大多数现代系统（如 Java 7 及更高版本）都支持这些模式。

此外，必须使用访问密钥 ID 和与 IAM 委托人关联的秘密访问密钥来对请求进行签名。或者，您可以使用 [Amazon Security Token Service \(Amazon STS\)](#) 生成临时安全凭证来对请求进行签名。

## 最佳实践

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

### 主题

- [Amazon CloudFormation \(p. 21\)](#)
- [证书固定 \(p. 21\)](#)
- [域验证 \(p. 22\)](#)
- [添加或删除域名 \(p. 22\)](#)
- [选择退出证书透明度日志记录 \(p. 22\)](#)
- [启用 Amazon CloudTrail。\(p. 23\)](#)

## Amazon CloudFormation

利用 Amazon CloudFormation，您可以创建一个模板来描述要使用的 Amazon 资源。随后，Amazon CloudFormation 将为您预置和配置这些资源。Amazon CloudFormation 可以为您预置 ACM 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 [Amazon Certificate Manager 集成的服务 \(p. 3\)](#)。

如果您使用 Amazon 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配额。有关更多信息，请参阅 [配额 \(p. 6\)](#)。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，如果您为只有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 Amazon 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 [ACM 证书的托管续订 \(p. 40\)](#) 以在 Amazon 颁发的 SSL/TLS 证书过期前进行续订这些证书。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：

- 将**您自己的证书**导入到 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。
- 如果您使用的是公有证书，则将您的应用程序固定到所有可用的 [Amazon 根证书](#)。如果您使用的是私有证书，则将您的应用程序固定到 CA 的根证书。

## 域验证

Amazon 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 [DNS 验证 \(p. 30\)](#) 和 [电子邮件验证 \(p. 33\)](#)。

## 添加或删除域名

您无法在现有 ACM 证书中添加或删除域名。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。

如果改为使用 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 不再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 已开始至少在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 \(p. 117\)](#)。

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` Amazon CLI 命令的 `options` 参数或 `RequestCertificate` API 操作。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 操作以选择退出。

### 限制

- 您无法使用控制台启用或禁用透明度日志记录。
- 在证书进入续订期（通常在证书过期前 60 天）后，您无法更改日志记录状态。在状态更改失败时不会生成错误消息。

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。

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 \  
--options DisableCertificateTransparencyLogging
```

```
--validation-method DNS \  
--options CertificateTransparencyLoggingPreference=DISABLED \  

```

上述命令输出新证书的 ARN。

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

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/certificate_ID \  
--options CertificateTransparencyLoggingPreference=DISABLED
```

## 启用 Amazon 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 Amazon API 调用历史记录来监控您的 Amazon 部署，包括通过 Amazon 管理控制台、Amazon 开发工具包、Amazon Command Line Interface 以及更高级的 Amazon Web Services 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。您可以使用 API 将 CloudTrail 集成到应用程序、为您的企业自动创建跟踪、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 [将 CloudTrail 用于 Amazon Certificate Manager \(p. 66\)](#) 以查看 ACM 操作的示例跟踪记录。

# 设置

借助 Amazon Certificate Manager (ACM)，您可以为基于 Amazon 的网站和应用程序预置和管理 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 Amazon 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 Amazon Certificate Manager 集成的服务 \(p. 3\)](#)。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 \(p. 48\)](#)。

## 主题

- [设置 Amazon 和 IAM \(p. 24\)](#)
- [注册域名 \(p. 25\)](#)
- ( 可选 ) [为域配置电子邮件 \(p. 25\)](#)
- ( 可选 ) [配置 CAA 记录 \(p. 26\)](#)

## 设置 Amazon 和 IAM

您必须首先注册 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以执行的操作。

## 注册 Amazon

如果您还不是 Amazon (Amazon Web Services) 的客户，则必须先进行注册才能使用 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 Amazon 新客户，还可以免费试用。有关更多信息，请参阅 [Amazon 免费套餐](#)。

### 如何注册 Amazon 账户

1. 转到 <http://aws.amazon.com/> 并选择 Sign Up ( 注册 )。
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 Amazon 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，我们建议您删除您的根用户访问密钥。然后创建 Amazon Identity and Access Management (IAM) 用户凭证来执行与 Amazon 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 Amazon 账户 \( 根 \) 访问密钥](#)。

## Note

您可能需要 Amazon Web Services 账户 根用户访问权限来执行特定任务，例如更改 Amazon 支持计划或关闭您的账户。在这些情况下，请使用您的电子邮件地址和密码登录 Amazon Web Services Management Console。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Amazon账户根用户的Amazon任务](#)。

通过 IAM，您可以安全地控制用户对 Amazon 账户中 Amazon 服务和资源的访问。例如，如果您需要管理员级别权限，则可以创建 IAM 用户，为该用户授予完全访问权限，然后使用这些凭证与 Amazon 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 Amazon 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 Amazon 账户中的资源具有只读访问权限的 IAM 用户，并将这些凭证分发给您的用户。

ACM 还提供两个[Amazon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 Amazon 账户拥有者。

## 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](#)。

## ( 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[电子邮件验证 \(p. 33\)](#)。

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[DNS 验证 \(p. 30\)](#)。

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拥有者或指定代表批准 ACM 证书的说明。有关更多信息，请参阅[电子邮件验证 \(p. 33\)](#)。如果您对验证电子邮件有疑问，请参阅[排查电子邮件验证的问题 \(p. 104\)](#)。

## 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 向域或子域的以下五个常用系统管理地址发送电子邮件：

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

### Tip

您可以覆盖您的验证消息发送到的域。如果您使用 `ResendValidationEmail` API 操作或 `resend-validation-email` Amazon CLI 命令，Amazon 不会执行 MX 查找。相反，您提供您的注册域名和自定义验证域的名称，并且 Amazon 会将前面的五封电子邮件发送到验证域而不是您的注册域。

无论您使用的是控制台、API 还是 Amazon CLI，ACM 始终将验证电子邮件发送到前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，Amazon 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 \(p. 104\)](#) 了解有关可能的原因及解决方法的信息。

## ( 可选 ) 配置 CAA 记录

您可以选择配置证书颁发机构授权 (CAA) DNS 记录，以指定允许 Amazon 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。

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。

### 标签

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。

## 值

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。

当存在不包含 ACM CA 值的 issuewild CAA 记录时，ACM 不能颁发任何通配符证书。如果没有 issuewild，但对于 ACM 有一个 issue CAA 记录，则 ACM 可以颁发通配符证书。

## 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。

# 颁发和管理证书

ACM 证书可用于跨 Internet 或在内部网络中建立安全通信。您可以直接从 ACM 请求公开信任证书（“ACM 证书”），也可以导入由第三方颁发的公开信任证书。也支持自签名证书。要预置企业的内部 PKI，您可以颁发由 Private Certificate Authority (CA) 签名的 ACM 证书，并由 [ACM Private CA](#) 进行管理。CA 可能位于您的账户中，也可能由其他账户与您共享。

## Note

公有 ACM 证书可以安装在连接到 [Nitro Enclave \(p. 5\)](#) 的 Amazon EC2 实例上，但不能安装到其他 Amazon EC2 实例上。有关在未连接到 Nitro Enclave 的 Amazon EC2 实例上设置独立 Web 服务器的信息，请参阅教程：[在 Amazon Linux 2 上安装 LAMP Web 服务器](#) 或者教程：[使用 Amazon Linux AMI 安装 LAMP Web 服务器](#)。

## Note

因为由私有 CA 签名的证书预设情况下不受信任，管理员必须在客户端信任存储中安装这些证书。

要开始颁发证书，请登录 Amazon 管理控制台并从以下位置打开 ACM 控制台：<https://console.aws.amazon.com/acm/home>。如果显示介绍页面，请选择 Get Started（开始使用）。否则，请在左侧导航窗格中选择 Certificate Manager 或 Private CAs。

## 主题

- [请求公有证书 \(p. 28\)](#)
- [验证域所有权 \(p. 30\)](#)
- [列出由 ACM 管理的证书 \(p. 35\)](#)
- [描述 ACM 证书 \(p. 36\)](#)
- [删除 ACM 管理的证书 \(p. 39\)](#)
- [安装 ACM 证书 \(p. 39\)](#)

## 请求公有证书

以下各部分讨论如何使用 ACM 控制台或 Amazon CLI 来请求公有 ACM 证书。

如果在请求证书时遇到问题，请参阅[排查证书请求问题 \(p. 100\)](#)。

## Note

除非您选择退出，否则公开受信任的 ACM 证书将自动记录在至少两个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 Amazon CLI 或 ACM API。有关更多信息，请参阅[选择退出证书透明度日志记录 \(p. 22\)](#)。有关透明度日志的一般信息，请参阅[证书透明度日志 \(p. 117\)](#)。

## 主题

- [使用控制台请求公有证书 \(p. 29\)](#)
- [使用 CLI 请求公有证书 \(p. 30\)](#)

## 使用控制台请求公有证书

### 请求 ACM 公有证书 (控制台)

1. 登录Amazon管理控制台并从以下位置打开 ACM 控制台：<https://console.aws.amazon.com/acm/home>。

选择请求证书。

2. 在 Domain names ( 域名 ) 部分，键入您的域名。

您可以使用完全限定域名 (FQDN) ( 例如 `www.example.com` )，或者裸域名或顶点域名 ( 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` 可以保护 `corp.example.com` 和 `images.example.com`。通配符名称将显示在 ACM 证书的 Subject ( 主题 ) 字段和 Subject Alternative Name ( 主题替代名称 ) 扩展中。

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` 可以保护 `login.example.com` 和 `test.example.com`，但不能保护 `test.login.example.com`。另请注意，`*.example.com` 仅保护 `example.com` 的子域，而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。

#### Note

为满足 [RFC 5280](#) 要求，在此步骤中输入的域名 ( 技术术语为“公用名” ) 的长度不能超过 64 个八位字节 ( 字符 )，包括句点。但是，您 ( 在下一步中 ) 提供的每个后续主题替代名称 (SAN) 的长度最多可达 253 个八位字节。

3. 要添加其他名称，请选择 Add another name to this certificate ( 向此证书添加另一个名称 )，然后在文本框中键入名称。这对于同时保护裸域或顶点域 ( 例如 `example.com` ) 及其子域 ( 例如 `*.example.com` ) 非常有用。
4. 在 Select validation method ( 选择验证方法 ) 部分，根据您的需要选择 DNS validation ( DNS 验证 ) 或 Email validation ( 电子邮件验证 )。

#### 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[DNS 验证 \(p. 30\)](#)。

ACM 在颁发证书之前，会验证您是否拥有或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[电子邮件验证 \(p. 33\)](#)。

如果使用 DNS 验证，则只需将 ACM 提供的别名记录写入您的 DNS 配置。有关 DNS 验证的更多信息，请参阅[DNS 验证 \(p. 30\)](#)。

5. 在 Tags ( 标签 ) 页面上，您可以选择为证书添加标签。标签是键值对，用作标识和组织 Amazon 资源的元数据。有关 ACM 标签参数的列表以及有关如何在创建证书后向证书添加标签的说明，请参阅[标记 Amazon Certificate Manager 证书 \(p. 55\)](#)。

完成添加标签后，选择 Request ( 请求 )。

6. 处理请求后，控制台将返回证书列表，其中会显示有关新证书的信息。

在接到请求时，证书的状态将变为 Pending validation ( 等待验证 )，除非因故障排除主题证书请求失败中列出的任何原因导致请求失败。ACM 重复尝试验证证书达 72 小时，然后超时。如果证书的状态显示为 Failed ( 已失败 ) 或 Validation timed out ( 验证超时 )，则删除请求，更正 [DNS 验证](#) 或 [电子邮件验证](#) 问题，然后重试。如果验证成功，则证书的状态将变为 Issued ( 已颁发 )。

## Note

根据您的列表排序的方式，您要查找的证书可能不会立即可见。您可以点击右侧的黑色三角形来更改顺序。您还可以使用右上角的页码浏览多页证书。

## 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
```

此命令输出新公有证书的 Amazon Resource Name (ARN)。

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

## 验证域所有权

在 Amazon 证书颁发机构 (CA) 能够为您的网站颁发证书以前，Amazon 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。在申请证书时，您可以选择通过域名系统 (DNS) 验证或电子邮件验证来证明您的所有权。

### Note

验证仅适用于 ACM 颁发的公开信任证书。ACM 不会验证[导入的证书 \(p. 48\)](#)或由私有 CA 签名的证书的域所有权。ACM 无法验证 Amazon VPC [私有托管区](#)或任何其他私有域中的资源。有关更多信息，请参阅[排查证书验证问题 \(p. 101\)](#)。

一般来说，我们建议使用 DNS 验证而不是电子邮件验证，原因如下：

- 如果您使用 Amazon Route 53 管理您的公有 DNS 记录，则可以直接通过 ACM 更新您的记录。
- 只要证书正在使用中，并且别名记录保持存在，ACM 就会自动续订 DNS 验证的证书。
- 要续订，通过电子邮件验证的证书需要域所有者执行操作。ACM 会在到期前 45 天开始发送续订通知。这些通知将发送到域的 WHOIS 邮箱地址和五个常用管理员地址。通知中包含一个链接，域所有者可以单击该链接以轻松续订。验证所有列出的域后，ACM 会颁发具有相同 ARN 的续订证书。

如果您没有编辑域的 DNS 数据库的授权，则必须使用[电子邮件验证 \(p. 33\)](#)。

### 主题

[DNS 验证 \(p. 30\)](#)

[电子邮件验证 \(p. 33\)](#)

## DNS 验证

域名系统 (DNS) 是连接到网络的资源的目录服务。DNS 提供商维护一个包含定义域的记录的数据库。如果选择 DNS 验证，ACM 会提供一条或多条别名记录，这些记录必须添加到此数据库。这些记录包含一个唯一的键值对，用于证明您对该域具有控制权。

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 为您创建两条别名记录。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 Amazon 域的别名，ACM 使用该域自动续订证书。别名记录只需添加到 DNS 数据库中一次。只要证书正在使用中，并且别名记录保持存在，ACM 就会自动续订证书。

### Important

如果您使用 Amazon Route 53 管理您的公有 DNS 记录，请联系您的 DNS 提供商了解如何添加记录。如果您没有编辑域的 DNS 数据库的权限，则必须使用 [电子邮件验证 \(p. 33\)](#)。

无需重复验证，只要别名记录仍然存在，您就可以为您的完全限定域名 (FQDN) 申请额外的 ACM 证书。也就是说，您可以创建具有相同域名的替换证书，或者是覆盖不同子域的证书。由于别名记录验证令牌适用于任何 Amazon 区域中，您可以在多个区域中重新创建相同的证书。您还可以替换已删除的证书。

您可以通过从证书关联的 Amazon 服务删除证书或通过删除别名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记录。如果 Route 53 是您的提供商，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 [ACM 证书的托管续订 \(p. 40\)](#)。

### Note

如果在 DNS 配置中链接到五个以上的别名记录，则别名记录解析将失败。如果您需要更长的链接，我们建议使用 [电子邮件验证 \(p. 33\)](#)。

## ACM 别名记录的工作原理

### Note

本部分适用于不使用 Route 53 作为其 DNS 提供商的客户。

如果您不使用 Route 53 作为 DNS 提供商，则需要（通常通过网站）手动将 ACM 提供的别名记录输入到提供商的数据库中。别名记录用于多种目的，包括作为重新导向机制和供应商特定元数据的容器。对于 ACM，这些记录允许初始域所有权验证和持续的自动证书续订。

下表显示了六个域名的示例别名记录。每条记录的记录名称-记录值对用于验证域名所有权。

在表中，请注意，前两个记录名称-记录值对是相同的。这说明了对于 `*.example.com` 等通配符域，ACM 创建的字符串与其基域 `example.com` 创建的字符串相同。否则，配对的记录名称和记录值将会因每个域名而异。

### 别名记录示例

域名	记录名称	记录值	Comment
<code>*.example.com</code>	<code>_<b>x1</b>.example.com.</code>	<code>_<b>x2</b>.acm- validations.aws.</code>	相同
<code>example.com</code>	<code>_<b>x1</b>.example.com.</code>	<code>_<b>x2</b>.acm- validations.aws.</code>	
<code>www.example.com</code>	<code>_<b>x3</b>.www.example.com.</code>	<code>_<b>x4</b>.acm- validations.aws.</code>	唯一
<code>host.example.com</code>	<code>_<b>x5</b>.host.example.com.</code>	<code>_<b>x6</b>.acm- validations.aws.</code>	唯一
<code>subdomain.example.com</code>	<code>_<b>x7</b>.subdomain.example.com.</code>	<code>_<b>x8</b>.acm- validations.aws.</code>	唯一
<code>host.subdomain.example.com</code>	<code>_<b>x9</b>.host.subdomain.example.com.</code>	<code>_<b>x10</b>.acm- validations.aws.</code>	唯一

下划线 ( \_ ) 之后的 **xN** 值是由 ACM 生成的长随机字符串。例如，

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

代表生成的记录名称。关联的记录值可能是

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

用于同一 DNS 记录。

#### Note

如果您的 DNS 提供商不支持带有前导下划线的别名记录值，请参阅[排查 DNS 验证问题 \(p. 102\)](#)。

当您请求证书并指定 DNS 验证时，ACM 会提供以下格式的别名记录信息：

域名	记录名称	记录类型	记录值
example.com	_a79865eb4cd1a6ab990a45779b4e0b96	别名记录	example.com.424c7224e9b0146f9a8808af955727d0.acm-validations.aws.

域名是与证书关联的 FQDN。记录名称唯一标识记录，用作键值对的键。记录值用作键值对的值。

必须在 DNS 提供商用于添加 DNS 记录的 Web 界面的相应字段中输入所有这三个值（域名、记录名称和记录值）。提供商对于记录名称（或只是“名称”）字段的处理方式并不相同。在某些情况下，您需要提供上面所示的整个字符串。其他提供商会自动将域名附加到您输入的任何字符串，这意味着（在本示例中）您只应输入

```
_a79865eb4cd1a6ab990a45779b4e0b96
```

到名称字段中。如果您猜错了这一点，并输入包含域名的记录名称（例如 `.example.com`），您的最终结果可能如下所示：

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

在这种情况下，验证将失败。因此，您应该尝试提前确定您的提供商期望的输入类型。

## 设置 DNS 验证

此部分介绍如何将公有证书配置为使用 DNS 验证。

在控制台中设置 DNS 验证

#### Note

此过程假设您已经创建至少一个证书，并且您正在创建证书的 Amazon 区域工作。如果您尝试打开控制台并看到首次使用屏幕，或者您已成功打开控制台，但没有在列表中看到您的证书，请确认您指定了正确的区域。

1. 通过以下网址打开 ACM 控制台：<https://console.aws.amazon.com/acm/>。
2. 在证书列表中，请选择要配置的状态为 Pending validation（等待验证）的证书的 Certificate ID（证书 ID）。此时将打开证书的详细信息页面。
3. 在 Domains（域）部分，完成下列两个过程之一：

- a. (可选) 使用 Route 53 进行验证。

如果满足以下条件，则会显示活动的 Create records in Route 53 (在 Route 53 中创建记录) 按钮：

- 您使用 Route 53 作为 DNS 提供商。
- 您有权写入由 Route 53 托管的区域。
- 您的 FQDN 尚未经过验证。

#### Note

如果您在使用 Route 53 但 Create record in Route 53 (在 Route 53 中创建记录) 按钮缺失或已禁用，请参阅 [ACM 控制台不显示“Create record in Route 53” \(在 Route 53 中创建记录\) 按钮 \(p. 103\)](#)。

选择 Create records in Route 53 (在 Route 53 中创建记录) 按钮，然后选择 Create records (创建记录)。此时会打开 Certificate status (证书状态) 页面，状态横幅将报告 Successfully created DNS records (已成功创建 DNS 记录)。

您的新证书可能会继续显示 Pending validation (等待验证) 最多 30 分钟。

#### Tip

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，您可以向 Route 53 发出一个 Amazon CLI 或 API 调用以在 Route 53 DNS 数据库中创建记录。有关 Route 53 记录集的更多信息，请参阅 [使用资源记录集](#)。

- b. (可选) 如果您没有使用 Route 53 作为 DNS 提供商，则必须检索 CNAME 信息，并将其添加到您的 DNS 数据库中。您可以通过两种方式之一来执行此操作：

- 复制 Domains (域) 部分中显示的 CNAME 组件。此信息需要手动添加到您的 DNS 数据库。
- 或者，选择 Export to CSV (导出到 CSV)。结果文件中的信息需要手动添加到 DNS 数据库中。

#### Important

要避免验证问题，请查看 [ACM 别名记录的工作原理 \(p. 31\)](#) 然后将信息添加到 DNS 提供商的数据库。如果遇到问题，请参阅 [排查 DNS 验证问题 \(p. 102\)](#)。

如果 ACM 无法在生成别名记录值后的 72 小时内验证域名，ACM 会将证书状态更改为 Validation timed out (验证超时)。导致此结果的最可能原因是您未使用 ACM 生成的值成功更新 DNS 配置。要解决这个问题，您必须在查看别名记录说明后请求新的证书。

## 电子邮件验证

Amazon 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 [DNS 验证 \(p. 30\)](#)。

#### Note

您需要一个在您的域中注册的工作电子邮件地址才能使用电子邮件验证。设置电子邮件地址的过程不在本指南的讨论范围内。

ACM 证书的有效期为 13 个月 (395 天)。要续订，通过电子邮件验证的证书需要域所有者执行操作。ACM 在过期前 45 天开始向域的 WHOIS 邮箱地址和五个常用管理员地址发送续订通知。通知中包含一个链接，域所有者可以单击该链接以轻松续订。验证所有列出的域后，ACM 会颁发具有相同 ARN 的续订证书。

如果在使用电子邮件验证时遇到问题，请参阅[排查电子邮件验证的问题 \(p. 104\)](#)。

#### Note

验证仅适用于 ACM 颁发的公开信任证书。ACM 不会验证[导入的证书 \(p. 48\)](#)或由私有 CA 签名的证书的域所有权。ACM 无法验证 Amazon VPC [私有托管区](#)或任何其他私有域中的资源。有关更多信息，请参阅[排查证书验证问题 \(p. 101\)](#)。

ACM 将电子邮件发送到 WHOIS 数据库中列出的 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，将发送最多八封电子邮件。例如，如果您仅指定一个域名，您最多将收到八封电子邮件。要进行验证，您必须在 72 小时内对这八封电子邮件中的一封进行操作。如果您指定三个域名，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三封电子邮件（指定的每个名称对应一封电子邮件）进行操作。

电子邮件将发送到 WHOIS 中的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用系统地址。如果您使用 [RequestCertificate API](#) 或 [request-certificate Amazon CLI 命令](#)，则 ACM 不会执行 MX 查找。它会将电子邮件发送到您在 `DomainName` 参数或在 `ValidationDomain` 可选参数中指定的域名称。有关更多信息，请参阅[MX 记录 \(p. 26\)](#)。

- `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 \(p. 25\)](#)。

此过程的例外情况

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 `ValidationDomainRequestCertificate API` 或 `request-certificate`<https://docs.amazonaws.cn/cli/latest/reference/acm/request-certificate.html> 命令中指定 Amazon CLI 选项。如果使用控制台请求证书，目前还无法使用此功能。即使所有邮件都发送到一个电子邮件地址，您也必须响应每个域或子域的一封邮件，以便验证该域并生成证书。

## 证书过期和续订

ACM 证书的有效期为 13 个月 ( 395 天 )。要续订，通过电子邮件验证的证书需要域所有者执行操作。ACM 在过期前 45 天开始向域的 WHOIS 邮箱地址和五个常用管理员地址发送续订通知。通知中包含一个链接，域所有者可以单击该链接以轻松续订。验证所有列出的域后，ACM 会颁发具有相同 ARN 的续订证书。

有关更多信息，请参阅上面的[电子邮件验证 \(p. 33\)](#)。

### ( 可选 ) 重新发送验证邮件

每封验证电子邮件都包含一个令牌，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。

有关电子邮件验证的持久性问题，请参阅[问题排查 \(p. 100\)](#)中的[排查电子邮件验证的问题 \(p. 104\)](#)部分。

#### Note

以下信息仅适用于 ACM 提供的证书，以及使用电子邮件验证的证书。私有 PKI 证书或[您导入到 ACM 中的证书 \(p. 48\)](#)不需要验证电子邮件。有关 DNS 域验证的信息，请参阅[DNS 验证 \(p. 30\)](#)。

使用控制台重新发送验证电子邮件

1. 登录 Amazon 管理控制台并从以下位置打开 ACM 控制台：<https://console.aws.amazon.com/acm/home>。
2. 在证书列表中，请选择要验证的证书的 Certificate ID ( 证书 ID )。此操作将打开详细信息页面。

#### Note

根据您的列表排序的方式，您要查找的证书可能不会立即可见。您可以点击右侧的黑色三角形来更改顺序。您还可以使用右上角的页码浏览多页证书。

3. 在 Domains ( 域 ) 部分，选择 Resend validation email ( 重新发送验证电子邮件 )，选择每个需要验证的域，然后选择 Resend ( 重新发送 )。此时应该会显示 Successfully resent validation emails ( 已成功重新发送验证电子邮件 ) 横幅。

要使用 Amazon CLI 重新发送验证电子邮件

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/certificate_ID --domain www.example.com --  
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。

## 列出由 ACM 管理的证书

您可以使用 ACM 控制台或 Amazon CLI 列出由 ACM 管理的证书。

#### Note

如果您管理 100 个以上的证书，我们建议您选择使用 Amazon CLI。

## 使用控制台列出证书

1. 通过以下网址打开 ACM 控制台：<https://console.aws.amazon.com/acm/>。
2. 查看证书列表中的信息。您可以使用右上角的页码浏览多页证书。每个证书占据一行，默认情况下为每个证书显示以下列：

- Domain Name ( 域名 ) – 证书的完全限定域名 (FQDN)。
- Type ( 类型 ) – 证书的类型。可能的值包括：Amazon issued ( Amazon 已颁发 ) | Private ( 私有 ) | Imported ( 已导入 )
- Status ( 状态 ) – 证书状态。可能的值包括：Pending validation ( 等待验证 ) | Issued ( 已颁发 ) | Inactive ( 非活动 ) | Expired ( 已过期 ) | Revoked ( 已吊销 ) | Failed ( 失败 ) | Validation timed out ( 验证超时 )
- 正在使用？ – ACM 证书是否主动与 Elastic Load Balancing 或 CloudFront 等 Amazon 服务关联。值可以是 No 或 Yes。

### Note

您可以通过选择控制台右上角的齿轮图标来自定义要显示的列以及其他设置。有关可用证书详细信息的更多信息，请参阅 [描述 ACM 证书 \(p. 36\)](#)。

要使用 Amazon CLI 列出证书

使用 `list-certificates` 命令列出 ACM 管理的证书，如以下示例所示：

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

命令返回类似于下文的信息：

```
{
  "CertificateSummaryList": [
    {
      "CertificateArn": "arn:aws:acm:region:account:certificate/certificate_ID_1",
      "DomainName": "example.com"
    },
    {
      "CertificateArn": "arn:aws:acm:region:account:certificate/certificate_ID_2",
      "DomainName": "mydomain.com"
    }
  ]
}
```

预设情况下，只返回具有 `keyTypes RSA_1024` 或 `RSA_2048` 并且至少有一个指定域的证书。要查看您控制的其他证书，例如无域证书或使用不同算法或位大小的证书，请提供 `--includes` 参数，如以下示例所示。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 Amazon CLI 列出有关证书的详细元数据。

在控制台中查看证书详细信息

1. 通过 <https://console.aws.amazon.com/acm/> 打开 ACM 控制台，以显示您的证书。您可以使用右上角的页码浏览多页证书。

2. 要显示所列证书的详细元数据，请选择证书 ID。此时将打开页面，其中显示以下信息：

- 证书状态
  - Identifier (标识符) – 证书的 32 字节十六进制唯一标识符
  - ARN - `arn:aws:acm:region:account:certificate/certificate_ID` 格式的 Amazon Resource Name (ARN)
  - Type (类型) - 标识 ACM 证书的管理类别。可能的值有：Amazon Issued (Amazon 已颁发) | Private (私有) | Imported (导入)。有关更多信息，请参阅 [请求公有证书 \(p. 28\)](#) 或 [将证书导入到 Amazon Certificate Manager 中 \(p. 48\)](#)。
  - Status (状态) – 证书状态。可能的值包括：Expired (已过期) | Failed (已失败) | Inactive (非活动) | Issued (已颁发) | Pending validation (等待验证)
  - Detailed status (详细状态) - 颁发或导入证书的日期和时间
- 域
  - Domain (域) – 证书的完全限定域名 (FQDN)。
  - Status (状态) – 域验证状态。可能的值包括：Pending validation (等待验证) | Revoked (已吊销) | Failed (失败) | Validation timed out (验证超时) | Success (成功)
- 详细信息
  - 正在使用？ – 证书是否与 [Amazon 集成服务 \(p. 3\)](#) 关联，可能的值有：Yes (是) | No (否)
  - Domain name (域名) – 证书的完全限定域名 (FQDN)。
  - Number of additional names (其他名称的数量) – 证书对其有效的域名数
  - Serial number (序列号) - 证书的 16 字节十六进制序列号
  - Public key info (公有密钥信息) - 生成密钥对的加密算法
  - Signature algorithm (签名算法) - 用于对证书进行签名的加密算法。
  - Can be used with (可以用于) - 支持带有这些参数的证书的 ACM [集成服务](#) 列表
  - Requested at (请求时间) – 颁发请求的日期和时间
  - Issued at (颁发时间) – 颁发的日期和时间 (如果适用)
  - Imported at (导入时间) – 导入的日期和时间 (如果适用)
  - Not before (不早于) – 证书有效期的开始时间
  - Not after (不晚于) - 证书的到期日期和时间。
  - Renewal eligibility (续订资格) – 可能的值包括：Eligible (有资格) | Ineligible (没有资格)
  - CA – 签名 CA 的 ARN
- 标签
  - Key (密钥)
  - 值
- Validation state (验证状态) - 如果适用，可能的值为：
  - Pending (挂起) – 已请求验证，但尚未完成。
  - Validation timed out (验证超时) - 请求的验证已超时，但您可以重复该请求。
  - None (无) - 证书用于私有 PKI 或自签名，不需要验证。

要使用 Amazon CLI 查看证书详细信息

使用 Amazon CLI 中的 `describe-certificate` 显示证书详细信息，如以下命令所示：

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

命令返回类似于下文的信息：

```
{
```

```
"Certificate": {
  "CertificateArn": "arn:aws:acm:region:account:certificate/certificate_ID",
  "Status": "EXPIRED",
  "Options": {
    "CertificateTransparencyLoggingPreference": "ENABLED"
  },
  "SubjectAlternativeNames": [
    "example.com",
    "www.example.com"
  ],
  "DomainName": "gregpe.com",
  "NotBefore": 1450137600.0,
  "RenewalEligibility": "INELIGIBLE",
  "NotAfter": 1484481600.0,
  "KeyAlgorithm": "RSA-2048",
  "InUseBy": [
    "arn:aws:cloudfront::account:distribution/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 ACM 管理的证书

您可以使用 ACM 控制台或 Amazon CLI 删除证书。

### Important

删除由私有证书颁发机构 (CA) 颁发的证书对 CA 没有影响。您将继续为 CA 支付费用，直到删除该 CA 为止。有关更多信息，请参阅 [Amazon Certificate Manager Private Certificate Authority 用户指南](#) 中的 [删除私有 CA](#)。

您无法删除正在由其他 Amazon 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。这是使用相关服务的控制台或 CLI 完成的。

### 使用控制台删除证书

1. 通过以下网址打开 ACM 控制台：<https://console.aws.amazon.com/acm/>。
2. 在证书列表中，选中 ACM 证书对应的复选框，然后选择 Delete (删除)。

### Note

根据您的列表排序的方式，您要查找的证书可能不会立即可见。您可以点击右侧的黑色三角形来更改顺序。您还可以使用右上角的页码浏览多页证书。

### 要使用 Amazon CLI 删除证书

使用 `delete-certificate` 命令删除证书，如以下命令所示：

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

## 安装 ACM 证书

您无法直接在基于 Amazon 的网站或应用程序中使用 ACM 安装公有证书。您必须使用与 ACM 集成的服务之一。有关更多信息，请参阅 [与 Amazon Certificate Manager 集成的服务 \(p. 3\)](#)。

ACM Private CA 中由 CA 签署并用于您的私有 PKI 的 ACM 证书，可以 [导出](#) 并手动安装在您具有管理权限的任何系统上。这些证书在公有 Internet 上不受信任。

# ACM 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。这意味着 ACM 将自动续订您的证书（如果您使用 DNS 验证），或者在接近过期时向您发送电子邮件通知。对于公有和私有 ACM 证书，都提供这些服务。

证书可自动续订，但需注意以下几点：

- 如果它与其他 Amazon 服务（例如 Elastic Load Balancing 或 CloudFront）关联，则符合条件。
- 如果它是自颁发或上次续订后导出的，则符合条件。
- 如果是 [导入 \(p. 48\)](#) 的证书，则不符合条件。
- 如果已过期，则不符合条件。

此外，还必须满足以下与 [国际化域名](#) 有关的 [Punycode](#) 要求：

1. 以“<character><character>--”模式开头的域名必须与“xn--”一致。
2. 以“xn--”开头的域名也必须是有效的国际化域名。

## Punycode 示例

域名	满足条件 1	满足条件 2	已允许	注意
example.com	不适用	不适用	✓	不是以“<character><character>--”开头
a--example.com	不适用	不适用	✓	不是以“<character><character>--”开头
abc--example.com	不适用	不适用	✓	不是以“<character><character>--”开头
xn--xyz.com	是	是	✓	有效的国际化域名（解析为简.com）
xn--example.com	是	否	✗	不是有效的国际化域名
ab--example.com	否	否	✗	必须以“xn--”开头

当 ACM 续订证书时，证书的 Amazon Resource Name (ARN) 保持不变。此外，ACM 证书是 [区域性资源 \(p. 3\)](#)。如果您在多个 Amazon 区域中具有带同一域名的证书，则必须单独续订每个证书。

## 主题

- [续订公开信任的证书 \(p. 40\)](#)
- [在私有 PKI 中续订证书 \(p. 43\)](#)
- [检查证书的续订状态 \(p. 45\)](#)

## 续订公开信任的证书

在颁发托管式的公开信任证书时，Amazon Certificate Manager 会要求您证明自己是域拥有者。这可以通过 [DNS 验证 \(p. 30\)](#) 或 [电子邮件验证 \(p. 33\)](#) 的方式进行。当证书需要续订时，ACM 会使用您之前选择的相同方法来重新验证您的所有权。以下主题说明了续订过程在各种情况下的工作机制。

## 主题

- [续订通过 DNS 验证的域 \(p. 41\)](#)
- [续订通过电子邮件验证的域 \(p. 41\)](#)

## 续订通过 DNS 验证的域

对于最初使用 [DNS 验证 \(p. 30\)](#) 颁发的 ACM 证书，托管续订是完全自动化的。

在过期前 60 天，ACM 会检查以下续订条件：

- 该证书当前正由 Amazon 服务使用中。
- ACM 提供的所有必需 DNS CNAME 记录（每个唯一的主题备用名称一个）都存在并可以通过公共 DNS 访问。

如果满足这些条件，ACM 将认为域名已通过验证并续订证书。

如果 ACM 无法通过 CNAME 记录自动验证域名，则会通知您使用与该域关联的电子邮件地址续订失败。这些通知将在过期前 45 天、30 天、15 天、7 天、3 天和 1 天的时候发送。自动验证失败的最常见原因是无意中更改或删除了所需的 CNAME 记录。

## 续订通过电子邮件验证的域

ACM 证书的有效期为 13 个月（395 天）。要续订，通过电子邮件验证的证书需要域拥有者执行操作。ACM 在过期前 45 天开始向域的 WHOIS 邮箱地址和五个常用管理员地址发送续订通知。通知中包含一个链接，域拥有者可以单击该链接以轻松续订。验证所有列出的域后，ACM 会颁发具有相同 ARN 的续订证书。

有关验证电子邮件的更多信息，请参阅 [电子邮件验证 \(p. 33\)](#)。

要了解如何以编程方式回复验证电子邮件，请参阅 [自动化电子邮件验证 \(p. 42\)](#)。

## 请求域验证电子邮件

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 \(p. 25\)](#)），您可以使用 Amazon 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 \(p. 45\)](#)。
- 您未收到或无法找到 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. 打开 Amazon Certificate Manager 控制台，网址：<https://console.aws.amazon.com/acm/home>。
2. 选择需要验证的证书的证书 ID。
3. 选择 Resend validation email（重新发送验证电子邮件）。

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。以下示例说明如何使用 Amazon CLI 执行该操作。此示例包含换行符以便于阅读。

```
$ aws acm resend-validation-email \
```

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

## 自动化电子邮件验证

通过电子邮件验证的 ACM 证书通常需要域拥有者的手动干预。处理大量经电子邮件验证证书的企业可能更愿意创建一个可以自动执行所需响应的解析器。为了帮助客户使用电子邮件验证，本节中的信息介绍了用于域验证电子邮件的模板以及完成验证过程所涉及的工作流。

### 验证电子邮件模板

验证电子邮件采用以下格式。突出显示的字符串的内容应替换为特定于正在验证的域的值。

```
Greetings from Amazon Web Services,  
  
We received a request to issue an SSL/TLS certificate for requested_domain.  
  
Verify that the following domain, Amazon Web Services account ID, and certificate  
identifier correspond  
to a request from you or someone in your organization.  
  
Domain: fqdn  
Amazon Web Services account ID: account_id  
Amazon Web Services Region name: region_name  
Certificate Identifier: certificate_identifier  
  
To approve this request, go to Amazon Certificate Approvals  
(https://region_name.acm-certificates.amazonaws.cn/approvals?  
code=validation_code&context=validation_context)  
and follow the instructions on the page.  
  
This email is intended solely for authorized individuals for fqdn. To express any  
concerns  
about this email or if this email has reached you in error, forward it along with a  
brief  
explanation of your concern to validation-questions@amazon.com.  
  
Sincerely,  
Amazon Web Services
```

#### Note

一旦您收到来自 Amazon 的新验证消息，我们建议您将其用作解析器的最新和最权威的模板。

使用 2020 年 11 月之前设计的消息解析器的客户应注意，可能已对模板进行如下更改。

- 电子邮件主题行现在显示为“Certificate request for domain name”而不是“Certificate approval for domain name”。
- 现在，Amazon account ID 在显示时不带破折号或连字符。
- Certificate Identifier 现在会显示完整的证书 ARN 而不是简写格式，例如，将显示 `arn:aws:acm:us-east-1:000000000000:certificate/3b4d78e1-0882-4f51-954a-298ee44ff369` 而不是 `3b4d78e1-0882-4f51-954a-298ee44ff369`。
- 证书批准 URL 现在包含 `acm-certificates.amazonaws.cn` 而不是 `certificates.amazon.com`。
- 通过单击证书批准 URL 打开的审批表单现在包含批准按钮。批准按钮 div 的名称现在是 `approve-button` 而不是 `approval_button`。
- 新请求的证书和续订证书的验证消息具有相同的电子邮件格式。

## 验证工作流程

本节提供有关经电子邮件验证的证书的续订工作流程相关信息。

- ACM 控制台处理多域证书请求时，它会将验证电子邮件发送到它找到的第一个包含 MX 记录的域。域拥有者需要验证每个域的电子邮件，然后 ACM 才能颁发证书。有关更多信息，请参阅[使用电子邮件验证域所有权](#)。
- 使用 ACM API 或 CLI 对多域证书请求进行电子邮件验证，会导致预设情况下将电子邮件发送到顶端域和每个子域。域拥有者需要验证每个域的电子邮件消息，然后 ACM 才能颁发证书。

### Note

在 2020 年 11 月之前，客户只需验证顶端域，ACM 将颁发一份同时涵盖任何子域的证书。使用此时间之前设计的消息解析器的客户应注意电子邮件验证工作流程的更改。

- 使用 ACM API 或 CLI，您可以强制将多域证书请求的所有验证电子邮件发送到顶端域。在 API 中，使用 `RequestCertificate` 操作的 `DomainValidationOptions` 参数来指定 `ValidationDomain` 的值，它是 `DomainValidationOption` 类型的成员。在 CLI 中，使用 `request-certificate` 命令的 `--domain-validation-options` 参数指定 `ValidationDomain` 的值。

## 在私有 PKI 中续订证书

由 ACM Private CA 中的私有 CA 签名的 ACM 证书符合条件进行托管续订。与公开受信任 ACM 证书不同，私有 PKI 的证书不需要验证。当管理员在客户端信任存储区中安装相应的根 CA 证书时，就会建立信任。

### Note

只有使用 ACM 控制台或 ACM API 的 `RequestCertificate` 操作获得的证书才符合托管续订条件。使用 PCA API 的 `IssueCertificate` 操作直接从 ACM Private CA 颁发的证书不由 ACM 管理。

当托管式证书还剩 60 天过期时，ACM 会自动尝试续订。这包括手动导出和安装的证书（例如，在本地数据中心中）。客户也可以随时使用 ACM API 的 `RenewCertificate` 操作强制续订。有关强制续订的 Java 实现示例，请参阅[续订证书 \(p. 92\)](#)。

续订后，证书部署到服务的方式有如下几种：

- 如果证书与 ACM [集成服务](#) 关联，则新证书将替换旧证书，而无需额外的客户操作。
- 如果证书不与 ACM [集成服务](#) 关联，则需要客户操作才能导出并安装续订的证书。您可以手动执行这些操作，或借助 [Amazon Health](#)、[Amazon EventBridge](#) 和 [Amazon Lambda](#) 执行，如下所示：有关更多信息，请参阅[自动导出已续订的证书 \(p. 43\)](#)

## 自动导出已续订的证书

以下过程提供了一个示例解决方案，用于在 ACM 续订私有 PKI 证书时自动导出这些证书。此示例仅从 ACM 中导出证书及其私有密钥；导出后，证书仍必须安装在其目标设备上。

使用控制台自动导出证书

1. 按照《Amazon Lambda 开发人员指南》中的步骤操作，创建并配置调用 ACM 导出 API 的 Lambda 函数。
  - a. [创建一个 Lambda 函数](#)。
  - b. 为您的函数[创建一个 Lambda 执行角色](#)并添加以下信任策略。该策略将向函数中的代码授予调用 ACM API 的 `ExportCertificate` 操作，从而检索已续订的证书和私有密钥的权限。

```
{
```

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

- c. 将应用程序代码添加到函数中，例如以下 Python 语言示例。此代码会检查将触发函数调用 `ExportCertificate` 操作的事件。有关更多信息，请参阅[使用 Python 构建 Lambda 函数](#)。

```
import json
import boto3

acm = boto3.client('acm')
passphrase = b'\x01\x02\x03\x04\x05\x06\x07\x08\x09\x00'

def lambda_handler(event, context):
    context.log("Incoming Event : " + json.dumps(event) + "\n");

    certificateArn = event['resources'][0];

    context.log("Renewed Certificate ARN : " + certificateArn + "\n");

    description = event['detail']['eventDescription'][0]['latestDescription'];

    if "completed the renewal" in description:
        response =
acm.export_certificate(CertificateArn=certificateArn,Passphrase=passphrase)

        certificate = response['Certificate']
        certificateChain = response['CertificateChain']
        encrPrivateKey = response['PrivateKey']

        context.log("Certificate : " + certificate + "\n");
        context.log("Certificate Chain : " + certificateChain + "\n");
        context.log("Encrypted Private Key : " + encrPrivateKey + "\n");

        #TODO Install the private key, cert and chain to where it will be consumed

    return {
        'statusCode': 200,
    }
```

2. 在 [Amazon EventBridge 中创建一条规则](#) 以侦听 ACM 运行状况事件，并在检测到相关事件时调用您的 Lambda 函数。ACM 每次尝试续订证书时都写入一个 Amazon Health 事件。有关这些通知的更多信息，请参阅 [使用 Personal Health Dashboard \(PHD\) 检查状态 \(p. 47\)](#)。

通过添加以下事件模式来配置规则。

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

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

3. 通过在目标系统上手动安装证书来完成续订过程。

## 测试私有 PKI 证书的托管式续订

您可以使用 ACM API 或 Amazon CLI 手动测试 ACM 托管续订工作流的配置。通过这样做，您可以确认您的证书将在过期前由 ACM 自动续订。

### Note

您只能测试由 ACM Private CA 颁发的证书的续订。

当您使用下面描述的 API 操作或 CLI 命令时，ACM 会尝试续订证书。如果续订成功，ACM 会更新管理控制台或 API 输出中显示的证书元数据。如果证书与 ACM [集成服务](#) 关联，则会部署新证书并在 Amazon CloudWatch Events 中生成续订事件。如果续订失败，ACM 将返回错误并建议采取补救措施。（您可以使用 [describe-certificate](#) 命令查看此信息。）如果证书不是通过集成服务部署的，您仍然需要将其导出并手动将其安装到资源上。

### Important

要使用 ACM 续订您的 ACM Private CA 证书，您必须先向 ACM 服务委托人授予执行此操作的权限。有关更多信息，请参阅 [将证书续订权限分配给 ACM](#)。

### 手动测试证书续订 (Amazon CLI)

1. 使用 [renew-certificate](#) 证书续订导出的私有证书。

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

2. 然后，使用 [describe-certificate](#) 命令确认已更新该证书的续订详细信息。

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

### 手动测试证书续订 (ACM API)

- 发送 [RenewCertificate](#) 请求，并指定要续订的私有证书的 ARN。然后，使用 [DescribeCertificate](#) 操作来确认已更新该证书的续订详细信息。

## 检查证书的续订状态

您可以使用 Amazon Certificate Manager 控制台、ACM API、Amazon CLI 或 Amazon Health Dashboard 检查 ACM 证书的续订状态。如果您使用控制台、Amazon CLI 或 ACM API，则证书续订可以具有下列四个可能的状态值之一。如果使用 Amazon 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[续订通过 DNS 验证的域 \(p. 41\)](#)。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。

### 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 \(p. 28\)](#)。

如果某个证书与其他 Amazon 服务（例如 Elastic Load Balancing 或 CloudFront）关联，或者它在颁发或上次续订之后已进行导出，则它符合续订的条件。

### 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 \(控制台\) \(p. 46\)](#)
- [检查状态 \(API\) \(p. 46\)](#)
- [检查状态 \(CLI\) \(p. 46\)](#)
- [使用 Personal Health Dashboard \(PHD\) 检查状态 \(p. 47\)](#)

## 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. 打开 Amazon Certificate Manager 控制台，网址：<https://console.aws.amazon.com/acm/home>。
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 \(p. 83\)](#)。

## 检查状态 (CLI)

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

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

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。

## 使用 Personal Health Dashboard (PHD) 检查状态

ACM 在过期前 60 天会尝试自动续订您的 ACM 证书。如果 ACM 无法自动续订您的证书，它会在过期前 45 天、30 天、15 天、7 天、3 天和 1 天向 Amazon Health Dashboard 发送证书续订事件通知，通知您需要执行相关操作。Amazon Health Dashboard 是 Amazon 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 [Amazon Health 用户指南](#)。

### Note

ACM 将连续续订事件通知写入 PHD 时间线中的单个事件。每个通知都会覆盖前一个通知，直到续订成功为止。

要使用 Amazon Health Dashboard，请执行以下操作：

1. 登录 Amazon Health Dashboard，网址为 <https://phd.aws.amazon.com/phd/home/>。
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 (应用)。
6. 对于 Event category，选择 Scheduled Change。
7. 选择 Apply (应用)。

# 将证书导入到 Amazon Certificate Manager 中

除了请求 Amazon Certificate Manager (ACM) 提供的 SSL/TLS 证书以外，您还可以导入您在 Amazon 外部获取的证书。您可能要执行此操作的原因包括：您已从第三方证书颁发机构 (CA) 获取证书，或者由 ACM 颁发的证书不能满足您的应用程序特定要求。

您可以将导入的证书用于任何与 [ACM 集成的 Amazon 服务](#) (p. 3)。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. 40)。

若要续订导入的证书，您可以从证书发布者处获取新证书，然后手动将其 [重新导入](#) 到 ACM。此操作将保留证书的关联及其 Amazon Resource name (ARN)。另外，您也可以导入全新的证书。可以导入多个具有相同域名的证书，但必须将其逐个导入。

## 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。您可以通过使用 Amazon CloudWatch Events 在导入的证书接近过期时发送通知来简化此任务。有关更多信息，请参阅 [使用 CloudWatch Events](#) (p. 58)。

ACM 中的所有证书都是区域性资源，包括您导入的证书。若要将同一证书用于不同 Amazon 区域中的 Elastic Load Balancing 负载均衡器，您必须将证书导入到您要在其中使用它的每个区域。要将证书用于 Amazon CloudFront，您必须将证书导入美国东部（弗吉尼亚北部）区域。有关更多信息，请参阅 [支持的区域](#) (p. 3)。

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 [证书导入问题](#) (p. 111)。

## 主题

- [导入证书的先决条件](#) (p. 48)
- [证书和密钥的导入格式](#) (p. 49)
- [导入证书](#) (p. 50)
- [重新导入证书](#) (p. 51)

## 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。要导入由非 Amazon 证书颁发机构 (CA) 签署的证书，您还须包含该证书的私有密钥、该证书的公有密钥以及证书链。您的证书必须满足本主题中描述的所有条件。

对于所有导入的证书，必须指定加密算法和密钥大小。ACM 支持以下算法（括号中为 API 名称）：

- 1024 位 RSA (RSA\_1024)
- 2048 位 RSA (RSA\_2048)
- 3072 位 RSA (RSA\_3072)
- 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)

另请注意以下额外要求：

- ACM [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。例如，CloudFront 仅支持 1024 位 RSA、2048 位 RSA 和 Elliptic Prime Curve 256 位密钥，而 Application Load Balancer 支持 ACM 提供的所有算法。有关详细信息，请参阅您使用的服务的文档。
- 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) 或 IP 地址以及有关发布者的信息。
- 证书可以由您拥有的私有密钥或发证 CA 的私有密钥进行自签名。

如果证书为自签名证书，则必须提供私有密钥。私有密钥不得超过 5KB (5,120 字节)，且必须未加密。

如果证书由 CA 签名，则必须提供证书链，并且证书的加密算法必须与 CA 的算法匹配。例如，如果 CA 密钥类型为 RSA，则该证书的密钥类型也必须为 RSA。

- 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。
- 需要的所有证书材料（证书、私有密钥和证书链）均采用 PEM 编码。上传 DER 编码的材料会导致错误。有关更多信息以及示例，请参阅 [证书和密钥的导入格式](#) (p. 49)。
- 续订（重新导入）证书时，如果先前导入的证书中没有扩展，则无法添加 KeyUsage 或 ExtendedKeyUsage 扩展。

## 证书和密钥的导入格式

ACM 要求您单独导入证书、证书链和私有密钥（如有），并以 PEM 格式对每个组件进行编码。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。

### Note

Amazon 不提供用于操作 PEM 文件或其他证书格式的实用程序。以下示例依赖于通用文本编辑器进行简单操作。如果您需要执行更复杂的任务（例如转换文件格式或提取密钥），随时可以使用免费的开源工具（如 [OpenSSL](#)）。

下面的示例介绍了要导入的文件格式。如果在单个文件中向您提供这些组件，请使用文本编辑器将它们拆分成三个文件（务必仔细）。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 1. PEM 编码的证书

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

### Example 2. 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。如要导入私有证书，请最后复制根证书。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 3. PEM 编码的私有密钥 ( 仅限私有证书 )

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-公有密钥对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 Amazon Web Services Management Console、Amazon CLI 或 ACM API 将从外部获取的证书 ( 即由第三方信任服务提供商提供的证书 ) 导入 ACM。以下主题向您介绍如何使用 Amazon Web Services Management Console 和 Amazon CLI。从非 Amazon 发布者获取证书的程序不在本指南的讨论范围之内。

### Important

您选择的签名算法必须满足 [导入证书的先决条件 \(p. 48\)](#)。

### 主题

- [导入 \( 控制台 \) \(p. 50\)](#)
- [导入 \(Amazon CLI\) \(p. 51\)](#)

## 导入 ( 控制台 )

以下示例说明如何使用 Amazon Web Services Management Console 导入证书。

1. 从以下位置打开 ACM 控制台：<https://console.aws.amazon.com/acm/home>。如果您是首次使用 ACM，请查找 Amazon Certificate Manager 标题，并选择其下方的 Get started ( 开始使用 ) 按钮。
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。它应以 -----BEGIN CERTIFICATE----- 开头并以 -----END CERTIFICATE----- 结尾。
  - b. 对于 Certificate private key (证书私有密钥)，粘贴证书的 PEM 编码的未加密私有密钥。它应以 -----BEGIN PRIVATE KEY----- 开头并以 -----END PRIVATE KEY----- 结尾。
  - c. (可选) 对于 Certificate chain (证书链)，粘贴 PEM 编码的证书链。
4. 选择 Review and import。
  5. 在 Review and import (查看并导入) 页面上，检查显示的有关证书的元数据，以确保它符合您的期望。这些字段包括：
    - Domains – 由证书验证的完全限定域名 (FQDN) 的列表
    - Expires in – 证书过期前的天数
    - Public key info – 用于生成密钥对的加密算法
    - Signature algorithm – 用于创建证书签名的加密算法
    - Can be used with – 支持您要导入的证书类型的 ACM [集成服务](#) 的列表

如果一切正确，请选择 Import (导入)。

## 导入 (Amazon CLI)

以下示例说明如何使用 [Amazon Command Line Interface \(Amazon CLI\)](#) 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 fileb://Certificate.pem \  
  --certificate-chain fileb://CertificateChain.pem \  
  --private-key fileb://PrivateKey.pem
```

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

## 重新导入证书

如果您已导入一个证书并将该证书与其他 Amazon 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 Amazon 服务关联。有关与 ACM 集成的 Amazon 服务的更多信息，请参阅 [与 Amazon Certificate Manager 集成的服务 \(p. 3\)](#)。

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。
- 如果原始导入的证书中存在 Key Usage (密钥用法) 扩展，您可以添加新的扩展值，但不能删除现有的扩展值。
- 如果原始导入的证书中存在 Extended Key Usage (扩展密钥用法) 扩展，您可以添加新的扩展值，但不能删除现有的扩展值。

- 密钥类型和大小不能更改。
- 您不能在重新导入证书时应用资源标签。

#### 主题

- [重新导入 \(控制台\) \(p. 52\)](#)
- [重新导入 \(Amazon CLI\) \(p. 52\)](#)

## 重新导入 (控制台)

以下示例说明如何使用 Amazon Web Services Management Console 重新导入证书。

1. 从以下位置打开 ACM 控制台：<https://console.aws.amazon.com/acm/home>。
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。
6. (可选) 对于 Certificate chain (证书链)，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。

## 重新导入 (Amazon CLI)

以下示例说明如何使用 Amazon Command Line Interface (Amazon 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。
- (仅限私有证书) 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 fileb://Certificate.pem \  
  --certificate-chain fileb://CertificateChain.pem \  
  --private-key fileb://PrivateKey.pem \  
  --certificate-arn arn:aws:acm:region:123456789012:certificate/12345678-1234-1234-1234-12345678901
```

如果 import-certificate 命令成功完成，则将返回证书的 Amazon Resource Name (ARN)。

# 导出私有证书

您可以导出 ACM Private CA 颁发的证书，以在私有 PKI 环境中的任何位置使用。导出的文件包含证书、证书链和加密的私有密钥。此文件必须安全存储。有关私有 ACM Private CA 的更多信息，请参阅 [Amazon Certificate Manager Private Certificate Authority 用户指南](#)。

## Note

不能导出公开信任的 ACM 证书或其私有密钥。

## 主题

- [导出私有证书 \(控制台\) \(p. 53\)](#)
- [导出私有证书 \(CLI\) \(p. 53\)](#)

## 导出私有证书 (控制台)

1. 登录 Amazon 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 Certificate Manager。
3. 选择您要导出的证书。
4. 在 Actions 菜单上，选择 Export (private certificates only)。
5. 输入并确认私有密钥的密码。

## Note

创建密码短语时，您可以使用除 #、\$ 或 % 之外的任何 ASCII 字符。

6. 选择 Generate PEM Encoding。
7. 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。
8. 选择完成。

## 导出私有证书 (CLI)

使用 `export-certificate` 命令导出私有证书和私有密钥。运行命令时，您必须指定密码。为了提高安全性，请使用文件编辑器将密码短语存储在文件中，然后通过提供文件来提供密码短语。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。

## Note

包含该密码的文件不得以行终止符结尾。您可以如下所示检查密码文件：

```
$ file -k passphrase.txt
passphrase.txt: ASCII text, with no line terminators
```

以下示例将命令输出发送到 `jq` 以便应用 PEM 格式。

```
[Linux]
$ aws acm export-certificate \
  --certificate-arn arn:aws:acm:region:account:certificate/certificate_ID \
  --passphrase fileb://path-to-passphrase-file \
```

```
| jq -r '"\"(.Certificate)\(.CertificateChain)\(.PrivateKey)\"'

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

这会输出 base64 编码的 PEM 格式证书，还包含证书链和加密的私有密钥，如下面的简短示例所示。

```
-----BEGIN CERTIFICATE-----
MIIDTCCAjSgAwIBAgIRANWuFpqA16g3IwStE3vVpTwwDQYJKoZIhvcNAQELBQAw
EzERMA8GA1UECgwIdHJvbG9sb2wwHhcNMtkwNzE5MTYxNTU1WhcNMjAwODE5MTcx
NTU1WjAXMRUwEwYDVQDDAx3d3cuc3B1ZHMuaW8wggeiMAOGCSqGSIb3DQEBAQUA
...
8UNFQvNo01VtICL4cwW0dLokxpwkKwTcEkQuHE1v5Vn6HpbfFmxkdPEasoDhthH
FFWIf4/+VOLbDLgjU4HgtmV4IJDtqM9rGOZ42eFYmmc3eQO0GmigBBwwXp3j6hoi
74YM+igvtILnbYkPYhY9qz8h7lHUmansS8j6YxmtppY=
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIC8zCCAdugAwIBAgIRAM/jQ/6h2/MI1NYWX3dDaZswDQYJKoZIhvcNAQELBQAw
EzERMA8GA1UECgwIdHJvbG9sb2wwHhcNMtkwNjE5MTk0NTE2WhcNMjkwNjE5MjA0
NTE2WjAtMREwDwYDVQDDAh0cm9sb2xvbDCCASIdwDQYJKoZIhvcNAQEBBQADggEP
...
j2PAOviqIXjwr08Zo/rTy/8m6LASmm3LVVYKLyPdl+KB6M/+H93Z1/Bs8ERqqga/
6lFm6iw2JHtkW+q4WexvQSoqRXFhCZWBWPZTUpBS0d4/Y5q92S3iJLra/JQ0d4U1
tWZyqJ2rj2RL+h7CE71XIAM//oHGcDDPaQBFD2DTisB/+ppGeDuB
-----END CERTIFICATE-----
-----BEGIN ENCRYPTED PRIVATE KEY-----
MIIFKzBVBGkqhkiG9w0BBQ0wSDAnBgkqhkiG9w0BBQwwGgQUMrZb7kZJ8nTZg7aB
1zmaQh4vwloCAGgAMB0GCWCGSAFlAwQBKqQDViroIHStQgNOjr6nTUnuwSCBNAN
JM4SG202YPUiddWeWmX/RKGg3lIdE+AOWLTPskNcCAHqdhOSqBwt65qUTZe3gBt
...
ZGipF/DobHDMkpwiaRR5sz6nG4wcki0ryYjAQrdGsR6EVvUUXADkrrrxuHTWjF1
wEuqyd8X/ApkQsYFX/nhepOEIGWf8Xu0nrjQo77/evhG0sHXborGzGJwKuimPVy
Fs5kw5mvEoe5DAe3rSKsSUJ1tM4RagJj2WH+BC04SZWNH8kxfOC1E/GSLBCixv3v
+Lwq38CEJRQJLdpta8NcLKnFBwmmVs9OV/VXzNuHYg==
-----END ENCRYPTED PRIVATE KEY-----
```

要将所有内容输出到文件，请将 > 重定向器附加到上面的示例中，从而生成如下结果。

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

# 标记 Amazon Certificate Manager 证书

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 Amazon Certificate Manager 控制台、Amazon Command Line Interface (Amazon 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 Amazon 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 \(p. 55\)](#)
- [管理标签 \(p. 55\)](#)

## 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。
- 标签键和值区分大小写。
- 保留 `aws:` 前缀以供 Amazon 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 Amazon Web Services Management Console 的 [Resource Groups](#) 和 [标签编辑器](#) 中使用。

有关 Amazon 标记约定的一般信息，请参阅 [标记 Amazon 资源](#)。

## 管理标签

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

## 管理标签 (控制台)

您可以使用 Amazon Web Services Management Console 添加、删除或编辑标签。您也可以在列中显示标签。

### 添加标签

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. 登录到 Amazon Web Services Management Console 并打开 Amazon Certificate Manager 控制台，网址是 <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. 选择保存。

### 删除标签

使用 ACM 控制台通过以下过程删除标签。

#### 删除标签 (控制台)

1. 登录到 Amazon Web Services Management Console 并打开 Amazon Certificate Manager 控制台，网址是 <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. 选择保存。

### 编辑标签

使用 ACM 控制台通过以下过程编辑标签。

#### 编辑标签 (控制台)

1. 登录到 Amazon Web Services Management Console 并打开 Amazon Certificate Manager 控制台，网址是 <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. 选择保存。

### 在列中显示标签

在 ACM 控制台中使用以下过程在列中显示标签。

### 在列中显示标签 (控制台)

1. 登录到 Amazon Web Services Management Console 并打开 Amazon Certificate Manager 控制台，网址是 <https://console.aws.amazon.com/acm/home>。
2. 通过选择控制台右上角的齿轮图标  来选择您想显示为列的标签。
3. 选中要在列中显示的标签旁的复选框。

## 管理标签 (CLI)

请参阅以下主题以了解如何使用 Amazon 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 (ACM 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](#)

# Amazon Certificate Manager 监控和日志记录

监控是保持 Amazon Certificate Manager 和您的 Amazon 解决方案的可靠性、可用性和性能的重要部分。您应从 Amazon 解决方案的所有部分收集监控数据，以便更轻松地调试出现的多点故障。

以下主题介绍了可与 ACM 配合使用的 Amazon 云监控工具。

## 主题

- [支持的 CloudWatch 指标 \(p. 58\)](#)
- [使用 CloudWatch Events \(p. 58\)](#)
- [将 CloudTrail 用于 Amazon Certificate Manager \(p. 66\)](#)

## 支持的 CloudWatch 指标

Amazon CloudWatch 是一项针对 Amazon 资源的监控服务。您可以使用 CloudWatch 收集和跟踪指标，设置警报，并自动应对您的 Amazon 资源的变化。Amazon Certificate Manager 支持以下 CloudWatch 指标。

AWS/CertificateManager 命名空间包括以下指标。

指标	描述	单位	维度
DaysToExpiry	证书过期之前的天数。ACM 会在证书过期后停止发布此指标。	整数	CertificateArn • 值：证书的 ARN。

有关 CloudWatch 指标的更多信息，请参阅以下主题：

- [使用 Amazon CloudWatch 指标](#)
- [创建 Amazon CloudWatch 警报](#)

## 使用 CloudWatch Events

您可以使用 [Amazon CloudWatch Events](#) 自动执行您的 Amazon 服务并自动响应系统事件，例如应用程序可用性问题或资源更改。Amazon 服务中的事件将近乎实时传输到 CloudWatch Events。您可以编写简单规则来指示您关注的事件，并指示要在事件匹配规则时执行的自动化操作。有关更多信息，请参阅 [创建触发事件的 CloudWatch Events 规则](#)。

CloudWatch Events 将使用 Amazon EventBridge 转换为操作。借助 EventBridge，您可以使用事件触发目标，包括 Amazon Lambda 函数、Amazon Batch 任务、Amazon SNS 主题以及许多其他目标。有关更多信息，请参阅 [For more information, see 什么是 Amazon EventBridge ?](#)

## 主题

- [Amazon EventBridge 对 ACM 的支持 \(p. 59\)](#)

- [使用 ACM 中的 CloudWatch Events 触发操作 \(p. 60\)](#)

## Amazon EventBridge 对 ACM 的支持

本主题列出并描述了 Amazon EventBridge 支持的 ACM 相关事件。

### Amazon 运行状况事件

系统将为符合续订条件的 ACM 证书生成 Amazon 运行状况事件。有关续订资格的信息，请参阅 [ACM 证书的托管续订 \(p. 40\)](#)。

系统会在两种情况下生成运行状况事件：

- 成功续订公有或私有证书时。
- 何时客户必须完成操作后才能成功续订。这可能意味着需要点击电子邮件中的链接（用于通过电子邮件进行验证的证书），或者解决某个错误。每个事件都包含以下事件类型代码中的一个。这些代码以变量的形式公开，供您用于筛选。
  - `AWS_ACM_RENEWAL_STATE_CHANGE`（证书已续订、已过期或即将过期）
  - `CAA_CHECK_FAILURE`（CAA 检查失败）
  - `AWS_ACM_RENEWAL_FAILURE`（由私有 CA 签名的证书）

运行状况事件具有如下结构。在此示例中，`AWS_ACM_RENEWAL_STATE_CHANGE` 事件已生成。

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

### ACM 过期事件

ACM 将从证书过期前 45 天开始，为所有有效的证书（公有、私有和导入的证书）每日发送过期事件。此时间可以使用 ACM API 的 [PutAccountConfiguration](#) 操作来更改。

ACM 会为由其颁发的符合条件的证书自动启动续订，但导入的证书需要在过期之前重新颁发并重新导入 ACM，以避免发生中断。有关更多信息，请参阅 [重新导入证书](#)。您可以使用过期事件来设置自动化以将证书重新导入 ACM。有关使用 Amazon Lambda 实现自动化的示例，请参见 [使用 ACM 中的 CloudWatch Events 触发操作 \(p. 60\)](#)。

过期事件具有如下结构。

```
{
```

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

## 使用 ACM 中的 CloudWatch Events 触发操作

您可以基于这些事件创建 CloudWatch 规则，并使用 CloudWatch 控制台配置检测到事件时执行的操作。本节提供了配置 CloudWatch 规则和生成的操作的示例过程。

主题

- [通过 Amazon SNS 响应事件 \(p. 60\)](#)
- [使用 Lambda 函数响应事件 \(p. 61\)](#)

## 通过 Amazon SNS 响应事件

本节介绍如何配置 Amazon SNS，以在每当 ACM 生成运行状况事件时发送文本通知。

完成以下过程来配置响应。

创建 CloudWatch Events 规则并触发操作

1. 创建 CloudWatch Events 规则。有关更多信息，请参阅[创建触发事件的 CloudWatch Events 规则](#)。
  - a. 在 CloudWatch 控制台（位置：<https://console.aws.amazon.com/cloudwatch/>）中，导航到事件 > 规则页面，然后选择 Create rule（创建规则）。
  - b. 在创建规则页面上，选择 Event Pattern（事件模式）。
  - c. 对于服务名称，从菜单中选择 Health（运行状况）。
  - d. 对于事件类型，选择 Specific Health events（特定运行状况事件）。
  - e. 选择 Specific service(s)（特定服务），然后从菜单中选择 ACM。
  - f. 选择 Specific event type category(s)（特定事件类型类别），然后选择 accountNotification。
  - g. 选择 Any event type code（任何事件类型代码）。
  - h. 选择 Any resource（任何资源）。
  - i. 在事件模式预览编辑器中，粘贴事件发出的 JSON 模式。此示例使用 [Amazon 运行状况事件 \(p. 59\)](#) 部分中的模式。

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

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

## 2. 配置操作。

在目标部分，您可以从许多可以立即使用您的事件的服务中进行选择，例如 Amazon Simple Notification Service (SNS)，也可以选择 Lambda 函数将事件传递到自定义的可执行代码。有关 Amazon Lambda 实现的示例，请参阅 [使用 Lambda 函数响应事件](#) (p. 61)。

## 使用 Lambda 函数响应事件

此过程演示如何使用 Amazon Lambda 以监听 CloudWatch Events、使用 Amazon Simple Notification Service (SNS) 创建通知，并将结果发布到 Amazon Security Hub，为管理员和安全团队提供可见性。

### 设置 Lambda 函数和 IAM 角色

1. 首先配置 Amazon Identity and Access Management (IAM) 角色并定义 Lambda 函数所需的权限。通过此安全性最佳实践，您可以灵活地指定谁有权调用该函数，并限制授予该用户的权限。不建议直接在用户账户下运行大多数 Amazon 操作，特别是不要在管理员账户下运行。

通过以下网址打开 IAM 控制台：<https://console.aws.amazon.com/iam/>。

2. 使用 JSON 策略编辑器创建在以下模板中定义的策略。提供您自己的区域和 Amazon 账户详细信息。有关更多信息，请参阅在“JSON”选项卡上创建策略。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "LambdaCertificateExpiryPolicy1",
      "Effect": "Allow",
      "Action": "logs:CreateLogGroup",
      "Resource": "arn:aws:logs:<region>:<AWS-ACCT-NUMBER>:*"
    },
    {
      "Sid": "LambdaCertificateExpiryPolicy2",
      "Effect": "Allow",
      "Action": [
        "logs:CreateLogStream",
        "logs:PutLogEvents"
      ],
      "Resource": [
        "arn:aws:logs:<region>:<AWS-ACCT-NUMBER>:log-group:/aws/lambda/handle-expiring-certificates:*"
      ]
    },
    {
      "Sid": "LambdaCertificateExpiryPolicy3",
      "Effect": "Allow",
      "Action": [
        "acm:DescribeCertificate",
        "acm:GetCertificate",
        "acm:ListCertificates",

```

```
        "acm:ListTagsForCertificate"
    ],
    "Resource": "*"
  },
  {
    "Sid": "LambdaCertificateExpiryPolicy4",
    "Effect": "Allow",
    "Action": "SNS:Publish",
    "Resource": "*"
  },
  {
    "Sid": "LambdaCertificateExpiryPolicy5",
    "Effect": "Allow",
    "Action": [
      "SecurityHub:BatchImportFindings",
      "SecurityHub:BatchUpdateFindings",
      "SecurityHub:DescribeHub"
    ],
    "Resource": "*"
  },
  {
    "Sid": "LambdaCertificateExpiryPolicy6",
    "Effect": "Allow",
    "Action": "cloudwatch:ListMetrics",
    "Resource": "*"
  }
]
}
```

3. 创建 IAM 角色并向其附加新策略。有关创建 IAM 角色和附加策略的信息，请参阅[为 Amazon 服务创建角色 \(控制台\)](#)。
4. 打开 Amazon Lambda 控制台，地址：<https://console.aws.amazon.com/lambda/>。
5. 创建 Lambda 函数。有关更多信息，请参阅[使用控制台创建 Lambda 函数](#)。完成以下步骤：
  - a. 在创建函数页面上，选择 Author from scratch (从头开始创作) 选项以创建函数。
  - b. 在 Function name (函数名称) 字段中指定一个名称，如“handle-expiring-certificates”。
  - c. 在 Runtime (运行时) 列表中，选择“Python 3.8”。
  - d. 展开 Change default execution role (更改默认执行角色)，然后选择 Use an existing role (使用现有角色)。
  - e. 从 Existing role (现有角色) 列表中选择您先前创建的角色。
  - f. 选择 Create function (创建函数)。
  - g. 在 Function code (函数代码) 下，插入以下代码。

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

```
# HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
# OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
# SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

import json
import boto3
import os
from datetime import datetime, timedelta, timezone
# -----
# setup global data
# -----
utc = timezone.utc
# make today timezone aware
today = datetime.now().replace(tzinfo=utc)
# set up time window for alert - default to 45 if its missing
if os.environ.get('EXPIRY_DAYS') is None:
    expiry_days = 45
else:
    expiry_days = int(os.environ['EXPIRY_DAYS'])
expiry_window = today + timedelta(days = expiry_days)
def lambda_handler(event, context):
    # if this is coming from the ACM event, its for a single certificate
    if (event['detail-type'] == "ACM Certificate Approaching Expiration"):
        response = handle_single_cert(event, context.invoked_function_arn)
        return {
            'statusCode': 200,
            'body': response
        }
def handle_single_cert(event, context_arn):
    cert_client = boto3.client('acm')
    cert_details =
    cert_client.describe_certificate(CertificateArn=event['resources'][0])
    result = 'The following certificate is expiring within ' + str(expiry_days) + '
days: ' + cert_details['Certificate']['DomainName']
    # check the expiry window before logging to Security Hub and sending an SNS
    if cert_details['Certificate']['NotAfter'] < expiry_window:
        # This call is the text going into the SNS notification
        result = result + ' (' + cert_details['Certificate']['CertificateArn'] + '
'
        # this call is publishing to SH
        result = result + ' - ' + log_finding_to_sh(event, cert_details,
context_arn)
        # if there's an SNS topic, publish a notification to it
        if os.environ.get('SNS_TOPIC_ARN') is None:
            response = result
        else:
            sns_client = boto3.client('sns')
            response = sns_client.publish(TopicArn=os.environ['SNS_TOPIC_ARN'],
Message=result, Subject='Certificate Expiration Notification')
        return result
def log_finding_to_sh(event, cert_details, context_arn):
    # setup for security hub
    sh_region = get_sh_region(event['region'])
    sh_hub_arn = "arn:aws:securityhub:{0}:{1}:hub/default".format(sh_region,
event['account'])
    sh_product_arn = "arn:aws:securityhub:{0}:{1}:product/{1}/
default".format(sh_region, event['account'])
    # check if security hub is enabled, and if the hub arn exists
    sh_client = boto3.client('securityhub', region_name = sh_region)
    try:
        sh_enabled = sh_client.describe_hub(HubArn = sh_hub_arn)
        # the previous command throws an error indicating the hub doesn't exist or
lambda doesn't have rights to it so we'll stop attempting to use it
    except Exception as error:
        sh_enabled = None
    print ('Default Security Hub product doesn\'t exist')
```

```
        response = 'Security Hub disabled'
    # This is used to generate the URL to the cert in the Security Hub Findings to
    link directly to it
    cert_id = right(cert_details['Certificate']['CertificateArn'], 36)
    if sh_enabled:
        # set up a new findings list
        new_findings = []
        # add expiring certificate to the new findings list
        new_findings.append({
            "SchemaVersion": "2018-10-08",
            "Id": cert_id,
            "ProductArn": sh_product_arn,
            "GeneratorId": context_arn,
            "AwsAccountId": event['account'],
            "Types": [
                "Software and Configuration Checks/AWS Config Analysis"
            ],
            "CreatedAt": event['time'],
            "UpdatedAt": event['time'],
            "Severity": {
                "Original": '89.0',
                "Label": 'HIGH'
            },
            "Title": 'Certificate expiration',
            "Description": 'cert expiry',
            "Remediation": {
                "Recommendation": {
                    "Text": 'A new certificate for ' + cert_details['Certificate']
['DomainName'] + ' should be imported to replace the existing imported certificate
before expiration',
                    "Url": "https://console.aws.amazon.com/acm/home?region=" +
event['region'] + "#/?id=" + cert_id
                }
            },
            "Resources": [
                {
                    "Id": event['id'],
                    "Type": 'ACM Certificate',
                    "Partition": 'aws',
                    "Region": event['region']
                }
            ],
            "Compliance": {'Status': 'WARNING'}
        })
    # push any new findings to security hub
    if new_findings:
        try:
            response = sh_client.batch_import_findings(Findings=new_findings)
            if response['FailedCount'] > 0:
                print("Failed to import {}
findings".format(response['FailedCount']))
        except Exception as error:
            print("Error: ", error)
            raise
    return json.dumps(response)
# function to setup the sh region
def get_sh_region(event_region):
    # security hub findings may need to go to a different region so set that here
    if os.environ.get('SECURITY_HUB_REGION') is None:
        sh_region_local = event_region
    else:
        sh_region_local = os.environ['SECURITY_HUB_REGION']
    return sh_region_local
# quick function to trim off right side of a string
def right(value, count):
    # To get right part of string, use negative first index in slice.
```

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

h. 在 Environment variables ( 环境变量 ) 下, 选择 Edit ( 编辑 ) 并可选择添加以下变量。

- ( 可选 ) EXPIRY\_DAYS

指定发送证书过期通知之前的准备时间 ( 天数 )。此函数默认为 45 天, 但您可以指定自定义值。

- ( 可选 ) SNS\_TOPIC\_ARN

指定 Amazon SNS 的 ARN。以 `arn:aws:sns:<##>:<####>:<####>` 格式提供完整的 ARN。

- ( 可选 ) SECURITY\_HUB\_REGION

在不同的区域指定一个 Amazon Security Hub。如果未指定此选项, 则使用正在运行的 Lambda 函数的区域。如果该函数在多个区域中运行, 则可能需要将所有证书消息都转到单个区域中的 Security Hub。

i. 在 Basic settings ( 基本设置 ) 下, 将 Timeout ( 超时 ) 设置为 30 秒。

j. 在页面顶部, 选择 Deploy ( 部署 )。

完成以下过程中的任务以开始使用此解决方案。

### 自动发送过期电子邮件通知

在此示例中, 我们在通过 CloudWatch Events 引发事件时, 为每个即将过期的证书提供一个电子邮件。预设情况下, ACM 每天都会为距离过期日 45 天或更短时间的证书引发一个事件。( 此时间段可以使用 ACM API 的 `PutAccountConfiguration` 操作自定义。 ) 这些事件中的每一个都会触发以下级联的自动操作 :

```
ACM raises CloudWatch event #
>>>>>> events

    Event matches CloudWatch rule #

        Rule calls Lambda function #

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

1. 创建 Lambda 函数并配置权限。( 已完成 – 请参阅 [设置 Lambda 函数和 IAM 角色 \(p. 61\)](#) )。
2. 为 Lambda 函数创建用于发送通知的标准 SNS 主题。有关更多信息, 请参阅 [创建 Amazon SNS 主题](#)。
3. 为任何感兴趣的人订阅新 SNS 主题。有关更多信息, 请参阅 [订阅 Amazon SNS 主题](#)。
4. 创建 CloudWatch Events 规则以触发 Lambda 函数。有关更多信息, 请参阅 [创建触发事件的 CloudWatch Events 规则](#)。

在 CloudWatch 控制台 ( 位置 : <https://console.aws.amazon.com/cloudwatch/> ) 中, 导航到事件 > 规则页面, 然后选择 Create rule ( 创建规则 )。指定服务名称、事件类型和 Lambda 函数。在事件模式预览编辑器中, 粘贴以下代码 :

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

显示示例事件下会显示如 Lambda 接收这样的事件 :

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

## 清理

一旦您不再需要示例配置或任何配置，最佳实践是删除该配置的所有痕迹，以避免安全问题和以后的意外费用：

- IAM 策略和角色
- Lambda 函数
- CloudWatch Events 规则
- 与 Lambda 关联的 CloudWatch Logs
- SNS 主题

## 将 CloudTrail 用于 Amazon Certificate Manager

Amazon Certificate Manager 已与 Amazon CloudTrail 集成，后者作为一项服务提供 ACM 中由用户、角色或 Amazon 服务所执行的操作的记录。您的 Amazon 账户上默认会启用 CloudTrail。CloudTrail 将以事件的形式捕获 ACM 的所有 API 调用，包括来自 ACM 控制台的调用以及对 ACM API 操作的代码调用。如果您配置跟踪记录，则可以使 CloudTrail 事件持续传送到 Amazon S3 存储桶（包括 ACM 的事件）。如果您不配置跟踪记录，则仍可在 CloudTrail 控制台中的 Event history（事件历史记录）中查看最新事件。

使用 CloudTrail 收集的信息，您可以确定向 ACM 发出了什么请求、发出请求的 IP 地址、何人发出的请求、请求的发出时间以及其他详细信息。有关更多信息，请参阅[使用 CloudTrail 事件历史记录查看事件](#)。当 ACM 中发生受支持的事件活动时，该活动将记录在 CloudTrail 事件中，并与其他 Amazon 服务事件一同保存在事件历史记录中。您可以在 Amazon 账户中查看、搜索和下载最新事件。

此外，您可以配置其它 Amazon 服务，进一步分析在 CloudTrail 日志中收集的事件数据并采取行动。

有关 CloudTrail 的更多信息，请参阅以下文档：

- [Amazon CloudTrail 用户指南](#)。
- [创建跟踪概览](#)
- [CloudTrail 支持的服务和集成](#)
- [为 CloudTrail 配置 Amazon SNS 通知](#)
- [从多个区域接收 CloudTrail 日志文件和从多个账户接收 CloudTrail 日志文件](#)

## 主题

- [CloudTrail 日志记录中支持的 ACM API 操作 \(p. 67\)](#)
- [记录集成服务的 API 调用 \(p. 76\)](#)

## CloudTrail 日志记录中支持的 ACM API 操作

ACM 支持将以下操作记录为 CloudTrail 日志文件中的事件：

每个事件或日志条目都包含有关生成请求的人员信息。身份信息可帮助您确定以下内容：

- 请求是使用根用户凭证还是 Amazon Identity and Access Management (IAM) 用户凭证发出的
- 请求是使用角色还是联合身份用户的临时安全凭证发出的
- 请求是否由其它 Amazon 服务发出

有关更多信息，请参阅 [CloudTrail userIdentity 元素](#)。

以下部分提供了支持的 API 操作的示例日志。

- [向证书添加标签 \(AddTagsToCertificate\) \(p. 67\)](#)
- [删除证书 \(DeleteCertificate\) \(p. 68\)](#)
- [描述证书 \(DescribeCertificate\) \(p. 68\)](#)
- [导出证书 \(ExportCertificate\) \(p. 69\)](#)
- [导入证书 \(ImportCertificate\) \(p. 70\)](#)
- [列出证书 \(ListCertificates\) \(p. 72\)](#)
- [列出证书的标签 \(ListTagsForCertificate\) \(p. 72\)](#)
- [从证书中删除标签 \(RemoveTagsFromCertificate\) \(p. 73\)](#)
- [请求证书 \(RequestCertificate\) \(p. 74\)](#)
- [重新发送验证电子邮件 \(ResendValidationEmail\) \(p. 74\)](#)
- [检索证书 \(GetCertificate\) \(p. 75\)](#)

### 向证书添加标签 (AddTagsToCertificate)

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/
fedcba98-7654-3210-fedc-ba9876543210"
  },
  "responseElements":null,
  "requestID":"fedcba98-7654-3210-fedc-ba9876543210",
  "eventID":"fedcba98-7654-3210-fedc-ba9876543210",
  "eventType":"AwsApiCall",
  "recipientAccountId":"123456789012"
}
]
```

## 删除证书 (DeleteCertificate)

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/
fedcba98-7654-3210-fedc-ba9876543210"
      },
      "responseElements":null,
      "requestID":"01234567-89ab-cdef-0123-456789abcdef",
      "eventID":"01234567-89ab-cdef-0123-456789abcdef",
      "eventType":"AwsApiCall",
      "recipientAccountId":"123456789012"
    }
  ]
}
```

## 描述证书 (DescribeCertificate)

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

### Note

`DescribeCertificate` 操作的 CloudTrail 日志不会显示有关您指定的 ACM 证书的信息。您可以使用控制台、Amazon 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/fedcba98-7654-3210-fedc-ba9876543210"
      },
      "responseElements": null,
      "requestID": "fedcba98-7654-3210-fedc-ba9876543210",
      "eventID": "fedcba98-7654-3210-fedc-ba9876543210",
      "eventType": "AwsApiCall",
      "recipientAccountId": "123456789012"
    }
  ]
}
```

## 导出证书 (ExportCertificate)

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

```
{
  "Records": [
    {
      "version": "0",
      "id": "01234567-89ab-cdef-0123-456789abcdef",
      "detail-type": "AWS API Call via CloudTrail",
      "source": "aws.acm",
      "account": "123456789012",
      "time": "2018-05-24T15:28:11Z",
      "region": "us-east-1",
      "resources": [
      ],
      "detail": {
        "eventVersion": "1.04",
        "userIdentity": {
          "type": "Root",
          "principalId": "123456789012",
          "arn": "arn:aws:iam::123456789012:user/Alice",
          "accountId": "123456789012",
          "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
          "userName": "Alice"
        },
        "eventTime": "2018-05-24T15:28:11Z",
        "eventSource": "acm.amazonaws.com",
        "eventName": "ExportCertificate",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "192.0.2.0",
      }
    }
  ]
}
```

```

        "userAgent": "aws-cli/1.15.4 Python/2.7.9 Windows/8 botocore/1.10.4",
        "requestParameters": {
            "passphrase": {
                "hb": [
                    42,
                    42,
                    42,
                    42,
                    42,
                    42,
                    42,
                    42,
                    42,
                    42,
                    42,
                    42
                ],
                "offset": 0,
                "isReadOnly": false,
                "bigEndian": true,
                "nativeByteOrder": false,
                "mark": -1,
                "position": 0,
                "limit": 10,
                "capacity": 10,
                "address": 0
            },
            "certificateArn": "arn:aws:acm:us-east-1:123456789012:certificate/
fedcba98-7654-3210-fedc-ba9876543210"
        },
        "responseElements": {
            "certificateChain": {
                "-----BEGIN CERTIFICATE-----
                base64 certificate
                -----END CERTIFICATE-----
                -----BEGIN CERTIFICATE-----
                base64 certificate
                -----END CERTIFICATE-----",
            "privateKey": "*****",
            "certificate": {
                "-----BEGIN CERTIFICATE-----
                base64 certificate
                -----END CERTIFICATE-----"
            }
        },
        "requestID": "01234567-89ab-cdef-0123-456789abcdef",
        "eventID": "fedcba98-7654-3210-fedc-ba9876543210",
        "eventType": "AwsApiCall"
    }
}
]
}

```

## 导入证书 (ImportCertificate)

以下示例显示了记录对 ACM `ImportCertificate` API 操作的调用 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/01234567-89ab-
cdef-0123-456789abcdef"
},
"requestID":"01234567-89ab-cdef-0123-456789abcdef",
"eventID":"01234567-89ab-cdef-0123-456789abcdef",
"eventType":"AwsApiCall",
```

```
} "recipientAccountId": "111122223333"  
}
```

## 列出证书 (ListCertificates)

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

### Note

[ListCertificates](#) 操作的 CloudTrail 日志不会显示您的 ACM 证书。您可以使用控制台、Amazon 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": "cdf1051-88aa-4aa3-8c33-a325270bff21",  
      "eventType": "AwsApiCall",  
      "recipientAccountId": "123456789012"  
    }  
  ]  
}
```

## 列出证书的标签 (ListTagsForCertificate)

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

### Note

[ListTagsForCertificate](#) 操作的 CloudTrail 日志不会显示您的标签。您可以使用控制台、Amazon 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": "2016-03-18T00:00:43Z",  
      "eventSource": "acm.amazonaws.com",  
      "eventName": "ListTagsForCertificate",  
      "awsRegion": "us-east-1",  
      "sourceIPAddress": "192.0.2.0",  
      "userAgent": "aws-cli/1.9.15",  
      "requestParameters": {  
        "certificateId": "123456789012",  
        "maxItems": 1000  
      },  
      "responseElements": null,  
      "requestID": "74c99844-ec9c-11e5-ac34-d1e4dfe1a11b",  
      "eventID": "cdf1051-88aa-4aa3-8c33-a325270bff21",  
      "eventType": "AwsApiCall",  
      "recipientAccountId": "123456789012"  
    }  
  ]  
}
```

```
    "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 (RemoveTagsFromCertificate)

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": "0cfa142e-ef74-4b21-9515-47197780c424",
      "eventType": "AwsApiCall",
      "recipientAccountId": "123456789012"
    }
  ]
}
```

## 请求证书 (RequestCertificate)

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 (ResendValidationEmail)

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 (GetCertificate)

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"  
  }  
]  
}
```

## 记录集成服务的 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 [Amazon CloudTrail 用户指南](#)。以下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 Amazon 资源。

### 主题

- [创建负载均衡器 \(p. 76\)](#)

## 创建负载均衡器

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 [Amazon CloudTrail 用户指南](#)。以下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 Amazon 资源。

### 主题

- [创建负载均衡器 \(p. 76\)](#)
- [使用负载均衡器注册 Amazon EC2 实例 \(p. 77\)](#)
- [加密私有密钥 \(p. 78\)](#)
- [解密私有密钥 \(p. 78\)](#)

## 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 Amazon Command Line Interface 来完成。以下示例显示了对 Amazon 账户 123456789012 上负载均衡器 LinuxTest 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 Amazon 中执行。

```
{
  "Records":[
    {
      "eventVersion":"1.03",
      "userIdentity":{
        "type":"IAMUser",
        "principalId":"AIDACKCEVSQ6C2EXAMPLE",
        "arn":"arn:aws:iam::111122223333:user/acm",
        "accountId":"111122223333",
        "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
        "userName":"acm"
      },
      "eventTime":"2016-01-05T18:36:29Z",
      "eventSource":"kms.amazonaws.com",
      "eventName":"Encrypt",
      "awsRegion":"us-east-1",
      "sourceIPAddress":"AWS Internal",
      "userAgent":"aws-internal",
      "requestParameters":{
        "keyId":"arn:aws:kms:us-east-1:123456789012:alias/aws/acm",
        "encryptionContext":{
          "aws:acm:arn":"arn:aws:acm:us-east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
        }
      },
      "responseElements":null,
      "requestID":"3c417351-b3db-11e5-9a24-7d9457362fcc",
      "eventID":"1794fe70-796a-45f5-811b-6584948f24ac",
      "readOnly":true,
      "resources":[
        {
          "ARN":"arn:aws:kms:us-east-1:123456789012:key/87654321-4321-4321-4321-210987654321",
          "accountId":"123456789012"
        }
      ],
      "eventType":"AwsServiceEvent",
      "recipientAccountId":"123456789012"
    }
  ]
}
```

## 解密私有密钥

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 Amazon 中进行，并且解密的密钥绝不会离开 Amazon。

```
{
  "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 API ( Java 示例 )

您可使用 Amazon 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 [Amazon Certificate Manager API 参考](#)。

除 Web API ( 或 HTTP API ) 以外，您还可以使用 Amazon 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 [用于 Amazon Web Services 的工具](#)。

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

## 主题

- [向证书添加标签 \(p. 80\)](#)
- [删除证书 \(p. 81\)](#)
- [描述证书 \(p. 83\)](#)
- [导出证书 \(p. 84\)](#)
- [检索证书和证书链 \(p. 86\)](#)
- [导入证书 \(p. 88\)](#)
- [列出证书 \(p. 90\)](#)
- [续订证书 \(p. 92\)](#)
- [列出证书标签 \(p. 93\)](#)
- [从证书中删除标签 \(p. 95\)](#)
- [请求证书 \(p. 96\)](#)
- [重新发送验证电子邮件 \(p. 98\)](#)

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

import com.amazonaws.auth.AWSStaticCredentialsProvider;
import com.amazonaws.auth.BasicAWSCredentials;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.certificatemanager.AWSCertificateManager;
import com.amazonaws.services.certificatemanager.AWSCertificateManagerClientBuilder;
import com.amazonaws.services.certificatemanager.model.ImportCertificateRequest;
import com.amazonaws.services.certificatemanager.model.ImportCertificateResult;
/**
 * This sample demonstrates how to use the ImportCertificate function in the Amazon
 * Certificate Manager
 * service.
 *
 * Input parameters:
 * Accesskey - Amazon access key
 * SecretKey - Amazon secret key
 * CertificateArn - Use to reimport a certificate (not included in this example).
```

```
* region - Amazon region
* Certificate - PEM file that contains the certificate to import. Ex: /data/certs/
servercert.pem
* CertificateChain - The certificate chain, not including the end-entity certificate.
* PrivateKey - The private key that matches the public key in the certificate.
*
* Output parameter:
* CertificateArn - The ARN of the imported certificate.
*
*/
public class AWSCertificateManagerSample {

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

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

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

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

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

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

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

        client.addTagsToCertificate(addTagsToCertificateRequest);
    }

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

## 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.AWSStaticCredentialsProvider;
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.regions.Regions;

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

/**
 * This sample demonstrates how to use the DeleteCertificate function in the Amazon
 * 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 Amazon  
 * 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)。(公有证书无论是由 ACM 颁发还是导入均无法导出。) 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.AWSStaticCredentialsProvider;
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.regions.Regions;

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

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

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

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

public class ExportCertificate {

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

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

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

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

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

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

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

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

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

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

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

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

// Export the certificate.
ExportCertificateResult result = null;

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

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

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

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

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

## 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.AWSStaticCredentialsProvider;
import com.amazonaws.auth.AWSCredentials;

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

/**
 * This sample demonstrates how to use the GetCertificate function in the Amazon
 * 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.AWSStaticCredentialsProvider;
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.regions.Regions;

import com.amazonaws.services.certificatemanager.model.ImportCertificateRequest;
import com.amazonaws.services.certificatemanager.model.ImportCertificateResult;
import com.amazonaws.services.certificatemanager.model.LimitExceededException;
import com.amazonaws.services.certificatemanager.model.ResourceNotFoundException;
import com.amazonaws.AmazonClientException;
import java.io.FileNotFoundException;
import java.io.IOException;

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

/**
```

```
* This sample demonstrates how to use the ImportCertificate function in the Amazon
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 AWSStaticCredentialsProvider(credentials))
            .build();

        // Initialize the file descriptors.
        RandomAccessFile file_certificate = null;
        RandomAccessFile file_chain = null;
        RandomAccessFile file_key = null;

        // Initialize the buffers.
        ByteBuffer buf_certificate = null;
        ByteBuffer buf_chain = null;
        ByteBuffer buf_key = null;

        // Create the file streams for reading.
        try {
            file_certificate = new RandomAccessFile("C:\\Temp\\certificate.pem", "r");
            file_chain = new RandomAccessFile("C:\\Temp\\chain.pem", "r");
            file_key = new RandomAccessFile("C:\\Temp\\private_key.pem", "r");
        }
        catch (IllegalArgumentException ex) {
            throw ex;
        }
        catch (SecurityException ex) {
            throw ex;
        }
        catch (FileNotFoundException ex) {
            throw ex;
        }

        // Create channels for mapping the files.
        FileChannel channel_certificate = file_certificate.getChannel();
        FileChannel channel_chain = file_chain.getChannel();
        FileChannel channel_key = file_key.getChannel();
    }
}
```

```
// Map the files to buffers.
try {
    buf_certificate = channel_certificate.map(FileChannel.MapMode.READ_ONLY, 0,
channel_certificate.size());
    buf_chain = channel_chain.map(FileChannel.MapMode.READ_ONLY, 0,
channel_chain.size());
    buf_key = channel_key.map(FileChannel.MapMode.READ_ONLY, 0, channel_key.size());

    // The files have been mapped, so clean up.
    channel_certificate.close();
    channel_chain.close();
    channel_key.close();
    file_certificate.close();
    file_chain.close();
    file_key.close();
}
catch (IOException ex)
{
    throw ex;
}

// Create a request object and set the parameters.
ImportCertificateRequest req = new ImportCertificateRequest();
req.setCertificate(buf_certificate);
req.setCertificateChain(buf_chain);
req.setPrivateKey(buf_key);

// Import the certificate.
ImportCertificateResult result = null;
try {
    result = client.importCertificate(req);
}
catch(LimitExceededException ex)
{
    throw ex;
}
catch (ResourceNotFoundException ex)
{
    throw ex;
}

// Clear the buffers.
buf_certificate.clear();
buf_chain.clear();
buf_key.clear();

// Retrieve and display the certificate ARN.
String arn = result.getCertificateArn();
System.out.println(arn);
}
}
```

## 列出证书

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

import com.amazonaws.AmazonClientException;

import java.util.Arrays;
import java.util.List;

/**
 * This sample demonstrates how to use the ListCertificates function in the Amazon
 * 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
  }
}]
}
```

## 续订证书

以下示例说明如何使用 [RenewCertificate](#) 函数。该函数续订由私有证书颁发机构 (CA) 颁发并通过 [ExportCertificate](#) 函数导出的私有证书。此时，使用此函数仅可续订导出的私有证书。要使用 ACM 续订您的 ACM PCA 证书，您必须先向 ACM 服务委托人授予执行此操作的权限。有关更多信息，请参阅[将证书续订权限分配给 ACM](#)。

```
package com.amazonaws.samples;

import com.amazonaws.AmazonClientException;

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

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

import com.amazonaws.services.certificatemanager.model.RenewCertificateRequest;
import com.amazonaws.services.certificatemanager.model.RenewCertificateResult;

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

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

public class RenewCertificate {
```

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

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

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

    // Create a request object and specify the ARN of the certificate to renew.
    RenewCertificateRequest req = new RenewCertificateRequest();
    req.withCertificateArn("arn:aws:acm:region:account:"
        +"certificate/M12345678-1234-1234-1234-123456789012");

    // Renew the certificate.
    RenewCertificateResult result = null;
    try {
        result = client.renewCertificate(req);
    }
    catch(InvalidArnException ex)
    {
        throw ex;
    }
    catch (ResourceNotFoundException ex)
    {
        throw ex;
    }
    catch (ValidationException ex)
    {
        throw ex;
    }

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

## 列出证书标签

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.AWSStaticCredentialsProvider;
import com.amazonaws.regions.Regions;

/**
 * This sample demonstrates how to use the ListTagsForCertificate function in the Amazon
 * Certificate
 * Manager service.
 *
 * Input parameter:
 * CertificateArn - The ARN of the certificate whose tags you want to list.
 */

public class AWSCertificateManagerExample {

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

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

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

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

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

        // Create a result object.
        ListTagsForCertificateResult result = null;
        try {
            result = client.listTagsForCertificate(req);
        }
        catch(InvalidArnException ex) {
            throw ex;
        }
        catch(ResourceNotFoundException ex) {
            throw ex;
        }

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

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}]}
```

## 从证书中删除标签

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

import java.util.ArrayList;

/**
 * This sample demonstrates how to use the RemoveTagsFromCertificate function in the Amazon
 * 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 Amazon
 * 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.AWSSStaticCredentialsProvider;  
import com.amazonaws.auth.AWSCredentials;  
import com.amazonaws.regions.Regions;  
  
/**  
 * This sample demonstrates how to use the ResendValidationEmail function in the Amazon  
 * 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 AWSSStaticCredentialsProvider(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。

# 问题排查

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

## Note

如果您在本节中未找到您的问题，建议您访问[Amazon 知识中心](#)。

## 主题

- [排查证书请求问题 \(p. 100\)](#)
- [排查证书验证问题 \(p. 101\)](#)
- [排查托管证书续订的问题 \(p. 106\)](#)
- [排查其他问题 \(p. 111\)](#)
- [处理异常 \(p. 3\)](#)

## 排查证书请求问题

如果在请求 ACM 证书时遇到问题，请参考以下主题。

## 主题

- [证书请求超时 \(p. 100\)](#)
- [证书请求失败 \(p. 100\)](#)

## 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。要更正此情况，请打开控制台，找到证书的记录，单击其复选框，选择 Actions (操作)，然后选择 Delete (删除)。然后，选择 Actions (操作) 和 Request a certificate (请求证书)，再次开始。有关更多信息，请参阅 [DNS 验证 \(p. 30\)](#) 或 [电子邮件验证 \(p. 33\)](#)。建议您尽量使用 DNS 验证。

## 证书请求失败

如果您的请求失败，ACM 和您会收到以下错误消息之一，请按照建议的步骤解决此问题。您无法重新提交失败的证书请求，请在解决问题后提交新的请求。

## 主题

- [错误消息：没有可用的联系人 \(p. 100\)](#)
- [错误消息：域被禁用 \(p. 101\)](#)
- [错误消息：需要其他验证 \(p. 101\)](#)
- [错误消息：无效的公有域 \(p. 101\)](#)
- [错误消息：其他 \(p. 101\)](#)

## 错误消息：没有可用的联系人

您在请求证书时选择了电子邮件验证，但 ACM 无法找到用于验证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 \(p. 33\)](#)发送到的位置。

完成上述任务之一便可解决此问题；您无需同时完成两项任务。在解决此问题后，请求一个新证书。

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 \(p. 25\)](#) 或 [未收到验证电子邮件 \(p. 104\)](#)。如果您遵循这些步骤并继续收到无可联系消息，请[将此情况报告给 Amazon](#)，以便我们可以进行调查。

## 错误消息：域被禁用

VirusTotal 将证书请求中的一个或多个域名报告为不安全域。要更正问题，请尝试以下操作：

- 在 [VirusTotal](#) 网站上搜索您的域名。如果您的域被报告为可疑，请参阅 [Google 针对被黑客攻击网站的帮助](#) 以了解可以执行的操作。
- 如果您认为结果为误报，请通知报告了您的域的组织。VirusTotal 是多个防病毒和 URL 扫描程序的集合体，本身并不能将您的域从黑名单中删除。

在您更正问题并更新了 VirusTotal 注册表之后，请求一个新的证书。

如果您看到此错误但您的域未包含在 VirusTotal 列表中，请访问[Amazon 支持中心](#)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](#)。

## 错误消息：需要其他验证

ACM 需要其他信息来处理此证书请求。如果您的域名属于 [Alexa 1000 强网站](#)，这将用作防欺诈措施。要提供此必要信息，请使用[支持中心](#)联系 Amazon Web Services Support。如果您没有支持计划，请在 [ACM 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。

## 错误消息：无效的公有域

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。再次尝试请求证书，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均属于有效的顶级域。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 [ACM 开发论坛](#)中发布新话题。

## 错误消息：其他

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。再次尝试请求证书，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消息，请使用[支持中心](#)联系 Amazon Web Services Support。如果您没有支持计划，请在 [ACM 开发论坛](#)中发布新话题。

# 排查证书验证问题

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[电子邮件验证 \(p. 33\)](#)。如果选择 DNS 验证，则必须将 ACM 为您创建的别名记录写入您的 DNS 数据库。有关更多信息，请参阅[DNS 验证 \(p. 30\)](#)。

#### 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 \(p. 104\)](#)。如果选择 DNS 验证，则必须为每个域创建一条 CNAME 记录。

#### Note

公有 ACM 证书可以安装在连接到 [Nitro Enclave \(p. 5\)](#) 的 Amazon EC2 实例上，但不能安装到其他 Amazon EC2 实例上。有关在未连接到 Nitro Enclave 的 Amazon EC2 实例上设置独立 Web 服务器的信息，请参阅[教程：在 Amazon Linux 2 上安装 LAMP Web 服务器](#)或者[教程：使用 Amazon Linux AMI 安装 LAMP Web 服务器](#)。

建议使用 DNS 验证而不是电子邮件验证。

如果遇到 DNS 验证问题，请参考以下主题。

#### 主题

- [排查 DNS 验证问题 \(p. 102\)](#)
- [排查电子邮件验证的问题 \(p. 104\)](#)

## 排查 DNS 验证问题

如果在使用 DNS 验证证书时遇到问题，请参阅以下指南。

#### Tip

DNS 问题排查的第一步是使用以下所示工具检查域的当前状态：

- dig – [Linux](#)、[Windows](#)
- nslookup – [Linux](#)、[Windows](#)
- whois – [Linux](#)、[Windows](#)

#### 主题

- [DNS 提供商禁用下划线 \(p. 102\)](#)
- [DNS 提供商添加的默认尾部句点 \(p. 103\)](#)
- [GoDaddy 上的 DNS 验证失败 \(p. 103\)](#)
- [ACM 控制台不显示“Create record in Route 53”（在 Route 53 中创建记录）按钮 \(p. 103\)](#)
- [私有（不受信任）域上的 Route 53 验证失败 \(p. 103\)](#)
- [VPN 上的 DNS 服务器验证失败 \(p. 104\)](#)

## DNS 提供商禁用下划线

如果您的 DNS 提供商禁止在别名记录值中使用前导下划线，您可以删除 ACM 提供的值内的下划线，并在没有该下划线的情况下验证域。例如，为了进行验证，可将别名记录值 `_x2.acm-validations.aws` 更改为 `x2.acm-validations.aws`。但是，别名记录名称参数必须始终以前导下划线开始。

您可以使用下表右侧两个值中的任意一个来验证域。

名称	类型	值
<code>_&lt;random value&gt;.example.com.</code>	别名记录	<code>&lt;random value&gt;.acm-validations.aws.</code>
<code>&lt;random value&gt;.example.com.</code>	别名记录	<code>&lt;random value&gt;.acm-validations.aws.</code>

## DNS 提供商添加的默认尾部句点

预设情况下，某些 DNS 提供商会向您提供的别名记录值添加尾部句点。因此，您自己添加句点可能会导致错误。例如，“`<random_value>.acm-validations.aws.`”将被拒绝，而“`<random_value>.acm-validations.aws`”将被接受。

## GoDaddy 上的 DNS 验证失败

除非您修改 ACM 提供的别名记录值，否则注册到 GoDaddy 和其他注册管理机构的域名的 DNS 验证可能会失败。以 .com 域名为例，发布的 CNAME 记录格式如下：

```
NAME: _ho9hv39800vb3examplew3vnewoib3u.example.com.
VALUE: _cjhvou20vhu2exampleuw20vuyb2ovb9.j9s73ucn9vy.acm-validations.aws.
```

您可以通过截断“NAME (名称)”字段末尾的顶点域（包括句点），创建与 GoDaddy 兼容的 CNAME 记录，如下所示：

```
NAME: _ho9hv39800vb3examplew3vnewoib3u
VALUE: _cjhvou20vhu2exampleuw20vuyb2ovb9.j9s73ucn9vy.acm-validations.aws.
```

## ACM 控制台不显示“Create record in Route 53”（在 Route 53 中创建记录）按钮

如果您选择 Amazon Route 53 作为您的 DNS 提供商，Amazon Certificate Manager 可以直接与其交互来验证您的域所有权。在某些情况下，控制台的 Create record in Route 53（在 Route 53 中创建记录）按钮可能未按预期可用。如果发生这种情况，请检查以下可能原因。

- 在验证页面上，您没有单击域名旁边的向下箭头。
- 您没有使用 Route 53 作为您的 DNS 提供商。
- 您使用不同账户登录 ACM 和 Route 53。
- 您缺少在 Route 53 托管的区域中创建记录的 IAM 权限。
- 您或其他人已验证该域。
- 该域不可公开寻址。

## 私有（不受信任）域上的 Route 53 验证失败

在 DNS 验证过程中，ACM 在公有托管区域中搜索别名记录。如果没有找到，则在 72 小时后超时，状态为 Validation timed out (验证超时)。您不能使用它来托管私有域的 DNS 记录，包括 Amazon VPC [私有托管区](#)中的资源、私有 PKI 中的不受信任域或自行签名的证书。

Amazon 确实通过 [ACM 私有 CA](#)服务为公开不受信任的域提供支持。

## VPN 上的 DNS 服务器验证失败

如果在 VPN 上找到 DNS 服务器，而 ACM 无法针对该服务器验证证书，请检查该服务器是否可以公开访问。使用 ACM DNS 验证颁发的公有证书要求域记录可以通过公有 Internet 进行解析。

## 排查电子邮件验证的问题

如果在使用电子邮件验证证书域时遇到问题，请参阅以下指南。

### 主题

- [未收到验证电子邮件 \(p. 104\)](#)
- [已发送到子域的电子邮件 \(p. 105\)](#)
- [隐藏的联系人信息 \(p. 105\)](#)
- [证书续订 \(p. 106\)](#)
- [WHOIS 限制 \(p. 106\)](#)
- [用于电子邮件验证的持久初始时间戳 \(p. 106\)](#)
- [排查 .IO 顶级域的问题 \(p. 106\)](#)

## 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[电子邮件验证 \(p. 33\)](#)。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 Amazon 账户所有者，除非该所有者也作为域联系人在 WHOIS 中列出。检查在 ACM 控制台中显示（或者从 CLI 或 API 中返回）的电子邮件地址列表，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 Amazon 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，预设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 [Amazon 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. 25) 处的说明。如果您使用的是 Amazon WorkMail，请参阅 Amazon WorkMail 管理员指南中的使用用户。

在提供 Amazon 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，您可以通过 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 Amazon Web Services Management Console 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（申请证书）向导的 Step 3: Validate（步骤 3：验证）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电子邮件验证 (p. 33) 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 Amazon 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 Amazon CLI，Amazon 不会执行 MX 查找。

#### 联系支持中心

如果在查看上述指导后，您仍无法收到域验证电子邮件，请访问 Amazon Web Services 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` 以及电子邮件验证 (p. 33) 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. 25)。

您可以使用 RequestCertificate API 或 request-certificate Amazon 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 Amazon CLI，Amazon 不会执行 MX 查找。

## 隐藏的联系人信息

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，以及由您的 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联系人电子邮件地址。

## 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 Amazon 外部。也就是说，Amazon 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 [Amazon Web Services Support 中心](#) 创建一个案例以寻求解决方法。

## 用于电子邮件验证的持久初始时间戳

证书的第一个电子邮件验证请求的时间戳在后续验证续订请求中一直保留。这不是 ACM 操作中存在错误的证据。

## 排查 .IO 顶级域的问题

.IO 顶级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 [DNS 验证 \(p. 30\)](#)。

## 排查托管证书续订的问题

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 [ACM 证书的托管续订 \(p. 40\)](#) 有任何疑问，请参阅以下主题。

## 准备进行自动域验证

要让 ACM 自动续订您的证书，必须满足以下条件：

- 您的证书必须和与 ACM 集成的某项 Amazon 服务关联。有关 ACM 支持的资源的信息，请参阅 [与 Amazon Certificate Manager 集成的服务 \(p. 3\)](#)。
- 对于使用电子邮件验证的证书，ACM 必须能够通过证书中所列每个域的管理员电子邮件地址与您联系。将尝试的电子邮件地址列在 [电子邮件验证 \(p. 33\)](#) 中。
- 对于使用 DNS 验证的证书，请确保您的 DNS 配置包含正确的 CNAME 记录，如 [DNS 验证 \(p. 30\)](#) 中所述。

## 处理托管证书续订失败

当证书还剩 60 天过期时，ACM 会自动每小时尝试续订。如果 ACM 在 15 天后无法续订证书，您将收到一封电子邮件，其中包含有关如何手动解决续订问题的进一步说明。此过程因最初验证证书的方式而异。

### 针对电子邮件验证证书的托管证书续订

ACM 证书的有效期为 13 个月 (395 天)。要续订，通过电子邮件验证的证书需要域所有者执行操作。ACM 在过期前 45 天开始向域的 WHOIS 邮箱地址和五个常用管理员地址发送续订通知。通知中包含一个链接，域所有者可以单击该链接以轻松续订。验证所有列出的域后，ACM 会颁发具有相同 ARN 的续订证书。

请参阅 [使用电子邮件验证 \(p. 33\)](#)，了解有关识别哪些域处于 PENDING\_VALIDATION 状态并重复这些域的验证过程的说明。

### 针对 DNS 验证证书的托管证书续订

ACM 不会尝试对 DNS 验证的证书进行 TLS 验证。如果 ACM 无法续订您通过 DNS 验证来验证的证书，则很可能是由于 DNS 配置中缺少别名记录或别名记录不准确。如果发生这种情况，ACM 会通知您无法自动续订证书。

#### Important

您必须将正确的 CNAME 记录插入到 DNS 数据库中。请咨询您的域名注册商以了解如何执行此操作。

您可以通过在 ACM 控制台中展开证书及其域条目，找到域的 CNAME 记录。有关详细信息，请参考下面的图。您还可以使用 ACM API 中的 [DescribeCertificate](#) 操作，或是 ACM CLI 中的 [describe-certificate](#) 命令，来检索别名记录。有关更多信息，请参阅 [DNS 验证 \(p. 30\)](#)。

<input type="checkbox"/>	Name ▾	Domain name ▾	Additional names
<input type="checkbox"/>		amzn1.example.biz	
<input type="checkbox"/>		amzn2.example.biz	
<input type="checkbox"/>		amzn3.example.biz	

### Status

**Status** Issued  
**Detailed status** The certificate was issued at 2018-03-22T22:42:...

**Domain**

amzn3.example.biz

[Export DNS configuration to a file](#) You can export all of the CNAME records to a file

### Details

<b>Type</b>	Amazon Issued
<b>In use?</b>	No
<b>Domain name</b>	amzn3.example.biz
<b>Number of additional names</b>	0
<b>Identifier</b>	<a href="#">1fae4ec1-6db6-4d3d-967a-eec5e53ecd45</a>
<b>Serial number</b>	0e:10:30:f3:1c:b4:1e:b7:54:bb:f3:99:62:5b:7f:fb

### Tags

<input type="text" value="Name"/>	<input type="text" value=""/>
-----------------------------------	-------------------------------

从控制台中选择目标证书。



amzn3.example

## Status

<b>Status</b>	Issued
<b>Detailed status</b>	The certificate 2018-03-22

### Domain

▼ amzn3.example.biz

Add the following CNAME record to the  
your DNS service Provider. [Learn more.](#)

#### Name

\_dc8d107e33e2a83816b6a2a395a  
ample.biz.

**Note:** Changing the DNS configuration  
You can revoke permission at any time

展开证书窗口以查找证书的 CNAME 信息。

如果问题依旧存在，请联系[支持中心](#)。

## 了解续订计时

ACM 证书的托管续订 (p. 40) 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会有延迟。ACM 获取续订的证书的时间与将证书部署到使用它的 Amazon 资源的时间之间可能出现额外延迟。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。

## 排查其他问题

本部分提供了与颁发或验证 ACM 证书无关的问题的指导。

### 主题

- [证书颁发机构授权 \(CAA\) 问题 \(p. 111\)](#)
- [证书导入问题 \(p. 111\)](#)
- [证书固定问题 \(p. 112\)](#)
- [API Gateway 问题 \(p. 112\)](#)
- [工作证书意外失败时该怎么办 \(p. 112\)](#)
- [ACM 服务关联角色 \(SLR\) 问题 \(p. 113\)](#)

## 证书颁发机构授权 (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错误消息，显示 One or more domain names have failed validation due to a Certification Authority Authentication (CAA) error [由于证书颁发机构认证 (CAA) 错误，一个或多个域名未能通过验证]，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。CAA 记录中的 value (值) 字段必须包含以下域名中的一个：

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

有关创建 CAA 记录的更多信息，请参阅 [\(可选\) 配置 CAA 记录 \(p. 26\)](#)。

### Note

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。

## 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。
- 如果您的证书由 CA 签名，则必须包含中间证书链，该证书链可提供到达证书颁发机构根目录的路径。

- 如果您的证书为自签名证书，则必须以明文形式包含私有密钥。
- 链中的每个证书都必须直接认证前一个证书。
- 请勿在中间证书链中包含最终实体证书。
- 证书、证书链和私有密钥（如有）必须采用 PEM 编码。通常，PEM 编码由 Base64 编码的 ASCII 文本块组成，这些文本块以明文页眉和页脚行开头和结尾。在复制或上载 PEM 文件时，不得添加行或空格或对文件进行任何其他更改。您可以使用 [OpenSSL 验证实用程序](#) 验证证书链。
- 不得加密您的私有密钥（如有）。（提示：如果它有密码短语，则会加密。）
- 与 ACM 集成的服务必须使用 ACM 支持的算法和密钥大小。请参阅 Amazon Certificate Manager 用户指南及各项服务的文档，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导入 IAM 还是 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。但在预设情况下，如果调用 [ListCertificates](#) API 或 [list-certificates](#) Amazon CLI 命令，但未指定 `keyTypes` 筛选条件，则将只显示 RSA\_1024 或 RSA\_2048 证书。

## 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 \(p. 21\)](#)（有时称为 SSL 固定）来固定 ACM 证书，则在 Amazon 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：

- 将您自己的证书导入到 ACM ([p. 48](#))，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。
- 如果您使用的是公有证书，则将您的应用程序固定到所有可用的 [Amazon 根证书](#)。如果您使用的是私有证书，则将您的应用程序固定到 CA 的根证书。

## API Gateway 问题

当您部署边缘优化的 API 终端节点时，API Gateway 将为您设置 CloudFront 分配。CloudFront 分配由 API Gateway 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 Gateway 自定义域。

当您部署区域 API 终端节点时，API Gateway 将代表您创建一个 Application Load Balancer (ALB)。该负载均衡器由 API Gateway 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 Gateway 自定义域。

## 工作证书意外失败时该怎么办

如果您已成功将 ACM 证书与集成服务关联，但证书停止工作，并且集成服务开始返回错误，则原因可能是服务使用 ACM 证书所需的权限发生了更改。

例如，Elastic Load Balancing (ELB) 需要权限才能对 Amazon KMS key 进行解密，而它反过来又能解密证书的私有密钥。此权限由基于资源的策略授予，当您证书与 ELB 关联时 ACM 将应用该策略。如果 ELB 失去对该权限的授予，则将在下次尝试解密证书密钥时失败。

若要调查该问题，请使用位于 <https://console.amazonaws.cn/kms> 的 Amazon KMS 控制台检查您的授权状态。执行下列操作之一：

- 如果您认为已撤销授予集成服务的权限，请访问集成服务的控制台，取消证书与服务的关联，然后重新关联它。这将重新应用基于资源的策略，并设置新的授权。
- 如果您认为已撤销授予 ACM 的权限，请通过以下地址联系 Amazon Web Services Support：<https://console.aws.amazon.com/support/home#/>。

## ACM 服务关联角色 (SLR) 问题

当您颁发由另一个账户与您共享的私有 CA 签名的证书时，ACM 会在首次使用时尝试设置与服务关联的角色 (SLR)，作为委托人与 ACM Private CA [基于资源的访问策略](#) 进行交互。如果您从共享 CA 颁发私有证书，而不存在 SLR，ACM 将无法为您自动续订该证书。

ACM 可能会提示您，它无法确定您的账户中是否存在 SLR。如果所需的 `iam:GetRole` 权限已被授予您账户的 ACM SLR，则在创建 SLR 后不会再发出提示。如果提示再次发生，那么您或您的账户管理员可能需要授予 `iam:GetRole` 访问 ACM 的权限，或者将您的账户与 ACM 托管策略 `AWSCertificateManagerFullAccess` 关联。

有关更多信息，请参阅 IAM 用户指南中的 [服务相关角色权限](#)。

## 处理异常

Amazon Certificate Manager 命令可能出于几个原因而失败。有关每个异常的信息，请参阅下表。

### 私有证书异常处理

当您尝试续订由 ACM Private CA 颁发的私有 PKI 证书时，可能会出现以下异常。

#### Note

中国（北京）区域和中国（宁夏）区域不支持 ACM Private CA。

ACM 故障代码	Comment
PCA_ACCESS_DENIED	私有 CA 未授予 ACM 权限。这将触发 <code>PCA AccessDeniedException</code> 故障代码。  若要解决此问题，请使用 <code>PCA CreatePermission</code> 操作向 ACM 服务委托人授予必要的权限。
PCA_INVALID_DURATION	请求的证书的有效期限超过了颁发的私有 CA 的有效期限。这将触发 <code>PCA ValidationException</code> 故障代码。  要解决此问题，请 <a href="#">安装新 CA 证书</a> （具有适当的有效期限）。
PCA_INVALID_STATE	将调用的私有 CA 未处于执行请求的 ACM 操作所需的正确状态。这将触发 <code>PCA InvalidStateException</code> 故障代码。  按如下方式解决此问题： <ul style="list-style-type: none"><li>• 如果 CA 的状态为 <code>CREATING</code>，请等待创建操作完成，然后安装 CA 证书。</li><li>• 如果 CA 的状态为 <code>PENDING_CERTIFICATE</code>，请安装 CA 证书。</li><li>• 如果 CA 的状态为 <code>DISABLED</code>，请将其更新为 <code>ACTIVE</code> 状态。</li><li>• 如果 CA 的状态为 <code>DELETED</code>，请将其还原。</li><li>• 如果 CA 的状态为 <code>EXPIRED</code>，请安装新证书</li></ul>

ACM 故障代码	Comment
	<ul style="list-style-type: none"><li>如果 CA 的状态为 FAILED，而您无法解决问题，请联系 <a href="#">Amazon Web Services Support</a>。</li></ul>
PCA_LIMIT_EXCEEDED	<p>私有 CA 已达到颁发配额。这将触发 PCA LimitExceededException 故障代码。在继续使用此帮助之前，请尝试重复您的请求。</p> <p>如果错误仍然存在，请联系 <a href="#">Amazon Web Services Support</a> 以请求提高配额。</p>
PCA_REQUEST_FAILED	<p>网络或系统出错。这将触发 PCA RequestFailedException 故障代码。在继续使用此帮助之前，请尝试重复您的请求。</p> <p>如果错误仍然存在，请联系 <a href="#">Amazon Web Services Support</a>。</p>
PCA_RESOURCE_NOT_FOUND	<p>私有 CA 已被永久删除。这将触发 PCA ResourceNotFoundException 故障代码。验证是否使用了正确的 ARN。如果验证失败，您将无法使用此 CA。</p> <p>要解决此问题，请 <a href="#">创建新 CA</a>。</p>
SLR_NOT_FOUND	<p>为了续订由位于另一个账户中的私有 CA 签名的证书，ACM 要求证书所在的账户上具有服务关联角色 (SLR)。如果需要重新创建已删除的 SLR，请参阅 <a href="#">为 ACM 创建 SLR (p. 18)</a>。</p>

# Concepts

本节提供了 Amazon Certificate Manager 所用概念的定义。

## 主题

- [ACM 证书 \(p. 115\)](#)
- [ACM 根 CA \(p. 116\)](#)
- [顶级域 \(p. 117\)](#)
- [非对称密钥加密 \(p. 117\)](#)
- [证书颁发机构 \(p. 117\)](#)
- [证书透明度日志 \(p. 117\)](#)
- [域名系统 \(p. 118\)](#)
- [域名 \(p. 118\)](#)
- [加密和解密 \(p. 119\)](#)
- [完全限定域名 \(FQDN\) \(p. 119\)](#)
- [公有密钥基础设施 \(p. 119\)](#)
- [根证书 \(p. 119\)](#)
- [安全套接字层 \(SSL\) \(p. 119\)](#)
- [安全 HTTPS \(p. 119\)](#)
- [SSL 服务器证书 \(p. 120\)](#)
- [对称密钥加密 \(p. 120\)](#)
- [传输层安全性 \(TLS\) \(p. 120\)](#)
- [Trust \(p. 120\)](#)

## ACM 证书

ACM 生成 X.509 版本 3 证书。每个证书有效期为 13 个月 ( 395 天 ) ，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。

ACM 颁发的证书的纯文本类似于以下示例：

```
Certificate:
  Data:
    Version: 3 (0x2)
    Serial Number:
      f2:16:ad:85:d8:42:d1:8a:3f:33:fa:cc:c8:50:a8:9e
    Signature Algorithm: sha256WithRSAEncryption
    Issuer: O=Example CA
    Validity
      Not Before: Jan 30 18:46:53 2018 GMT
```

```
Not After : Jan 31 19:46:53 2018 GMT
Subject: C=US, ST=VA, L=Herndon, O=Amazon, OU=AWS, CN=example.com
Subject Public Key Info:
  Public Key Algorithm: rsaEncryption
  Public-Key: (2048 bit)
  Modulus:
    00:ba:a6:8a:aa:91:0b:63:e8:08:de:ca:e7:59:a4:
    69:4c:e9:ea:26:04:d5:31:54:f5:ec:cb:4e:af:27:
    e3:94:0f:a6:85:41:6b:8e:a3:c1:c8:c0:3f:1c:ac:
    a2:ca:0a:b2:dd:7f:c0:57:53:0b:9f:b4:70:78:d5:
    43:20:ef:2c:07:5a:e4:1f:d1:25:24:4a:81:ab:d5:
    08:26:73:f8:a6:d7:22:c2:4f:4f:86:72:0e:11:95:
    03:96:6d:d5:3f:ff:18:a6:0b:36:c5:4f:78:bc:51:
    b5:b6:36:86:7c:36:65:6f:2e:82:73:1f:c7:95:85:
    a4:77:96:3f:c0:96:e2:02:94:64:f0:3a:df:e0:76:
    05:c4:56:a2:44:72:6f:8a:8a:a1:f3:ee:34:47:14:
    bc:32:f7:50:6a:e9:42:f5:f4:1c:9a:7a:74:1d:e5:
    68:09:75:19:4b:ac:c6:33:90:97:8c:0d:d1:eb:8a:
    02:f3:3e:01:83:8d:16:f6:40:39:21:be:1a:72:d8:
    5a:15:68:75:42:3e:f0:0d:54:16:ed:9a:8f:94:ec:
    59:25:e0:37:8e:af:6a:6d:99:0a:8d:7d:78:0f:ea:
    40:6d:3a:55:36:8e:60:5b:d6:0d:b4:06:a3:ac:ab:
    e2:bf:c9:b7:fe:22:9e:2a:f6:f3:42:bb:94:3e:b7:
    08:73
  Exponent: 65537 (0x10001)
X509v3 extensions:
  X509v3 Basic Constraints:
    CA:FALSE
  X509v3 Authority Key Identifier:
    keyid:84:8C:AC:03:A2:38:D9:B6:81:7C:DF:F1:95:C3:28:31:D5:F7:88:42
  X509v3 Subject Key Identifier:
    97:06:15:F1:EA:EC:07:83:4C:19:A9:2F:AF:BA:BB:FC:B2:3B:55:D8
  X509v3 Key Usage: critical
    Digital Signature, Key Encipherment
  X509v3 Extended Key Usage:
    TLS Web Server Authentication, TLS Web Client Authentication
  X509v3 CRL Distribution Points:
    Full Name:
      URI:http://example.com/crl

Signature Algorithm: sha256WithRSAEncryption
69:03:15:0c:fb:a9:39:a3:30:63:b2:d4:fb:cc:8f:48:a3:46:
69:60:a7:33:4a:f4:74:88:c6:b6:b6:b8:ab:32:c2:a0:98:c6:
8d:f0:8f:b5:df:78:a1:5b:02:18:72:65:bb:53:af:2f:3a:43:
76:3c:9d:d4:35:a2:e2:1f:29:11:67:80:29:b9:fe:c9:42:52:
cb:6d:cd:d0:e2:2f:16:26:19:cd:f7:26:c5:dc:81:40:3b:e3:
d1:b0:7e:ba:80:99:9a:5f:dd:92:b0:bb:0c:32:dd:68:69:08:
e9:3c:41:2f:15:a7:53:78:4d:33:45:17:3e:f2:f1:45:6b:e7:
17:d4:80:41:15:75:ed:c3:d4:b5:e3:48:8d:b5:0d:86:d4:7d:
94:27:62:84:d8:98:6f:90:1e:9c:e0:0b:fa:94:cc:9c:ee:3a:
8a:6e:6a:9d:ad:b8:76:7b:9a:5f:d1:a5:4f:d0:b7:07:f8:1c:
03:e5:3a:90:8c:bc:76:c9:96:f0:4a:31:65:60:d8:10:fc:36:
44:8a:c1:fb:9c:33:75:fe:a6:08:d3:89:81:b0:6f:c3:04:0b:
a3:04:a1:d1:1c:46:57:41:08:40:b1:38:f9:57:62:97:10:42:
8e:f3:a7:a8:77:26:71:74:c2:0a:5b:9e:cc:d5:2c:c5:27:c3:
12:b9:35:d5
```

## ACM 根 CA

由 ACM 颁发的公有终端实体证书从以下 Amazon 根 CA 获得其信任：

可分辨名称	加密算法
CN=Amazon Root CA 1,O=Amazon,C=US	2048 位 RSA (RSA_2048)
CN=Amazon Root CA 2,O=Amazon,C=US	4096 位 RSA (RSA_4096)
CN=Amazon Root CA 3,O=Amazon,C=US	Elliptic Prime Curve 256 位 (EC_prime256v1)
CN=Amazon Root CA 4,O=Amazon,C=US	Elliptic Prime Curve 384 位 (EC_secp384r1)

ACM 所颁发证书的默认信任根为 CN=Amazon Root CA 1,O=Amazon,C=US，这提供了 2048 位的 RSA 安全性。其他根保留以供将来使用。所有根都由 Starfield Services Root Certificate Authority 证书交叉签名。

有关更多信息，请参阅 [Amazon Trust Services](#)。

## 顶级域

请参阅 [域名 \(p. 118\)](#)。

## 非对称密钥加密

非对称加密不同于[对称密钥加密 \(p. 120\)](#)，它使用不同的但在数学上相关的密钥加密和解密内容。密钥之一是公有密钥，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) 的身份绑定到公有密钥。

ACM 证书是 X.509 SSL/TLS 证书，它将您网站的身份和企业的详细信息绑定到证书中包含的公有密钥。ACM 使用 Amazon KMS key 加密私有密钥。有关更多信息，请参阅 [ACM 私有密钥安全性 \(p. 10\)](#)。

## 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有密钥。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三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 \(p. 22\)](#)。

## 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 `111.122.133.144`)。不过，域的 DNS 数据库包含大量其他用途的记录。例如，通过 ACM，您可以使用别名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[DNS 验证 \(p. 30\)](#)。

## 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。

### Subdomain

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` 是适用于 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。

#### Registration

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. 118)。

## 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。它们都使用 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 \(p. 117\)](#)。

## 传输层安全性 (TLS)

请参阅 [安全套接字层 \(SSL\) \(p. 119\)](#)。

## Trust

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。

# 文档历史记录

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

变更	说明	日期
<a href="#">更新用于导入的密钥算法类型 (p. 121)</a>	导入 ACM 的证书现在拥有的密钥可能具有其他 RSA 和椭圆曲线算法。有关当前支持的密钥算法的列表，请参阅 <a href="https://docs.amazonaws.cn/acm/latest/userguide/import-certificate-prerequisites.html">https://docs.amazonaws.cn/acm/latest/userguide/import-certificate-prerequisites.html</a> 。	2021 年 7 月 14 日
<a href="#">将“监控和日志”提升为单独一章。(p. 121)</a>	将监控和日志文档移至其自己的章节。此更改涵盖 CloudWatch Metrics、CloudWatch Events/EventBridge 和 CloudTrail。有关更多信息，请参阅 <a href="https://docs.amazonaws.cn/acm/latest/userguide/monitoring-and-logging.html">https://docs.amazonaws.cn/acm/latest/userguide/monitoring-and-logging.html</a> 。	2021 年 3 月 23 日
<a href="#">增加了 CloudWatch 指标和事件支持 (p. 121)</a>	增加了 DaysToExpiry 指标和事件以及支持 API。有关更多信息，请参阅 <a href="https://docs.amazonaws.cn/acm/latest/userguide/cloudwatch-metrics.html">https://docs.amazonaws.cn/acm/latest/userguide/cloudwatch-metrics.html</a> 和 <a href="https://docs.amazonaws.cn/acm/latest/userguide/cloudwatch-events.html">https://docs.amazonaws.cn/acm/latest/userguide/cloudwatch-events.html</a> 。	2021 年 3 月 3 日
<a href="#">增加了跨账户支持 (p. 121)</a>	增加了对使用 ACM Private CA 的私有 CA 的跨账户支持。有关更多信息，请参阅 <a href="https://docs.amazonaws.cn/acm/latest/userguide/ca-access.html">https://docs.amazonaws.cn/acm/latest/userguide/ca-access.html</a> 。	2020 年 8 月 17 日
<a href="#">增加了区域支持 (p. 121)</a>	增加了对 Amazon 中国（北京和宁夏）地区的区域支持。有关支持区域的完整列表，请参阅 <a href="https://docs.amazonaws.cn/general/latest/gr/rande.html#acm-pca_region">https://docs.amazonaws.cn/general/latest/gr/rande.html#acm-pca_region</a> 。	2020 年 3 月 4 日
<a href="#">增加了续订工作流程测试 (p. 121)</a>	客户现在可以手动测试 ACM 托管续订工作流程的配置。有关更多信息，请参阅 <a href="#">测试 ACM 的托管续订配置</a> 。	2019 年 3 月 14 日
<a href="#">现在默认启用证书透明度日志 (p. 121)</a>	增加了预设情况下将 ACM 公有证书发布到证书透明度日志的功能。	2018 年 2 月 4 日
<a href="#">启动 ACM Private CA (p. 121)</a>	发布了 ACM Private Certificate Manager (CM) 和 Amazon Certificate Manager 扩展，允许用	2018 年 4 月 4 日

证书透明度日志 (p. 121)	户构建安全的托管基础设施以用于颁发和撤销私有数字证书。有关更多信息，请参阅 <a href="#">Amazon Private Certificate Authority</a> 。	在最佳实践中增加了证书透明度日志记录。 2018 年 3 月 27 日
------------------	--	-------------------------------------

下表介绍了 2018 年之前的 Amazon Certificate Manager 文档发布历史记录。

更改	描述	发行日期
新增内容	在 <a href="#">DNS 验证 (p. 30)</a> 中添加了 DNS 验证相关内容。	2017 年 11 月 21 日
新增内容	向 <a href="#">使用 API (Java 示例) (p. 80)</a> 添加了新的 Java 代码示例。	2017 年 10 月 12 日
新增内容	向 <a href="#">(可选) 配置 CAA 记录 (p. 26)</a> 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 <a href="#">问题排查 (p. 100)</a> 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 <a href="#">重新导入证书 (p. 51)</a> 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 <a href="#">最佳实践 (p. 21)</a> 和 <a href="#">问题排查 (p. 100)</a> 中。	2017 年 07 月 7 日
新增内容	已将 Amazon CloudFormation 添加到 <a href="#">与 Amazon Certificate Manager 集成的服务 (p. 3)</a> 。	2017 年 5 月 27 日
更新	已将更多信息添加到 <a href="#">配额 (p. 6)</a> 。	2017 年 5 月 27 日
新增内容	添加了有关 <a href="#">Amazon Certificate Manager 的身份和访问管理 (p. 10)</a> 的文档。	2017 年 4 月 28 日
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 <a href="#">电子邮件验证 (p. 33)</a> 。	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 <a href="#">(可选) 为域配置电子邮件 (p. 25)</a> 。	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 <a href="#">检查证书的续订状态 (p. 45)</a> 。	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	增加了对 Amazon Elastic Beanstalk 和 Amazon API	2017 年 3 月 21 日

更改	描述	发行日期
	Gateway 的支持。请参阅 <a href="#">与 Amazon Certificate Manager 集成的服务 (p. 3)</a> 。	
更新	更新了有关 <a href="#">托管续订 (p. 40)</a> 的文档。	2017 年 2 月 20 日
新增内容	添加了有关 <a href="#">导入证书 (p. 48)</a> 的文档。	2016 年 10 月 13 日
新增内容	增加了对 ACM 操作的 Amazon CloudTrail 支持。请参阅 <a href="#">将 CloudTrail 用于 Amazon Certificate Manager (p. 66)</a> 。	2016 年 3 月 25 日
新指南	此版本引入了 Amazon Certificate Manager。	2016 年 1 月 21 日