aws

用户指南

AWS 证书 Manager

版本 1.0

AWS 证书 Manager: 用户指南

Table of Contents

什么是 AWS Certificate Manager？

AWS Certificate Manager (ACM) 负责创建、存储和续订保护您的网站和应用程序的公共和私有 SSL/TLS X.509 证书和密钥的复杂性。 AWS 您可以直接通过 ACM 签发证书，或者通过将第三方证书导入 ACM 管理系统中，为集成 AWS 服务提供证书。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。您也可以导出由签名的 ACM 证书，AWS 私有 CA 以便在内部 PKI 中的任何地方使用。

> **Note**
>
> ACM 并不合适独立 webserver 使用。如果您想在 Amazon EC2 实例上设置独立的安全服务器，以下教程包含相关说明：在 Amazon Linux 2023 上配置 SSL/TLS。

主题

ACM 服务是否适合我？

AWS 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. AWS Certificate Manager (ACM)-此服务适用于需要使用 TLS 实现安全网站的企业客户。ACM 证书通过 Elastic Load Balancing、Amazon CloudFront、Amazon API Gateway 和其他集成 AWS 服务部署。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。
2. AWS 私有 CA—此服务适用于在 AWS 云端构建公钥基础架构 (PKI) 并供组织内部私人使用的企业客户。使用 AWS 私有 CA，您可以创建自己的证书颁发机构 (CA) 层次结构，并使用它颁发证书，

用于对用户、计算机、应用程序、服务、服务器和其他设备进行身份验证。无法在 Internet 上使用私有 CA 所颁发的证书。有关更多信息，请参阅 AWS 私有 CA 《用户指南》。

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。

证书颁发机构和层次结构

您通过 ACM 申请的公共证书是从 Amazon Trust Services 获得的，这是 Amazon 管理的公共证书颁发机构 (CA)。Amazon Root CA 1 到 4 由名为 Starfield G2 Root Certificate Authority - G2 的早期根交叉签名。Starfield 根在 Android 设备上受到信任，从较高版本的 Gingerbread 开始，而在 iOS 上则从 4.1 版本开始受到信任。iOS 从版本 11 开始信任 Amazon 根。任何包含 Amazon 或 Starfield 根的浏览器、应用程序或操作系统都将信任从 ACM 获得的公共证书。

ACM 向客户颁发的叶或终端实体证书的授权来自 Amazon Trust Services 根 CA，通过多个中间 CA 中的任何一个提供。ACM 根据申请的证书类型（RSA 或 ECDSA）随机分配中间 CA。由于中间 CA 是在生成请求后随机选择的，因此 ACM 不提供中间 CA 信息。

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。

中间 CA 和根 CA 轮换

为了保持弹性和灵活的证书基础设施，Amazon 可以随时选择终止中间 CA，恕不另行通知。此类变更对客户没有影响。有关更多信息，请参阅博客文章 "Amazon introduces dynamic intermediate certificate authorities"（Amazon 引入动态中间证书颁发机构）。

万一 Amazon 终止根 CA，则将在情况需要时尽快执行此类变更。由于此类变更的影响很大，Amazon将使用所有可用的机制来通知 AWS 客户 AWS Health Dashboard，包括向账户所有者发送电子邮件以及联系技术客户经理。

用于吊销的防火墙访问权

如果终端实体证书不再可信，则该证书将被吊销。OCSP 和 CRL 是用于验证证书是否已被吊销的标准机制。OCSP 和 CRL 是用于发布吊销信息的标准机制。某些客户防火墙可能需要额外的规则才能使这些机制发挥作用。

以下示例 URL 通配符模式可用于识别吊销流量。星号 (*) 通配符代表一个或多个字母数字字符，问号 (?) 代表单个字母数字字符，哈希标记 (#) 则代表数字。

- OCSP

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

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

- CRL

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

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

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 和 DNS 验证。

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。

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。

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。

密钥算法

证书必须指定算法和密钥大小。目前，ACM 支持以下 RSA 和椭圆曲线数字签名算法 (ECDSA) 公有密钥算法。ACM 可以使用标有星号 (*) 的算法请求颁发新证书。导入的证书仅支持其余算法。

> **Note**
>
> 当您请求由 CA 签名的私有 PKI 证书时 AWS Private CA，指定的签名算法系列（RSA 或 ECDSA）必须与 CA 密钥的算法系列相匹配。

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

ECDSA 密钥更小，提供的安全性与 RSA 密钥相当，但计算效率更高。但是，并非所有网络客户端都支持 ECDSA。下表改编自 NIST，显示了 RSA 和 ECDSA 具有代表性的安全强度以及各种大小的密钥。所有值均以位为单位。

比较算法和密钥的安全性

安全强度	RSA 密钥大小	ECDSA 密钥大小
128	3072	256
192	7680	384
256	15360	512

安全强度被理解为 2 的幂，与破解加密所需的猜测次数有关。例如，3072 位 RSA 密钥和 256 位 ECDSA 密钥都可以通过不超过 2^{128} 次猜测来检索。

有关帮助您选择算法的信息，请参阅中的 AWS 博客文章《如何评估和使用 ECDSA 证书》。 AWS Certificate Manager

> **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。
> - 有关信息 CloudFront，请参阅支持的 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	备注
example.com	不适用	不适用	✓	不是以“<character><character>--”开头

域名	满足条件 1	满足条件 2	已允许	备注
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	不支持	✗	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。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 与之集成的服务 AWS Certificate Manager。

支持的区域

访问 AWS 一般参考 中的 AWS 区域和端点或 AWS 区域表以查看 ACM 的区域可用性。

ACM 中的证书属于区域性资源。要将证书与 Elastic Load Balancing 一起用于多个区域中的相同完全限定域名 (FQDN) 或一组 FQDN，您必须为每个 AWS 区域申请或导入证书。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 上使用 ACM 证书 CloudFront，您必须在美国东部（弗吉尼亚北部）地区申请或导入该证书。此区域中与分配关联的 ACM 证书将 CloudFront 分发到为该分配配置的所有地理位置。

与之集成的服务 AWS Certificate Manager

AWS Certificate Manager 支持越来越多的 AWS 服务。您不能将您的 ACM 证书或私有 AWS 私有 CA 证书直接安装在您的网站 AWS 或应用程序上。

> **Note**
>
> 公有 ACM 证书可以安装在连接到 Nitro Enclave 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 是一项网络服务，它通过从全球边缘站点网络交付您的内容，加快向最终用户分发动态和静态网页内容的速度。当最终用户请求您提供的内容时 CloudFront，该用户将被路由到延迟最低的边缘站点。这样可以确保尽可能以最佳性能传输内容。如果内容当前位于该边缘位置，则立即 CloudFront 交付。如果内容当前不在该边缘位置，则会将其从您已确定为最终内容来

源的 Amazon S3 存储桶或 Web 服务器中 CloudFront 检索。有关更多信息 CloudFront，请参阅《Amazon CloudFront 开发者指南》。

要通过 SSL/TLS 提供安全内容， CloudFront 需要在 CloudFront 分发版或支持的内容源上安装 SSL/TLS 证书。ACM 已与集成， CloudFront 以便在发行版上部署 ACM 证书。 CloudFront 有关更多信息，请参阅获取 SSL/TLS 证书。

> **Note**
>
> 要将 ACM 证书与一起使用 CloudFront，您必须在美国东部（弗吉尼亚北部）地区申请或导入该证书。

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。用户可以使用您的 AWS 账户 凭据直接登录，也可以通过第三方（例如Facebook、亚马逊、谷歌或苹果）登录。有关 Amazon Cognito 的更多信息，请参阅《Amazon Cognito 开发人员指南》。

当您将 Cognito 用户池配置为使用 Amazon CloudFront 代理时， CloudFront 可以设置 ACM 证书来保护自定义域。在这种情况下，请注意，必须先删除证书与的关联， CloudFront 然后才能将其删除。

AWS Elastic Beanstalk

Elastic Beanstalk 可帮助您在云端部署和管理应用程序 AWS ，而不必担心运行这些应用程序的基础架构。 AWS Elastic Beanstalk 降低了管理复杂性。您只需上载应用程序，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 AWS 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。

AWS App Runner

App Runner 是一项 AWS 服务，它提供了一种快速、简单且经济实惠的方式，可将源代码或容器映像直接部署到 AWS 云中可扩展且安全的 Web 应用程序。您无需学习新技术、决定使用哪种计算服务，也不需要知道如何预置和配置 AWS 资源。有关 App Runner 的更多信息，请参阅 AWS App Runner 开发人员指南。

当您将自定义域名与 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 AWS 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 AWS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》。

AWS Nitro 飞地

AWS Nitro Enclaves 是 Amazon EC2 的一项功能，允许您从 Amazon EC2 实例创建隔离的执行环境，称为安全区。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 AWS Certificate Manager。

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

AWS CloudFormation

AWS CloudFormation 帮助您建模和设置亚马逊 Web Services 资源。您可以创建一个描述要使用的 AWS 资源的模板，例如 Elastic Load Balancing 或 API Gateway。然后， AWS 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 AWS 资源，也不需要弄清楚哪些资源依赖于什么； AWS CloudFormation 可以处理所有这些。ACM 证书作为模板资源提供，这意味着它 AWS CloudFormation 可以请求 ACM 证书，您可以将这些证书与 AWS 服务一起使用以启用安全连接。此外，您可以设置的许多 AWS 资源中都包含了 ACM 证书。 AWS CloudFormation

有关的一般信息 CloudFormation，请参阅《AWS CloudFormation 用户指南》。有关所支持的 ACM 资源的信息 CloudFormation，请参阅AWS::CertificateManager::Certificate。

借助提供的强大自动化功能 AWS CloudFormation，很容易超过您的证书配额，对于新 AWS 账户尤其如此。我们建议您遵循以下方面的 ACM 最佳实践。 AWS CloudFormation

> **Note**
>
> 如果您使用创建 ACM 证书，则 AWS CloudFormation 堆栈将保持 AWS CloudFormation 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。

AWS 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其上构建全栈应用程序。 AWS 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 使用一组界面组件构建前端 ready-to-use 用户界面，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《AWS 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。

亚马逊 OpenSearch 服务

Amazon S OpenSearch ervice 是一个搜索和分析引擎，用于日志分析、实时应用程序监控和点击流分析等用例。有关更多信息，请参阅《亚马逊 OpenSearch 服务开发者指南》。

创建包含自定义域和终端节点的 OpenSearch 服务集群时，可以使用 ACM 为关联的 Application Load Balancer 配置证书。

AWS Network Firewall

AWS Network Firewall 是一项托管服务，可让您轻松为您的所有 Amazon 虚拟私有云 (VPC) 部署基本网络保护。有关更多信息，请参阅 AWS Network Firewall 开发人员指南。

Network Firewall 防火墙与 ACM 集成，用于进行 TLS 检查。如果您在 Network Firewall 中使用 TLS 检查，则必须配置 ACM 证书才能解密和重新加密通过您防火墙的 SSL/TLS 流量。有关 Network Firewall 如何使用 ACM 进行 TLS 检查，请参阅《AWS Network Firewall 开发人员指南》中的使用 SSL/TLS 证书的 TLS 检查配置要求。

网站签章和信任徽标

Amazon 不提供网站签章，也不允许其商标用作以下用途之一：

- AWS Certificate Manager (ACM) 不提供可在您的网站上使用的安全网站印章。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 AWS Certificate Manager (ACM) 服务配额适用于每个 AWS 账户的每个 AWS 区域。

要了解可以调整哪些配额，请参阅 AWS 一般参考指南中的 ACM 配额表。要请求增加配额，请在 AWS 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。 来自 CA 签名的证书 AWS 私有 CA 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	您的账户配额的两倍

项目	默认配额
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 AWS Support 中心。 来自 CA 签名的证书 AWS 私有 CA 不计入此总数。	
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
每份 ACM 证书的域名数量 每份 ACM 证书的默认配额为 10 个域名。您的配额可以提升。 您提交的第一个域名作为证书的主题公用名 (CN) 包含在内。所有名称都包含在主题替代名称扩展中。 您最多可以请求 100 个域名。要申请增加配额，请在 Service Quotas 控制台中为 ACM 服务创建请求。但在创建案例之前，请务必了解在使用电子邮件验证的情况下，添加更多域名会给您带来更多的管理工作。有关更多信息，请参阅域验证。 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。	10

项目	默认配额
私有 CA 的数量 ACM 已与 AWS Private Certificate Authority (AWS 私有 CA) 集成。您可以使用 ACM 控制台或 ACM API 向托管的现有私有证书颁发机构 (CA) 申请私有证书。 AWS CLI AWS 私有 CA这些证书在 ACM 环境中进行管理，具有与 ACM 颁发的公有证书相同的限制。有关更多信息，请参阅 请求私有 PKI 证书。您也可以使用独立 AWS 私有 CA 服务颁发私有证书。有关更多信息，请参阅颁发私有最终实体证书。 已删除的私有 CA 将计入配额内，直到还原周期结束为止。有关更多信息，请参阅删除私有 CA。	200
每个 CA 的私有证书数量（生命周期）	1000000

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 会拒绝原本是有效的请求。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 AWS 私有 CA的外部 `IssueCertificate` 操作。有关 up-to-date 速率配额的信息`IssueCertificate`，请参阅终端节点和配额 AWS 私有 CA。

每个 ACM API 操作的 R equests-per-second 配额

API 调用	每秒请求数
AddTagsToCertificate	5
DeleteCertificate	10
DescribeCertificate	10
ExportCertificate	5
GetAccountConfiguration	1
GetCertificate	10
ImportCertificate	1
ListCertificates	8
ListTagsForCertificate	10
PutAccountConfiguration	1
RemoveTagsFromCertificate	5
RenewCertificate	5
RequestCertificate	5
ResendValidationEmail	1
UpdateCertificateOptions	5

有关更多信息，请参阅 AWS Certificate Manager API 参考。

的定价 AWS Certificate Manager

对于使用 AWS Certificate Manager管理的 SSL/TLS 证书，您不需要支付额外费用。您只需为为运行网站或应用程序而创建的 AWS 资源付费。有关最新的 ACM 定价信息，请参阅 AWS 网站上的AWS Certificate Manager 服务定价页面。

安全性 AWS Certificate Manager

云安全 AWS 是重中之重。作为 AWS 客户，您可以受益于专为满足大多数安全敏感型组织的要求而构建的数据中心和网络架构。

安全是双方共同承担 AWS 的责任。责任共担模式将其描述为云的安全性和云中的安全性：

- 云安全 — AWS 负责保护在 AWS 云中运行 AWS 服务的基础架构。 AWS 还为您提供可以安全使用的服务。作为AWS 合规计划合规计划合规计划合的一部分，第三方审计师定期测试和验证我们安全的有效性。要了解适用的合规计划 AWS Certificate Manager，请参阅按合规计划划分的范围内的 AWSAWS 服务按合规计划。
- 云端安全-您的责任由您使用的 AWS 服务决定。您还需要对其他因素负责，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。

本文档可帮助您了解在使用 AWS Certificate Manager (ACM) 时如何应用分担责任模型。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。您还将学习如何使用其他 AWS 服务来帮助您监控和保护您的 ACM 资源。

主题

中的数据保护 AWS Certificate Manager

分 AWS 担责任模型适用于中的数据保护 AWS Certificate Manager。如本模型所述 AWS ，负责保护运行所有内容的全球基础架构 AWS Cloud。您负责维护对托管在此基础架构上的内容的控制。您还负责您所使用的 AWS 服务 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。有关欧洲数据保护的信息，请参阅 AWS Security Blog 上的 AWS Shared Responsibility Model and GDPR 博客文章。

出于数据保护目的，我们建议您保护 AWS 账户 凭证并使用 AWS IAM Identity Center 或 AWS Identity and Access Management (IAM) 设置个人用户。这样，每个用户只获得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数据：

- 对每个账户使用多重身份验证（MFA）。
- 使用 SSL/TLS 与资源通信。 AWS 我们要求使用 TLS 1.2，建议使用 TLS 1.3。
- 使用设置 API 和用户活动日志 AWS CloudTrail。
- 使用 AWS 加密解决方案以及其中的所有默认安全控件 AWS 服务。
- 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 Amazon S3 中的敏感数据。
- 如果您在 AWS 通过命令行界面或 API 进行访问时需要经过 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（如您客户的电子邮件地址）放入标签或自由格式文本字段（如名称字段）。这包括您 AWS 服务 使用控制台、API 或 AWS SDK 与 ACM 或其他人合作时。AWS CLI在用于名称的标签或自由格式文本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网址，强烈建议您不要在网址中包含凭证信息来验证对该服务器的请求。

证书私有密钥的安全性

当您请求公共证书时， AWS Certificate Manager (ACM) 会生成公钥/私钥对。对于导入的证书，您可生成密钥对。公有密钥将成为证书的一部分。ACM 存储证书及其相应的私钥，并使用 AWS Key Management Service (AWS KMS) 来帮助保护私钥。该过程的工作方式如下所示：

1. 首次在某个 AWS 区域申请或导入证书时，ACM 会创建一个别名为 aw s/acm AWS KMS key 的托管证书。此 KMS 密钥在每个 AWS 账户和每个 AWS 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 使用相同的 KMS 密钥来加密特定 AWS 账户和特定 AWS 区域中所有证书的私钥。
3. 将证书关联到与 AWS Certificate Manager集成的服务时，ACM 会将证书和加密的私有密钥发送给服务。中还创建了一个授权 AWS KMS ，允许服务使用 KMS 密钥解密证书的私钥。有关授权的更多信息，请参阅 AWS 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 与之集成的服务 AWS Certificate Manager。

> **Note**
>
> 您可以控制自动创建的 AWS 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。

适用于 Identity and Access Managem AWS Certificate Manager

AWS Identity and Access Management (IAM) AWS 服务 可帮助管理员安全地控制对 AWS 资源的访问权限。IAM 管理员控制可以通过身份验证（登录）和授权（具有权限）使用 ACM 资源的人员。您可以使用 IAM AWS 服务 ，无需支付额外费用。

主题

受众

您的使用方式 AWS Identity and Access Management (IAM) 会有所不同，具体取决于您在 ACM 中所做的工作。

服务用户 – 如果使用 ACM 服务来完成任务，则您的管理员会为您提供所需的凭证和权限。当您使用更多 ACM 功能来完成工作时，您可能需要额外权限。了解如何管理访问权限有助于您向管理员请求适合的权限。如果您无法访问 ACM 中的功能，请参阅 对 AWS Certificate Manager 身份和访问进行故障排除。

服务管理员 – 如果您在公司负责管理 ACM 资源，则您可能具有 ACM 的完全访问权限。您有责任确定您的服务用户应访问哪些 ACM 功能和资源。然后，您必须向 IAM 管理员提交请求以更改服务用户的

权限。请查看该页面上的信息以了解 IAM 的基本概念。要了解有关您的公司如何将 IAM 与 ACM 搭配使用的更多信息，请参阅 如何 AWS Certificate Manager 与 IAM 配合使用。

IAM 管理员 – 如果您是 IAM 管理员，您可能希望了解有关如何编写策略以管理对 ACM 的访问权限的详细信息。要查看您可在 IAM 中使用的 ACM 基于身份的策略示例，请参阅 基于身份的策略示例 AWS Certificate Manager。

使用身份进行身份验证

身份验证是您 AWS 使用身份凭证登录的方式。您必须以 IAM 用户身份或通过担 AWS 账户根用户任 IAM 角色进行身份验证（登录 AWS）。

您可以使用通过身份源提供的凭据以 AWS 联合身份登录。 AWS IAM Identity Center （IAM Identity Center）用户、贵公司的单点登录身份验证以及您的 Google 或 Facebook 凭据就是联合身份的示例。当您以联合身份登录时，您的管理员以前使用 IAM 角色设置了身份联合验证。当你使用联合访问 AWS 时，你就是在间接扮演一个角色。

根据您的用户类型，您可以登录 AWS Management Console 或 AWS 访问门户。有关登录的更多信息 AWS，请参阅《AWS 登录 用户指南》中的如何登录到您 AWS 账户的。

如果您 AWS 以编程方式访问，则会 AWS 提供软件开发套件 (SDK) 和命令行接口 (CLI)，以便使用您的凭据对请求进行加密签名。如果您不使用 AWS 工具，则必须自己签署请求。有关使用推荐的方法自行签署请求的更多信息，请参阅 IAM 用户指南中的签署 AWS API 请求。

无论使用何种身份验证方法，您可能需要提供其他安全信息。例如， AWS 建议您使用多重身份验证 (MFA) 来提高账户的安全性。要了解更多信息，请参阅《AWS IAM Identity Center 用户指南》中的 多重身份验证和《IAM 用户指南》中的 在 AWS中使用多重身份验证（MFA）。

AWS 账户 root 用户

创建时 AWS 账户，首先要有一个登录身份，该身份可以完全访问账户中的所有资源 AWS 服务 和资源。此身份被称为 AWS 账户 root 用户，使用您创建账户时使用的电子邮件地址和密码登录即可访问该身份。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《IAM 用户指南》中的需要根用户凭证的任务。

联合身份

作为最佳实践，要求人类用户（包括需要管理员访问权限的用户）使用与身份提供商的联合身份验证 AWS 服务 通过临时证书进行访问。

联合身份是指您的企业用户目录、Web 身份提供商、Identity C enter 目录中的用户，或者任何使用 AWS 服务 通过身份源提供的凭据进行访问的用户。 AWS Directory Service当联合身份访问时 AWS 账户，他们将扮演角色，角色提供临时证书。

要集中管理访问权限，建议您使用 AWS IAM Identity Center。您可以在 IAM Identity Center 中创建用户和群组，也可以连接并同步到您自己的身份源中的一组用户和群组，以便在您的所有 AWS 账户 和应用程序中使用。有关 IAM Identity Center 的信息，请参阅《AWS IAM Identity Center 用户指南》中的什么是 IAM Identity Center？

IAM 用户和群组

I AM 用户是您 AWS 账户 内部对个人或应用程序具有特定权限的身份。在可能的情况下，我们建议使用临时凭证，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，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

I AM 角色是您内部具有特定权限 AWS 账户 的身份。它类似于 IAM 用户，但与特定人员不关联。您可以 AWS Management Console 通过切换角色在中临时担任 IAM 角色。您可以通过调用 AWS CLI 或 AWS API 操作或使用自定义 URL 来代入角色。有关使用角色的方法的更多信息，请参阅《IAM 用户指南》中的使用 IAM 角色。

具有临时凭证的 IAM 角色在以下情况下很有用：

- 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。如果您使用 IAM Identity Center，则需要配置权限集。为控制您的身份在进行身份验证后可以访问的内容，IAM Identity Center 将权限集与 IAM 中的角色相关联。有关权限集的信息，请参阅《AWS IAM Identity Center 用户指南》中的 权限集。
- 临时 IAM 用户权限 – IAM 用户可代入 IAM 用户或角色，以暂时获得针对特定任务的不同权限。

- 跨账户存取 – 您可以使用 IAM 角色以允许不同账户中的某个人（可信主体）访问您的账户中的资源。角色是授予跨账户访问权限的主要方式。但是，对于某些资源 AWS 服务，您可以将策略直接附加到资源（而不是使用角色作为代理）。要了解用于跨账户访问的角色和基于资源的策略之间的差别，请参阅《IAM 用户指南》中的 IAM 中的跨账户资源访问。
- 跨服务访问 — 有些 AWS 服务 使用其他 AWS 服务服务中的功能。例如，当您在某个服务中进行调用时，该服务通常会在 Amazon EC2 中运行应用程序或在 Simple Storage Service (Amazon S3) 中存储对象。服务可能会使用发出调用的主体的权限、使用服务角色或使用服务相关角色来执行此操作。
  - 转发访问会话 (FAS) — 当您使用 IAM 用户或角色在中执行操作时 AWS，您被视为委托人。使用某些服务时，您可能会执行一个操作，然后此操作在其他服务中启动另一个操作。FAS 使用调用委托人的权限以及 AWS 服务 向下游服务发出请求的请求。 AWS 服务只有当服务收到需要与其他 AWS 服务 或资源交互才能完成的请求时，才会发出 FAS 请求。在这种情况下，您必须具有执行这两个操作的权限。有关发出 FAS 请求时的策略详情，请参阅转发访问会话。
  - 服务角色 - 服务角色是服务代表您在您的账户中执行操作而分派的 IAM 角色。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 AWS 服务委派权限的角色。
  - 服务相关角色-服务相关角色是一种链接到的服务角色。 AWS 服务服务可以代入代表您执行操作的角色。服务相关角色出现在您的中 AWS 账户 ，并且归服务所有。IAM 管理员可以查看但不能编辑服务相关角色的权限。
- 在 A@@ mazon 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 AWS CLI 或 AWS API 请求的应用程序的临时证书。这优先于在 EC2 实例中存储访问密钥。要向 EC2 实例分配 AWS 角色并使其可供其所有应用程序使用，您需要创建附加到该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。

要了解是使用 IAM 角色还是 IAM 用户，请参阅IAM 用户指南中的何时创建 IAM 角色（而不是用户）。

使用策略管理访问

您可以 AWS 通过创建策略并将其附加到 AWS 身份或资源来控制中的访问权限。策略是其中的一个对象 AWS ，当与身份或资源关联时，它会定义其权限。 AWS 在委托人（用户、root 用户或角色会话）发出请求时评估这些策略。策略中的权限确定是允许还是拒绝请求。大多数策略都以 JSON 文档的 AWS 形式存储在中。有关 JSON 策略文档的结构和内容的更多信息，请参阅 IAM 用户指南中的 JSON 策略概览。

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

默认情况下，用户和角色没有权限。要授予用户对所需资源执行操作的权限，IAM 管理员可以创建 IAM 策略。管理员随后可以向角色添加 IAM policy，用户可以代入角色。

IAM 策略定义操作的权限，无关乎您使用哪种方法执行操作。例如，假设您有一个允许 `iam:GetRole` 操作的策略。拥有该策略的用户可以从 AWS Management Console AWS CLI、或 AWS API 获取角色信息。

基于身份的策略

基于身份的策略是可附加到身份（如 IAM 用户、用户组或角色）的 JSON 权限策略文档。这些策略控制用户和角色可在何种条件下对哪些资源执行哪些操作。要了解如何创建基于身份的策略，请参阅《IAM 用户指南》中的创建 IAM policy。

基于身份的策略可以进一步归类为内联策略或托管策略。内联策略直接嵌入单个用户、组或角色中。托管策略是独立的策略，您可以将其附加到中的多个用户、群组和角色 AWS 账户。托管策略包括 AWS 托管策略和客户托管策略。要了解如何在托管式策略和内联策略之间进行选择，请参阅 IAM 用户指南中的在托管式策略与内联策略之间进行选择。

基于资源的策略

基于资源的策略是附加到资源的 JSON 策略文档。基于资源的策略的示例包括 IAM 角色信任策略和 Amazon S3 存储桶策略。在支持基于资源的策略的服务中，服务管理员可以使用它们来控制对特定资源的访问。对于在其中附加策略的资源，策略定义指定主体可以对该资源执行哪些操作以及在什么条件下执行。您必须在基于资源的策略中指定主体。委托人可以包括账户、用户、角色、联合用户或 AWS 服务。

基于资源的策略是位于该服务中的内联策略。您不能在基于资源的策略中使用 IAM 中的 AWS 托管策略。

访问控制列表 (ACL)

访问控制列表（ACL）控制哪些主体（账户成员、用户或角色）有权访问资源。ACL 与基于资源的策略类似，尽管它们不使用 JSON 策略文档格式。

Amazon S3 和 Amazon VPC 就是支持 ACL 的服务示例。 AWS WAF要了解有关 ACL 的更多信息，请参阅《Amazon Simple Storage Service 开发人员指南》中的访问控制列表（ACL）概览。

其他策略类型

AWS 支持其他不太常见的策略类型。这些策略类型可以设置更常用的策略类型向您授予的最大权限。

- 权限边界：权限边界是一个高级特征，用于设置基于身份的策略可以为 IAM 实体（IAM 用户或角色）授予的最大权限。您可为实体设置权限边界。这些结果权限是实体基于身份的策略及其权限边界的交集。在 `Principal` 中指定用户或角色的基于资源的策略不受权限边界限制。任一项策略中的显式拒绝将覆盖允许。有关权限边界的更多信息，请参阅《IAM 用户指南》中的 IAM 实体的权限边界。
- 服务控制策略 (SCP)-SCP 是 JSON 策略，用于指定组织或组织单位 (OU) 的最大权限。 AWS Organizations AWS Organizations 是一项用于对您的企业拥有的多 AWS 账户 项进行分组和集中管理的服务。如果在组织内启用了所有功能，则可对任意或全部账户应用服务控制策略 (SCP)。SCP 限制成员账户中的实体（包括每个 AWS 账户根用户实体）的权限。有关 Organizations 和 SCP 的更多信息，请参阅《AWS Organizations 用户指南》中的 SCP 的工作原理。
- 会话策略 – 会话策略是当您以编程方式为角色或联合用户创建临时会话时作为参数传递的高级策略。结果会话的权限是用户或角色的基于身份的策略和会话策略的交集。权限也可以来自基于资源的策略。任一项策略中的显式拒绝将覆盖允许。有关更多信息，请参阅《IAM 用户指南》中的会话策略。

多个策略类型

当多个类型的策略应用于一个请求时，生成的权限更加复杂和难以理解。要了解在涉及多种策略类型时如何 AWS 确定是否允许请求，请参阅 IAM 用户指南中的策略评估逻辑。

如何 AWS Certificate Manager 与 IAM 配合使用

在使用 IAM 管理对 ACM 的访问之前，您应该了解哪些 IAM 功能可用于 ACM。

您可以搭配使用的 IAM 功能 AWS Certificate Manager

IAM 功能	ACM 支持
基于身份的策略	是
基于资源的策略	否
策略操作	是

IAM 功能	ACM 支持
策略资源	是
策略条件键（特定于服务）	是
ACL	否
ABAC（策略中的标签）	部分
临时凭证	是
主体权限	是
服务角色	否
服务相关角色	是

要全面了解 ACM 和其他 AWS 服务如何与大多数 IAM 功能配合使用，请参阅 IAM 用户指南中的与 IAM 配合使用的AWS 服务。

ACM 的基于身份的策略

支持基于身份的策略	是

基于身份的策略是可附加到身份（如 IAM 用户、用户组或角色）的 JSON 权限策略文档。这些策略控制用户和角色可在何种条件下对哪些资源执行哪些操作。要了解如何创建基于身份的策略，请参阅《IAM 用户指南》中的创建 IAM policy。

通过使用 IAM 基于身份的策略，您可以指定允许或拒绝的操作和资源以及允许或拒绝操作的条件。您无法在基于身份的策略中指定主体，因为它适用于其附加的用户或角色。要了解可在 JSON 策略中使用的所有元素，请参阅《IAM 用户指南》中的 IAM JSON 策略元素引用。

ACM 的基于身份的策略示例

要查看 ACM 基于身份的策略的示例，请参阅 基于身份的策略示例 AWS Certificate Manager。

ACM 内基于资源的策略

支持基于资源的策略	否

基于资源的策略是附加到资源的 JSON 策略文档。基于资源的策略的示例包括 IAM 角色信任策略和 Amazon S3 存储桶策略。在支持基于资源的策略的服务中，服务管理员可以使用它们来控制对特定资源的访问。对于在其中附加策略的资源，策略定义指定主体可以对该资源执行哪些操作以及在什么条件下执行。您必须在基于资源的策略中指定主体。委托人可以包括账户、用户、角色、联合用户或 AWS 服务。

要启用跨账户存取，您可以将整个账户或其他账户中的 IAM 实体指定为基于资源的策略中的主体。将跨账户主体添加到基于资源的策略只是建立信任关系工作的一半而已。当委托人和资源处于不同位置时 AWS 账户，可信账户中的 IAM 管理员还必须向委托人实体（用户或角色）授予访问资源的权限。他们通过将基于身份的策略附加到实体以授予权限。但是，如果基于资源的策略向同一个账户中的主体授予访问权限，则不需要额外的基于身份的策略。有关更多信息，请参阅 IAM 用户指南中的跨账户访问 IAM 中的资源。

ACM 的策略操作

支持策略操作	是

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

JSON 策略的 `Action` 元素描述可用于在策略中允许或拒绝访问的操作。策略操作通常与关联的 AWS API 操作同名。有一些例外情况，例如没有匹配 API 操作的仅限权限 操作。还有一些操作需要在策略中执行多个操作。这些附加操作称为相关操作。

在策略中包含操作以授予执行关联操作的权限。

要查看 ACM 操作的列表，请参阅《服务授权参考》中的 AWS Certificate Manager 定义的操作。

ACM 中的策略操作在操作前使用以下前缀：

```
acm
```

要在单个语句中指定多项操作，请使用逗号将它们隔开。

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

要查看 ACM 基于身份的策略的示例，请参阅 基于身份的策略示例 AWS Certificate Manager。

ACM 的策略资源

支持策略资源	是

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体 可以对什么资源执行操作，以及在什么条件下执行。

`Resource` JSON 策略元素指定要向其应用操作的一个或多个对象。语句必须包含 `Resource` 或 `NotResource` 元素。作为最佳实践，请使用其 Amazon 资源名称 (ARN) 指定资源。对于支持特定资源类型（称为资源级权限）的操作，您可以执行此操作。

对于不支持资源级权限的操作（如列出操作），请使用通配符 (*) 指示语句应用于所有资源。

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

要查看 ACM 的资源类型及其 ARN 的列表，请参阅《服务授权参考》中的 AWS Certificate Manager定义的资源。要了解可以在哪些操作中指定每个资源的 ARN，请参阅 AWS Certificate Manager定义的操作。

要查看 ACM 基于身份的策略的示例，请参阅 基于身份的策略示例 AWS Certificate Manager。

ACM 的策略条件密钥

支持特定于服务的策略条件键	是

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

在 Condition 元素（或 Condition 块）中，可以指定语句生效的条件。Condition 元素是可选的。您可以创建使用条件运算符（例如，等于或小于）的条件表达式，以使策略中的条件与请求中的值相匹配。

如果您在一个语句中指定多个 Condition 元素，或在单个 Condition 元素中指定多个键，则 AWS 使用逻辑 AND 运算评估它们。如果您为单个条件键指定多个值，则使用逻辑OR运算来 AWS 评估条件。在授予语句的权限之前必须满足所有的条件。

在指定条件时，您也可以使用占位符变量。例如，只有在使用 IAM 用户名标记 IAM 用户时，您才能为其授予访问资源的权限。有关更多信息，请参阅《IAM 用户指南》中的 IAM policy 元素：变量和标签。

AWS 支持全局条件密钥和特定于服务的条件密钥。要查看所有 AWS 全局条件键，请参阅 IAM 用户指南中的AWS 全局条件上下文密钥。

有关 ACM 条件密钥的列表，请参阅《服务授权参考》中的 AWS Certificate Manager 的条件密钥。要了解可以使用条件键的操作和资源，请参阅由定义的操作 AWS Certificate Manager。

要查看 ACM 基于身份的策略的示例，请参阅 基于身份的策略示例 AWS Certificate Manager。

ACM 中的 ACL

支持 ACL	否

访问控制列表 (ACL) 控制哪些主体（账户成员、用户或角色）有权访问资源。ACL 与基于资源的策略类似，尽管它们不使用 JSON 策略文档格式。

ABAC 以及 ACM

支持 ABAC（策略中的标签）	部分

基于属性的访问权限控制（ABAC）是一种授权策略，该策略基于属性来定义权限。在中 AWS，这些属性称为标签。您可以向 IAM 实体（用户或角色）和许多 AWS 资源附加标签。标记实体和资源是

ABAC 的第一步。然后设计 ABAC 策略，以在主体的标签与他们尝试访问的资源标签匹配时允许操作。

ABAC 在快速增长的环境中非常有用，并在策略管理变得繁琐的情况下可以提供帮助。

要基于标签控制访问，您需要使用 `aws:ResourceTag/key-name`、`aws:RequestTag/key-name` 或 `aws:TagKeys` 条件键在策略的条件元素中提供标签信息。

如果某个服务对于每种资源类型都支持所有这三个条件键，则对于该服务，该值为是。如果某个服务仅对于部分资源类型支持所有这三个条件键，则该值为部分。

有关 ABAC 的更多信息,请参阅《IAM 用户指南》中的什么是 ABAC？。要查看设置 ABAC 步骤的教程，请参阅《IAM 用户指南》中的使用基于属性的访问权限控制（ABAC）。

将临时凭证用于 ACM

支持临时凭证	是

当你使用临时证书登录时，有些 AWS 服务 不起作用。有关更多信息，包括哪些 AWS 服务 适用于临时证书，请参阅 IAM 用户指南中的AWS 服务 与 IAM 配合使用的信息。

如果您使用除用户名和密码之外的任何方法登录，则 AWS Management Console 使用的是临时证书。例如，当您 AWS 使用公司的单点登录 (SSO) 链接进行访问时，该过程会自动创建临时证书。当您以用户身份登录控制台，然后切换角色时，您还会自动创建临时凭证。有关切换角色的更多信息，请参阅《IAM 用户指南》中的 切换到角色（控制台）。

您可以使用 AWS CLI 或 AWS API 手动创建临时证书。然后，您可以使用这些临时证书进行访问 AWS。 AWS 建议您动态生成临时证书，而不是使用长期访问密钥。有关更多信息，请参阅 IAM 中的临时安全凭证。

ACM 的跨服务主体权限

支持转发访问会话 (FAS)	是

当您使用 IAM 用户或角色在中执行操作时 AWS，您被视为委托人。使用某些服务时，您可能会执行一个操作，然后此操作在其他服务中启动另一个操作。FAS 使用调用委托人的权限以及 AWS 服务 向下

游服务发出请求的请求。 AWS 服务只有当服务收到需要与其他 AWS 服务 或资源交互才能完成的请求时，才会发出 FAS 请求。在这种情况下，您必须具有执行这两个操作的权限。有关发出 FAS 请求时的策略详情，请参阅转发访问会话。

ACM 的服务角色

支持服务角色	否

服务角色是由一项服务担任、代表您执行操作的 IAM 角色。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 AWS 服务委派权限的角色。

> **Warning**
> 更改服务角色的权限可能会破坏 ACM 的功能。仅当 ACM 提供相关指导时才编辑服务角色。

ACM 的服务相关角色

支持服务相关角色	是

服务相关角色是一种链接到的服务角色。 AWS 服务服务可以代入代表您执行操作的角色。服务相关角色出现在您的中 AWS 账户 ，并且归服务所有。IAM 管理员可以查看但不能编辑服务相关角色的权限。

有关创建或管理服务相关角色的详细信息，请参阅能够与 IAM 搭配使用的AWS 服务。在表中查找服务相关角色列中包含 Yes 的表。选择是链接以查看该服务的服务相关角色文档。

基于身份的策略示例 AWS Certificate Manager

默认情况下，用户和角色没有创建或修改 ACM 资源的权限。他们也无法使用 AWS Management Console、 AWS Command Line Interface (AWS CLI) 或 AWS API 执行任务。要授予用户对所需资源执行操作的权限，IAM 管理员可以创建 IAM policy。管理员随后可以向角色添加 IAM 策略，用户可以代入角色。

要了解如何使用这些示例 JSON 策略文档创建基于 IAM 身份的策略，请参阅 IAM 用户指南中的 创建 IAM 策略。

有关 ACM 定义的操作和资源类型的详细信息，包括每种资源类型的 ARN 格式，请参阅《服务授权参考》中的 AWS Certificate Manager的操作、资源和条件密钥。

主题

策略最佳实践

基于身份的策略确定某个人是否可以创建、访问或删除您账户中的 ACM 资源。这些操作可能会使 AWS 账户产生成本。创建或编辑基于身份的策略时，请遵循以下指南和建议：

- 开始使用 AWS 托管策略并转向最低权限权限 — 要开始向用户和工作负载授予权限，请使用为许多常见用例授予权限的AWS 托管策略。它们在你的版本中可用 AWS 账户。我们建议您通过定义针对您的用例的 AWS 客户托管策略来进一步减少权限。有关更多信息，请参阅《IAM 用户指南》中的 AWS 托管策略 或 工作职能的AWS 托管策略。
- 应用最低权限 – 在使用 IAM policy 设置权限时，请仅授予执行任务所需的权限。为此，您可以定义在特定条件下可以对特定资源执行的操作，也称为最低权限许可。有关使用 IAM 应用权限的更多信息，请参阅《IAM 用户指南》中的 IAM 中的策略和权限。
- 使用 IAM policy 中的条件进一步限制访问权限 – 您可以向策略添加条件来限制对操作和资源的访问。例如，您可以编写策略条件来指定必须使用 SSL 发送所有请求。如果服务操作是通过特定 AWS 服务的（例如）使用的，则也可以使用条件来授予对服务操作的访问权限 AWS CloudFormation。有关更多信息，请参阅《IAM 用户指南》中的 IAM JSON 策略元素：条件 。
- 使用 IAM Access Analyzer 验证您的 IAM policy，以确保权限的安全性和功能性 – IAM Access Analyzer 会验证新策略和现有策略，以确保策略符合 IAM policy语言 (JSON) 和 IAM 最佳实践。IAM Access Analyzer 提供 100 多项策略检查和可操作的建议，以帮助您制定安全且功能性强的策略。有关更多信息，请参阅《IAM 用户指南》中的 IAM Acess Analyzer 策略验证。
- 需要多重身份验证 (MFA)-如果 AWS 账户您的场景需要 IAM 用户或根用户，请启用 MFA 以提高安全性。若要在调用 API 操作时需要 MFA，请将 MFA 条件添加到您的策略中。有关更多信息，请参阅《IAM 用户指南》中的 配置受 MFA 保护的 API 访问。

有关 IAM 中的最佳实操的更多信息，请参阅《IAM 用户指南》中的 IAM 中的安全最佳实操。

使用 ACM 控制台

要访问 AWS Certificate Manager 控制台，您必须拥有一组最低权限。这些权限必须允许您列出和查看有关您 AWS 账户的 ACM 资源的详细信息。如果创建比必需的最低权限更为严格的基于身份的策略，对于附加了该策略的实体（用户或角色），控制台将无法按预期正常运行。

对于仅调用 AWS CLI 或 AWS API 的用户，您无需为其设置最低控制台权限。相反，只允许访问与其尝试执行的 API 操作相匹配的操作。

为确保用户和角色仍然可以使用 ACM 控制台，还要将 ACM *AWSCertificateManagerReadOnly* AWS 托管策略附加到实体。有关更多信息，请参阅《IAM 用户指南》中的为用户添加权限。

允许用户查看他们自己的权限

该示例说明了您如何创建策略，以允许 IAM 用户查看附加到其用户身份的内联和托管策略。此策略包括在控制台上或使用 AWS CLI 或 AWS API 以编程方式完成此操作的权限。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
```

```
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

列出证书

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显示在 Elastic Load Balancing 和 CloudFront 控制台中。

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"
            }
            }
```

ACM API 权限：操作和资源参考

在设置和编写您可附加到 IAM 用户或角色的访问控制和写入权限策略，可以使用下表作为参考。该表的第一列中列出了每个 AWS 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
AddTagsToCertificate	acm:AddTagsToCertificate	arn:aws:acm: *region*:*account*:certificate/ *certificate_ID*
DeleteCertificate	acm:DeleteCertificate	arn:aws:acm: *region*:*account*:certificate/ *certificate_ID*
DescribeCertificate	acm:DescribeCertificate	arn:aws:acm: *region*:*account*:certificate/ *certificate_ID*
ExportCertificate	acm:ExportCertificate	arn:aws:acm: *region*:*account*:certificate/ *certificate_ID*
GetAccountConfiguration	acm:GetAccountConfiguration	*
GetCertificate	acm:GetCertificate	arn:aws:acm: *region*:*account*:certificate/ *certificate_ID*
ImportCertificate	acm:ImportCertificate	arn:aws:acm: *region*:*account*:certificate/* or *
ListCertificates	acm:ListCertificates	*
ListTagsForCertificate	acm:ListTagsForCertificate	arn:aws:acm: *region*:*account*:certificate/ *certificate_ID*

ACM API 操作	必需的权限（API 操作）	资源
PutAccountConfiguration	acm:PutAccountConfiguration	*
RemoveTagsFromCertificate	acm:RemoveTagsFromCertificate	arn:aws:acm: *region*:*account*:certificate/ *certificate_ID*
RequestCertificate	acm:RequestCertificate	arn:aws:acm: *region*:*account*:certificate/* or *
ResendValidationEmail	acm:ResendValidationEmail	arn:aws:acm: *region*:*account*:certificate/ *certificate_ID*
UpdateCertificateOptions	acm:UpdateCertificateOptions	arn:aws:acm: *region*:*account*:certificate/ *certificate_ID*

AWS适用于 AWS Certificate Manager 的托管策略

AWS 托管式策略是由 AWS 创建和管理的独立策略。AWS 托管式策略旨在为许多常见用例提供权限，以便您可以开始为用户、组和角色分配权限。

请记住，AWS 托管式策略可能不会为您的特定使用场景授予最低权限，因为它们可供所有 AWS 客户使用。我们建议通过定义特定于您的使用场景的客户管理型策略来进一步减少权限。

您无法更改 AWS 托管策略中定义的权限。如果 AWS 更新在 AWS 托管式策略中定义的权限，则更新会影响该策略所附加到的所有主体身份（用户、组和角色）。当新的 AWS 服务启动或新的 API 操作可用于现有服务时，AWS 最有可能更新 AWS 托管式策略。

有关更多信息，请参阅《IAM 用户指南》中的 AWS 托管式策略。

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

ACM 对AWS托管策略的更新

查看有关 ACM 的AWS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 页面上的 RSS 源。

更改	说明	日期
增加了对 AWSCertificateManagerReadOnly 策略的 GetAccountConfiguration 支持。	AWSCertificateManagerReadOnly 策略现在包含调用 GetAccountConfiguration API 操作的权限。	2021 年 3 月 3 日

更改	说明	日期
ACM 开始跟踪更改	ACM 开始跟踪AWS托管策略的更改。	2021 年 3 月 3 日

在 ACM 中使用条件密钥

AWS Certificate Manager 使用 AWS Identity and Access Management（IAM）条件密钥来限制对证书请求的访问权限。使用 IAM policy 或服务控制策略（SCP）中的条件密钥，您可以创建符合组织准则的证书请求。

> **Note**
>
> 将 ACM 条件密钥与 AWS 全局条件密钥（例如 `aws:PrincipalArn`）结合，从而进一步将操作限制为特定用户或角色。

支持的 ACM 条件

ACM API 操作和支持的条件

条件键	支持的 ACM API 操作	类型	描述
`acm:ValidationMethod`	RequestCertificate	字符串（EMAIL，DNS）	依据 ACM 验证方法筛选请求
`acm:DomainNames`	RequestCertificate	字符串数组	依据 ACM 请求中的域名筛选
`acm:KeyAlgorithm`	RequestCertificate	字符串	依据 ACM 密钥算法和大小筛选请求
`acm:CertificateTransparencyLogging`	RequestCertificate	字符串（ENABLED，DISABLEI	依据 ACM 证书透明度日志记录首选项筛选请求

条件键	支持的 ACM API 操作	类型	描述
acm:CertificateAuthority	RequestCertificate	ARN	依据 ACM 请求中的证书颁发机构筛选请求

示例 1：限制验证方法

以下策略使用电子邮件验证方法来拒绝新的证书请求，但使用 arn:aws:iam::123456789012:role/AllowedEmailValidation 角色发出的请求除外。

```
{
    "Version":"2012-10-17",
    "Statement":{
        "Effect":"Deny",
        "Action":"acm:RequestCertificate",
        "Resource":"*",
        "Condition":{
            "StringLike" : {
                "acm:ValidationMethod":"EMAIL"
            },
            "ArnNotLike": {
                "aws:PrincipalArn": [ "arn:aws:iam::123456789012:role/
AllowedEmailValidation"]
            }
        }
    }
}
```

示例 2：阻止通配符域

以下策略拒绝使用通配符域的任何新 ACM 证书请求。

```
{
    "Version":"2012-10-17",
    "Statement":{
        "Effect":"Deny",
```

```
        "Action":"acm:RequestCertificate",
        "Resource":"*",
        "Condition": {
            "ForAnyValue:StringLike": {
                "acm:DomainNames": [
                    "${*}.*"
                ]
            }
        }
    }
}
```

示例 3：限制证书域

以下策略拒绝非 *.amazonaws.com 结尾的域的任何新 ACM 证书请求

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

该策略可以进一步限制为特定的子域。此政策仅允许其中每个域都与至少一个条件域名匹配的请求。

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

```
            "ForAllValues:StringNotLike": {
                "acm:DomainNames": ["support.amazonaws.com", "developer.amazonaws.com"]
            }
        }
    }
}
```

示例 4：限制密钥算法

以下策略使用条件密钥 StringNotLike 来仅允许使用 ECDSA 384 位 (EC_secp384r1) 密钥算法请求的证书。

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

以下策略使用条件密钥 StringLike 和通配符 * 匹配来阻止在 ACM 中使用任何 RSA 密钥算法的新证书。

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

```
            }
        }
    }
}
```

示例 5：限制证书颁发机构

以下策略仅允许使用提供的私有证书颁发机构（PCA）ARN 的私有证书请求。

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

此策略使用 `acm:CertificateAuthority` 条件来仅允许 Amazon Trust Services 颁发的公开信任证书的请求。将证书颁发机构 ARN 设置为 `false` 能阻止来自 PCA 的私有证书请求。

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

```
    }
}
```

将服务相关角色 (SLR) 用于 ACM

AWS Certificate Manager 使用 AWS Identity and Access Management (IAM) 服务相关角色启用托管 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 AWS 服务所需的一切权限。

SLR 可让您更轻松地设置 ACM，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，而且权限策略不能附加到任何其它 IAM 实体。

有关支持 SLR 的其他服务的信息，请参阅可与 IAM 配合使用的AWS 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。

S AWSServiceRoleForCertificateManager LR 信任以下服务来担任该角色：

- `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 AWS Management Console、或 AWS API 颁发 ACM 证书时 AWS CLI，ACM 会在您首次选择私有 CA 来签署证书时为您创建 SLR。

如果您收到消息称 ACM 无法确定您的账户中是否存在 SLR，则可能意味着您的账户未授予所需的读取权限。 AWS 私有 CA 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。

> **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。另外，如果您在 2017 年 1 月 1 日开始支持 SLR 之前使用 ACM 服务，那么 ACM 会在您的账户中创建该 AWSServiceRoleForCertificateManager 角色。要了解更多信息，请参阅我的 IAM 账户中的新角色。

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，选择角色、创建角色、Certi ficate Manager，CertificateManagerServiceRolePolicy使用该用例创建新角色。
- 使用 IAM API CreateServiceLinkedRole或相应的 AWS CLI 命令 create-service-linked-role，使用`acm.amazonaws.com`服务名称创建一个 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。不过，您可以使用 IAM 编辑角色的说明。有关更多信息，请参阅《IAM 用户指南》中的编辑服务相关角色。

删除 ACM 的 SLR

您通常不需要删除 AWSServiceRoleForCertificateManager 单反相机。但是，您可以使用 IAM 控制台、 AWS CLI 或 AWS API 手动删除角色。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 AWS 私有 CA 的所有地区使用单反相机。有关更多信息，请参阅 AWS 区域和端点。

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
US East（Ohio）	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
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
亚太地区（悉尼）	ap-southeast-2	是
亚太地区（东京)	ap-northeast-1	是
加拿大（中部）	ca-central-1	是
欧洲地区（法兰克福）	eu-central-1	是
欧洲（苏黎世）	eu-central-2	是
Europe（Ireland）	eu-west-1	是
欧洲地区（伦敦）	eu-west-2	是
欧洲地区（巴黎）	eu-west-3	是
南美洲（圣保罗）	sa-east-1	是
AWS GovCloud （美国西部）	us-gov-west-1	是

区域名称	区域标识	ACM 中支持
AWS GovCloud （美国东部）东部	us-gov-east-1	是

对 AWS Certificate Manager 身份和访问进行故障排除

使用以下信息可帮助您诊断和修复在使用 ACM 和 IAM 时可能遇到的常见问题。

主题

我无权在 ACM 中执行操作

如果您收到错误提示，表明您无权执行某个操作，则您必须更新策略以允许执行该操作。

当 mateojackson IAM 用户尝试使用控制台查看有关虚构 *my-example-widget* 资源的详细信息，但不拥有虚构 acm:*GetWidget* 权限时，会发生以下示例错误。

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

在此情况下，必须更新 mateojackson 用户的策略，以允许使用 acm:*GetWidget* 操作访问 *my-example-widget* 资源。

如果您需要帮助，请联系您的 AWS 管理员。您的管理员是提供登录凭证的人。

我无权在 ACM 中申请证书

如果您收到此错误，则说明您的 ACM 或 PKI 管理员已设置规则，防止您在当前状态下请求证书。

当 IAM 用户尝试通过控制台，使用由组织管理员配置为 DENY 的选项请求证书时，会出现以下示例错误。

```
User: arn:aws:sts::account::ID: is not authorized to perform: acm:RequestCertificate
on resource: arn:aws:acm:region:account:certificate/*
```

```
with an explicit deny in a service control policy
```

在这种情况下，应以符合管理员设置的策略的方式再次提出请求。或者，需要更新政策以允许请求证书。

我无权执行 iam：PassRole

如果您收到一个错误，表明您无权执行 `iam:PassRole` 操作，则必须更新策略以允许您将角色传递给 ACM。

有些 AWS 服务 允许您将现有角色传递给该服务，而不是创建新的服务角色或服务相关角色。为此，您必须具有将角色传递到服务的权限。

当名为 `marymajor` 的 IAM 用户尝试使用控制台在 ACM 中执行操作时，会发生以下示例错误。但是，服务必须具有服务角色所授予的权限才可执行此操作。Mary 不具有将角色传递到服务的权限。

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

在这种情况下，必须更新 Mary 的策略以允许她执行 `iam:PassRole` 操作。

如果您需要帮助，请联系您的 AWS 管理员。您的管理员是提供登录凭证的人。

我想允许我以外的人 AWS 账户 访问我的 ACM 资源

您可以创建一个角色，以便其他账户中的用户或您组织外的人员可以使用该角色来访问您的资源。您可以指定谁值得信赖，可以担任角色。对于支持基于资源的策略或访问控制列表（ACL）的服务，您可以使用这些策略向人员授予对您的资源的访问权。

要了解更多信息，请参阅以下内容：

- 要了解 ACM 是否支持这些功能，请参阅 如何 AWS Certificate Manager 与 IAM 配合使用。
- 要了解如何提供对您拥有的资源的访问权限 AWS 账户 ，请参阅 IAM 用户指南中的向您拥有 AWS 账户 的另一个 IAM 用户提供访问权限。
- 要了解如何向第三方提供对您的资源的访问权限 AWS 账户，请参阅 IAM 用户指南中的向第三方提供访问权限。 AWS 账户
- 要了解如何通过联合身份验证提供访问权限，请参阅《IAM 用户指南》中的为经过外部身份验证的用户（联合身份验证）提供访问权限。
- 要了解使用角色和基于资源的策略进行跨账户访问的区别，请参阅 IA M 用户指南中的跨账户资源访问。

韧性在 AWS Certificate Manager

AWS 全球基础设施是围绕 AWS 区域和可用区构建的。 AWS 区域提供多个物理隔离和隔离的可用区，这些可用区通过低延迟、高吞吐量和高度冗余的网络相连。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失效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 AWS 区域和可用区的更多信息，请参阅AWS 全球基础设施。

AWS Certificate Manager中的基础设施安全性

作为一项托管服务 AWS Certificate Manager ，受 AWS 全球网络安全的保护。有关 AWS 安全服务以及如何 AWS 保护基础设施的信息，请参阅AWS 云安全。要使用基础设施安全的最佳实践来设计您的 AWS 环境，请参阅 S AWS ecurity Pillar Well-Architected Fram ework 中的基础设施保护。

您可以使用 AWS 已发布的 API 调用通过网络访问 ACM。客户端必须支持以下内容：

- 传输层安全性协议 (TLS) 我们要求使用 TLS 1.2，建议使用 TLS 1.3。
- 具有完全向前保密 (PFS) 的密码套件，例如 DHE（临时 Diffie-Hellman）或 ECDHE（临时椭圆曲线 Diffie-Hellman）。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 主体关联的秘密访问密钥来对请求进行签名。或者，您可以使用 AWS Security Token Service（AWS STS）生成临时安全凭证来对请求进行签名。

授予对 ACM 的编程访问权限

如果用户想在 AWS 外部进行交互，则需要编程访问权限 AWS Management Console。授予编程访问权限的方式取决于正在访问的用户类型 AWS。

要向用户授予编程式访问权限，请选择以下选项之一。

哪个用户需要编程式访问权限？	目的	方式
人力身份 （在 IAM Identity Center 中管理的用户）	使用临时证书签署向 AWS CLI、 AWS 软件开发工具包或 AWS API 发出的编程请求。	按照您希望使用的界面的说明进行操作。 • 有关的 AWS CLI，请参阅《AWS Command Line

哪个用户需要编程式访问权限？	目的	方式
		Interface 用户指南》 AWS IAM Identity Center中的配置 AWS CLI 以使用。 • 有关 AWS 软件开发工具包、工具和 AWS API，请参阅《软件开发工具包和AWS 工具参考指南》中的 IAM 身份中心身份验证。
IAM	使用临时证书签署向 AWS CLI、 AWS 软件开发工具包或 AWS API 发出的编程请求。	按照 IAM 用户指南中的将临时证书与 AWS 资源配合使用中的说明进行操作。
IAM	（不推荐使用） 使用长期凭证签署向 AWS CLI、 AWS 软件开发工具包或 AWS API 发出的编程请求。	按照您希望使用的界面的说明进行操作。 • 有关信息 AWS CLI，请参阅用户指南中的使用 IAM 用户证书进行身份验证。AWS Command Line Interface • 有关 AWS SDK 和工具，请参阅 S AWS DK 和工具参考指南中的使用长期凭证进行身份验证。 • 有关 AWS API，请参阅 IAM 用户指南中的管理 IAM 用户的访问密钥。

最佳实践

最佳实践是可以帮助您更有效地使用 AWS Certificate Manager (AWS Certificate Manager) 的建议。以下最佳实践基于来自当前 ACM 客户的实际经验。

主题

- 账户级分离
- AWS CloudFormation
- 证书固定
- 域验证
- 添加或删除域名
- 选择退出证书透明度日志记录
- 开启 AWS CloudTrail

账户级分离

在策略中使用账户级别的分隔来控制谁可以在账户级别访问证书。将您的生产证书与测试和开发证书分开存放在不同的账户中。如果您无法使用账户级分离，则可以通过拒绝策略中的kms:CreateGrant操作来限制对特定角色的访问权限。这限制了账户中的哪些角色可以对证书进行高级签名。有关授权的信息，包括授权术语，请参阅《AWS Key Management Service 开发者指南》AWS KMS中的授权。

如果您想要更精细的控制而不是限制kms:CreateGrant按账户使用，则可以使用 k ms: EncryptionContext 条件密钥限制kms:CreateGrant特定证书。指定arn:aws:acm为密钥，并指定要限制的 ARN 的值。以下示例策略禁止使用特定证书，但允许使用其他证书。

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

AWS CloudFormation

AWS CloudFormation 您可以使用创建描述您要使用的 AWS 资源的模板。 AWS CloudFormation 然后为您配置和配置这些资源。 AWS CloudFormation 可以配置 ACM 支持的资源，例如 Elastic Load Balancing CloudFront、Amazon 和 Amazon API Gateway。有关更多信息，请参阅 与之集成的服务 AWS Certificate Manager。

如果您使用 AWS CloudFormation 快速创建和删除多个测试环境，我们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。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 AWS CloudFormation 用于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 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

在 Amazon 证书颁发机构 (CA) 为您的网站颁发证书之前， AWS Certificate Manager (ACM) 必须验证您是否拥有或控制您在请求中指定的所有域。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 和 电子邮件验证。

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 已开始在至少两个公有日志中发布所有新证书和续订。证书一旦记录，便无法删除。有关更多信息，请参阅 证书透明度日志。

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

要在申请证书时选择退出透明度记录，请使用请求证书 AWS CLI 命令或 RequestCertificateAPI 操作的options参数。如果您的证书是在 2018 年 4 月 24 日之前颁发的，并且您想确保续订期间未记录该证书，则可以使用update-certificate-options命令或 UpdateCertificateOptionsAPI 操作选择退出。

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 \
--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"
}
```

如果您已有证书，并且不希望在续订证书时将其记录下来，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
```

开启 AWS CloudTrail

在开始使用 ACM 之前，请开启 CloudTrail 日志功能。 CloudTrail 允许您通过检索账户的 AWS API 调用历史记录来监控您的 AWS 部署，包括通过 AWS 管理控制台、软件开发工具包和更高级别的

Amazon Web AWS Services 进行的 API 调用。 AWS Command Line Interface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 CloudTrail 集成到应用程序中，为您的组织自动创建跟踪，检查跟踪的状态，并控制管理员如何开启和关闭 CloudTrail 登录功能。有关更多信息，请参阅创建跟踪。转到 CloudTrail 与一起使用 AWS Certificate Manager 以查看 ACM 操作的示例跟踪记录。

设置

使用 AWS Certificate Manager (ACM)，您可以为 AWS 基于您的网站和应用程序配置和管理 SSL/TLS 证书。您可以使用 ACM 创建或导入证书，然后加以管理。您必须使用其他 AWS 服务将证书部署到您的网站或应用程序。有关与 ACM 集成的服务的更多信息，请参阅 与之集成的服务 AWS Certificate Manager。以下章节介绍在使用 ACM 之前需要执行的步骤。

主题

注册获取 AWS 账户

如果您没有 AWS 账户，请完成以下步骤来创建一个。

报名参加 AWS 账户

1. 打开 https://portal.aws.amazon.com/billing/signup。
2. 按照屏幕上的说明进行操作。

   在注册时，将接到一通电话，要求使用电话键盘输入一个验证码。

   当您注册时 AWS 账户，就会创建AWS 账户根用户一个。根用户有权访问该账户中的所有 AWS 服务 和资源。作为安全最佳实践，请为用户分配管理访问权限，并且只使用根用户来执行需要根用户访问权限的任务。

AWS 注册过程完成后会向您发送一封确认电子邮件。在任何时候，您都可以通过转至 https://aws.amazon.com/ 并选择我的账户来查看当前的账户活动并管理您的账户。

创建具有管理访问权限的用户

注册后，请保护您的安全 AWS 账户 AWS 账户根用户 AWS IAM Identity Center，启用并创建管理用户，这样您就不会使用 root 用户执行日常任务。

保护你的 AWS 账户根用户

1. 选择 Root 用户并输入您的 AWS 账户 电子邮件地址，以账户所有者的身份登录。AWS Management Console在下一页上，输入您的密码。

   要获取使用根用户登录方面的帮助，请参阅 《AWS 登录 用户指南》中的以根用户身份登录。
2. 为您的根用户启用多重身份验证 (MFA)。

   有关说明，请参阅 I A M 用户指南中的为 AWS 账户 根用户启用虚拟 MFA 设备（控制台）。

创建具有管理访问权限的用户

1. 启用 IAM Identity Center。

   有关说明，请参阅《AWS IAM Identity Center 用户指南》中的启用 AWS IAM Identity Center。
2. 在 IAM Identity Center 中，为用户授予管理访问权限。

   有关使用 IAM Identity Center 目录 作为身份源的教程，请参阅《用户指南》 IAM Identity Center 目录中的使用默认设置配置AWS IAM Identity Center 用户访问权限。

以具有管理访问权限的用户身份登录

- 要使用您的 IAM Identity Center 用户身份登录，请使用您在创建 IAM Identity Center 用户时发送到您的电子邮件地址的登录网址。

  有关使用 IAM Identity Center 用户登录的帮助，请参阅AWS 登录 用户指南中的登录 AWS 访问门户。

将访问权限分配给其他用户

1. 在 IAM Identity Center 中，创建一个权限集，该权限集遵循应用最低权限的最佳做法。

   有关说明，请参阅《AWS IAM Identity Center 用户指南》中的创建权限集。
2. 将用户分配到一个组，然后为该组分配单点登录访问权限。

   有关说明，请参阅《AWS IAM Identity Center 用户指南》中的添加组。

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。
> 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。

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

（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 AWS Certificate Manager (ACM) 为您的域或子域名颁发证书。验证域之后，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"
```

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

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

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

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

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

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 AWS 私有 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 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 AWS 管理控制台并打开 ACM 控制台，网址为 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

以下各节讨论如何使用 ACM 控制台或 AWS CLI 申请公有 ACM 证书。请求公有证书后，必须完成 验证域所有权 中所述的其中一个程序。

公有 ACM 证书遵循 X.509 标准，并具有以下限制：

- 名称：必须使用符合 DNS 的主题名称。有关更多信息，请参阅 域名。
- 算法：对于加密，证书私有密钥算法必须是 2048 位 RSA、256 位 ECDSA 或 384 位 ECDSA。
- 有效期：每个证书的有效期为 13 个月（395 天）。
- 续订：ACM 会在 11 个月后尝试自动续订私有证书。

如果在请求证书时遇到问题，请参阅排查证书请求问题。

要使用申请私有 PKI 的证书 AWS 私有 CA，请参阅请求私有 PKI 证书。

> **Note**
>
> 管理员可以使用 ACM 条件密钥政策控制最终用户如何颁发新证书。这些条件密钥能够对域、验证方法以及与证书请求相关的其他属性施加限制。

> **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 AWS 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。有关透明度日志的一般信息，请参阅证书透明度日志。

主题

- 使用控制台请求公有证书
- 使用 CLI 请求公有证书

使用控制台请求公有证书

请求 ACM 公有证书（控制台）

1. 登录 AWS 管理控制台并打开 ACM 控制台，网址为 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。

   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。

3. 在 Validation method（验证方法）部分，根据您的需要选择 DNS validation – recommended（DNS 验证 – 推荐）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。

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最多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。

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。

4. 在 Key algorithm（密钥算法）部分中，选择三种可用算法之一：

   - RSA 2048（默认）
   - ECDSA P 256
   - ECDSA P 384

   有关帮助您选择算法的信息，请参阅密钥算法中的 AWS 博客文章《如何评估和使用 ECDSA 证书》。 AWS Certificate Manager

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，用作识别和组织 AWS 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 AWS Certificate Manager 证书。

   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**
>
> 根据您对列表排序的方式，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。验证方法的可选值是 DNS 和 EMAIL（电子邮件）。密钥算法的可选值是 RSA_2048（如果未明确提供参数，则为默认值）、EC_prime256v1 和 EC_secp384r1。

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

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

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

请求私有 PKI 证书

如果您有权访问由创建的现有私有 CA AWS 私有 CA，ACM 可以申请适合在您的私有 PKI 中使用的证书。CA 可能位于您的账户中，也可能由其他账户与您共享。有关创建私有证书颁发机构的信息，请参阅创建私有证书颁发机构。

默认情况下，私有 CA 签署的证书不受信任，ACM 不支持对这些证书进行任何形式的验证。因此，管理员必须采取措施，将证书安装到您组织的客户端信任存储中。

私有 ACM 证书遵循 X.509 标准，并具有以下限制：

- 名称：必须使用符合 DNS 的主题名称。有关更多信息，请参阅 域名。

- 算法：对于加密，证书私有密钥算法必须是 2048 位 RSA、256 位 ECDSA 或 384 位 ECDSA。

> **Note**
>
> 指定的签名算法系列（RSA 或 ECDSA）必须与 CA 密钥的算法系列匹配。

- 有效期：每个证书的有效期为 13 个月（395 天）。签署的私有证书颁发机构证书的结束日期必须晚于请求的证书结束日期，否则证书请求将失败。
- 续订：ACM 会在 11 个月后尝试自动续订私有证书。

用于签署终端实体证书的私有证书颁发机构受其自身限制：

- 证书颁发机构的状态必须为“活跃”。
- 证书颁发机构私有密钥算法必须是 RSA 2048 或 RSA 4096。

> **Note**
>
> 与公开受信任的证书不同，由私有 CA 签署的证书不需要验证。

主题

配置对私有 CA 的访问

在以下两种情况下 AWS 私有 CA ，您可以使用对 ACM 证书进行签名：

- 单一账户：签名 CA 和颁发的 ACM 证书位于同一个 AWS 账户中。

  要启用单账户颁发和续订， AWS 私有 CA 管理员必须向 ACM 服务主体授予创建、检索和列出证书的权限。这是使用 AWS 私有 CA API 操作CreatePermission或 AWS CLI 命令创建权限完成的。账户拥有者将这些权限分配给负责颁发证书的 IAM 用户、组或角色。
- 跨账户：签名的 CA 和颁发的 ACM 证书位于不同的 AWS 账户中，并且证书所在的账户已被授予对 CA 的访问权限。

要启用跨账户签发和续订，AWS 私有 CA 管理员必须使用 AWS 私有 CA API 操作PutPolicy或命令 put-policy 将基于资源的策略附加到 CA。AWS CLI该策略指定其他账户中允许对 CA 进行有限访问的主体。有关更多信息，请参阅将基于资源的策略用于 ACM Private CA。

跨账户方案还要求 ACM 设置服务关联角色 (SLR)，以便作为主体与 PCA 策略进行交互。ACM 在颁发第一个证书时自动创建 SLR。

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，请参阅将服务相关角色用于 ACM。

> **Important**
>
> 您的 ACM 证书必须与支持的 AWS 服务主动关联，然后才能自动续订。有关 ACM 支持的资源的信息，请参阅 与之集成的服务 AWS Certificate Manager。

使用 ACM 控制台请求私有 PKI 证书

1. 登录 AWS 管理控制台并打开 ACM 控制台，网址为 https://console.aws.amazon.com/acm/home。

   选择请求证书。

2. 在 Request certificate（请求证书）页面上，选择 Request a private certificate（请求私有证书）和 Next（下一步）以继续。

3. 在 Certificate authority details（证书颁发机构详细信息）部分，单击 Certificate authority（证书颁发机构）菜单，然后选择其中一个可用的私有 CA。如果 CA 是从另一个账户共享的，则 ARN 的前面包含所有权信息。

   此时将显示有关 CA 的详细信息，以帮助您验证选择了正确的 CA：

   - 所有者
   - 类型
   - 公用名（CN）

- 组织 (O)
- 组织部门（OU）
- 国家/地区名称 (C)
- 州或省
- 所在地名称

4. 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。

   > **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**)。要同时保护二者，请参阅下一个步骤

   或者，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。这对于同时验证裸域或顶点域（例如 **example.com**）及其子域（例如 ***.example.com**）非常有用。

5. 在 Key algorithm（密钥算法）部分中，选择三种可用算法之一：

   - RSA 2048（默认）
   - ECDSA P 256
   - ECDSA P 384

   有关帮助您选择算法的信息，请参阅 密钥算法。

6. 在 Tags（标签）选择中，您可以选择为证书添加标签。标签是键值对，用作识别和组织 AWS 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 AWS Certificate Manager 证书。

7. 在 Certificate renewal permissions（证书续订权限）部分，确认有关证书续订权限的通知。这些权限允许自动续订使用所选 CA 签署的私有 PKI 证书。有关更多信息，请参阅将服务相关角色用于 ACM。
8. 提供所有必需信息后，选择 Request（请求）。控制台将返回证书列表，您可以在其中查看新证书。

> **Note**
>
> 根据您对列表排序的方式，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 请求私有 PKI 证书

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

> **Note**
>
> 当您请求由 CA 签名的私有 PKI 证书时 AWS Private CA，指定的签名算法系列（RSA 或 ECDSA）必须与 CA 密钥的算法系列相匹配。

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

此命令输出新私有证书的 Amazon Resource Name (ARN)。

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

在大多数情况下，ACM 会在您首次使用共享 CA 时自动将服务关联角色 (SLR) 附加到您的账户。SLR 允许自动续订您颁发的终端实体证书。要检查 SLR 是否存在，可使用以下命令查询 IAM：

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

如果存在 SLR，命令输出应类似如下所示：

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

如果缺少 SLR，请参阅将服务关联角色用于 ACM。

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， AWS Certificate Manager (ACM) 必须先确认您拥有或可以控制请求中指定的所有域名。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。

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拥有者执行操作。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 邮箱地址和最多五个常用管理员地址。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。

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。

> **Note**
>
> 在创建采用电子邮件验证的证书后，您无法切换到使用 DNS 对其进行验证。

主题

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。

> **Note**
>
> 在创建采用电子邮件验证的证书后，您无法切换到使用 DNS 对其进行验证。

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。该值是一个别名，指向 ACM 用来自动续订您的证书的 AWS 域。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。

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 CNAME 验证令牌适用于任何 AWS 区域，因此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 AWS 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。

> **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。

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 创建的字符串与为其基域 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	注释
*.example.com	*_x1*.example.com。	*_x2*.acm-validations.aws。	相同
example.com	*_x1*.example.com。	*_x2*.acm-validations.aws。	
www.example.com	*_x3*.www.example.com。	*_x4*.acm-validations.aws。	唯一
host.example.com	*_x5*.host.example.com。	*_x6*.acm-validations.aws。	唯一
subdomain.example.com	*_x7*.subdomain.example.com。	*_x8*.acm-validations.aws。	唯一
host.subdomain.example.com	*_x9*.host.subdomain.example.com。	*_x10*.acm-validations.aws。	唯一

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。

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.example.com。	别名记录	_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了至少一个证书，并且您正在创建该证书的 AWS 地区工作。如果您尝试打开主机并看到首次使用屏幕，或者成功打开主机但未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。

   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，您可以对 Route 53 进行 AWS CLI 或 API 调用，在 Route 53 的 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。在新证书的详细信息页面上，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 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。

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。要解决此问题，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

在 Amazon 证书颁发机构 (CA) 为您的网站颁发证书之前， AWS Certificate Manager (ACM) 必须验证您是否拥有或控制您在请求中指定的所有域。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。

请注意以下有关电子邮件验证的注意事项。

- 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。

- 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。
- 在创建采用电子邮件验证的证书后，您无法切换到使用 DNS 对其进行验证。

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。

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。

ACM 将电子邮件发送到您选择的超级域名。不超过最小网站地址的任何子域名都是有效的，并将用作电子邮件地址的域名作为 “@” 之后的后缀（例如，如果您将 example.com 指定为 subdomain.example.com 的验证域，则可以收到一封发送至 admin@example.com 的电子邮件）。

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**
>
> 我们强烈建议您配置和监控证书的五个常用系统地址。从 WHOIS 检索联系信息并不可靠。WHOIS 查询成功率很低（不到 5%），部分原因是为了遵循全球隐私保护法律的要求。

> **Important**
>
> 从 2024 年 6 月起，ACM 不再支持通过 WHOIS 联系地址进行新的电子邮件验证。对于现有证书，从 2024 年 10 月起，ACM 将不会向域名的 WHOIS 联系地址发送续订通知。ACM 将继续向所请求域的五个通用系统地址发送验证电子邮件。有关更多详细信息，请参阅AWS Certificate Manager 将停止 WHOIS 查询电子邮件验证证书

当您申请证书时，ACM 会向您在DomainName参数或可选ValidationDomain参数中指定的域名发送电子邮件。有关更多信息，请参阅 ???。

- 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。

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。这些地址是由在域名的其余部分预置管理员@、管理员@、hostmaster@、postmaster@ 和 webmaster@ 构成的。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 RequestCertificateAPI 或请求证书 AWS CLI 命令中指定ValidationDomain选项。如果使用控制台请求证书，目前还无法使用此功能。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。

有关更多信息，请参阅上面的电子邮件验证。

（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，请参阅故障排除中的排查电子邮件验证的问题部分。

> **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 中的证书不需要验证电子邮件。有关 DNS 域验证的信息，请参阅DNS 验证。

使用控制台重新发送验证电子邮件

1. 登录 AWS 管理控制台并打开 ACM 控制台，网址为 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**
   >
   > 根据您对列表排序的方式，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 AWS 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:444455556666:certificate/certificate_ID --domain www.example.com --
validation-domain example.com
```

> **Note**
>
> 该resend-validation-email命令仅适用于您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 AWS CLI 列出 ACM 管理的证书。控制台在一个页面中最多可以列出 500 个证书，CLI 最多可以列出 1000 个证书。

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 证书是否与诸如 Elastic Load Balancing 之类的 AWS 服务主动关联还是。CloudFront值可以是 No 或 Yes。
- Renewal eligibility（续订资格）– 当证书临近到期时，ACM 是否可以自动续订证书。可能的值为：Eligible（有资格）| Ineligible（没有资格）。有关资格规则，请参阅 ACM 证书的托管续订。

通过选择控制台右上角的设置图标，您可以自定义页面上显示的证书数量、指定单元格内容的换行行为以及显示其他信息字段。以下可选字段可用：

- Additional domain names（其他域名）– 证书中包含的一个或多个域名（主题备用名称）。
- Requested at（请求时间）– ACM 请求证书的时间。
- Issued at（颁发时间）– 颁发证书的时间。此信息仅适用于 Amazon 颁发的证书，不适用于导入的证书。

- Not before（不早于）– 证书无效之前的时间。
- Not after（不晚于）– 在该时间之后证书失效。
- Revoked at（吊销时间）– 对于已吊销的证书，这是指吊销时间。
- Name tag（名称标签）– 此证书上名为 Name 的标签的值（如果存在这样的标签）。
- Renewal status（续订状态）- 证书请求续订的状态。只有在请求续订时，此字段才会显示并具有值。可能的值为：Pending automatic renewal（待自动续订）| Pending validation（待验证）| Success（成功）| Failure（失败）。

> **Note**
>
> 对证书状态的更改可能需要数小时才能生效。如果遇到问题，则证书请求会在 72 小时后超时，并且必须从头开始重复颁发或续订过程。

Page size（页面大小）首选项指定了每个控制台页面上返回的证书数量。

有关可用证书详细信息的更多信息，请参阅 描述 ACM 证书。

要列出您的证书，请使用 AWS CLI

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:444455556666:certificate/certificate_ID",
            "DomainName": "example.com"
  "SubjectAlternativeNameSummaries": [
                "example.com",
                "other.example.com"
            ],
            "HasAdditionalSubjectAlternativeNames": false,
            "Status": "ISSUED",
            "Type": "IMPORTED",
```

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

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 控制台或列 AWS CLI 出有关您的证书的详细元数据。

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*`:`*444455556666*`: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、请求私有 PKI 证书 或 将证书导入 AWS Certificate Manager。
- 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）
- 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AWS 集成服务关联，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）- 支持具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）– 可能的值为：Eligible（有资格）| Ineligible（没有资格）。有关资格规则，请参阅 ACM 证书的托管续订。
  - Renewal status（续订状态）- 证书请求续订的状态。只有在请求续订时，此字段才会显示并具有值。可能的值为：Pending automatic renewal（待自动续订）| Pending validation（待验证）| Success（成功）| Failure（失败）。

> **Note**
>
> 对证书状态的更改可能需要数小时才能生效。如果遇到问题，则证书请求会在 72 小时后超时，并且必须从头开始重复颁发或续订过程。

  - CA – 签名 CA 的 ARN
- 标签
  - 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。

要查看证书详细信息，请使用 AWS CLI

使用中的描述证书显示证书详细信息，如以下命令所示： AWS CLI

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

命令返回类似于下文的信息：

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

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

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

删除 ACM 管理的证书

您可以使用 ACM 控制台或 AWS CLI 删除证书。

Important

- 您无法删除正在由其他 AWS 服务使用的 ACM 证书。要删除正在使用的证书，您必须先删除证书关联。这是使用相关服务的控制台或 CLI 完成的。
- 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《AWS Private Certificate Authority 用户指南》中的删除私有证书。

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

根据您对列表排序的方式，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。

要使用删除证书 AWS CLI

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:444455556666:certificate/certificate_ID
```

安装 ACM 证书

您不能使用 ACM 将公共证书直接安装到您的网站 AWS 或应用程序上。您必须使用与 ACM 集成的服务之一。有关更多信息，请参阅 与之集成的服务 AWS Certificate Manager。

由 CA 签名 AWS 私有 CA 并用于您的私有 PKI 的 ACM 证书可以手动导出并安装在您拥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：

- 如果与其他 AWS 服务（例如 Elastic Load Balancing 或）相关联，则符合资格 CloudFront。
- 如果它是自颁发或上次续订后导出的，则符合条件。
- 如果它是通过调用 ACM RequestCertificateAPI 颁发然后导出或与其他 AWS 服务关联的私有证书，则符合条件。
- 如果它是通过管理控制台颁发然后导出，或者是与其他 AWS 服务关联的私有证书，则符合条件。
- 如果它是通过调用 AWS 私有 CA IssueCertificateAPI 颁发的私有证书，则不符合资格。
- 如果是导入的证书，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	备注
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

域名	满足条件 1	满足条件 2	已允许	备注
xn--xyz.com	是	是	✓	有效的国际化域名（解析为简.com）
xn--example.com	是	不支持	✗	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。如果您在多个 AWS 区域拥有同一个域名的证书，则每个证书都必须单独续订。

主题

- 续订公开信任的证书
- 在私有 PKI 中续订证书
- 检查证书的续订状态

续订公开信任的证书

在颁发受管理的公开信任证书时， AWS Certificate Manager 需要您证明自己是域名所有者。这可以通过 DNS 验证或电子邮件验证的方式进行。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
- 续订通过电子邮件验证的域

续订通过 DNS 验证的域

对于最初使用 DNS 验证颁发的 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：

- 该证书目前正由 AWS 服务使用。
- 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。

如果在续订期间无法自动验证域名（例如，由于存在 CAA 记录），ACM 会发送 AWS Health EventBridge 事件和 Amazon 事件。这些事件将在过期前 45 天、30 天、15 天、7 天、3 天和 1 天发送。有关更多信息，请参阅 亚马逊对 ACM 的 EventBridge 支持。

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。

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 。

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），您可以使用 AWS Certificate Manager 控制台或 ACM API 来请求 ACM 向您发送证书续订的域验证电子邮件。您应在以下情况下执行此操作：

- 您最初请求 ACM 证书时使用的是电子邮件验证。
- 您的证书的续订状态为等待验证。有关确定证书的续订状态的信息，请参阅检查证书的续订状态。
- 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 AWS Certificate Manager 控制台，网址为 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 中使用该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 AWS 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
```

在私有 PKI 中续订证书

由来自的私有 CA 签署的 ACM 证书 AWS 私有 CA 有资格进行托管续订。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**
>
> 只有通过 ACM 控制台或 ACM API RequestCertificate操作获得的证书才有资格进行托管续订。 AWS 私有 CA 使用 AWS 私有 CA API IssueCertificate操作直接颁发的证书不由 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）。客户还可以随时使用 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。

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。您可以手动或在 Amazon 的帮助下AWS Health执行这些操作 EventBridge，AWS Lambda如下所示。有关更多信息，请参阅自动导出已续订的证书

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 AWS Lambda 开发者指南中的步骤，创建和配置一个调用 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。该策略允许函数中的代码通过调用 ACM API 的ExportCertificate操作来检索续订的证书和私钥。

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

2. 在 Amazon 中创建规则 EventBridge，用于监听 ACM 运行状况事件，并在检测到 Lambda 函数时调用该函数。ACM 每次尝试续订证书时都会写入 AWS 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。

   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 AWS CLI 手动测试 ACM 托管续订工作流程的配置。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**
>
> 您只能测试由颁发和导出的证书的续订 AWS 私有 CA。

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 集成服务相关联，则会部署新证书，并在 Amazon Events 中生成续订 CloudWatch 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 AWS 私有 CA 证书，必须先向 ACM 服务主体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 (AWS 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，指定要续订的私有证书的 ARN。然后使用该DescribeCertificate操作确认证书的续订详细信息已更新。

检查证书的续订状态

当您尝试续订证书时，ACM 会在证书详细信息中提供 Renewal status（续订状态）信息字段。您可以使用 AWS Certificate Manager 控制台、ACM API AWS CLI、或 AWS Health Dashboard 来检查 ACM 证书的续订状态。如果您使用控制台 AWS CLI、或 ACM API，则续订状态可以是下面列出的四个可能的状态值之一。如果使用 AWS 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。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。

如果证书与其他 AWS 服务（例如 Elastic Load Balancing 或）关联 CloudFront，或者该证书自签发或上次续订后已导出，则该证书有资格续订。

> **Note**
>
> 对续订状态的更改可能需要数小时才能生效。如果遇到问题，则续订请求会在 72 小时后超时，并且必须从头开始重复续订过程。有关问题排查帮助，请参阅排查证书请求问题。

主题

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 AWS Certificate Manager 控制台，网址为 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（续订状态）。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)

有关演示如何使用DescribeCertificate操作检查状态的 Java 示例，请参阅描述证书。

检查状态 (CLI)

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

```
$ aws acm describe-certificate \
 --certificate-arn arn:aws:acm:region: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，它会在证书到期后每隔 45 天、30 天、15 天、7 天、3 天和 1 天向您发送证书续订事件通知，告知您需要采取行动。 AWS Health Dashboard AWS Health Dashboard 是 AWS Health 服务的一部分。它不需要设置，您的账户中通过身份验证的任何用户都可以查看。有关更多信息，请参阅《AWS Health 用户指南》。

> **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 AWS Health Dashboard，请执行以下操作：

1. 登录网址为 AWS Health Dashboard 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 应用。
6. 对于 Event category，选择 Scheduled Change。
7. 选择 应用。

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手动操作。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

验证电子邮件具有以下两种格式之一，具体取决于是申请新证书还是续订现有证书。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。

验证新证书

电子邮件模板文本：

```
Greetings from Amazon Web Services,

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

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

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

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

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

Sincerely,
```

```
Amazon Web Services
```

验证证书以进行续订

电子邮件模板文本：

```
Greetings from Amazon Web Services,

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

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

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

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

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

Sincerely,
Amazon Web Services

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

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

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

收到来自的新验证消息后 AWS，我们建议您将其用作解析器的最 up-to-date 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，AWS 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.amazon.com 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。

- 当 ACM 控制台处理多域证书请求时，它会向 TODO 发送验证电子邮件。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参数为指定值ValidationDomain，

该值是该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。

将证书导入 AWS Certificate Manager

除了申请 AWS Certificate Manager (ACM) 提供的 SSL/TLS 证书外，您还可以导入在外部获得的证书。 AWS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AWS 服务。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。

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。您可以使用 Amazon CloudWatch Events 在导入的证书即将到期时发送通知，从而简化此任务。有关更多信息，请参阅 使用亚马逊 EventBridge。

ACM 中的所有证书都是区域性资源，包括您导入的证书。要将同一证书用于不同 AWS 区域的 Elastic Load Balancing 负载均衡器，您必须将该证书导入要使用的每个区域。要在 Amazon 上使用证书 CloudFront，您必须将其导入美国东部（弗吉尼亚北部）地区。有关更多信息，请参阅 支持的区域。

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。

主题

- 导入证书的先决条件
- 证书和密钥的导入格式
- 导入证书
- 重新导入证书

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AWS 证书颁发机构 (CA) 签发的证书，您还必须包括证书的私有密钥和公有密钥。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）：

- RSA 1024 位 (RSA_1024)
- RSA 2048 位 (RSA_2048)
- RSA 3072 位 (RSA_3072)
- RSA 4096 位 (RSA_4096)
- ECDSA 256 位 (EC_prime256v1)
- ECDSA 384 位 (EC_secp384r1)
- ECDSA 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、3072 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自签名，也可以由颁发 CA 的私有密钥签名。您必须提供私有密钥，该私有密钥不得超过 5 KB（5,120 字节），并且必须未加密。
- 如果证书由 CA 签名，并且您选择提供证书链，则该链必须采用 PEM 编码。
- 证书在导入时必须是有效的。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。
- 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。
- AWS CloudFormation 不支持将证书导入 ACM。

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**
>
> AWS 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

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、或 ACM API 将外部获得的证书（即第三方信任服务提供商提供的证书）导入 ACM。AWS Management Console AWS CLI以下主题向您展示了如何使用 AWS Management Console 和 AWS CLI。从非颁发者处获取证书的程序不在本指南的范围之内。AWS

> **Important**
>
> 您选择的签名算法必须满足 导入证书的先决条件。

主题

- 导入（控制台）
- 导入 (AWS CLI)

导入（控制台）

以下示例说明如何使用 AWS 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 AWS Certificate Manager 标题，并选择其下方的 Get started（开始使用）按钮。
2. 选择导入证书。
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. （可选）要向导入的证书添加标签，请选择标签。标签是您分配给 AWS 资源的标签。每个标签都包含定义的一个密钥和一个可选值。您可以使用标签来组织资源或跟踪 AWS 成本。
5. 选择 Import（导入）。

导入 (AWS CLI)

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

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

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

```
$ aws acm import-certificate --certificate 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

如果您导入了证书并将其与其他 AWS 服务关联，则可以在证书到期之前重新导入该证书，同时保留原始证书的 AWS 服务关联。有关与 ACM 集成的 AWS 服务的更多信息，请参阅与之集成的服务 AWS Certificate Manager。

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）

以下示例说明如何使用 AWS 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. 选择审核并导入。

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 (AWS CLI)

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

- PEM 编码的证书存储在名为 `Certificate.pem` 的文件中。
- PEM 编码的证书链存储在名为 `CertificateChain.pem` 的文件中。
- （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

您可以导出由颁发的证书，AWS 私有 CA 以便在私有 PKI 环境中的任何地方使用。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。有关的更多信息 AWS 私有 CA，请参阅《AWS Private Certificate Authority 用户指南》。

> **Note**
>
> 无论公开信任的证书或其私钥是由 ACM 颁发还是导入的，您都无法导出该证书。

主题

导出私有证书（控制台）

1. 登录 AWS 管理控制台并打开 ACM 控制台，网址为 https://console.aws.amazon.com/acm/home。
2. 选择 Certificate Manager。
3. 选择要导出的证书的链接。
4. 选择导出。
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:444455556666:certificate/certificate_ID \
     --passphrase fileb://path-to-passphrase-file  \
     | jq -r '"\(.Certificate)\(.CertificateChain)\(.PrivateKey)"'

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

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

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

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

标记 AWS 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 AWS Certificate Manager 控制台、AWS Command Line Interface (AWS CLI) 或 ACM API 来添加、查看或删除 ACM 证书的标签。您可以选择要在 ACM 控制台中显示的标签。

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

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

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

主题

标签限制

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 AWS 使用；您无法添加、编辑或删除其键以 `aws:` 开头的标签。以 `aws:` 开头的标签不计入每个资源的标签数限制。
- 如果您计划在多个服务和资源中使用添加标签方案，请记得其他服务可能对允许使用的字符有其他限制。请参阅该服务对应的文档。

- ACM 证书标签不可在 AWS Management Console 的 Resource Groups 和标签编辑器中使用。

有关AWS标记约定的一般信息，请参阅标记AWS资源。

管理标签

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

管理标签（控制台）

您可以使用 AWS 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 AWS Management Console 并打开 AWS Certificate Manager 控制台 (https://console.aws.amazon.com/acm/home)。
2. 选择要为其添加标签的证书旁的箭头。
3. 在详细信息窗格中，向下滚动到 Tags。
4. 选择 Edit 和 Add Tag。
5. 键入标签的键和值。
6. 选择 Save（保存）。

删除标签

使用 ACM 控制台通过以下过程删除标签。

删除标签 (控制台)

1. 登录 AWS Management Console 并打开 AWS Certificate Manager 控制台 (https://console.aws.amazon.com/acm/home)。
2. 选择要删除其标签的证书旁的箭头。

3. 在详细信息窗格中，向下滚动到 Tags。
4. 选择编辑。
5. 选择要删除的标签旁的 X。
6. 选择 Save（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 AWS Management Console 并打开 AWS Certificate Manager 控制台 (https://console.aws.amazon.com/acm/home)。
2. 选择要编辑的证书旁的箭头。
3. 在详细信息窗格中，向下滚动到 Tags。
4. 选择编辑。
5. 修改要更改的标签的键或值。
6. 选择 Save（保存）。

在列中显示标签

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 AWS Management Console 并打开 AWS Certificate Manager 控制台 (https://console.aws.amazon.com/acm/home)。
2. 通过选择控制台右上角的齿轮图标

   来选择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 AWS CLI 添加、列出和删除标签。

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

管理标签 (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

监控和记录 AWS Certificate Manager

监控是维护 AWS 解决方案的可靠性、可用性和性能的重要组成部分。 AWS Certificate Manager 您应该从 AWS 解决方案的所有部分收集监控数据，以便在出现多点故障时可以更轻松地进行调试。

以下主题介绍可用于 ACM 的 AWS 云监控工具。

主题

使用亚马逊 EventBridge

您可以使用 Amazon EventBridge（前身为 CloudWatch Events）实现 AWS 服务自动化，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。来自包括 ACM 在内的 AWS 服务的事件几乎实时地传送到 Amazon EventBridge 。您可以使用事件触发目标，包括 AWS Lambda 函数、 AWS Batch 作业、Amazon SNS 主题等。有关更多信息，请参阅什么是亚马逊 EventBridge？

主题

亚马逊对 ACM 的 EventBridge 支持

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

“ACM 证书即将到期”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。可以使用 ACM API 的PutAccountConfiguration操作来更改此时间。

ACM 会自动启动其颁发的符合条件的证书的续订，但是为了避免中断，导入的证书需要在到期之前重新签发并重新导入。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。有关使用自动化的示例 AWS Lambda，请参阅 EventBridge 在 ACM 中使用亚马逊触发操作。

ACM 证书即将到期事件具有以下结构。

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

“ACM 证书已过期”事件

> **Note**
>
> 证书过期事件不适用于导入的证书。

客户可以侦听此事件，以便在其账户中 ACM 颁发的公有或私有证书过期时收到提醒。

ACM 证书已过期事件具有以下结构。

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

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

“ACM 证书可用”活动

客户可以侦听此事件，以便在托管的公有或私有证书准备就绪时收到通知。该事件将在颁发、续订和导入时发布。对于私有证书，一旦它可用，仍然需要客户操作才能将其部署到主机。

ACM 证书可用事件具有以下结构。

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

“ACM 证书需要续订操作”事件

> **Note**
>
> 需要续订证书操作的事件不适用于导入的证书。

客户可以侦听此事件，以便在必须采取客户操作后才能续订证书时收到提醒。例如，如果客户添加了阻止 ACM 续订证书的 CAA 记录，则 ACM 将在到期前 45 天自动续订失败时发布此事件。如果客户未采取任何操作，则 ACM 将在 30 天、15 天、3 天和 1 天内再次尝试续订，或者直到客户采取操作、证书过期或证书不再有资格续订。每次续订尝试都会发布一个事件。

ACM 证书需要续订操作事件具有以下结构。

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

AWS 健康事件

AWS 系统会为符合续订条件的 ACM 证书生成运行状况事件。有关续订资格的信息，请参阅ACM 证书的托管续订。

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"
      ]
   }
}
```

EventBridge 在 ACM 中使用亚马逊触发操作

您可以根据这些事件创建 Amazon EventBridge 规则，并使用 Amazon EventBridge 控制台配置检测到事件时发生的操作。本节提供配置 Amazon EventBridge 规则和由此产生的操作的示例程序。

主题

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 Amazon EventBridge 规则并触发操作

1. 创建 Amazon EventBridge 规则。有关更多信息，请参阅创建对事件做出反应的 Amazon EventBridge 规则。

   a. 在亚马逊 EventBridge 控制台 https://console.aws.amazon.com/events/ 中，导航至事件 > 规则页面，然后选择创建规则。

   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 AWS 健康事件 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 AWS Lambda 实现的示例，请参阅 使用 Lambda 函数响应事件。

使用 Lambda 函数响应事件

此过程演示 AWS Lambda 如何使用在亚马逊上监听 EventBridge、使用亚马逊简单通知服务 (SNS) Simple Notification Service 创建通知以及如何向其发布调查结果 AWS Security Hub，从而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 AWS 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。不建议直接在用户帐户下运行大多数 AWS 操作，尤其不要在管理员帐户下运行。

   通过 https://console.aws.amazon.com/iam/ 打开 IAM 控制台。

2. 使用 JSON 策略编辑器创建在以下模板中定义的策略。提供您自己的地区和 AWS 账户详情。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 AWS 服务创建角色（控制台）。
4. 打开 AWS Lambda 控制台，网址为 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. 在函数名称字段中指定一个名称，例如 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 AWS Security Hub 中指定。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，当通过 Amazon EventBridge 发起活动时，我们为每份即将到期的证书提供一封电子邮件。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操作进行自定义。）这些事件中的每一个都会触发以下级联的自动操作：

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

          Event matches Amazon EventBridge rule #
```

```
                    Rule calls Lambda function #

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

1. 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）。
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. 创建用于触发 Lambda 函数的亚马逊 EventBridge 规则。有关更多信息，请参阅创建对事件做出反应的 Amazon EventBridge 规则。

   在亚马逊 EventBridge 控制台 https://console.aws.amazon.com/events/ 中，导航至事件 > 规则页面，然后选择创建规则。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 赛事规则
- CloudWatch 与 Lambda 关联的日志
- SNS 主题

CloudTrail 与一起使用 AWS Certificate Manager

AWS Certificate Manager 与 AWS CloudTrail一项服务集成，该服务提供用户、角色或 AWS 服务在 ACM 中执行的操作的记录。 CloudTrail 默认情况下，您的 AWS 账户已启用。 CloudTrail 将 ACM 的 API 调用捕获为事件，包括来自 ACM 控制台的调用和对 ACM API 操作的代码调用。如果您配置跟踪，则可以允许将 CloudTrail 事件持续传输到 Amazon S3 存储桶，包括 ACM 的事件。如果您未配置跟踪，您仍然可以在 CloudTrail 控制台的事件历史记录中查看最新的事件。

使用收集的信息 CloudTrail，您可以确定向 ACM 发出的请求、发出请求的 IP 地址、谁发出了请求、何时发出请求以及其他详细信息。有关更多信息，请参阅使用事件历史记录查看 CloudTrail 事件。当 ACM 中出现支持的事件活动时，该活动会与其他 AWS 服务 CloudTrail 事件一起记录在事件历史记录中。您可以在 AWS 账户中查看、搜索和下载最新事件。

此外，您可以配置其他 AWS 服务，以进一步分析和处理 CloudTrail 日志中收集的事件数据。

有关的更多信息 CloudTrail，请参阅以下文档：

- AWS CloudTrail 用户指南。
- 创建跟踪概述
- CloudTrail 支持的服务和集成

- 配置 Amazon SNS 通知 CloudTrail
- 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

- 日志记录中 CloudTrail 支持 ACM API 操作
- 记录集成服务的 API 调用

日志记录中 CloudTrail 支持 ACM API 操作

ACM 支持将以下操作作为事件记录在 CloudTrail 日志文件中：

每个事件或日记账条目都包含有关生成请求的人员信息。身份信息有助于您确定以下内容：

- 请求是使用 AWS 账户根用户 或 AWS Identity and Access Management (IAM) 用户证书发出的。
- 请求是使用角色还是联合用户的临时安全凭证发出的。
- 请求是否由其他 AWS 服务发出

有关更多信息，请参阅CloudTrail用户身份元素。

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)
- 删除证书 (DeleteCertificate)
- 描述证书 (DescribeCertificate)
- 导出证书 (ExportCertificate)
- 导入证书 (ImportCertificate)
- 列出证书 (ListCertificates)
- 列出证书的标签 (ListTagsForCertificate)
- 从证书中删除标签 (RemoveTagsFromCertificate)
- 请求证书 (RequestCertificate)
- 重新发送验证电子邮件 (ResendValidationEmail)
- 检索证书 (GetCertificate)

向证书添加标签 (AddTagsToCertificate)

以下 CloudTrail 示例显示了调用 AddTagsToCertificateAPI 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 示例显示了调用 DeleteCertificateAPI 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 示例显示了调用 DescribeCertificateAPI 的结果。

> **Note**
>
> `DescribeCertificate`操作 CloudTrail 日志不显示有关您指定的 ACM 证书的信息。您可以使用控制台 AWS Command Line Interface、或 DescribeCertificateAPI 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 示例显示了调用 ExportCertificateAPI 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
               ],
               "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 ACM ImportCertificateAPI 操作调用的 CloudTrail 日志条目。

```
{
   "eventVersion":"1.04",
   "userIdentity":{
      "type":"IAMUser",
      "principalId":"AIDACKCEVSQ6C2EXAMPLE",
      "arn":"arn:aws:iam::111122223333:user/Alice",
```

```
      "accountId":"111122223333",
      "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
      "userName":"Alice"
   },
   "eventTime":"2016-10-04T16:01:30Z",
   "eventSource":"acm.amazonaws.com",
   "eventName":"ImportCertificate",
   "awsRegion":"ap-southeast-2",
   "sourceIPAddress":"54.240.193.129",
   "userAgent":"Coral/Netty",
   "requestParameters":{
      "privateKey":{
         "hb":[
            "byte",
            "byte",
            "byte",
            "..."
         ],
         "offset":0,
         "isReadOnly":false,
         "bigEndian":true,
         "nativeByteOrder":false,
         "mark":-1,
         "position":0,
         "limit":1674,
         "capacity":1674,
         "address":0
      },
      "certificateChain":{
         "hb":[
            "byte",
            "byte",
            "byte",
            "..."
         ],
         "offset":0,
         "isReadOnly":false,
         "bigEndian":true,
         "nativeByteOrder":false,
         "mark":-1,
         "position":0,
         "limit":2105,
         "capacity":2105,
         "address":0
```

```
      },
      "certificate":{
         "hb":[
            "byte",
            "byte",
            "byte",
            "..."
         ],
         "offset":0,
         "isReadOnly":false,
         "bigEndian":true,
         "nativeByteOrder":false,
         "mark":-1,
         "position":0,
         "limit":2503,
         "capacity":2503,
         "address":0
      }
   },
   "responseElements":{
      "certificateArn":"arn:aws:acm:ap-
southeast-2:111122223333:certificate/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 示例显示了调用 ListCertificatesAPI 的结果。

> **Note**
>
> `ListCertificates`操作 CloudTrail 日志不显示您的 ACM 证书。您可以使用控制台 AWS Command Line Interface、或 ListCertificatesAPI 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":"cdfe1051-88aa-4aa3-8c33-a325270bff21",
        "eventType":"AwsApiCall",
        "recipientAccountId":"123456789012"
      }
   ]
}
```

列出证书的标签 (ListTagsForCertificate)

以下 CloudTrail 示例显示了调用 ListTagsForCertificateAPI 的结果。

> **Note**
>
> 该ListTagsForCertificate操作的 CloudTrail 日志不会显示您的标签。您可以使用控制台 AWS Command Line Interface、或 ListTagsForCertificateAPI 查看标签列表。

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

```
      {
         "eventVersion":"1.04",
         "userIdentity":{
            "type":"IAMUser",
            "principalId":"AIDACKCEVSQ6C2EXAMPLE",
            "arn":"arn:aws:iam::123456789012:user/Alice",
            "accountId":"123456789012",
            "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
            "userName":"Alice"
         },
         "eventTime":"2016-04-06T13:30:11Z",
         "eventSource":"acm.amazonaws.com",
         "eventName":"ListTagsForCertificate",
         "awsRegion":"us-east-1",
         "sourceIPAddress":"192.0.2.0",
         "userAgent":"aws-cli/1.10.16",
         "requestParameters":{
            "certificateArn":"arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
         },
         "responseElements":null,
         "requestID":"b010767f-fbfb-11e5-b596-79e9a97a2544",
         "eventID":"32181be6-a4a0-48d3-8014-c0d972b5163b",
         "eventType":"AwsApiCall",
         "recipientAccountId":"123456789012"
      }
   ]
}
```

从证书中删除标签 (RemoveTagsFromCertificate)

以下 CloudTrail 示例显示了调用 RemoveTagsFromCertificateAPI 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 示例显示了调用 RequestCertificateAPI 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 示例显示了调用 ResendValidationEmailAPI 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 示例显示了调用 GetCertificateAPI 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

您可以使用审核 CloudTrail 与 ACM 集成的服务发出的 API 调用。有关使用的更多信息 CloudTrail，请参阅《AWS 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 AWS 资源。

主题

- 创建负载均衡器

创建负载均衡器

您可以使用审核 CloudTrail 与 ACM 集成的服务发出的 API 调用。有关使用的更多信息 CloudTrail，请参阅《AWS CloudTrail 用户指南》。以下示例显示了可以生成的日志类型，具体取决于您配置 ACM 证书所依据的 AWS 资源。

主题

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 AWS Command Line Interface来完成。以下示例显示了对 AWS 账户 123456789012 LinuxTest 上名RegisterInstancesWithLoadBalancer为的负载均衡器的调用。

```
{
   "eventVersion":"1.03",
   "userIdentity":{
      "type":"IAMUser",
      "principalId":"AIDACKCEVSQ6C2EXAMPLE",
      "arn":"arn:aws:iam::123456789012:user/ALice",
      "accountId":"123456789012",
      "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
      "userName":"Alice",
      "sessionContext":{
         "attributes":{
            "mfaAuthenticated":"false",
            "creationDate":"2016-01-01T19:35:52Z"
         }
      },
      "invokedBy":"signin.amazonaws.com"
   },
   "eventTime":"2016-01-01T21:11:45Z",
```

```
   "eventSource":"elasticloadbalancing.amazonaws.com",
   "eventName":"RegisterInstancesWithLoadBalancer",
   "awsRegion":"us-east-1",
   "sourceIPAddress":"192.0.2.0/24",
   "userAgent":"signin.amazonaws.com",
   "requestParameters":{
      "loadBalancerName":"LinuxTest",
      "instances":[
         {
            "instanceId":"i-c67f4e78"
         }
      ]
   },
   "responseElements":{
      "instances":[
         {
            "instanceId":"i-c67f4e78"
         }
      ]
   },
   "requestID":"438b07dc-b0cc-11e5-8afb-cda7ba020551",
   "eventID":"9f284ca6-cbe5-42a1-8251-4f0e6b5739d6",
   "eventType":"AwsApiCall",
   "recipientAccountId":"123456789012"
}
```

加密私有密钥

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

```
{
   "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。解密是在内部执行的 AWS，解密后的密钥永远不会离开。 AWS

```
{
   "eventVersion":"1.03",
   "userIdentity":{
      "type":"AssumedRole",
      "principalId":"AIDACKCEVSQ6C2EXAMPLE:1aba0dc8b3a728d6998c234a99178eff",
      "arn":"arn:aws:sts::111122223333:assumed-role/
DecryptACMCertificate/1aba0dc8b3a728d6998c234a99178eff",
      "accountId":"111122223333",
      "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
      "sessionContext":{
```

```
         "attributes":{
            "mfaAuthenticated":"false",
            "creationDate":"2016-01-01T21:13:28Z"
         },
         "sessionIssuer":{
            "type":"Role",
            "principalId":"APKAEIBAERJR2EXAMPLE",
            "arn":"arn:aws:iam::111122223333:role/DecryptACMCertificate",
            "accountId":"111122223333",
            "userName":"DecryptACMCertificate"
         }
      }
   },
   "eventTime":"2016-01-01T21:13:28Z",
   "eventSource":"kms.amazonaws.com",
   "eventName":"Decrypt",
   "awsRegion":"us-east-1",
   "sourceIPAddress":"AWS Internal",
   "userAgent":"aws-internal/3",
   "requestParameters":{
      "encryptionContext":{
         "aws:elasticloadbalancing:arn":"arn:aws:elasticloadbalancing:us-
east-1:123456789012:loadbalancer/LinuxTest",
         "aws:acm:arn":"arn:aws:acm:us-
east-1:123456789012:certificate/87654321-4321-4321-4321-210987654321"
      }
   },
   "responseElements":null,
   "requestID":"809a70ff-b0cc-11e5-8f42-c7fdf1cb6e6a",
   "eventID":"7f89f7a7-baff-4802-8a88-851488607fb9",
   "readOnly":true,
   "resources":[
      {
         "ARN":"arn:aws:kms:us-
east-1:123456789012:key/12345678-1234-1234-1234-123456789012",
         "accountId":"123456789012"
      }
   ],
   "eventType":"AwsServiceEvent",
   "recipientAccountId":"123456789012"
}
```

支持的 CloudWatch 指标

Amazon CloudWatch 是一项 AWS 资源监控服务。您可以使用 CloudWatch 来收集和跟踪指标、设置警报以及自动对 AWS 资源变化做出反应。ACM 会每天为账户中的每个证书发布一次指标，直到到期。

`AWS/CertificateManager` 命名空间包括以下指标。

指标	描述	单位	尺寸
`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 API（Java 示例）

您可使用 AWS Certificate Manager API 通过发送 HTTP 请求以编程方式与该服务进行交互。有关详细信息，请参阅 AWS Certificate Manager API 参考。

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

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

主题

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 AWS
 Certificate Manager
 * service.
 *
 * Input parameters:
 *   Accesskey - AWS access key
 *   SecretKey - AWS secret key
 *   CertificateArn - Use to reimport a certificate (not included in this example).
 *   region - AWS region
 *   Certificate - PEM file that contains the certificate to import. Ex: /data/certs/
servercert.pem
 *   CertificateChain - The certificate chain, not including the end-entity
 certificate.
 *   PrivateKey - The private key that matches the public key in the certificate.
 *
 * Output parameter:
 *   CertificcateArn - The ARN of the imported certificate.
 *
 */
public class AWSCertificateManagerSample {

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

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

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

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

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

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

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

    client.addTagsToCertificate(addTagsToCertificateRequest);
   }

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

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 AWS
 Certificate
 * Manager service.
 *
 * Input parameter:
 *   CertificateArn - The ARN of the certificate to delete.
 *
 */

public class AWSCertificateManagerExample {

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

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

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

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

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

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

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

   }
}
```

描述证书

以下示例说明如何使用 DescribeCertificate 函数。

```
package com.amazonaws.samples;

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

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

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

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

public class AWSCertificateManagerExample {

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

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

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

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

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

      DescribeCertificateResult result = null;
      try{
```

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

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

    }
}
```

如果成功，上述示例将显示类似于以下内容的信息。

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

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

导出证书

以下示例演示如何使用 ExportCertificate 函数。该函数将导出由私有证书颁发机构 (CA) 颁发的私有证书（PKCS #8 格式）。（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 AWS
 Certificate
 * Manager service.
 *
 * Input parameter:
 *   CertificateArn - The ARN of the certificate to retrieve.
 *
 * Output parameters:
 *   Certificate - A base64-encoded certificate in PEM format.
 *   CertificateChain - The base64-encoded certificate chain in PEM format.
 *
 */
```

```
public class AWSCertificateManagerExample {

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

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

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

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

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

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

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

         timeSlept += sleepInterval;
      }

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

前面的示例将创建类似于以下内容的输出。

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

导入证书

以下示例演示如何使用 ImportCertificate 函数。

```
package com.amazonaws.samples;

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

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

import com.amazonaws.services.certificatemanager.model.ImportCertificateRequest;
import com.amazonaws.services.certificatemanager.model.ImportCertificateResult;
import com.amazonaws.services.certificatemanager.model.LimitExceededException;
```

```
import com.amazonaws.services.certificatemanager.model.ResourceNotFoundException;
import com.amazonaws.AmazonClientException;
import java.io.FileNotFoundException;
import java.io.IOException;

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

/**
 * This sample demonstrates how to use the ImportCertificate function in the AWS
 Certificate Manager
 * service.
 *
 * Input parameters:
 *   Certificate - PEM file that contains the certificate to import.
 *   CertificateArn - Use to reimport a certificate (not included in this example).
 *   CertificateChain - The certificate chain, not including the end-entity
 certificate.
 *   PrivateKey - The private key that matches the public key in the certificate.
 *
 * Output parameter:
 *   CertificcateArn - The ARN of the imported certificate.
 *
 */
public class AWSCertificateManagerSample {

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

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

      // Create a client.
      AWSCertificateManager client = AWSCertificateManagerClientBuilder.standard()
              .withRegion(Regions.US_EAST_1)
```

```
                .withCredentials(new AWSStaticCredentialsProvider(credentials))
                .build();

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

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

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

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

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

           // The files have been mapped, so clean up.
```

```
            channel_certificate.close();
            channel_chain.close();
            channel_key.close();
            file_certificate.close();
            file_chain.close();
            file_key.close();
        }
        catch (IOException ex)
        {
            throw ex;
        }

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

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

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

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

列出证书

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 AWS
 Certificate
 * Manager service.
 *
 * Input parameters:
 *   CertificateStatuses - An array of strings that contains the statuses to use for
 filtering.
 *   MaxItems - The maximum number of certificates to return in the response.
 *   NextToken - Use when paginating results.
 *
 * Output parameters:
 *   CertificateSummaryList - A list of certificates.
 *   NextToken - Use to show additional results when paginating a truncated list.
 *
 */

public class AWSCertificateManagerExample {

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

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

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

      // Create a request object and set the parameters.
      ListCertificatesRequest req = new ListCertificatesRequest();
      List<String> Statuses = Arrays.asList("ISSUED", "EXPIRED", "PENDING_VALIDATION",
 "FAILED");
      req.setCertificateStatuses(Statuses);
      req.setMaxItems(10);

      // Retrieve the list of certificates.
      ListCertificatesResult result = null;
      try {
         result = client.listCertificates(req);
      }
      catch (Exception ex)
      {
         throw ex;
      }

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

前面的示例将创建类似于以下内容的输出。

```
{
```

```
    CertificateSummaryList: [{
        CertificateArn:
 arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012,
        DomainName: www.example1.com
    },
    {
        CertificateArn:
 arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012,
        DomainName: www.example2.com
    },
    {
        CertificateArn:
 arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012,
        DomainName: www.example3.com
    }]
}
```

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 AWS 私有 CA 证书，您必须先向 ACM 服务主体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 AWS
 Certificate
 * Manager service.
 *
 * Input parameter:
```

```
 *   CertificateArn - The ARN of the certificate whose tags you want to list.
 *
*/

public class AWSCertificateManagerExample {

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

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

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

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

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

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

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

   }
}
```

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
 AWS Certificate
 * Manager service.
 *
 * Input parameters:
```

```
 *   CertificateArn - The ARN of the certificate from which you want to remove one or
 more tags.
 *   Tags - A collection of key-value pairs that specify which tags to remove.
 *
*/

public class AWSCertificateManagerExample {

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

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

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

      // Specify the tags to remove.
      Tag tag1 = new Tag();
      tag1.setKey("Short_Name");
      tag1.setValue("My_Cert");

      Tag tag2 = new Tag()
            .withKey("Purpose")
            .withValue("Test");

      // Add the tags to a collection.
      ArrayList<Tag> tags = new ArrayList<Tag>();
      tags.add(tag1);
      tags.add(tag2);

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

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

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

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

请求证书

以下示例说明如何使用 RequestCertificate 函数。

```
package com.amazonaws.samples;

import com.amazonaws.services.certificatemanager.AWSCertificateManagerClientBuilder;
import com.amazonaws.services.certificatemanager.AWSCertificateManager;
import com.amazonaws.services.certificatemanager.model.RequestCertificateRequest;
import com.amazonaws.services.certificatemanager.model.RequestCertificateResult;

import
 com.amazonaws.services.certificatemanager.model.InvalidDomainValidationOptionsException;
import com.amazonaws.services.certificatemanager.model.LimitExceededException;
import com.amazonaws.AmazonClientException;
```

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

import java.util.ArrayList;

/**
 * This sample demonstrates how to use the RequestCertificate function in the AWS
 Certificate
 * Manager service.
 *
 * Input parameters:
 *   DomainName - FQDN of your site.
 *   DomainValidationOptions - Domain name for email validation.
 *   IdempotencyToken - Distinguishes between calls to RequestCertificate.
 *   SubjectAlternativeNames - Additional FQDNs for the subject alternative names
 extension.
 *
 * Output parameter:
 *   Certificate ARN - The Amazon Resource Name (ARN) of the certificate you requested.
 *
*/

public class AWSCertificateManagerExample {

   public static void main(String[] args) {

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

      // Create a client.
      AWSCertificateManager client = AWSCertificateManagerClientBuilder.standard()
              .withRegion(Regions.US_EAST_1)
```

```
                .withCredentials(new AWSStaticCredentialsProvider(credentials))
                .build();

        // Specify a SAN.
        ArrayList<String> san = new ArrayList<String>();
        san.add("www.example.com");

        // Create a request object and set the input parameters.
        RequestCertificateRequest req = new RequestCertificateRequest();
        req.setDomainName("example.com");
        req.setIdempotencyToken("1Aq25pTy");
        req.setSubjectAlternativeNames(san);

        // Create a result object and display the certificate ARN.
        RequestCertificateResult result = null;
        try {
           result = client.requestCertificate(req);
        }
        catch(InvalidDomainValidationOptionsException ex)
        {
           throw ex;
        }
        catch(LimitExceededException ex)
        {
           throw ex;
        }

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

    }

}
```

前面的示例将创建类似于以下内容的输出。

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

重新发送验证电子邮件

以下示例向您说明如何使用 ResendValidationEmail 函数。

```
package com.amazonaws.samples;

import com.amazonaws.services.certificatemanager.AWSCertificateManagerClientBuilder;
import com.amazonaws.services.certificatemanager.AWSCertificateManager;
import com.amazonaws.services.certificatemanager.model.ResendValidationEmailRequest;
import com.amazonaws.services.certificatemanager.model.ResendValidationEmailResult;

import
 com.amazonaws.services.certificatemanager.model.InvalidDomainValidationOptionsException;
import com.amazonaws.services.certificatemanager.model.ResourceNotFoundException;
import com.amazonaws.services.certificatemanager.model.InvalidStateException;
import com.amazonaws.services.certificatemanager.model.InvalidArnException;
import com.amazonaws.AmazonClientException;

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

/**
 * This sample demonstrates how to use the ResendValidationEmail function in the AWS
 Certificate
 * Manager service.
 *
 * Input parameters:
 *   CertificateArn - Amazon Resource Name (ARN) of the certificate request.
 *   Domain - FQDN in the certificate request.
 *   ValidationDomain - The base validation domain that is used to send email.
 *
*/

public class AWSCertificateManagerExample {

   public static void main(String[] args) {

      // Retrieve your credentials from the C:\Users\name\.aws\credentials file in
 Windows
      // or the ~/.aws/credentials file in Linux.
      AWSCredentials credentials = null;
      try {
          credentials = new ProfileCredentialsProvider().getCredentials();
      }
      catch (Exception ex) {
```

```
            throw new AmazonClientException("Cannot load your credentials from file.",
 ex);
      }

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

      // Create a request object and set the input parameters.
      ResendValidationEmailRequest req = new ResendValidationEmailRequest();

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

      // Create a result object.
      ResendValidationEmailResult result = null;
      try {
         result = client.resendValidationEmail(req);
      }
      catch(ResourceNotFoundException ex)
      {
         throw ex;
      }
      catch (InvalidStateException ex)
      {
         throw ex;
      }
      catch (InvalidArnException ex)
      {
         throw ex;
      }
      catch (InvalidDomainValidationOptionsException ex)
      {
         throw ex;
      }

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

   }
```

```
}
```

前面的示例将重新发送验证电子邮件并显示一个空集。

故障排除

如果在使用 AWS 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AWS 知识中心。

主题

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

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 验证或电子邮件验证。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

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 或 未收到验证电子邮件。如果您遵循这些步骤并继续收到无可用联系人消息，请将此情况报告给 AWS，以便我们可以进行调查。

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 AWS 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 AWS 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。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。

> **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。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 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。

如果您遇到验证问题，请查阅以下主题。

主题

- 排查 DNS 验证问题
- 排查电子邮件验证的问题

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。

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

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` 可更改为 `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	Type	值
`_<random value>.example.com.`	别名记录	`_<random value>.acm-validations.aws.`
`_<random value>.example.com.`	别名记录	`<random value>.acm-validations.aws.`

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。

开启的 DNS 验证 GoDaddy 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:
 _cjhwou20vhu2exampleuw20vuyb2ovb9.j9s73ucn9vy.acm-validations.aws.
```

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

```
NAME: _ho9hv39800vb3examplew3vnewoib3u VALUE:
 _cjhwou20vhu2exampleuw20vuyb2ovb9.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，则 AWS Certificate Manager 可以直接与其交互以验证您的域名所有权。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。

- 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。

AWS 确实通过该AWS 私有 CA服务为公开不受信任的域提供支持。

验证成功，但签发或续订失败

如果证书颁发失败，显示“待验证”，则即使 DNS 正确也请检查证书颁发机构授权 (CAA) 记录没有阻止颁发证书。有关更多信息，请参阅 （ 可选 ）配置 CAA 记录。

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

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。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。除非 AWS 账户所有者也在 WHOIS 中被列为域名联系人，否则不会向账户所有者发送电子邮件。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 中的域名注册人、技术和管理联系人可能不公开， AWS 因此无法联系到这些联系人。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，域名联系人信息可能使用隐私地址，例如通过 WhoisGuard 或提供的隐私地址 PrivacyGuard。

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`、`contact.gandi.net` 或 `identity-protect.org` 电子邮件地址。确保您的域注册者文件上的注册者电子邮件是最新的，以便发送到这些隐藏的电子邮件地址的电子邮件可转发到您控制的电子邮件地址。

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

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

在提供 AWS 将验证电子邮件发送到的八个电子邮件地址中的至少一个地址并确认您可以收到该地址的电子邮件后，您便可以通过 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 AWS 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**
>
> 要绕过要求 MX 记录的要求，您可以使用 RequestCertificateAPI 中的`ValidationDomain`选项或 request-certification AWS CLI 命令来指定 ACM 向其发送验证电子邮件的域名。如果您使用 API 或 AWS CLI，则 AWS 不执行 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 AWS Support 中心并创建案例。如果您没有支持协议，请将消息发布到 ACM 开发论坛。

已发送到子域的电子邮件

如果您使用控制台并为子域名称（如 `sub.test.example.com`）请求证书，则 ACM 会进行检查，查看 `sub.test.example.com` 是否存在 MX 记录。如果不存在，则检查父域 `test.example.com`，依此类推，直至基础域 `example.com`。如果找到了 MX 记录，则搜索将停止，并且验证电子邮件将发送到子域的常用管理地址。例如，如果找到了 `test.example.com` 的 MX 记录，则电子邮件将发送到 admin@test.example.com、administrator@test.example.com 以及电子邮件验证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。

您可以不使用控制台，而是使用 RequestCertificateAPI 中的`ValidationDomain`选项或 request-certificat AWS CLI e 命令来指定 ACM 向其发送验证电子邮件的域名。如果您使用 API 或 AWS CLI，则 AWS 不执行 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。这个问题是外部的 AWS。也就是说， AWS 不会控制 WHOIS 服务器，也无法阻止 WHOIS 服务器限制。如果您遇到此问题，请在 AWS Support 中心创建一个案例以寻求解决方法。

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。如果禁用隐私保护，注册者可能会在自己的 WHOIS 输出中显示此信息，但这种做法因不同的注册者而异。如果 WHOIS 中的注册者不提供，则 ACM 无法向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。

我无法切换到 DNS 验证

在创建采用电子邮件验证的证书后，您无法切换到使用 DNS 对其进行验证。

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：

- 您的证书必须与与 ACM 集成的 AWS 服务相关联。有关 ACM 支持的资源的信息，请参阅 与之集成的服务 AWS Certificate Manager。
- 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 中所述。

处理托管证书续订失败

当证书即将到期时（DNS 为 60 天，电子邮件为 45 天，私有证书为 60 天），如果证书符合资格标准，ACM 会尝试续订证书。您可能需要执行相关操作才能成功续订。有关更多信息，请参阅 ACM 证书的托管续订。

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。

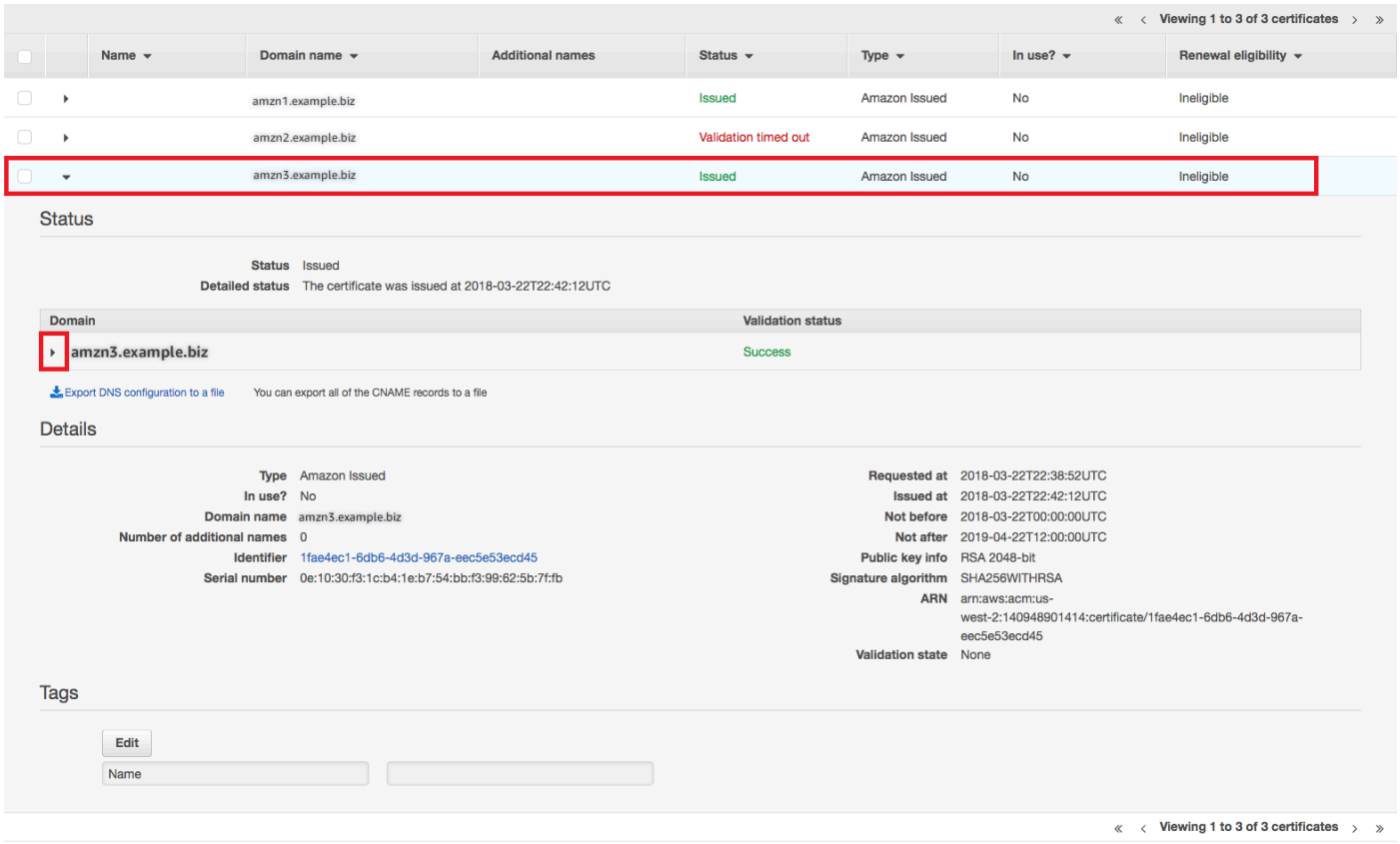
请参阅使用电子邮件验证，了解有关识别哪些域处于 PENDING_VALIDATION 状态并重复这些域的验证过程的说明。

针对 DNS 验证证书的托管证书续订

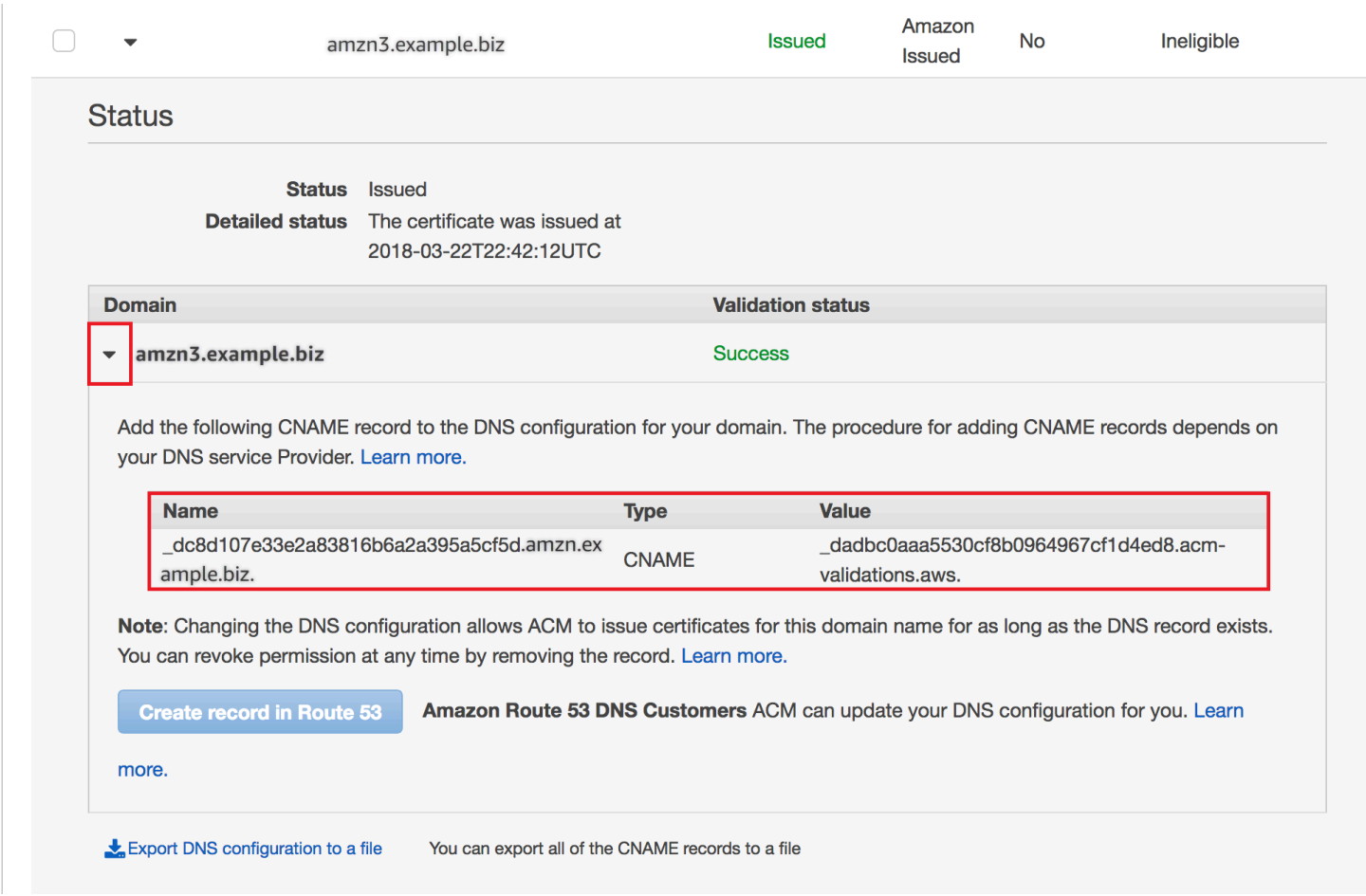
ACM 不会尝试对 DNS 验证的证书进行 TLS 验证。如果 ACM 无法续订您通过 DNS 验证来验证的证书，则很可能是由于 DNS 配置中缺少别名记录或别名记录不准确。如果发生这种情况，ACM 会通知您无法自动续订证书。

> **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 ACM CLI 中的 desc ribe- certificate 命令来检索 CNAME 记录。有关更多信息，请参阅 DNS 验证。



从控制台中选择目标证书。



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 AWS 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

- 证书导入问题
- 证书固定问题
- API Gateway 问题
- 工作证书意外失败时该怎么办
- ACM 服务关联角色 (SLR) 问题

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。

> **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该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《AWS Certificate Manager 用户指南》和每项服务的文档，确保您的证书能够正常使用。
- 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。但是，默认情况下，如果您在未指定`keyTypes`筛选条件的情况下调用 ListCertificatesAPI 或 list-cer tificates AWS CLI 命令，则只会`RSA_1024`显示或`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（有时称为 SSL 固定）来固定 ACM 证书，则在 AWS 续订证书后，应用程序可能无法连接到您的域。为此，我们建议您不要固定 ACM 证书。如果您的应用程序必须固定证书，您可以执行以下操作：

- 将您自己的证书导入到 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 端点时，API Gateway 会为您设置 CloudFront 分配。该 CloudFront 分配归 API Gateway 所有，而不是归您的账户所有。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) 需要解密权限 AWS KMS key ，然后才能解密证书的私钥。此权限由基于资源的策略授予，当您将证书与 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。

要调查问题，请使用 AWS KMS 控制台查看您的授权状态，网址为https://console.aws.amazon.com/kms。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。
- 如果您认为授予 ACM 的权限已被撤销，请通过 AWS Support https://console.aws.amazon.com/support/home#/ 联系。

ACM 服务关联角色 (SLR) 问题

当您颁发由私有 CA 签名并由其他账户与您共享的证书时，ACM 会在首次使用时尝试设置服务相关角色 (SLR)，以委托人身份与基于 AWS 私有 CA 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

AWS Certificate Manager 命令可能由于多种原因而失败。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 AWS 私有 CA颁发的私有 PKI 证书时，可能会出现以下异常。

> **Note**
>
> AWS 私有 CA 不支持中国（北京）区域和中国（宁夏）区域。

ACM 故障代码	注释
PCA_ACCESS_DENIED	私有 CA 未授予 ACM 权限。这会触发 AWS 私有 CA AccessDeniedException 失败代码。 要解决此问题，请使用该 AWS 私有 CA CreatePermission操作向 ACM 服务主体授予必要的权限。
PCA_INVALID_DURATION	请求的证书的有效期超过了颁发的私有 CA 的有效期。这会触发 AWS 私有 CA ValidationException 失败代码。 要解决此问题，请安装新 CA 证书（具有适当的有效期）。
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。这会触发 AWS 私有 CA InvalidStateException 失败代码。 按如下方式解决此问题： • 如果 CA 的状态为 CREATING，请等待创建操作完成，然后安装 CA 证书。 • 如果 CA 的状态为 PENDING_CERTIFICATE ，请安装 CA 证书。 • 如果 CA 的状态为 DISABLED，请将其更新为 ACTIVE 状态。 • 如果 CA 的状态为 DELETED，请将其还原。 • 如果 CA 的状态为 EXPIRED，请安装新证书

ACM 故障代码	注释
	• 如果 CA 的状态为 FAILED，而您无法解决问题，请联系 AWS Support。
PCA_LIMIT_EXCEEDED	私有 CA 已达到颁发配额。这会触发 AWS 私有 CA LimitExceededException 失败代码。在继续使用此帮助之前，请尝试重复您的请求。 如果错误仍然存在，请联系 AWS Support 以请求提高配额。
PCA_REQUEST_FAILED	网络或系统出错。这会触发 AWS 私有 CA RequestFailedException 失败代码。在继续使用此帮助之前，请尝试重复您的请求。 如果错误仍然存在，请联系 AWS Support。
PCA_RESOURCE_NOT_FOUND	私有 CA 已被永久删除。这会触发 AWS 私有 CA ResourceNotFoundException 失败代码。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。 要解决此问题，请创建新 CA。
SLR_NOT_FOUND	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 为 ACM 创建 SLR。

概念

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

主题

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)

可分辨名称	加密算法
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。

非对称密钥加密

非对称加密不同于对称密钥加密，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 AWS KMS key 加密私有密钥。有关更多信息，请参阅 证书私有密钥的安全性。

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。

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。

域名

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

TLD

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

顶级域

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

子域

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

超级域名

子域所属的域。

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，$[subdomain_1].[subdomain_2]...[subdomain_n].[apex\ domain].[top-level\ domain]$ 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 $[subdomain_1.subdomain_2.]$ 这样的名称就是 PQDN，这是因为无法确定根域。

注册

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

加密和解密

加密是提供数据机密性的过程。解密将反转此过程并恢复原始数据。未加密的数据通常称为“明文”，无论它是否为文本。加密的数据通常称为“密文”。客户端与服务器之间的消息的 HTTPS 加密使用算法和密钥。算法定义了将纯文本数据转换为密文（加密）和将密文转换回原始纯文本（解密）的 step-by-step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。

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。

传输层安全性协议（TLS）

请参阅安全套接字层 (SSL)。

信任

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

文档历史记录

下表描述了 2018 年 AWS Certificate Manager 开始的文档发布历史。

变更	说明	日期
弃用邮件交换器 (MX) 电子邮件验证	ACM 不再支持邮件交换器 (MX)。而是使用 DNS 验证或指定超级域名来接收电子邮件验证。	2024 年 6 月 27 日
添加有关账户级别分离的最佳实践	尽可能在保单中使用账户级别的分离。如果不可能，您可以在账户级别或通过策略中的加密上下文条件密钥来限制权限。	2024 年 6 月 11 日
即将弃用 WHOIS 电子邮件验证	添加了关于从 2024 年 6 月起停用 WHOIS 电子邮件验证的说明。	2024年2月5日
添加了条件密钥支持	在请求 ACM 证书时添加了对 IAM 条件密钥的支持。有关受支持的条件列表，请参阅 https://docs.aws.amazon.com/acm/latest/userguide/acm-conditions.html#acm-conditions-supported。	2023 年 8 月 24 日
添加了 ECDSA 支持	在请求公有 ACM 证书时添加了对椭圆曲线数字签名算法 (ECDSA) 的支持。有关支持的密钥算法的列表，请参阅 https://docs.aws.amazon.com/acm/latest/userguide/acm-certificate.html#algorithms。	2022 年 11 月 8 日

新 CloudWatch 活动	添加了“ACM 证书已过期”、“ACM 证书可用”以及“ACM 证书需要续订操作”事件。有关支持 CloudWatch 的事件列表，请参阅https://docs.aws.amazon.com/acm/latest/userguide/cloudwatch-events.html。	2022 年 10 月 27 日
更新用于导入的密钥算法类型	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 https://docs.aws.amazon.com/acm/latest/userguide/import-certificate-prerequisites.html。	2021 年 7 月 14 日
将“监控和日志”提升为单独一章	将监控和日志文档移至其自己的章节。此更改涵盖 CloudWatch指标、 CloudWatch 事件/ 和EventBridge。CloudTrail有关更多信息，请参阅 https://docs.aws.amazon.com/acm/latest/userguide/monitoring-and-logging.html。	2021 年 3 月 23 日
添加了 CloudWatch 指标和事件支持	添加了 DaysToExpiry 指标和事件以及支持 API。有关更多信息，请参阅 https://docs.aws.amazon.com/acm/latest/userguide/cloudwatch-metrics.html 和 https://docs.aws.amazon.com/acm/latest/userguide/cloudwatch-events.html。	2021 年 3 月 3 日

增加了跨账户支持	为使用来自 AWS 私有 CA的私有 CA 添加了跨账户支持。有关更多信息，请参阅 https://docs.aws.amazon.com/acm/latest/userguide/ca-access.html。	2020 年 8 月 17 日
增加了区域支持	增加了对 AWS 中国（北京和宁夏）区域的区域支持。有关支持区域的完整列表，请参阅 https://docs.aws.amazon.com/general/latest/gr/rande.html#acm-pca_region。	2020 年 3 月 4 日
增加了续订工作流测试	客户现在可以手动测试 ACM 托管续订工作流的配置。有关更多信息，请参阅测试 ACM 的托管续订配置。	2019 年 3 月 14 日
现在默认启用证书透明度日志	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 24 月 4 日
正在启动 AWS 私有 CA	推出 ACM Private Certificate Manager (CM) AWS Certificate Manager ，其扩展允许用户建立安全的托管基础架构，用于颁发和撤销私有数字证书。有关更多信息，请参阅 AWS Private Certificate Authority。	2018 年 4 月 4 日
证书透明度日志	在最佳实践中增加了证书透明度日志记录。	2018 年 3 月 27 日

下表描述了 2018 年 AWS Certificate Manager 之前的文档发布历史。

更改	描述	发行日期
新增内容	在DNS 验证中添加了 DNS 验证相关内容。	2017 年 11 月 21 日
新增内容	向 使用 API（Java 示例） 添加了新的 Java 代码示例。	2017 年 10 月 12 日
新增内容	向 （可选）配置 CAA 记录 添加了有关 CAA 记录的信息。	2017 年 9 月 21 日
新增内容	已将有关 .IO 域的信息添加到故障排除中。	2017 年 07 月 7 日
新增内容	已将有关重新导入证书的信息添加到重新导入证书 中。	2017 年 07 月 7 日
新增内容	已将有关证书固定的信息添加到最佳实践和故障排除中。	2017 年 07 月 7 日
新增内容	已添加 AWS CloudFormation 到与之集成的服务 AWS Certificate Manager。	2017 年 5 月 27 日
更新	已将更多信息添加到配额。	2017 年 5 月 27 日
新增内容	添加了有关适用于 Identity and Access Managem AWS Certificate Manager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 电子邮件验证。	2017 年 4 月 21 日
更新	添加了有关为您的域设置电子邮件的信息。请参阅 （可选）为域配置电子邮件。	2017 年 4 月 6 日

更改	描述	发行日期
更新	添加了有关在控制台中检查证书续订状态的信息。请参阅 检查证书的续订状态。	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 AWS Elastic Beanstalk 和 Amazon API Gateway 的支持。请参阅 与之集成的服务 AWS Certificate Manager。	2017 年 3 月 21 日
更新	更新了有关托管续订的文档。	2017 年 2 月 20 日
新增内容	添加了有关导入证书的文档。	2016 年 10 月 13 日
新增内容	增加了对 ACM 操作的 AWS CloudTrail 支持。请参阅 CloudTrail 与一起使用 AWS Certificate Manager。	2016 年 3 月 25 日
新指南	此版本引入了 AWS Certificate Manager。	2016 年 1 月 21 日

本文属于机器翻译版本。若本译文内容与英语原文存在差异，则一律以英文原文为准。