AWS Certificate Manager

User Guide

Version 1.0

AWS Certificate Manager: User Guide

Table of Contents

What is AWS Certificate Manager? ... 1
- Concepts ... 2
- ACM Certificate Characteristics ... 3
- Managed Renewal ... 4
- Supported Regions ... 5
- Supported Services ... 5
- Limits ... 6
  - Number of Domain Names per ACM Certificate ... 6
- Pricing ... 7

Setting Up ... 8
- Set Up AWS and IAM ... 8
  - Sign Up for AWS ... 8
  - Create an IAM User ... 9
- Register a Domain Name ... 9
- Configure Email ... 9
- Set Up Your Website or Application ... 10
  - Linux Quickstart ... 10
  - Windows Quickstart ... 10

Getting Started ... 11
- Request and Manage Certificates ... 11
  - Request a Certificate ... 11
  - Validate Domain Ownership ... 12
  - Manage ACM Certificates ... 17
- Elastic Load Balancing ... 20
- Amazon CloudFront ... 22

Importing Certificates ... 23
- Prerequisites for Importing Certificates ... 23
- Importing Certificates (AWS Management Console) ... 24
- Importing Certificates (ACM API) ... 24
- Troubleshooting ... 25

Tagging ACM Certificates ... 27
- Tag Restrictions ... 27
- Managing Tags ... 28
  - Managing Tags (Console) ... 28
  - Managing Tags (AWS Command Line Interface) ... 29
  - Managing Tags (AWS Certificate Manager API) ... 29

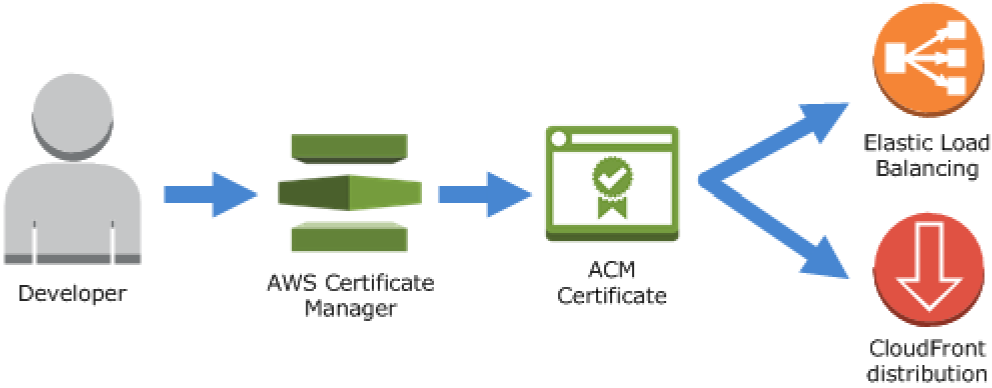
Permissions and Policies ... 30
- Managed Policies ... 30
  - AWSCertificateManagerReadOnly ... 30
  - AWSCertificateManagerFullAccess ... 31
  - Attaching a Managed Policy Using the AWS Management Console ... 31
- Inline Policies ... 31
  - Listing Certificates ... 32
  - Retrieving a Certificate ... 32
  - Deleting a Certificate ... 32
  - Read-only Access to ACM ... 33
  - Full Access to ACM ... 33
  - Administrator Access to All AWS Resources ... 33
- Integrated Resource Policies ... 34
  - Example IAM Policies for Elastic Load Balancing ... 34
  - Example IAM Policies for Amazon CloudFront ... 34

Using AWS CloudTrail ... 35
- Logging ACM API Calls ... 35
  - Adding Tags ... 36
  - Deleting a Certificate ... 37

Describing a Certificate ... 38
Retrieving a Certificate ... 38
Import a Certificate ... 39
Listing Certificates ... 41
Listing Tags ... 42
Removing Tags ... 42
Requesting a Certificate ... 43
Resending Validation Email ... 44
Logging ACM-Related API Calls ... 44
Creating a Load Balancer ... 45
Registering Amazon EC2 ... 45
Encrypting a Private Key ... 46
Decrypting a Private Key ... 47
Using the ACM API ... 49
DeleteCertificate ... 49
DescribeCertificate ... 50
GetCertificate ... 51
Listing ACM Certificates ... 53
RequestCertificate ... 54
ResendValidationEmail ... 55
ACM Private Key Security ... 57
Troubleshooting ... 58
Not Receiving Validation Email ... 58
Email Sent to Subdomain ... 59
Certificate Request Timed Out ... 59
Certificate Request Failed ... 59
No Available Contacts ... 60
Domain Not Allowed ... 60
Additional Verification Required ... 60
Invalid Public Domain ... 61
Other ... 61
Validation Not Complete ... 61
Document History ... 62

What is AWS Certificate Manager?

AWS Certificate Manager (ACM) handles the complexity of provisioning, deploying, and managing SSL/TLS certificates for your AWS-based websites and applications. You can use certificates provided by ACM (p. 11) (ACM Certificates) or certificates that you import into ACM (p. 23). You use ACM to request, import, and manage the certificate and then use other AWS services to associate the certificate with your website or application. As shown in the following illustration, you can use certificates stored in ACM with Elastic Load Balancing and Amazon CloudFront. You cannot use Amazon-issued certificates outside of AWS.



For more information about the services integrated with ACM, see the following topics:

- Getting Started with Elastic Load Balancing (p. 20)
- Getting Started with Amazon CloudFront (p. 22)

For general information about ACM, see the following topics.

Topics

- Concepts (p. 2)
- ACM Certificate Characteristics (p. 3)
- Managed Renewal (p. 4)

- Supported Regions (p. 5)
- Supported Services (p. 5)
- Limits (p. 6)
- Pricing (p. 7)

Concepts

This section introduces basic terms and concepts related to AWS Certificate Manager (ACM).

Certificate Authority

A certificate authority (CA) is an entity that issues digital certificates. Commercially, the most common type of digital certificate is based on the ISO X.509 standard. The CA issues signed digital certificates that affirm the identity of the certificate subject and bind that identity to the public key contained in the certificate. The CA also typically manages certificate revocation.

Domain Name System

The Domain Name System (DNS) is a hierarchical distributed naming system for computers and other resources connected to the Internet or a private network. DNS translates textual domain names, such as http://aws.amazon.com, into numerical IP (Internet Protocol) addresses.

Domain Name

A fully qualified domain name (FQDN) is the complete, human-readable name for a computer or other resource connected to the Internet. For example aws.amazon.com is the FQDN for Amazon Web Services. An FQDN is made up of various parts. In the preceding example, *aws* is the name of the host located in the domain amazon.com, and *.com* is called the top-level domain. The .com suffix is generally used to represent commercial activity. There are many different top-level domains including .net and .edu.

Encryption and Decryption

Encryption is the process of providing data confidentiality. Decryption reverses the process and recovers the original data. Unencrypted data is typically called plaintext whether it is text or not. Encrypted data is typically called ciphertext. HTTPS encryption of messages between clients and servers uses algorithms and keys. Algorithms define the step-by-step procedure by which plaintext data is converted into ciphertext (encryption) and ciphertext is converted back into the original plaintext (decryption). Keys are used by algorithms during the encryption or decryption process. Keys can be either private or public.

Fully Qualified Domain Name (FQDN)

See Domain Name (p. 2).

Public Key Infrastructure

A public key infrastructure (PKI) consists of hardware, software, people, policies, documents, and procedures needed to create, issue, manage, distribute, use, store, and revoke digital certificates. PKI facilitates the secure transfer of information across computer networks.

Root Certificate

A certificate authority typically exists within a hierarchical structure that contains multiple CAs with clearly defined parent-child relationships between them. Child subordinate CAs are certified by their parent CAs, creating a certificate chain. The CA at the top of the hierarchy is referred to as the root CA, and its certificate is called the root certificate. This certificate is typically self-signed.

Secure Sockets Layer (SSL)

Secure Sockets Layer (SSL) and Transport Layer Security (TLS) are cryptographic protocols that provide communication security over a computer network. TLS is the successor of SSL. They both use X.509 certificates to authenticate the server, and both protocols negotiate a symmetric key between the client and the server that is used to encrypt data flowing between the two entities.

Secure HTTPS

HTTPS stands for HTTP over SSL/TLS, a secure form of HTTP that is supported by all major browsers and servers. All HTTP requests and responses are encrypted before being sent across a network. HTTPS combines the HTTP protocol with symmetric, asymmetric, and X.509 certificate-based cryptographic techniques. HTTPS works by inserting a cryptographic security layer below the HTTP application layer and above the TCP transport layer in the Open Systems Interconnection (OSI) model. The security layer uses the Secure Sockets Layer (SSL) protocol or the Transport Layer Security (TLS) protocol.

SSL Server Certificates

HTTPS transactions require server certificates to authenticate a server. A server certificate is an X.509 v3 data structure that binds the public key in the certificate to the subject of the certificate. An SSL/TLS certificate is signed by a certificate authority (CA) and contains the name of the server, the validity period, the public key, the signature algorithm, and more.

Symmetric Key Cryptography

Symmetric key cryptography uses the same key to both encrypt and decrypt digital data.

Transport Layer Security (TLS)

See Secure Sockets Layer (SSL) (p. 2).

Trust

In order for a web browser to trust the identity of a website, the browser must be able to verify the website's certificate. Browsers, however, trust only a small number of certificates known as CA root certificates. A trusted third party, known as a certificate authority (CA), validates the identity of the website and issues a signed digital certificate to the website's operator. The browser can then check the digital signature to validate the identity of the website. If validation is successful, the browser displays a lock icon in the address bar.

ACM Certificate Characteristics

Certificates provided by ACM have the characteristics described in this section.

> **Note**
>
> These characteristics apply only to certificates provided by ACM. They might not apply to certificates that you import into ACM (p. 23).

Domain Validation (DV)

ACM Certificates are domain validated. That is, the subject field of an ACM Certificate identifies a domain name and nothing more. Email is sent to the registered owner for each domain name in the request. The domain owner or an authorized representative can approve the certificate request by following the instructions in the email. For more information, see Validate Domain Ownership (p. 12).

Validity Period

The validity period for ACM Certificates is currently 13 months.

Managed Renewal and Deployment

ACM manages the process of renewing ACM Certificates and provisioning the certificates after they are renewed. Automatic renewal can help you avoid downtime due to misconfigured, revoked, or expired certificates. For more information, see Managed Renewal (p. 4).

Browser and Application Trust

ACM Certificates are trusted by all major browsers including Google Chrome, Microsoft Internet Explorer and Microsoft Edge, Mozilla Firefox, and Apple Safari. Browsers that trust ACM Certificates display a lock icon in their status bar or address bar when connected by SSL/TLS to sites that use ACM Certificates. ACM Certificates are also trusted by Java.

Multiple Domain Names

Each ACM Certificate must include at least one fully qualified domain name (FQDN), and you can add additional names if you want. For example, when you are creating an ACM Certificate for `www.example.com`, you can also add the name `www.example.net` if customers can reach your site by using either name. This is also true of bare domains (also known as the zone apex or naked domains). That is, you can request an ACM Certificate for www.example.com and add the name example.com. For more information, see Request a Certificate (p. 11).

Wildcard Names

ACM allows you to use an asterisk (*) in the domain name to create an ACM Certificate containing a wildcard name that can protect several sites in the same domain. For example, `*.example.com` protects `www.example.com` and `images.example.com`.

> **Note**
>
> When you request a wildcard certificate, the asterisk (*****) must be in the leftmost position of the domain name and can protect only one subdomain level. For example, **`*.example.com`** can protect **`login.example.com`** and **`test.example.com`**, but it cannot protect **`test.login.example.com`**. Also note that **`*.example.com`** protects *only* the subdomains of **`example.com`**, it does not protect the bare or apex domain (**`example.com`**). However, you can request a certificate that protects a bare or apex domain and its subdomains by specifying multiple domain names in your request. For example, you can request a certificate that protects **`example.com`** and **`*.example.com`**.

Algorithms

Currently, ACM supports the RSA-2048 encryption and SHA-256 hashing algorithms.

Exceptions

Note the following:

- ACM does not provide extended validation (EV) certificates or organization validation (OV) certificates.
- ACM does not provide certificates for anything other than the SSL/TLS protocols.
- You cannot use ACM Certificates for code signing or email encryption.
- ACM allows only UTF-8 encoded ASCII for domain names, including labels that contain "xn--" (Punycode). ACM does not accept Unicode input (u-labels) for domain names.
- ACM does not currently permit you to opt out of managed certificate renewal (p. 4) for certificates provided by ACM. Managed renewal is not available for certificates that you import into ACM.
- You cannot request certificates for Amazon-owned domain names such as those ending in amazonaws.com, cloudfront.net, or elasticbeanstalk.com.
- You cannot download the private key for an ACM Certificate.
- You cannot associate ACM Certificates with Amazon Elastic Compute Cloud (Amazon EC2) instances.

Managed Renewal

By default, ACM automatically renews certificates provided by ACM that are used with other AWS services, such as Elastic Load Balancing and CloudFront. Managed renewal makes configuring and maintaining SSL/TLS for a secure website or application easier and less error prone than manual renewal processes. Managed renewal can help you avoid downtime due to misconfigured, revoked, or expired certificates. Further, managed renewal doesn't require you to install or maintain a software client or agent on your website. Instead, because ACM is integrated with other AWS services, you can centrally manage and deploy ACM Certificates on the AWS platform from the console, AWS CLI, or API of the integrated service. For a list of supported services, see Services Integrated with AWS Certificate Manager (p. 5).

> **Note**
>
> Managed renewal applies only to certificates provided by ACM. ACM does not attempt to renew certificates that you import into ACM (p. 23).

ACM attempts to perform automatic renewals on all ACM Certificates before they expire. If ACM is unable to do so, it falls back on alternate renewal methods such as sending validation email to domain registrants. Certificates that can be renewed automatically include those that are being used by AWS resources on a publicly accessible site. This includes certificates for bare domains such as `example.com`.

The following conditions must be met before an ACM Certificate can be renewed:

- DNS must be configured to resolve all of the fully qualified domain names (FQDNs) included in the certificate to the AWS resource with which the certificate is associated. ACM checks that each FQDN in the certificate maps to an Amazon-controlled public IP address. ACM does not check DNS resolution for wildcard domain names such as `*.example.com`.
- The AWS resource must be configured so that AWS can make an SSL/TLS connection to it from the internet.

If an ACM Certificate cannot be automatically renewed, ACM sends email validation requests to the domain owner or to an authorized representative. The email contains instructions about how to renew the ACM Certificate.

ACM does not attempt to renew certificates that are not in use. To be considered in use, an ACM Certificate must be associated with an AWS service such as Elastic Load Balancing or CloudFront.

If an ACM Certificate is in use but cannot be publicly accessed by using the DNS name(s) in the certificate, ACM attempts to renew the certificate through email validation. If email validation fails, ACM notifies you by creating a support ticket that sends email to the address registered with your AWS account.

ACM begins the renewal process up to 60 days prior to the certificate’s expiration date. The validity period for certificates provided by ACM is currently 13 months.

ACM generates a new key pair when renewing the ACM Certificate. AWS will issue a new certificate with a new key pair without changing other certificate fields.

Supported Regions

Visit AWS Regions and Endpoints in the *AWS General Reference* or the AWS Region Table to see the regional availability for ACM.

Like most AWS resources, certificates in ACM are regional resources. To use a certificate with Elastic Load Balancing for the same fully qualified domain name (FQDN) or set of FQDNs in more than one AWS region, you must request or import a certificate for each region. For certificates provided by ACM, this means you must revalidate each domain name in the certificate for each region. You cannot copy a certificate between regions.

To use an ACM Certificate with Amazon CloudFront, you must request or import the certificate in the US East (N. Virginia) region. ACM Certificates in this region that are associated with a CloudFront distribution are distributed to all the geographic locations configured for that distribution.

Services Integrated with AWS Certificate Manager

Elastic Load Balancing

Elastic Load Balancing automatically distributes your incoming application traffic across multiple Amazon EC2 instances. It detects unhealthy instances and reroutes traffic to healthy instances until the unhealthy instances have been restored. Elastic Load Balancing automatically scales its request handling capacity in response to incoming traffic. For more information about load balancing, see the Elastic Load Balancing User Guide. In general, to serve secure content over SSL/TLS, load balancers require that SSL/TLS certificates be installed on either the load balancer or the back-end Amazon EC2 instance. ACM integrates with Elastic Load Balancing to deploy ACM Certificates on the load balancer. For a quick start on how to use Elastic Load Balancing, see Getting Started with Elastic Load Balancing (p. 20).

Amazon CloudFront

Amazon CloudFront is a web service that speeds up distribution of your dynamic and static web content to end users by delivering your content from a worldwide network of edge locations. When an end user requests content that you're serving by using CloudFront, the user is routed to the edge location that provides the lowest latency so that content is delivered with the best possible performance. If the content is currently at that edge location, CloudFront delivers it immediately. If the content is not currently at that edge location, CloudFront retrieves it from the Amazon S3 bucket or web server that you have identified as the definitive content source. For more information about CloudFront, see the Amazon CloudFront Developer Guide. To serve secure content over SSL/TLS, CloudFront requires that SSL/TLS certificates be installed on either the CloudFront distribution or on the back-end content source. ACM integrates with CloudFront to deploy ACM Certificates on the CloudFront distribution. For a quick start on how to use CloudFront, see Getting Started with Amazon CloudFront (p. 22).

Note

To use an ACM Certificate with CloudFront, you must request or import the certificate in the US East (N. Virginia) region.

Limits

The following AWS Certificate Manager (ACM) limits apply to each AWS region and each AWS account. To request higher limits, create a case at the AWS Support Center. New AWS accounts might start with limits that are lower than the limits described here.

Item	Default Limit
Number of ACM-provided certificates	100
Number of imported certificates	100
Number of domain names per ACM-provided certificate	10. See the information following this table.

Note

The limit for the number of domain names per ACM Certificate applies only to certificates provided by ACM. This limit does not apply to certificates that you import into ACM (p. 23). The following section applies only to certificates provided by ACM.

Number of Domain Names per ACM Certificate

The default limit is 10 domain names per ACM Certificate. To request a higher limit, read the following information and then create a case at the AWS Support Center.

You cannot add or remove domain names from an existing ACM Certificate. Instead, you must request a new certificate with the revised list of domain names. When you do, you must validate ownership (p. 12) of all the domain names in the request, including the domain names that you previously validated for the original certificate. For each domain name in the request, you receive up to 8 validation emails, at least 1 of which must be acted upon within 72 hours.

For example, when you request a certificate with 5 domain names, you receive up to 40 validation emails, at least 5 of which must be acted upon within 72 hours. To add 4 more domain names to the certificate, you must request a new certificate with all 9 domain names, resulting in up to 72 emails, at least 9 of which must be acted upon within 72 hours. As the number of domain names in the certificate request increases, so does the work required to validate ownership of the domain names each time you want to make a change to the domain names in the certificate.

Before requesting an increase to the number of domain names allowed per ACM Certificate, consider the management overhead required to validate ownership of the domain names each time you want to make a change to the domain names in the certificate. You might find it easier to request a high number of certificates with only one or two domain names in each rather than request a low number of certificates with many domain names in each.

Pricing for AWS Certificate Manager

You are not charged by AWS for the SSL/TLS certificates that you manage with AWS Certificate Manager. You pay only for the AWS resources that you create to run your website or application. For the latest ACM pricing information, see the AWS Certificate Manager Service Pricing page on the AWS website.

Setting Up

With AWS Certificate Manager (ACM) you can provision, manage, and deploy SSL/TLS certificates for your AWS-based websites and applications. You use ACM to provision and manage the certificate and then use other AWS services to deploy the certificate for your website or application. For more information about the services integrated with ACM, see Services Integrated with AWS Certificate Manager (p. 5). The following topics discuss the steps you need to perform before using ACM.

> **Note**
> In addition to using certificates provided by ACM, you can also import certificates into ACM. For more information, see Importing Certificates (p. 23).

Topics

- Set Up AWS and IAM (p. 8)
- Register a Domain Name (p. 9)
- Configure Email for Your Domain (p. 9)
- Set Up Your Website or Application (p. 10)

Set Up AWS and IAM

Before you can use ACM, you must sign up for Amazon Web Services. You can optionally create an IAM user to limit the actions your users can perform.

Sign Up for AWS

If you are not already an Amazon Web Services (AWS) customer, you must sign up to be able to use ACM. Your account is automatically signed up for all available services, but you are charged for only the services that you use. Also, if you are a new AWS customer, you can get started for free. For more information, see AWS Free Tier.

To sign up for an AWS account

1. Go to https://aws.amazon.com/ and choose **Sign Up**.
2. Follow the on-screen instructions.

> **Note**
> Part of the sign-up procedure includes receiving an automated telephone call and entering the supplied PIN on the telephone keypad. You must also supply a credit card number even if you are signing up for the free tier.

Create an IAM User

Amazon Web Services require that you provide credentials when you access them so that each service can determine whether you have permission to use its resources. The console requires your user name and password, and you can create access keys to use the command line interface or API. But because your root credentials permit unlimited access, we don't recommend that you use them with AWS. Instead, we recommend that you use AWS Identity and Access Management (IAM) to create users with more limited permissions. That is, create an IAM user and add this user to an IAM group that has a specific set of permissions. You can then access AWS using a special URL and the user's credentials.

For example, to create a group for administrator's and add users to this group, see Creating an Administrator's Group in the AWS Identity and Access Management User Guide. The user can then sign into the account with a special URL. For more information, see How Users Sign In to Your Account.

See Permissions and Policies (p. 30) for examples of IAM policies that can be used with ACM.

Register a Domain Name

A fully qualified domain name (FQDN) is the unique name of an organization or individual on the Internet followed by a top-level domain extension such as .com, or .edu. If you do not already have a registered domain name, you can register one through Amazon Route 53 or dozens of other commercial registrars. Typically you go to the registrar's website and request a domain name. The registrar queries WHOIS to determine whether the requested FQDN is available. If it is, the registrar usually lists related names that differ by domain extension and provides you an opportunity to acquire any of the available names. Registration usually lasts for a set period of time such as one or two years before it must be renewed.

For more information about Amazon Route 53, see Registering Domain Names Using Amazon Route 53.

Configure Email for Your Domain

Once you have a registered domain name, use your registrar's website to associate an email address with it. Domain name registrars usually require that you list contact information for your domain. The contact information is entered with WHOIS. ACM uses the contact information to identify who to contact when validating domain ownership. In the process of validating your identity, an email is sent to the following three registered contact addresses in WHOIS:

- Domain registrant
- Technical contact
- Administrative contact

> **Note**
> Some registrars allow you to hide your contact information in your WHOIS listing, and others allow you to substitute your real email address with a privacy (or proxy) address. To prevent problems with receiving the domain validation email from ACM, ensure your contact information is visible in WHOIS. If your WHOIS listing shows a privacy email address, ensure that email sent to the privacy address is forwarded to your real email address, or list your real email address instead.

Email is also sent to the following five common system administration addresses:

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

The validation email contains instructions for confirming that the domain owner or an appointed representative approves of the ACM Certificate. For more information about validation, see Validate Domain Ownership (p. 12).

Set Up Your Website or Application

You can install your website on an Amazon EC2 Linux or Windows instance. For more information about Linux Amazon EC2 instances, see Amazon Elastic Compute Cloud User Guide for Linux. For more information about Windows Amazon EC2 instances, see Amazon Elastic Compute Cloud User Guide for Microsoft Windows.

> **Note**
> Currently, ACM Certificates are associated with Elastic Load Balancing load balancers or Amazon CloudFront distributions. Although you install your website on an Amazon EC2 instance, you do not deploy an ACM Certificate there. Instead, deploy the ACM Certificate on your Elastic Load Balancing load balancer or on your CloudFront distribution.

To get your website up and running quickly on either Windows or Linux, see the following topics.

Topics

- Linux Quickstart (p. 10)
- Windows Quickstart (p. 10)

Linux Quickstart

To create your website or application on a Linux instance, you can choose a Linux Amazon Machine Image (AMI) and install an Apache web server on it. For more information, see Tutorial: Installing a LAMP Web Server on Amazon Linux in the *Amazon EC2 User Guide for Linux Instances*.

Windows Quickstart

To acquire a Microsoft Windows server on which you can install your website or application, choose a Windows Server AMI that comes bundled with a Microsoft Internet Information Services (IIS) web server. Then use the default website or create a new one.

Getting Started

You can use the console, the AWS Command Line Interface (AWS CLI), or the SDK to get started with AWS Certificate Manager and services integrated with ACM. For more information, see the following topics.

Topics

- Requesting and Managing ACM Certificates (p. 11)
- Getting Started with Elastic Load Balancing (p. 20)
- Getting Started with Amazon CloudFront (p. 22)

Requesting and Managing ACM Certificates

The following topics help you get started with AWS Certificate Manager by using the console or the AWS CLI.

Topics

- Request a Certificate (p. 11)
- Validate Domain Ownership (p. 12)
- Manage ACM Certificates (p. 17)

Request a Certificate

The following sections discuss how to use the ACM console or AWS CLI to request an ACM Certificate.

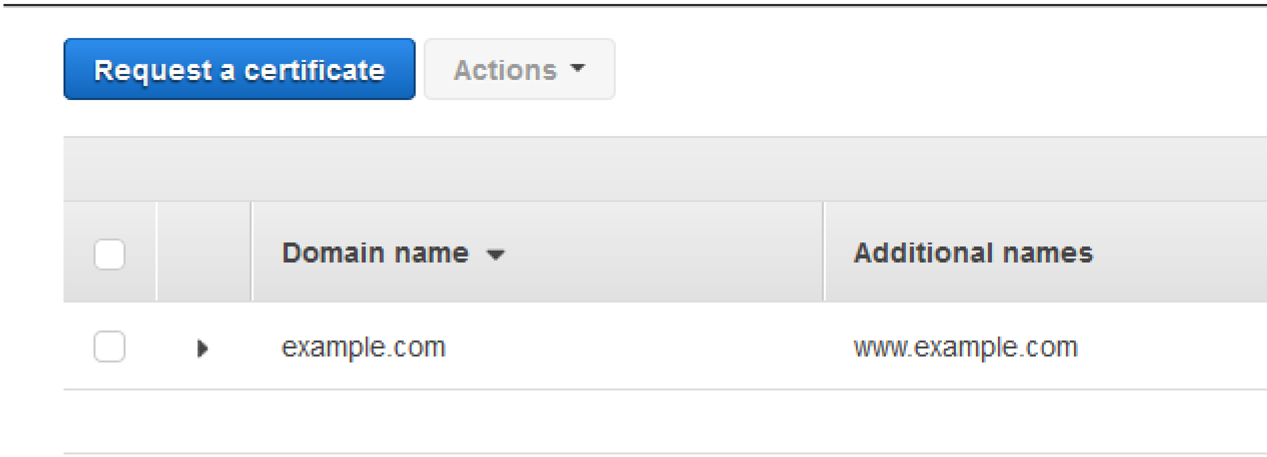
To request an ACM Certificate (console)

1. Sign into the AWS Management Console and open the ACM console at https://console.aws.amazon.com/acm/home. If the introductory page appears, choose **Get Started**. Otherwise, choose **Request a certificate**.
2. On the **Request a certificate** page, type your domain name. You can use a fully qualified domain name (FQDN) such as **`www.example.com`** or a bare or apex domain name such

as **`example.com`**. You can also use an asterisk (**`*`**) as a wildcard in the leftmost position to protect several site names in the same domain. For example, **`*.example.com`** protects **`corp.example.com`**, and **`images.example.com`**. The wildcard name will appear in the **Subject** field and the **Subject Alternative Names** extension of the ACM Certificate.

> **Note**
> When you request a wildcard certificate, the asterisk (**`*`**) must be in the leftmost position of the domain name and can protect only one subdomain level. For example, **`*.example.com`** can protect **`login.example.com`**, and **`test.example.com`**, but it cannot protect **`test.login.example.com`**. Also note that **`*.example.com`** protects *only* the subdomains of **`example.com`**, it does not protect the bare or apex domain (**`example.com`**). To protect both, see the next step.

3. To add more domain names to the ACM Certificate, choose **Add more names** and type another domain name in the text box that opens. This is useful for protecting both a bare or apex domain (like **`example.com`**) and its subdomains (**`*.example.com`**).
4. After you have entered valid domain names, choose **Review and Request** or choose **Cancel** to quit.
5. If the review page correctly contains the information you entered for your request, choose **Confirm and request**. The following page shows that your request status is pending validation.



□		Domain name	Additional names
□	▸	example.com	www.example.com

Before an ACM Certificate can be issued, an authorized representative must validate that it was requested. For more information, see Validate Domain Ownership (p. 12).

To request an ACM Certificate (AWS CLI)

Use the request-certificate command to request a new ACM Certificate on the command line.

```
aws acm request-certificate --domain-name www.example.com
```

See the AWS CLI reference for more information and examples.

Validate Domain Ownership

> **Note**
> The information in this section applies only to certificates provided by ACM. ACM does not validate domain ownership for certificates that you import into ACM (p. 23).

Before the Amazon certificate authority (CA) can issue a certificate for your site, AWS Certificate Manager (ACM) must verify that you own or control the domain for which the ACM Certificate will be issued. ACM does this by sending a domain validation email to an address that is registered to the domain. For more information, see Configure Email for Your Domain (p. 9). Email is sent to the following three registered contact addresses in WHOIS:

- Domain registrant
- Technical contact
- Administrative contact

> **Note**
> Some registrars allow you to hide your contact information in your WHOIS listing, and others allow you to substitute your real email address with a privacy (or proxy) address. To prevent problems with receiving the domain validation email from ACM, ensure your contact information is visible in WHOIS. If your WHOIS listing shows a privacy email address, ensure that email sent to the privacy address is forwarded to your real email address, or list your real email address instead.

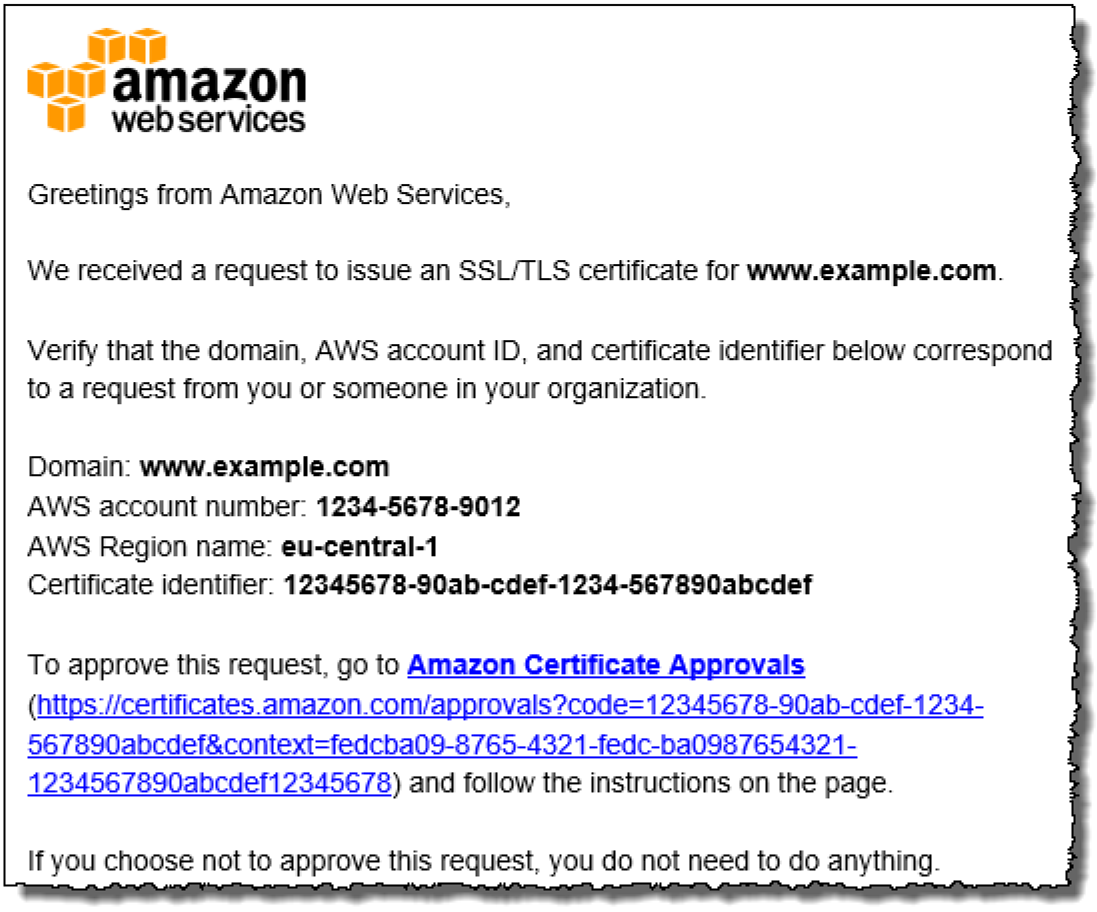
Email is also sent to the following five common system administration addresses where *`your_domain`* is the domain name that you entered when you initially requested the certificate.

- administrator@*`your_domain`*
- hostmaster@*`your_domain`*
- postmaster@*`your_domain`*
- webmaster@*`your_domain`*
- admin@*`your_domain`*

> **Note**
> There is an exception to the process described above. If you request an ACM Certificate for a domain name that begins with **`www`** or a wildcard asterisk (**`*`**), ACM removes the leading **`www`** or asterisk and sends email to the administrative addresses formed by pre-pending admin@, administrator@, hostmaster@, postmaster@, and webmaster@ to the remaining portion of the domain name. For example, if you request an ACM Certificate for www.example.com, email is sent to admin@example.com rather than to admin@www.example.com. Likewise, if you request an ACM Certificate for *.test.example.com, email is sent to admin@test.example.com. The remaining common administrative addresses are similarly formed.

> **Note**
> To ensure that email is sent to the administrative addresses for an apex domain, such as example.com, rather than to the administrative addresses for a subdomain, such as test.example.com, specify the `ValidationDomain` option in the RequestCertificate API or the request-certificate AWS CLI command. This feature is not currently supported in the console.

The following validation email is sent.



Greetings from Amazon Web Services,

We received a request to issue an SSL/TLS certificate for **www.example.com**.

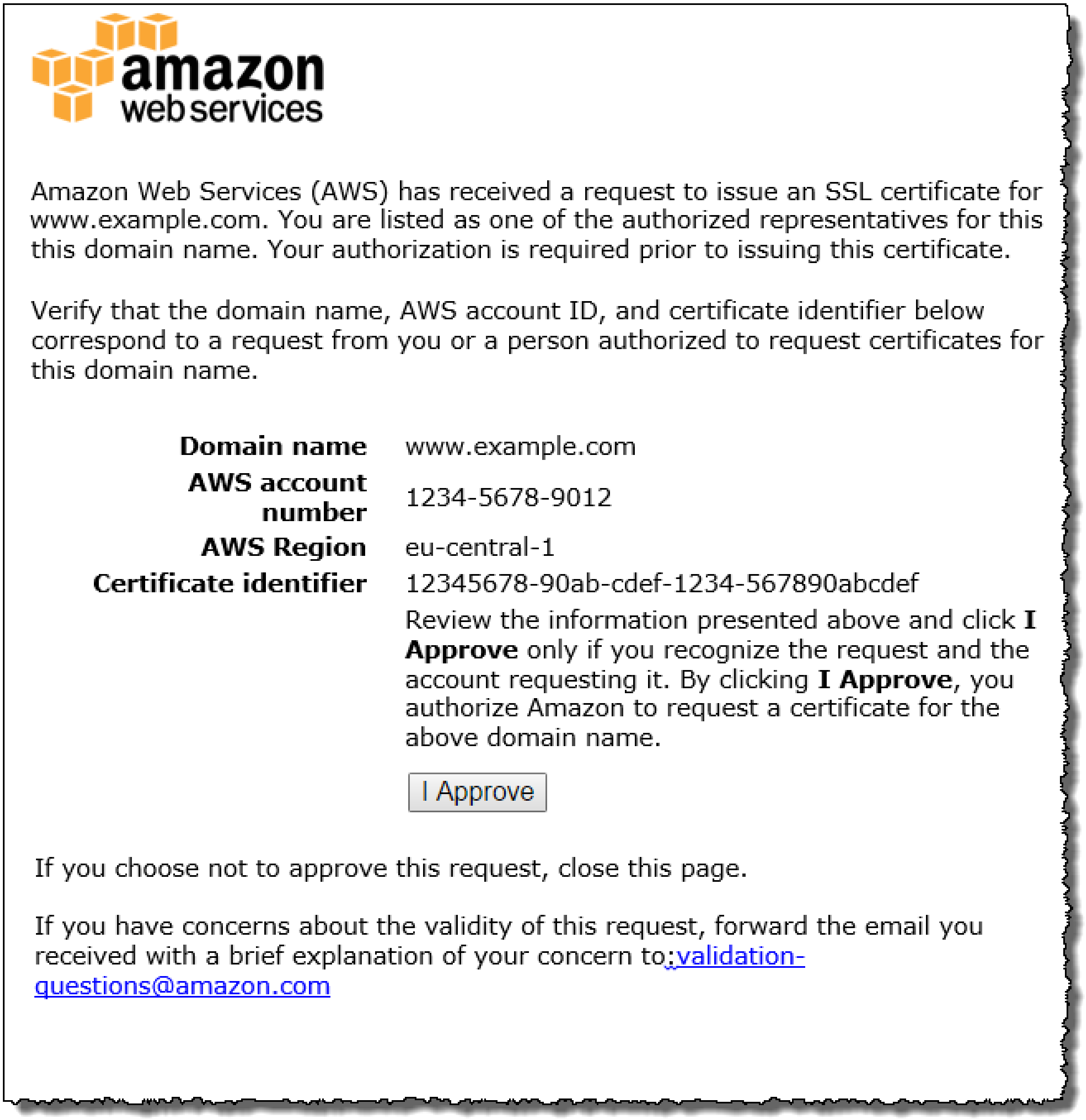
Verify that the domain, AWS account ID, and certificate identifier below correspond to a request from you or someone in your organization.

Domain: **www.example.com**
AWS account number: **1234-5678-9012**
AWS Region name: **eu-central-1**
Certificate identifier: **12345678-90ab-cdef-1234-567890abcdef**

To approve this request, go to **Amazon Certificate Approvals** (https://certificates.amazon.com/approvals?code=12345678-90ab-cdef-1234-567890abcdef&context=fedcba09-8765-4321-fedc-ba0987654321-1234567890abcdef12345678) and follow the instructions on the page.

If you choose not to approve this request, you do not need to do anything.

Choose the link that sends you to the Amazon Certificate Approvals website and then choose **I Approve**.



Amazon Web Services (AWS) has received a request to issue an SSL certificate for www.example.com. You are listed as one of the authorized representatives for this this domain name. Your authorization is required prior to issuing this certificate.

Verify that the domain name, AWS account ID, and certificate identifier below correspond to a request from you or a person authorized to request certificates for this domain name.

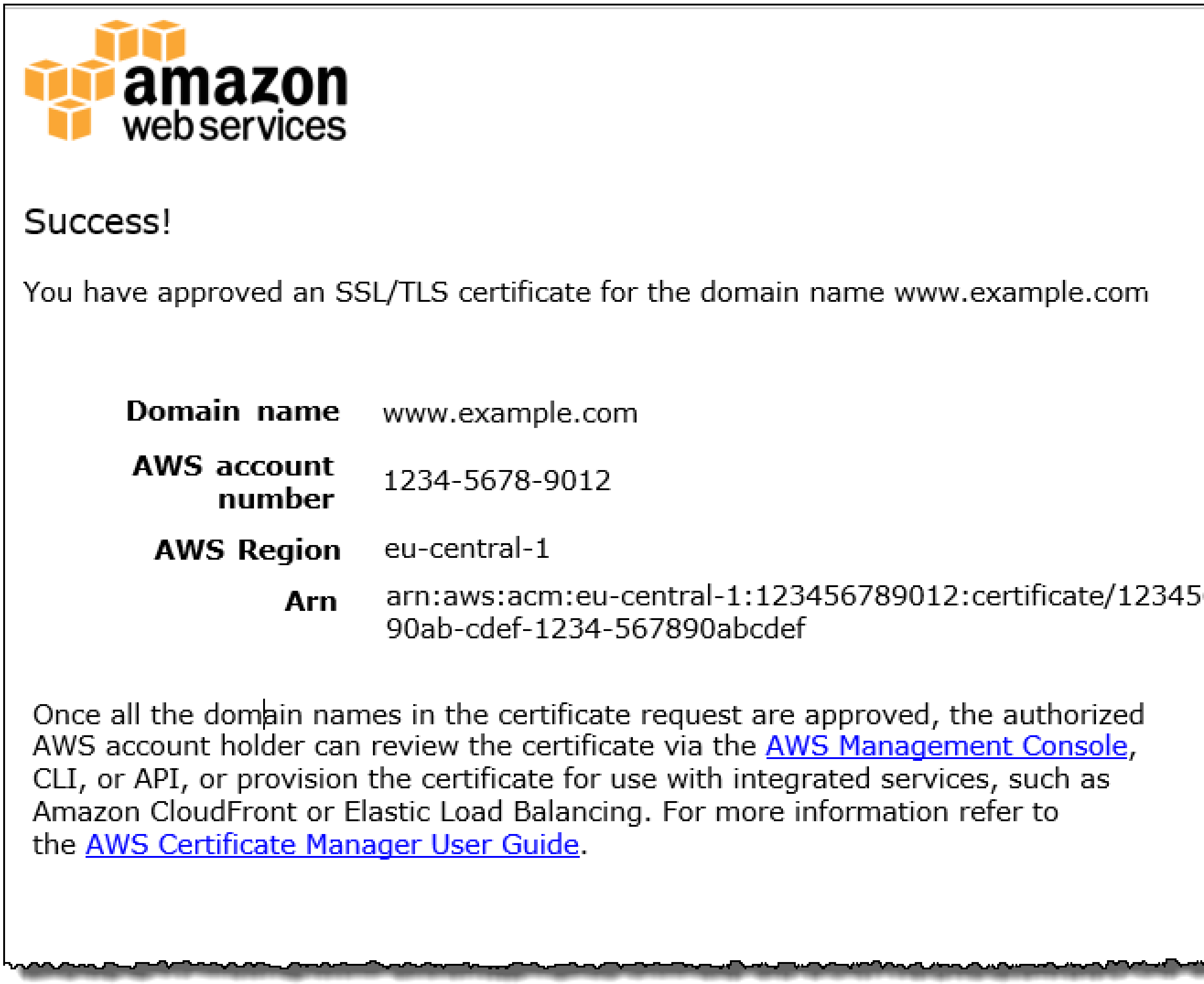
Domain name	www.example.com
AWS account number	1234-5678-9012
AWS Region	eu-central-1
Certificate identifier	12345678-90ab-cdef-1234-567890abcdef

Review the information presented above and click **I Approve** only if you recognize the request and the account requesting it. By clicking **I Approve**, you authorize Amazon to request a certificate for the above domain name.

If you choose not to approve this request, close this page.

If you have concerns about the validity of this request, forward the email you received with a brief explanation of your concern to:validation-questions@amazon.com

After choosing **I Approve**, a website opens to indicate that your request was successful.



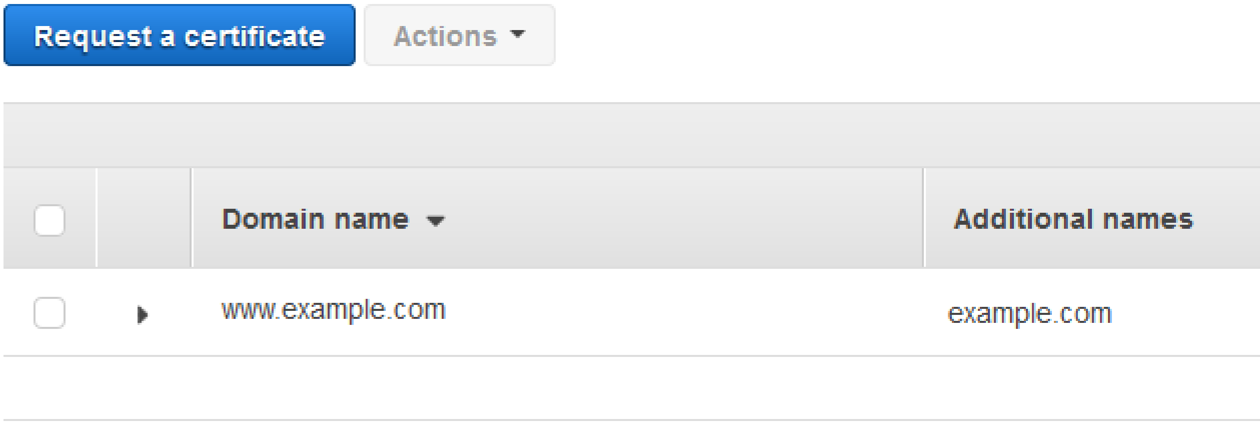
Success!

You have approved an SSL/TLS certificate for the domain name www.example.com

Domain name	www.example.com
AWS account number	1234-5678-9012
AWS Region	eu-central-1
Arn	arn:aws:acm:eu-central-1:123456789012:certificate/12345 90ab-cdef-1234-567890abcdef

Once all the domain names in the certificate request are approved, the authorized AWS account holder can review the certificate via the AWS Management Console, CLI, or API, or provision the certificate for use with integrated services, such as Amazon CloudFront or Elastic Load Balancing. For more information refer to the AWS Certificate Manager User Guide.

You can navigate back to the ACM console by clicking a link on the success page. The **Status** column in the console indicates that the ACM Certificate has been **Issued**.



□		Domain name	Additional names
□	▸	www.example.com	example.com

Manage ACM Certificates

After you have requested (p. 11) one or more certificates and AWS Certificate Manager has provided them, you can manage those certificates from the AWS Management Console or AWS CLI. You can also manage the certificates that you imported (p. 23).

Manage ACM Certificates (Console)

You can use the ACM console to get information about or delete an ACM Certificate. For certificates provided by ACM, you can also have ACM resend the validation email.

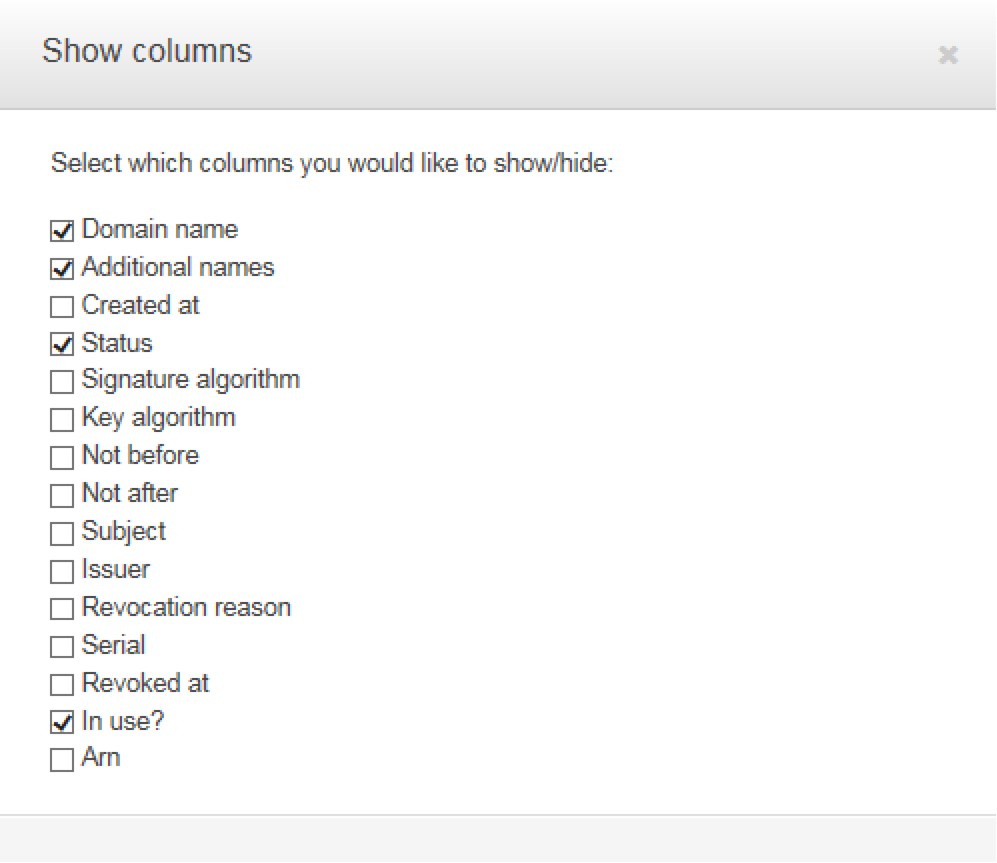
Display ACM Certificate Information

Each of the ACM Certificates occupies a row in the console. By default, the following columns are displayed for each certificate:

- **Domain Name** – The fully qualified domain name for the certificate.
- **Additional Names** – Additional names that are supported by this certificate.
- **Status** – Certificate status. This can be any of the following values:
  - Pending validation
  - Issued
  - Inactive
  - Expired
  - Revoked
  - Failed
  - Timed out
- **In Use?** – Whether the ACM Certificate is actively associated with an AWS service such as Elastic Load Balancing or CloudFront. The value can be **No** or **Yes**.

Customize Console Display

You can select the columns that you want to display by choosing the gear icon (⚙) in the upper right corner of the console. You can select from among the following columns.



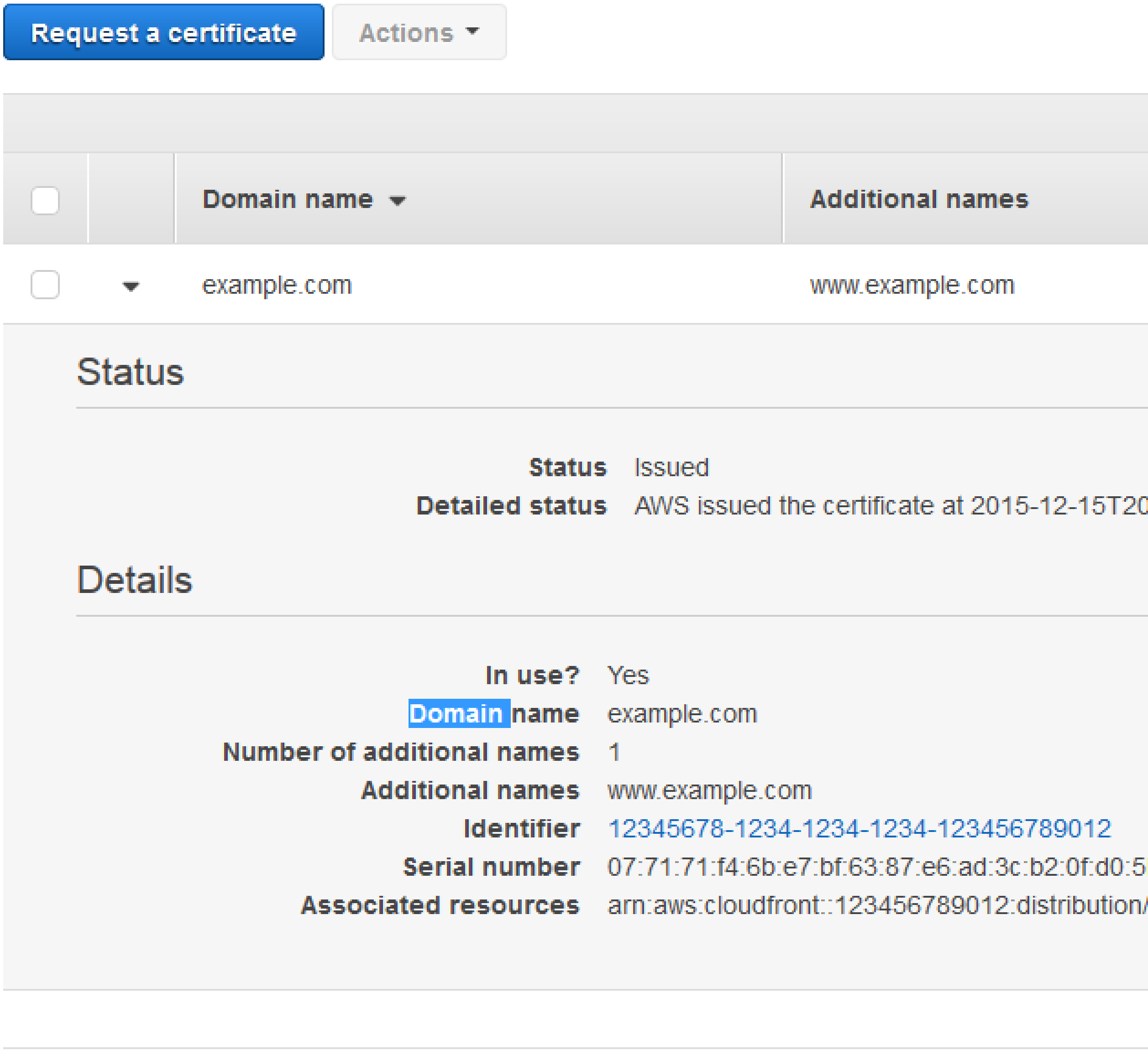
Show columns

Select which columns you would like to show/hide:

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

Display Certificate Metadata

To show ACM Certificate metadata, choose the arrow to the immediate left of the domain name. The console displays information similar to the following.



		Domain name	Additional names
□		example.com	www.example.com

Status

Status Issued
Detailed status AWS issued the certificate at 2015-12-15T20

Details

In use? Yes
Domain name example.com
Number of additional names 1
Additional names www.example.com
Identifier 12345678-1234-1234-1234-123456789012
Serial number 07:71:71:f4:6b:e7:bf:63:87:e6:ad:3c:b2:0f:d0:5
Associated resources arn:aws:cloudfront::123456789012:distribution/

Delete an ACM Certificate

In the list of certificates, select the check box for the ACM Certificate that you want to delete. For **Actions**, choose **Delete**.

Resend Validation Email (ACM-provided Certificates)

You approve an ACM Certificate request by using a validation token that ACM sends to the authorized representative. However, because the validation email required for the approval process can be blocked by spam filters or lost in transit, the validation token automatically expires after 72 hours. If the registered representative does not receive the original email or the token has expired, you can request that the email be resent by selecting the check box for the ACM Certificate, choosing the **Actions** button, and then choosing **Resend email**. If the 72 hour period has passed and the certificate status has changed to **Timed out**, you cannot resend validation email.

Note
The preceding information applies only to certificates provided by ACM. You cannot resend validation email for certificates that you imported into ACM (p. 23).

Manage ACM Certificates (AWS CLI)

You can use the AWS CLI to get information about an issued certificate, delete a certificate, or resend validation email.

Retrieve ACM Certificate Fields

You can use the describe-certificate command to retrieve information about a certificate.

```
aws acm describe-certificate --certificate-arn arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012
```

Delete an ACM Certificate

You can use the delete-certificate command to delete a certificate.

```
aws acm delete-certificate --certificate-arn arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012
```

Resend Validation Email (ACM-provided Certificates)

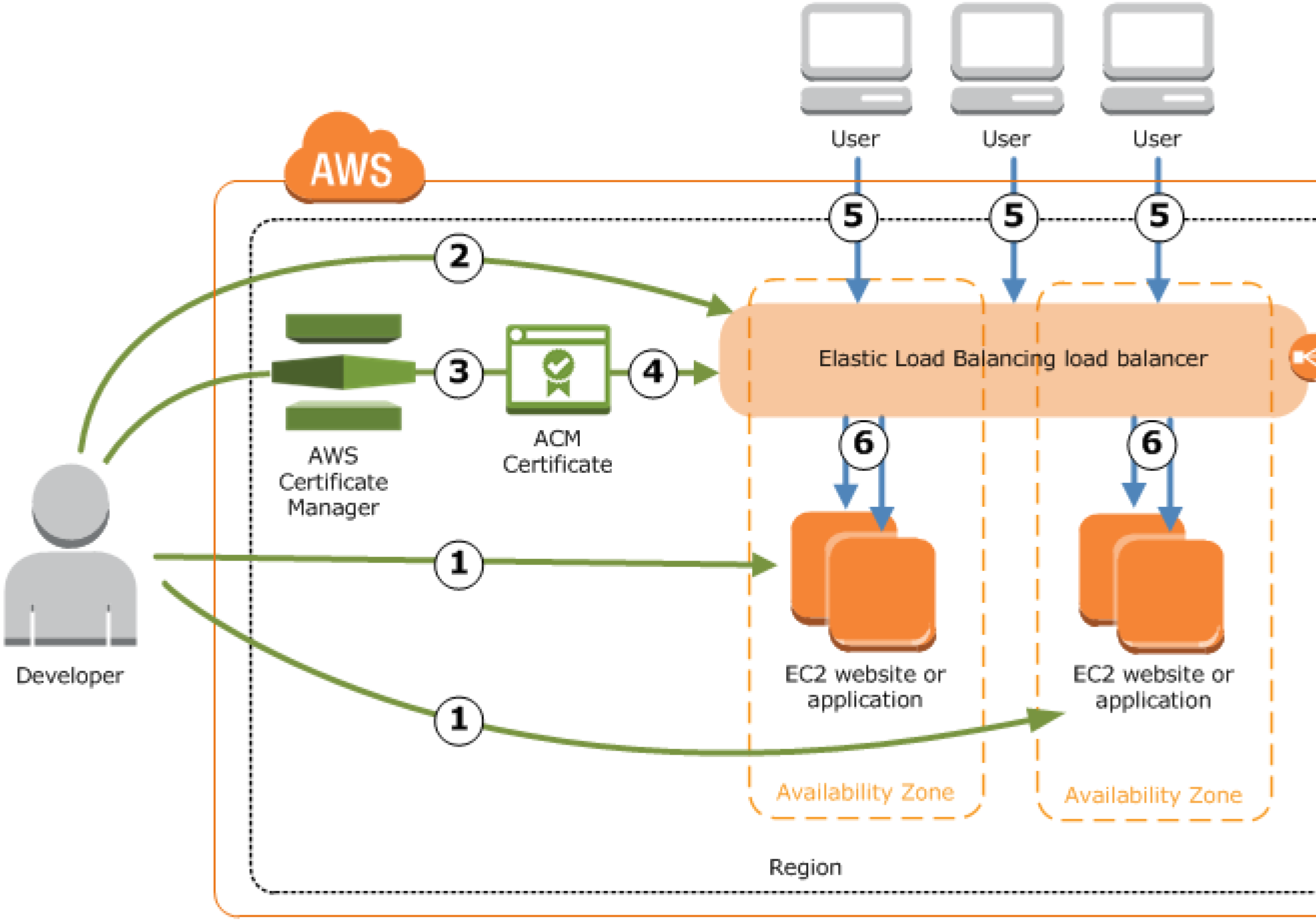
You can use the resend-validation-email command to send validation email again.

```
aws acm resend-validation-email --certificate-arn arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012 --
validation-domain example.com
```

Note
The preceding information applies only to certificates provided by ACM. You cannot resend validation email for certificates that you imported into ACM (p. 23).

Getting Started with Elastic Load Balancing

You do not install your ACM Certificate directly on the Amazon EC2 instances that contain your website or your application. Instead, you associate the ACM Certificate with an AWS service, such as Elastic Load Balancing. Elastic Load Balancing improves the availability of your website or application by automatically distributing incoming traffic across your Amazon EC2 instances. The load balancer serves as a single point of contact for clients. This is shown by the following illustration.



You must perform the following steps to use ACM with a load balancer.

1. Install a website or application on one or more Amazon EC2 instances.
2. Create an Elastic Load Balancing load balancer to route client traffic to the Amazon EC2 instances.
3. Use the ACM console, API, or AWS Command Line Interface to request an ACM Certificate (p. 11) or import a certificate into ACM (p. 23).
4. Use the Elastic Load Balancing console, API, or AWS CLI to provision the ACM Certificate on the load balancer.
5. Clients access the website through the load balancer.
6. The load balancer distributes client traffic to the Amazon EC2 instances.

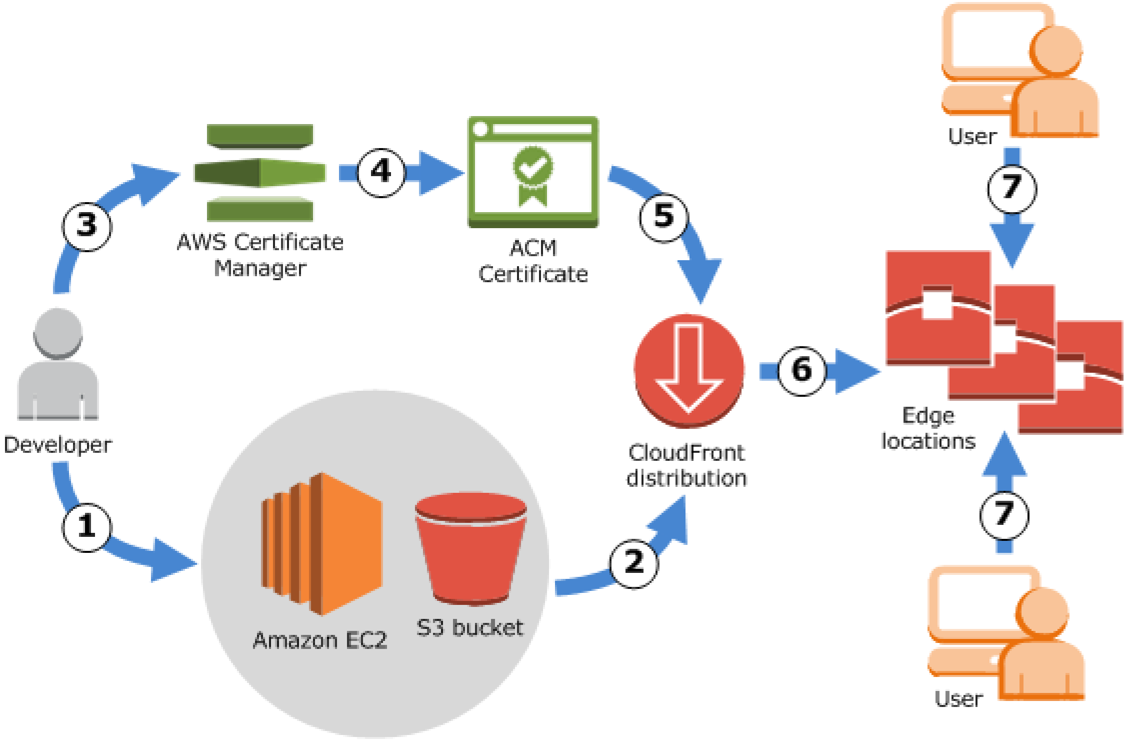
For more information about configuring your load balancer to use a certificate provided by ACM (step 4 in the preceding list), see one of the following topics:

- Create an HTTPS Listener for Your Application Load Balancer in the *Application Load Balancer Guide*
- HTTPS Listeners for Your Classic Load Balancer in the *Classic Load Balancer Guide*

Getting Started with Amazon CloudFront

Amazon CloudFront is a web service that speeds up distribution of your static and dynamic web content by delivering the content through a worldwide network of edge locations. For more information, see the Amazon CloudFront Developer Guide. For more information about using HTTPS with CloudFront, see Using HTTPS with CloudFront in the *Amazon CloudFront Developer Guide*.

The following illustration shows how ACM and CloudFront can be used together to deliver your content securely through edge locations.



You must perform the following steps to use ACM with CloudFront.

> **Note**
> To use an ACM Certificate with CloudFront, you must request or import the certificate in the US East (N. Virginia) region.

1. Configure your origin servers to store the original, definitive version of your content. Your origin can be either an Amazon S3 bucket or an Amazon EC2 web server instance.
2. Create a CloudFront **distribution** which tells CloudFront which origin servers to get your files from when users make requests through your web site or application.
3. Use the ACM console, ACM API, or AWS CLI to request a certificate (p. 11) or import a certificate into ACM (p. 23).
4. Use the CloudFront console, API, or AWS CLI to associate your ACM Certificate with the CloudFront **distribution**.
5. CloudFront sends your distribution's configuration (but not your content) to all of its edge locations. Edge locations are collections of servers in geographically dispersed data centers where CloudFront caches copies of your website or application files.
6. When a user accesses your website and requests one or more objects, DNS routes the user to the CloudFront edge location that can best serve the request. For more information about how CloudFront delivers content to your users, see How CloudFront Delivers Content in the *Amazon CloudFront Developer Guide*.

Importing Certificates into AWS Certificate Manager

In addition to requesting SSL/TLS certificates provided by AWS Certificate Manager (ACM), you can import certificates that you obtained outside of AWS. You might do this because you already obtained a certificate from a third-party issuer, or because the certificates provided by ACM do not meet your requirements.

After you import a certificate, you can use it with the AWS services that are integrated with ACM (p. 5). The certificates that you import work the same as those provided by ACM, with one important exception: ACM does not provide managed renewal (p. 4) for imported certificates.

> **Important**
> You are responsible for monitoring the expiration date of your imported certificates and for renewing them before they expire.

The ACM console displays a warning when an imported certificate is nearing its expiration date. To renew an imported certificate, you can obtain a new certificate from your certificate issuer and then import it to ACM. Or, you can request a new certificate (p. 11) from ACM.

All certificates in ACM are regional resources, including the certificates that you import. To use the same certificate with Elastic Load Balancing load balancers in different AWS regions, you must import the certificate into each region where you want to use it. To use a certificate with Amazon CloudFront, you must import it into the US East (N. Virginia) region. For more information, see Supported Regions (p. 5).

For information about how to import certificates into ACM, see the following topics.

Topics

- Prerequisites for Importing Certificates (p. 23)
- Importing Certificates (AWS Management Console) (p. 24)
- Importing Certificates (ACM API) (p. 24)
- Troubleshooting (p. 25)

Prerequisites for Importing Certificates

To import a certificate into ACM, you must provide the certificate and its matching private key. When the certificate is not self-signed, you must also provide a certificate chain. (You don't need a certificate chain when importing a self-signed certificate.) Before you import a certificate, ensure that you have all these items and that they meet the following criteria:

- The certificate must contain a 1024-bit or 2048-bit RSA public key.
- The certificate must be an SSL/TLS certificate with at least one fully qualified domain name. You cannot import a certificate for code signing, email encryption, or other uses.
- The certificate must be valid at the time of import. You cannot import a certificate before its validity period begins (the certificate's `NotBefore` date) or after it expires (the certificate's `NotAfter` date).
- The private key must be unencrypted. You cannot import a private key that is protected by a password or passphrase. For help decrypting an encrypted private key, see Troubleshooting (p. 25).
- The certificate, private key, and certificate chain must all be PEM-encoded. For help converting these items to PEM format, see Troubleshooting (p. 25).

Importing Certificates (AWS Management Console)

To import a certificate to ACM (console)

1. Open the ACM console at https://console.aws.amazon.com/acm/home.
2. Choose **Import a certificate**.
3. Do the following:

   a. For **Certificate body**, paste the PEM-encoded certificate to import.
   b. For **Certificate private key**, paste the PEM-encoded, unencrypted private key that matches the certificate's public key.
   c. (Optional) For **Certificate chain**, paste the PEM-encoded certificate chain.
4. Choose **Review and import**.
5. Review the information about your certificate, then choose **Import**.

Importing Certificates (ACM API)

To use the ACM API to import a certificate, send an ImportCertificate request. The following example shows how to do this with the AWS Command Line Interface (AWS CLI). The example assumes the following:

- The PEM-encoded certificate is stored in a file named `Certificate.pem`.
- The PEM-encoded certificate chain is stored in a file named `CertificateChain.pem`.
- The PEM-encoded, unencrypted private key is stored in a file named `PrivateKey.pem`.

To use the following example command, replace these file names with your own and type the command on one continuous line. The following example includes line breaks and extra spaces to make it easier to read.

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

When the preceding command is successful, it returns the Amazon Resource Name (ARN) of the imported certificate.

Troubleshooting

Before you can import a certificate into ACM, you must make sure that the certificate, private key, and certificate chain are all PEM-encoded. You must also ensure that the private key is unencrypted. See the following examples.

Example PEM-encoded certificate

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

Example PEM-encoded, unencrypted private key

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

Example PEM-encoded certificate chain

A certificate chain contains one or more certificates. The following example contains three certificates, but your certificate chain might contain more or fewer.

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

If these items are not in the right format for importing into ACM, you can use OpenSSL to convert them to the right format.

To convert a certificate or certificate chain from DER to PEM

Use the OpenSSL **x509** command, as in the following example. In the following example command, replace *Certificate.der* with the name of the file that contains your DER-encoded certificate. Replace *Certificate.pem* with the desired name of the output file to contain the PEM-encoded certificate.

```
$ openssl x509 -inform DER -in Certificate.der -outform PEM -
out Certificate.pem
```

To convert a private key from DER to PEM

Use the OpenSSL **rsa** command, as in the following example. In the following example command, replace *PrivateKey.der* with the name of the file that contains your DER-encoded private key.

Replace *PrivateKey.pem* with the desired name of the output file to contain the PEM-encoded private key.

```
$ openssl rsa -inform DER -in PrivateKey.der -outform PEM -
out PrivateKey.pem
```

To decrypt an encrypted private key (remove the password or passphrase)

Use the OpenSSL **rsa** command, as in the following example. To use the following example command, replace *EncryptedPrivateKey.pem* with the name of the file that contains your encrypted private key. Replace *PrivateKey.pem* with the desired name of the output file to contain the PEM-encoded unencrypted private key.

```
$ openssl rsa -in EncryptedPrivateKey.pem -out PrivateKey.pem
```

To convert a certificate bundle from PKCS#12 (PFX) to PEM

Use the OpenSSL **pkcs12** command, as in the following example. In the following example command, replace *CertificateBundle.p12* with the name of the file that contains your PKCS#12-encoded certificate bundle. Replace *CertificateBundle.pem* with the desired name of the output file to contain the PEM-encoded certificate bundle.

```
$ openssl pkcs12 -in CertificateBundle.p12 -out CertificateBundle.pem -
nodes
```

To convert a certificate bundle from PKCS#7 to PEM

Use the OpenSSL **pkcs7** command, as in the following example. In the following example command, replace *CertificateBundle.p7b* with the name of the file that contains your PKCS#7-encoded certificate bundle. Replace *CertificateBundle.pem* with the desired name of the output file to contain the PEM-encoded certificate bundle.

```
$ openssl pkcs7 -in CertificateBundle.p7b -print_certs -
out CertificateBundle.pem
```

Tagging AWS Certificate Manager Certificates

A *tag* is a label that you can assign to an ACM Certificate. Each tag consists of a *key* and a *value*. You can use the AWS Certificate Manager console, AWS Command Line Interface (AWS CLI), or ACM API to add, view, or remove tags for ACM Certificates. You can choose which tags to display in the ACM console.

You can create custom tags that suit your needs. For example, you could tag multiple ACM Certificates with an `Environment = Prod` or `Environment = Beta` tag to identify which environment each ACM Certificate is intended for. The following list includes a few additional examples of other custom tags:

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

Other AWS resources also support tagging. You can, therefore, assign the same tag to different resources to indicate whether those resources are related. For example, you can assign a tag such as `Website = example.com` to the ACM Certificate, the load balancer, and other resources used for your example.com website.

Topics

- Tag Restrictions (p. 27)
- Managing Tags (p. 28)

Tag Restrictions

The following basic restrictions apply to ACM Certificate tags:

- The maximum number of tags per ACM Certificate is 50.
- The maximum length of a tag key is 127 characters.
- The maximum length of a tag value is 255 characters.

- Tag keys and values are case sensitive.
- The `aws:` prefix is reserved for AWS use; you cannot add, edit, or delete tags whose key begins with `aws:`. Tags that begin with `aws:` do not count against your tags-per-resource limit.
- If you plan to use your tagging schema across multiple services and resources, remember that other services may have other restrictions for allowed characters. Refer to the documentation for that service.
- ACM Certificate tags are not available for use in the AWS Management Console's Resource Groups and Tag Editor.

Managing Tags

You can add, edit, and delete tags by using the AWS Management Console, the AWS Command Line Interface, or the AWS Certificate Manager API.

Managing Tags (Console)

You can use the AWS Management Console to add, delete, or edit tags. You can also display tags in columns.

Adding a Tag (Console)

Use the following procedure to add tags by using the ACM console.

To add a tag to a certificate (console)

1. Sign into the AWS Management Console and open the AWS Certificate Manager console at https://console.aws.amazon.com/acm/home.
2. Choose the arrow next to the certificate that you want to tag.
3. In the details pane, scroll down to **Tags**.
4. Choose **Edit** and **Add Tag**.
5. Type a key and a value for the tag.
6. Choose **Save**.

Deleting a Tag (Console)

Use the following procedure to delete tags by using the ACM console.

To delete a tag (console)

1. Sign into the AWS Management Console and open the AWS Certificate Manager console at https://console.aws.amazon.com/acm/home.
2. Choose the arrow next to the certificate with a tag that you want to delete.
3. In the details pane, scroll down to **Tags**.
4. Choose **Edit**.
5. Choose the **X** next to the tag you want to delete.
6. Choose **Save**.

Editing a Tag (Console)

Use the following procedure to edit tags by using the ACM console.

To edit a tag (console)

1. Sign into the AWS Management Console and open the AWS Certificate Manager console at https://console.aws.amazon.com/acm/home.
2. Choose the arrow next to certificate you want to edit.
3. In the details pane, scroll down to **Tags**.
4. Choose **Edit**.
5. Modify the key or value of the tag you want to change.
6. Choose **Save**.

Showing Tags in Columns (Console)

Use the following procedure to show tags in columns in the ACM console.

To display tags in columns (console)

1. Sign into the AWS Management Console and open the AWS Certificate Manager console at https://console.aws.amazon.com/acm/home.
2. Choose the tags that you want to display as columns by choosing the gear icon ⚙ in the upper right corner of the console.
3. Select the check box beside the tag that you want to display in a column.

Managing Tags (AWS Command Line Interface)

Refer to the following topics to learn how to add, list, and delete tags by using the AWS CLI.

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

Managing Tags (AWS Certificate Manager API)

Refer to the following topics to learn how to add, list, and delete tags by using the API.

- AddTagsToCertificate
- ListTagsForCertificate
- RemoveTagsFromCertificate

Permissions and Policies

AWS Certificate Manager (ACM) uses AWS Identity and Access Management (IAM) to manage user permissions. The following topics provide a brief overview that explains how you can use IAM in ACM. For a more complete discussion of IAM, see the AWS Identity and Access Management User Guide.

Topics

- Managed Policies (p. 30)
- Inline Policies (p. 31)
- Integrated Resource Policies (p. 34)

Managed Policies

Managed policies are standalone policies that you can attach to multiple users, groups, and roles in your AWS account. The following AWS-managed policies are available for ACM. For more information about attaching managed policies to a user, group, or role, see Working with Managed Policies in the IAM User Guide.

Topics

- AWSCertificateManagerReadOnly (p. 30)
- AWSCertificateManagerFullAccess (p. 31)
- Attaching a Managed Policy Using the AWS Management Console (p. 31)

AWSCertificateManagerReadOnly

This policy provides read-only access to ACM Certificates; it allows users to describe, list, and retrieve ACM Certificates.

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

To view this AWS-managed policy in the console, go to https://console.aws.amazon.com/iam/home#policies/arn:aws:iam::aws:policy/AWSCertificateManagerReadOnly.

AWSCertificateManagerFullAccess

This policy provides full access to all ACM actions and resources, and includes all of the permissions which are allowed under the AWSCertificateManagerReadOnly policy.

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

To view this AWS-managed policy in the console, go to https://console.aws.amazon.com/iam/home#policies/arn:aws:iam::aws:policy/AWSCertificateManagerFullAccess.

Attaching a Managed Policy Using the AWS Management Console

To use a managed policy, a user with administrative privileges must attach the policy to a user, role, or group. The following procedure discusses how to do this in the IAM management console.

1. Sign in to the Identity and Access Management (IAM) console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose **Policies**.
3. In the list of policies, select the check box next to the name of the policy to attach. You can use the **Filter** menu and the **Search** box to filter the list of policies.
4. Choose **Policy Actions**, and then choose **Attach**.
5. Select the user, group, or role to attach the policy to. You can use the **Filter** menu and the **Search** box to filter the list of principal entities. After selecting the principal entities to attach the policy to, choose **Attach Policy**.

Inline Policies

Inline policies are policies that you create and manage and embed directly into a single user, group, or role. The following policy examples show you how to assign permissions to perform ACM actions. For more information about attaching inline policies, see Working with Inline Policies in the IAM User Guide.

Topics

- Listing Certificates (p. 32)
- Retrieving a Certificate (p. 32)
- Deleting a Certificate (p. 32)
- Read-only Access to ACM (p. 33)
- Full Access to ACM (p. 33)
- Administrator Access to All AWS Resources (p. 33)

Listing Certificates

The following policy allows a user to list all of the ACM Certificates in the user's account.

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

Note
This permission is required for ACM Certificates to appear in the Elastic Load Balancing and CloudFront consoles.

Retrieving a Certificate

The following policy allows a user to retrieve a specific ACM Certificate.

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

Deleting a Certificate

The following policy allows a user to delete a specific ACM Certificate.

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

Read-only Access to ACM

The following policy allows a user to describe and list an ACM Certificate and to retrieve the ACM Certificate and certificate chain.

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

Note

This policy is available as an AWS-managed policy in the AWS Management Console. For more information, see AWSCertificateManagerReadOnly (p. 30). To view the managed policy in the console, go to https://console.aws.amazon.com/iam/home#policies/arn:aws:iam::aws:policy/AWSCertificateManagerReadOnly.

Full Access to ACM

The following policy allows a user to perform any ACM action.

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

Note

This policy is available as an AWS-managed policy in the AWS Management Console. For more information, see AWSCertificateManagerFullAccess (p. 31). To view the managed policy in the console, go to https://console.aws.amazon.com/iam/home#policies/arn:aws:iam::aws:policy/AWSCertificateManagerFullAccess.

Administrator Access to All AWS Resources

The following policy allows a user to perform any action on any AWS resource.

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

Note
This policy is available as an AWS-managed policy in the AWS Management Console. To view the managed policy in the console, go to https://console.aws.amazon.com/iam/home#policies/arn:aws:iam::aws:policy/AdministratorAccess.

Integrated Resource Policies

The following topics discuss how to provide IAM access to users, groups, and roles for the services integrated with ACM.

Example IAM Policies for Elastic Load Balancing

For more information and example policies detailing how to grant IAM permissions to Elastic Load Balancing users and groups, see Authentication and Access Control for Your Load Balancer.

Example IAM Policies for Amazon CloudFront

For more information and example policies detailing how to grant IAM permissions to CloudFront resources, see Authentication and Access Control for CloudFront.

Using AWS CloudTrail

You can use CloudTrail to record API calls that are made by AWS Certificate Manager and by services integrated with ACM as discussed in the following topics.

Topics

- Logging AWS Certificate Manager API Calls with AWS CloudTrail (p. 35)
- Logging ACM-Related API Calls (p. 44)

Logging AWS Certificate Manager API Calls with AWS CloudTrail

AWS Certificate Manager (ACM) is integrated with AWS CloudTrail, a service that captures API calls, delivers the log files to an Amazon Simple Storage Service (Amazon S3) bucket that you specify, and maintains API call history. CloudTrail captures API calls from the AWS Certificate Manager console, CLI, or from your code. Using the information collected by CloudTrail, you can determine the request that was made to ACM, the IP address from which the request was made, who made the request, when it was made, and so on.

To learn more about CloudTrail, including how to configure and enable it, see the AWS CloudTrail User Guide.

When you enable CloudTrail logging in your AWS account, API calls made to ACM actions are tracked in CloudTrail log files. The ACM records are written with other AWS service records. CloudTrail determines when to create and write to a new log file based on a time period and file size.

The following ACM actions are supported:

- AddTagsToCertificate
- DeleteCertificate
- DeleteCertificate
- DescribeCertificate
- GetCertificate
- ImportCertificate
- ListCertificates
- ListTagsForCertificate
- RemoveTagsFromCertificate
- RequestCertificate

- ResendValidationEmail

Every log entry contains information about who generated the request. The user identity information in the log entry helps you determine whether the request was made with root or with IAM user credentials, with temporary security credentials for a role or federated user, or by another AWS service. For more information, see the CloudTrail userIdentity Element.

You can store your log files in your bucket for as long as you want, but you can also define Amazon S3 lifecycle rules to archive or delete log files automatically. By default, your log files are encrypted using Amazon S3 server-side encryption (SSE).

You can choose to have CloudTrail publish Amazon SNS notifications when new log files are delivered if you want to take quick action upon log delivery. For more information, see Configuring Amazon SNS Notifications for CloudTrail in the *AWS CloudTrail User Guide*.

You can also aggregate AWS Certificate Manager log files from multiple AWS regions and multiple AWS accounts into a single Amazon S3 bucket. For more information, see Receiving CloudTrail Log Files from Multiple Regions and Receiving CloudTrail Log Files from Multiple Accounts.

CloudTrail log files contain one or more log entries where each entry lists multiple JSON-formatted events. A log entry represents a single request from any source and includes information about the requested action, the date and time of the action, request parameters and so on. The log entries are not guaranteed to be in any particular order. That is, they are not an ordered trace of the public API calls. For more information about the fields that make up a log entry, see the CloudTrail Event Reference.

For examples of possible ACM CloudTrail entries, see the following topics.

Topics

- Adding Tags to a Certificate (p. 36)
- Deleting a Certificate (p. 37)
- Describing a Certificate (p. 38)
- Retrieving a Certificate (p. 38)
- Import a Certificate (p. 39)
- Listing Certificates (p. 41)
- Listing Tags for a Certificate (p. 42)
- Removing Tags from a Certificate (p. 42)
- Requesting a Certificate (p. 43)
- Resending Validation Email (p. 44)

Adding Tags to a Certificate

The following CloudTrail example shows the results of a call to the AddTagsToCertificate API.

```
{
    Records: [{
        eventVersion: "1.04",
        userIdentity: {
            type: "IAMUser",
            principalId: "AIDACKCEVSQ6C2EXAMPLE",
            arn: "arn:aws:iam::123456789012:user/Alice",
            accountId: "123456789012",
            accessKeyId: "AKIAIOSFODNN7EXAMPLE",
            userName: "Alice"
        },
        eventTime: "2016-04-06T13:53:53Z",
        eventSource: "acm.amazonaws.com",
        eventName: "AddTagsToCertificate",
        awsRegion: "us-east-1",
        sourceIPAddress: "192.0.2.0",
        userAgent: "aws-cli/1.10.16",
        requestParameters: {
            tags: [{
                value: "Alice",
                key: "Admin"
            }],
            certificateArn: "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
        },
        responseElements: null,
        requestID: "ffd7dd1b-fbfe-11e5-ba7b-5f4e988901f9",
        eventID: "4e7b10bb-7010-4e60-8376-0cac3bc860a5",
        eventType: "AwsApiCall",
        recipientAccountId: "123456789012"
    }]
}
```

Deleting a Certificate

The following CloudTrail example shows the results of a call to the DeleteCertificate API.

```
{
    "Records": [{
        "eventVersion": "1.04",
        "userIdentity": {
            "type": "IAMUser",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:iam::123456789012:user/Alice",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "userName": "Alice"
        },
        "eventTime": "2016-03-18T00:00:26Z",
        "eventSource": "acm.amazonaws.com",
        "eventName": "DeleteCertificate",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "192.0.2.0",
        "userAgent": "aws-cli/1.9.15",
        "requestParameters": {
            "certificateArn": "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
        },
        "responseElements": null,
        "requestID": "6b0f5bb9-ec9c-11e5-a28b-51e7e3169e0f",
        "eventID": "08f18f8a-a827-4924-b864-afaf98517793",
        "eventType": "AwsApiCall",
        "recipientAccountId": "123456789012"
    }]
}
```

Describing a Certificate

The following CloudTrail example shows the results of a call to the DescribeCertificate API.

> **Note**
> The CloudTrail log for the `DescribeCertificate` action does not display information about the ACM Certificate you specify. You can view information about the certificate by using the console, the AWS Command Line Interface, or the DescribeCertificate API.

```
{
    "Records": [{
        "eventVersion": "1.04",
        "userIdentity": {
            "type": "IAMUser",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:iam::123456789012:user/Alice",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "userName": "Alice"
        },
        "eventTime": "2016-03-18T00:00:42Z",
        "eventSource": "acm.amazonaws.com",
        "eventName": "DescribeCertificate",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "192.0.2.0",
        "userAgent": "aws-cli/1.9.15",
        "requestParameters": {
            "certificateArn": "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
        },
        "responseElements": null,
        "requestID": "74b91d83-ec9c-11e5-ac34-d1e4dfe1a11b",
        "eventID": "7779b6da-75c2-4994-b8c1-af3ad47b518a",
        "eventType": "AwsApiCall",
        "recipientAccountId": "123456789012"
    }]
}
```

Retrieving a Certificate

The following CloudTrail example shows the results of a call to the GetCertificate API.

```
{
    "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-----
            MIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
            VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
            b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
            BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
            MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
            VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
            b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
            YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
            21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
            rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
            Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
            nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
            FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
            NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=\n-----END
 CERTIFICATE-----",
            "certificate": "-----BEGIN CERTIFICATE-----
            MIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
            VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
            b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
            BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
            MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
            VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
            b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
            YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
            21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
            rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
            Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
            nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
            FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
            NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=-----END
 CERTIFICATE-----"
        },
        "requestID": "744dd891-ec9c-11e5-ac34-d1e4dfe1a11b",
        "eventID": "7aa4f909-00dd-478a-9a00-b2709bcad2bb",
        "eventType": "AwsApiCall",
        "recipientAccountId": "123456789012"
    }]
}
```

Import a Certificate

The following example shows the CloudTrail log entry that records a call to the ACM ImportCertificate API operation.

```
{
  "eventVersion": "1.04",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
    "arn": "arn:aws:iam::111122223333:user/Alice",
    "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "userName": "Alice"
  },
  "eventTime": "2016-10-04T16:01:30Z",
  "eventSource": "acm.amazonaws.com",
  "eventName": "ImportCertificate",
  "awsRegion": "ap-southeast-2",
  "sourceIPAddress": "54.240.193.129",
  "userAgent": "Coral/Netty",
  "requestParameters": {
    "privateKey": {
      "hb": [
        byte,
        byte,
        byte,
        ...
      ],
      "offset": 0,
      "isReadOnly": false,
      "bigEndian": true,
      "nativeByteOrder": false,
      "mark": -1,
      "position": 0,
      "limit": 1674,
      "capacity": 1674,
      "address": 0
    },
    "certificateChain": {
      "hb": [
        byte,
        byte,
        byte,
        ...
      ],
      "offset": 0,
      "isReadOnly": false,
      "bigEndian": true,
      "nativeByteOrder": false,
      "mark": -1,
      "position": 0,
      "limit": 2105,
      "capacity": 2105,
      "address": 0
    },
    "certificate": {
      "hb": [
        byte,
        byte,
        byte,
        ...
      ],
      "offset": 0,
      "isReadOnly": false,
      "bigEndian": true,
      "nativeByteOrder": false,
      "mark": -1,
      "position": 0,
      "limit": 2503,
      "capacity": 2503,
      "address": 0
    }
  },
  "responseElements": {
    "certificateArn": "arn:aws:acm:ap-
southeast-2:111122223333:certificate/6ae06649-ea82-4b58-90ee-dc05870d7e99"
  },
  "requestID": "cf1f3db7-8a4b-11e6-88c8-196af94bb7be",
  "eventID": "fb443118-bfaa-4c90-95c1-beef21e07f8e",
  "eventType": "AwsApiCall",
  "recipientAccountId": "111122223333"
}
```

Listing Certificates

The following CloudTrail example shows the results of a call to the ListCertificates API.

> **Note**
> The CloudTrail log for the `ListCertificates` action does not display your ACM certificates. You can view the certificate list by using the console, the AWS Command Line Interface, or the ListCertificates API.

```

{
    "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"
    }]
}
```

Listing Tags for a Certificate

The following CloudTrail example shows the results of a call to the ListTagsForCertificate API.

> **Note**
> The CloudTrail log for the `ListTagsForCertificate` action does not display your tags. You can view the tag list by using the console, the AWS Command Line Interface, or the ListTagsForCertificate API.

```
{
    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"
    }]
}
```

Removing Tags from a Certificate

The following CloudTrail example shows the results of a call to the RemoveTagsFromCertificate API.

```
{
    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"
    }]
}
```

Requesting a Certificate

The following CloudTrail example shows the results of a call to the RequestCertificate API.

```

{
    "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"
    }]
}
```

Resending Validation Email

The following CloudTrail example shows the results of a call to the ResendValidationEmail API.

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

Logging ACM-Related API Calls

You can use CloudTrail to audit API calls made by services that are integrated with ACM. For more information about using CloudTrail, see the AWS CloudTrail User Guide. The following examples show the types of logs that can be generated depending on the AWS resources on which you provision the ACM Certificate.

Topics

- Creating a Load Balancer (p. 45)
- Registering an Amazon EC2 Instance with a Load Balancer (p. 45)

- Encrypting a Private Key (p. 46)
- Decrypting a Private Key (p. 47)

Creating a Load Balancer

The following example shows a call to the `CreateLoadBalancer` function by an IAM user named Alice. The name of the load balancer is `TestLinuxDefault`, and the listener is created using an ACM Certificate.

```
{
 "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"
}
```

Registering an Amazon EC2 Instance with a Load Balancer

When you provision your website or application on an Amazon Elastic Compute Cloud (Amazon EC2) instance, the load balancer must be made aware of that instance. This can be accomplished through the Elastic Load Balancing console or the AWS Command Line Interface. The following example shows a call to `RegisterInstancesWithLoadBalancer` for a load balancer named LinuxTest on AWS account 123456789012.

```
{
    "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"
}
```

Encrypting a Private Key

The following example shows an `Encrypt` call that encrypts the private key associated with an ACM Certificate. Encryption is performed within AWS.

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

Decrypting a Private Key

The following example shows a `Decrypt` call that decrypts the private key associated with an ACM Certificate. Decryption is performed within AWS, and the decrypted key never leaves 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"
    }
```

Using the ACM API

You can use the AWS Certificate Manager API to interact with the service programmatically by sending HTTP requests. For more information, see the AWS Certificate Manager API Reference.

In addition to the web API (or HTTP API), you can use the AWS SDKs and command line tools to interact with ACM and other services. For more information, see Tools for Amazon Web Services.

The following pages show you how to use one of the AWS SDKs, the AWS SDK for Java, to perform some of the available operations in the AWS Certificate Manager API.

Topics

- Deleting a Certificate (p. 49)
- Describing a Certificate (p. 50)
- Getting a Certificate Chain (p. 51)
- Listing ACM Certificates (p. 53)
- Requesting a Certificate (p. 54)
- Resending Validation Email (p. 55)

Deleting a Certificate

The following example shows how to use the DeleteCertificate function.

```
package com.amazonaws.samples;

import com.amazonaws.regions.Region;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.certificatemanager.AWSCertificateManagerClient;
import com.amazonaws.services.certificatemanager.AWSCertificateManager;
import
 com.amazonaws.services.certificatemanager.model.DeleteCertificateRequest;
import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.AmazonClientException;

 /**
  * This sample demonstrates how to use the DeleteCertificate function in the
  * AWS Certificate Manager service. This function deletes the certificate
  * specified by its ARN.
  *
  * Input Parameters:
  *    CertificateArn - String that contains the Amazon Resource Name (ARN)
 of
  *                     the certificate to be deleted.
  */
public class AWSCertificateManagerSample {

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

        // Retrieve the user's access key ID and secret access key.
        AWSCredentials credentials = null;
        try {
            credentials = new ProfileCredentialsProvider().getCredentials();
        } catch (Exception e) {
            throw new AmazonClientException(
                    "Cannot load the credentials from the credential profiles
 file. " +
                    "Please make sure that your credentials file is at the
 correct " +
                    "location (~/.aws/credentials in Linux or C:\\Users\
\your_user_name\\.aws" +
                    "in Windows), and is in a valid format.",
                    e);
        }

        // Create an AWSCertificateManager client and set the region.
        AWSCertificateManager acm = new
 AWSCertificateManagerClient(credentials);
        Region usEast1 = Region.getRegion(Regions.US_EAST_1);
        acm.setRegion(usEast1);

        // Specify the certificate ARN and call the deleteCertificate()
 function.
        String certARN = "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-12345678";
        DeleteCertificateRequest req = new
 DeleteCertificateRequest().withCertificateArn(certARN);
        acm.deleteCertificate(req);
    }
}
```

Describing a Certificate

The following example shows how to use the DescribeCertificate function.

```


package com.amazonaws.samples;

import com.amazonaws.regions.Region;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.certificatemanager.AWSCertificateManagerClient;
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.AWSCredentials;
import com.amazonaws.AmazonClientException;

 /**
  * This sample demonstrates how to use the DescribeCertificate function in
 the
  * AWS Certificate Manager service. This function returns a list of the
 fields
  * contained in the certificate specified by its ARN.
  *
  * Input Parameters:
  *    CertificateArn - String that contains the Amazon Resource Name (ARN)
 for
  *                     the certificate.
  */
public class AWSCertificateManagerSample {

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

        // Retrieve the user's access key ID and secret access key.
        AWSCredentials credentials = null;
        try {
            credentials = new ProfileCredentialsProvider().getCredentials();
        } catch (Exception e) {
            throw new AmazonClientException(
                    "Cannot load the credentials from the credential profiles
 file. " +
                    "Please make sure that your credentials file is at the
 correct " +
                    "location (~/.aws/credentials in Linux or C:\\Users\
\your_user_name\\.aws" +
                    "in Windows), and is in a valid format.",
                    e);
        }

        // Create an AWSCertificateManager client and set the region.
        AWSCertificateManager acm = new
 AWSCertificateManagerClient(credentials);
        Region usEast1 = Region.getRegion(Regions.US_EAST_1);
        acm.setRegion(usEast1);

        // Specify the certificate ARN and call the describeCertificate()
 function.
        String certARN = "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012";
        DescribeCertificateRequest certRequest = new
 DescribeCertificateRequest().withCertificateArn(certARN);
     DescribeCertificateResult res = acm.describeCertificate(certRequest);
        System.out.println(res);
    }
}
```

Getting a Certificate Chain

The following example shows how to use the GetCertificate function.

```
package com.amazonaws.samples;

import com.amazonaws.regions.Region;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.certificatemanager.AWSCertificateManagerClient;
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.AWSCredentials;
import com.amazonaws.AmazonClientException;

 /**
  * This sample demonstrates how to use the GetCertificate function in the
  * AWS Certificate Manager service. This function retrieves your SSL/TLS
  * certificate and certificate ARN.
  *
  * Input Parameters:
  *    CertificateArn - String that contains the ARN of the certificate to
  *                     be retrieved.
  *
  */
public class AWSCertificateManagerSample {

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

        // Retrieve the user's access key ID and secret access key.
        AWSCredentials credentials = null;
        try {
            credentials = new ProfileCredentialsProvider().getCredentials();
        } catch (Exception e) {
            throw new AmazonClientException(
                    "Cannot load the credentials from the credential profiles
 file. " +
                    "Please make sure that your credentials file is at the
 correct " +
                    "location (~/.aws/credentials in Linux or C:\\Users\
\your_user_name\\.aws" +
                    "in Windows), and is in a valid format.",
                    e);
        }

        // Create an AWSCertificateManager client and set the region.
        AWSCertificateManager acm = new
 AWSCertificateManagerClient(credentials);
        Region usEast1 = Region.getRegion(Regions.US_EAST_1);
        acm.setRegion(usEast1);

        // Specify the certificate ARN and call the getCertificate()
 function.
        String certARN = "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012";
        GetCertificateRequest req = new
 GetCertificateRequest().withCertificateArn(certARN);
     GetCertificateResult res = acm.getCertificate(req);
        System.out.println(res);
    }
}
```

Listing ACM Certificates

The following example shows how to use the ListCertificates function.

```
package com.amazonaws.samples;

import com.amazonaws.regions.Region;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.certificatemanager.AWSCertificateManagerClient;
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.AWSCredentials;
import com.amazonaws.AmazonClientException;

 /**
  * This sample demonstrates how to use the ListCertificates function in the
  * AWS Certificate Manager service. This function retrieves a list of your
  * certificate ARNS and their associated domain names.
  *
  * Input Parameters: - None
  *
  */
public class AWSCertificateManagerSample {

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

        // Retrieve the user's access key ID and secret access key.
        AWSCredentials credentials = null;
        try {
            credentials = new ProfileCredentialsProvider().getCredentials();
        } catch (Exception e) {
            throw new AmazonClientException(
                    "Cannot load the credentials from the credential profiles
 file. " +
                    "Please make sure that your credentials file is at the
 correct " +
                    "location (~/.aws/credentials in Linux or C:\\Users\
\your_user_name\\.aws" +
                    "in Windows), and is in a valid format.",
                    e);
        }

        // Create an AWSCertificateManager client and set the region.
        AWSCertificateManager acm = new
 AWSCertificateManagerClient(credentials);
        Region usEast1 = Region.getRegion(Regions.US_EAST_1);
        acm.setRegion(usEast1);
        // Call the listCertificates function.
        ListCertificatesResult res = acm.listCertificates(new
 ListCertificatesRequest());
        System.out.println(res);
    }
}
```

Requesting a Certificate

The following example shows how to use the RequestCertificate function.

```

package com.amazonaws.samples;

import com.amazonaws.regions.Region;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.certificatemanager.AWSCertificateManagerClient;
import com.amazonaws.services.certificatemanager.AWSCertificateManager;
import
 com.amazonaws.services.certificatemanager.model.RequestCertificateRequest;
import
 com.amazonaws.services.certificatemanager.model.RequestCertificateResult;
import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.AmazonClientException;

 /**
  * This sample demonstrates how to use the RequestCertificate function in
 the
  * AWS Certificate Manager service. This function requests an ACM SSL/TLS
  * certificate for your AWS account.
  *
  * Input Parameters:
  *    CertificateArn - String that contains the Amazon Resource Name (ARN)
 for
  *                    the certificate.
  */

public class AWSCertificateManagerSample {

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

        // Retrieve the user's access key ID and secret access key.
        AWSCredentials credentials = null;
        try {
            credentials = new ProfileCredentialsProvider().getCredentials();
        } catch (Exception e) {
            throw new AmazonClientException(
                    "Cannot load the credentials from the credential profiles
 file. " +
                    "Please make sure that your credentials file is at the
 correct " +
                    "location (~/.aws/credentials in Linux or C:\\Users\
\your_user_name\\.aws" +
                    "in Windows), and is in a valid format.",
                    e);
        }

        // Create an AWSCertificateManager client and set the region.
        AWSCertificateManager acm = new
 AWSCertificateManagerClient(credentials);
        Region usEast1 = Region.getRegion(Regions.US_EAST_1);
        acm.setRegion(usEast1);

        // Specify the arguments and call the requestCertificate() function.
        String domainName = "www.example.com";
        String idempotencyToken = "1AqO5pTy";
        String[] SAN = {"www.example.net"};
        RequestCertificateRequest req = new
 RequestCertificateRequest().withDomainName(domainName).withIdempotencyToken(idempotencyTok
        RequestCertificateResult res = acm.requestCertificate(req);
        System.out.println(res);

    }
}
```

Resending Validation Email

The following example shows you how to use the ResendValidationEmail function.

```
package com.amazonaws.samples;

import com.amazonaws.regions.Region;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.certificatemanager.AWSCertificateManagerClient;
import com.amazonaws.services.certificatemanager.AWSCertificateManager;
import
 com.amazonaws.services.certificatemanager.model.ResendValidationEmailRequest;
import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.AmazonClientException;

 /**
  * This sample demonstrates how to use the ResendValidationEmail function in
 the
  * AWS Certificate Manager service. This function resend the email that you
 can
  * use to validate.
  *
  * Input Parameters:
  *    CertificateArn - String that contains the Amazon Resource Name (ARN)
 for
  *                     the certificate.
  */
public class AWSCertificateManagerSample {

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

        // Retrieve the user's access key ID and secret access key.
        AWSCredentials credentials = null;
        try {
            credentials = new ProfileCredentialsProvider().getCredentials();
        } catch (Exception e) {
            throw new AmazonClientException(
                    "Cannot load the credentials from the credential profiles
 file. " +
                    "Please make sure that your credentials file is at the
 correct " +
                    "location (~/.aws/credentials in Linux or C:\\Users\
\your_user_name\\.aws" +
                    "in Windows), and is in a valid format.",
                    e);
        }

        // Create an AWSCertificateManager client and set the region.
        AWSCertificateManager acm = new
 AWSCertificateManagerClient(credentials);
        Region usEast1 = Region.getRegion(Regions.US_EAST_1);
        acm.setRegion(usEast1);

        // Specify the input parameters and call the resendValidationEmail()
 function.
        String certARN = "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012";
        String domain = "example.com";
        String validationDomain = "example.com";
        ResendValidationEmailRequest req = new
 ResendValidationEmailRequest().withCertificateArn(certARN).withDomain(domain).withValidati
        acm.resendValidationEmail(req);
    }
}
```

ACM Private Key Security

When you request a certificate (p. 11), AWS Certificate Manager (ACM) generates a public/private key pair. For imported certificates (p. 23), you generate the key pair. The public key becomes part of the certificate. ACM stores the certificate and its corresponding private key, and uses AWS Key Management Service (AWS KMS) to help protect the private key. The process works like this:

1. The first time you request or import a certificate in an AWS region, ACM creates an AWS-managed customer master key (CMK) in AWS KMS with the alias **aws/acm**. This CMK is unique in each AWS account and each AWS region.
2. ACM uses this CMK to encrypt the certificate's private key. ACM stores only an encrypted version of the private key (ACM does not store the private key in plaintext form). ACM uses the same CMK to encrypt the private keys for all certificates in a specific AWS account and a specific AWS region.
3. When you associate the certificate with an Elastic Load Balancing load balancer or an Amazon CloudFront distribution, ACM sends the certificate and the encrypted private key to the load balancer or distribution. You also implicitly create a grant in AWS KMS that allows the load balancer or distribution to use the CMK in AWS KMS to decrypt that specific certificate's private key. For more information about grants, see Using Grants in the *AWS Key Management Service Developer Guide*.
4. The load balancer or distribution uses the CMK in AWS KMS to decrypt the private key. Then the load balancer or distribution uses the certificate and the decrypted (plaintext) private key to establish secure communication channels (SSL/TLS sessions) with its clients.
5. When the certificate is disassociated from the load balancer or distribution, the grant (created at step 3) is retired. This means the load balancer or distribution can no longer use the CMK in AWS KMS to decrypt that specific certificate's private key.

Troubleshooting

Consult the following information if you encounter problems using AWS Certificate Manager.

Topics

- Not Receiving Validation Email (p. 58)
- Email Sent to Subdomain (p. 59)
- Certificate Request Timed Out (p. 59)
- Certificate Request Failed (p. 59)
- Validation Not Complete (p. 61)

Not Receiving Validation Email

When you request a certificate from ACM, domain validation email is sent to three contact addresses specified in WHOIS and to five common administrative addresses. For more information, see Validate Domain Ownership (p. 12). If you are experiencing problems receiving validation email, review the suggestions that follow.

Where to look for email

Validation email is sent to contact addresses listed in WHOIS and to common administrative addresses for the domain. Email is not sent to the AWS account owner unless the owner is also listed as a domain contact in WHOIS. Review the list of email addresses that are displayed in the ACM console (or returned from the CLI or API) to determine where you should be looking for validation email. To see the list, click the icon next to the domain name in the box labeled **Validation not complete**.

The email is marked as spam

Check your spam folder for the validation email.

GMail automatically sorts your email

If you are using GMail, the validation email may have been automatically sorted into the **Updates** or **Promotions** tabs.

The domain registrar does not display contact information or privacy protection is enabled

In some cases, the domain registrant, technical, and administrative contacts in WHOIS may not be publicly available, and AWS therefore cannot reach these contacts. At your discretion, you can choose to configure your registrar to list your email address in WHOIS, although not all registrars support this option. You may be required to make a change directly at your domain's registry. In other cases, the domain contact information may be using a privacy address, such as those provided through WhoisGuard or PrivacyGuard. For domains purchased from Amazon Route 53, privacy protection is enabled by default and your email address is mapped to a `whoisprivacyservice.org` or `contact.gandi.net` email address. Ensure that your registrant email address on file with your domain registrar is up to date so that the email sent to these obscured email addresses can be forwarded to an email address that you control.

If email contact information for your domain is not available through WHOIS, or if email sent to the contact information does not reach the domain owner or an authorized representative, we recommend that you configure your domain or subdomain to receive email sent to one or more of the common administrative addresses formed by prepending admin@, administrator@, hostmaster@, webmaster@, and postmaster@ to the requested domain name. For more information about configuring email for your domain, see the documentation for your email service provider and follow the instructions at Configure Email for Your Domain (p. 9). If you are using Amazon WorkMail, see Working with Users in the Amazon WorkMail Administrator Guide.

After making available at least one of the eight email addresses to which AWS sends validation email and confirming that you can receive email for that address, you are ready to request a certificate through ACM. After you make a certificate request, ensure the intended email address appears in the list of email addresses in the AWS Management Console. While the certificate is in the **Pending validation** state, you can expand the list to view it by clicking the icon next to the domain name in the box labeled **Validation not complete**. You can also view the list in **Step 3: Validate** of the ACM **Request a Certificate** wizard. The listed email addresses are the ones to which email was sent.

Contact the Support Center

If, after reviewing the preceding guidance, you still don't receive the domain validation email, please visit the AWS Support Center and create a case. If you don't have a support agreement, post a message to the ACM Discussion Forum.

Email Sent to Subdomain

If you request a certificate for a subdomain name such as `sub.test.example.com`, then ACM checks to see if there is an MX record for `sub.test.example.com`. If not, then the parent domain `test.example.com` is checked, and so on, up to the base domain `example.com`. If an MX record is found, the search stops and a validation email is sent to the common administration addresses for the subdomain. So if an MX record is found for `test.example.com` then an email is sent to admin@test.example.com, administrator@test.example.com, and the other administrative addresses specified in Validate Domain Ownership (p. 12). If an MX record is not found in any of the subdomains, then no email is sent. To have the email instead sent directly to the apex domain, such as example.com, specify the `ValidationDomain` option in theRequestCertificate API or the request-certificate AWS CLI command. This functionality is not currently supported in the console.

Certificate Request Timed Out

Requests for ACM Certificates time out if they are not validated within 72 hours. To correct this condition, delete your request and choose **Request a certificate** to begin again. For more information about how to approve a certificate request, see Validate Domain Ownership (p. 12).

Certificate Request Failed

A request for an ACM Certificate can fail. If that happens, the following explanations can help you understand why the request failed and suggest steps you can take to fix the problem.

Failure Reasons

- No Available Contacts (p. 60)

- Domain Not Allowed (p. 60)
- Additional Verification Required (p. 60)
- Invalid Public Domain (p. 61)
- Other (p. 61)

No Available Contacts

ACM could not find an email address to use for validating one or more of the domain names in the certificate request. To correct this problem, you can do one of the following:

- Ensure that you have a working email address that is registered in WHOIS and that the address is visible when performing a standard WHOIS lookup for the domain names in the certificate request. Typically, you do this through your domain registrar.
- Ensure your domain is configured to receive email. Your domain's name server must have a mail exchanger record (MX record) so ACM's email servers know where to send the domain validation email (p. 12).

Accomplishing one of the preceding tasks is enough to correct this problem; you don't need to do both. After you correct the problem, request a new certificate. You cannot resubmit a failed certificate request.

For more information about how to ensure that you receive domain validation emails from ACM, see Configure Email for Your Domain (p. 9) or Not Receiving Validation Email (p. 58). If you follow these steps and continue to get the **No Available Contacts** message, then report this to AWS so that we can investigate it.

Domain Not Allowed

ACM does not allow certificate requests for one or more of the domain names in the certificate request. Typically, this is because one or more of the domain names in the certificate request was found in the Google Safe Browsing list of unsafe websites or the PhishTank list of valid phishes. To correct this problem, you can do the following:

- Search for your domain name at the Google Safe Browsing Site Status website. If your domain is considered unsafe, see Google Help for Hacked Websites to learn what you can do. If you think your domain is safe, see Request a review to request a review from Google.
- Search for your domain name on the PhishTank home page. If your domain is considered a phish, see Google Help for Hacked Websites or StopBadware Webmaster Help to learn what you can do. If you think your domain is safe, see the PhishTank FAQ for information about how to report a false positive.

After you correct the problem, request a new certificate. You cannot resubmit a failed certificate request.

Additional Verification Required

ACM requires additional information to process this certificate request. To provide this information, use the Support Center to contact AWS Support. If you don't have a support plan, post a new thread in the AWS Certificate Manager discussion forum.

> **Note**
> You cannot request a certificate for Amazon-owned domain names such as those ending in amazonaws.com, cloudfront.net, or elasticbeanstalk.com. This failure reason occurs when your certificate request includes these domain names.

Invalid Public Domain

One or more of the domain names in the certificate request is not valid. Typically, this is because a domain name in the request is not a valid top-level domain. Try to request a certificate again, correcting any spelling errors or typos that were in the failed request, and ensuring that all domain names in the request are for valid top-level domains. For example, you cannot request an ACM Certificate for example.invalidpublicdomain because "invalidpublicdomain" is not a valid top-level domain. If you continue to receive this failure reason, use the Support Center to contact AWS Support. If you don't have a support plan, post a new thread in the AWS Certificate Manager discussion forum.

Other

Typically, this failure occurs when there is a typographical error in one or more of the domain names in the certificate request. Try to request a certificate again, correcting any spelling errors or typos that were in the failed request. If you continue to receive this failure reason, use the Support Center to contact AWS Support. If you don't have a support plan, post a new thread in the AWS Certificate Manager discussion forum.

Validation Not Complete

If the ACM Certificate request status is **Pending validation**, the request is awaiting approval. To approve the request, the authorized representative must respond to the validation email sent to the registered WHOIS contact addresses and other common email addresses for the requested domain. For more information about how to approve a request, see Validate Domain Ownership (p. 12).

> **Important**
> If your request includes more than one domain name in the certificate, then you must approve every domain name that you included. If you do not receive a validation email for each domain name included in the request, then see Not Receiving Validation Email (p. 58).

Document History

The following table describes the documentation release history of AWS Certificate Manager.

Latest documentation update: October 13, 2016

Change	Description	Release Date
New content	Added documentation about Importing Certificates (p. 23).	October 13, 2016
New content	Added AWS CloudTrail support for ACM actions. See Logging AWS Certificate Manager API Calls with AWS CloudTrail (p. 35).	March 25, 2016
New guide	This release introduces AWS Certificate Manager.	January 21, 2016