AWS Certificate Manager

API Reference

API Version 2015-12-08

AWS Certificate Manager: API Reference

Table of Contents

Welcome

Welcome to the AWS Certificate Manager (ACM) API Reference. This guide provides descriptions, syntax, and usage examples for each ACM API operation.

You can use ACM to manage SSL/TLS certificates for your AWS-based websites and applications. For general information about using ACM, see the AWS Certificate Manager User Guide.

Instead of using the ACM HTTP API directly, you can use one of the AWS SDKs or command line tools to interact with the ACM API. These tools are available for a variety of programming languages and platforms. For more information, see Tools for Amazon Web Services.

Signing API Requests

You must sign your HTTP API requests to ACM. When you use the AWS SDKs and command line tools, they sign API requests for you. If you do not use these tools, you must calculate the signature yourself. For more information, see Signature Version 4 Signing Process in the *Amazon Web Services General Reference*. ACM requires signature version 4.

Actions

The following actions are supported:

AddTagsToCertificate

Adds one or more tags to an ACM certificate. Tags are labels that you can use to identify and organize your AWS resources. Each tag consists of a `key` and an optional `value`. You specify the certificate on input by its Amazon Resource Name (ARN). You specify the tag by using a key-value pair.

You can apply a tag to just one certificate if you want to identify a specific characteristic of that certificate, or you can apply the same tag to multiple certificates if you want to filter for a common relationship among those certificates. Similarly, you can apply the same tag to multiple resources if you want to specify a relationship among those resources. For example, you can add the same tag to an ACM certificate and an Elastic Load Balancing load balancer to indicate that they are both used by the same website. For more information, see Tagging ACM certificates.

To remove one or more tags, use the RemoveTagsFromCertificate (p. 32) action. To view all of the tags that have been applied to the certificate, use the ListTagsForCertificate (p. 29) action.

Request Syntax

```
{
   "CertificateArn": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 64).

The request accepts the following data in JSON format.

Note
In the following list, the required parameters are described first.

CertificateArn (p. 3)

String that contains the ARN of the ACM certificate to which the tag is to be applied. This must be of the form:

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

For more information about ARNs, see Amazon Resource Names (ARNs) and AWS Service Namespaces.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=,.@-]+(/[\w+=,.@-]+)*`

Required: Yes

Tags (p. 3)

The key-value pair that defines the tag. The tag value is optional.

Type: Array of Tag (p. 63) objects

Array Members: Minimum number of 1 item. Maximum number of 50 items.

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 66).

InvalidArnException

The requested Amazon Resource Name (ARN) does not refer to an existing resource.

HTTP Status Code: 400

InvalidTagException

One or both of the values that make up the key-value pair is not valid. For example, you cannot specify a tag value that begins with `aws:`.

HTTP Status Code: 400

ResourceNotFoundException

The specified certificate cannot be found in the caller's account or the caller's account cannot be found.

HTTP Status Code: 400

TooManyTagsException

The request contains too many tags. Try the request again with fewer tags.

HTTP Status Code: 400

Example

Add two tags to an ACM certificate

Sample Request

```
POST / HTTP/1.1
Host: acm.us-east-1.amazonaws.com
X-Amz-Target: CertificateManager.AddTagsToCertificate
X-Amz-Date: 20160414T162438Z
User-Agent: aws-cli/1.10.20 Python/2.7.3 Linux/3.13.0-83-generic botocore/1.4.11
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20160414/us-east-1/acm/
aws4_request,
```

```
SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-target,
Signature=370a583d3532f14e0cb34ea51de782e9e5138171184bfede740f5f150251fa2f

{
  "CertificateArn": "arn:aws:acm:us-
east-1:111122223333:certificate/12345678-1234-1234-1234-123456789012",
  "Tags": [{
    "Key": "website",
    "Value": "example.com"
  },
  {
    "Key": "stack",
    "Value": "production"
  }]
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 640bd601-025d-11e6-baa2-cd9f4ef8cda6
Content-Type: application/x-amz-json-1.1
Content-Length: 0
Date: Thu, 14 Apr 2016 16:24:41 GMT
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

DeleteCertificate

Deletes a certificate and its associated private key. If this action succeeds, the certificate no longer appears in the list that can be displayed by calling the ListCertificates (p. 25) action or be retrieved by calling the GetCertificate (p. 18) action. The certificate will not be available for use by AWS services integrated with ACM.

Note
You cannot delete an ACM certificate that is being used by another AWS service. To delete a certificate that is in use, the certificate association must first be removed.

Request Syntax

```
{
   "CertificateArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 64).

The request accepts the following data in JSON format.

Note
In the following list, the required parameters are described first.

CertificateArn (p. 6)

String that contains the ARN of the ACM certificate to be deleted. This must be of the form:

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

For more information about ARNs, see Amazon Resource Names (ARNs) and AWS Service Namespaces.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=,.@-]+(/[\w+=,.@-]+)*`

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 66).

InvalidArnException

The requested Amazon Resource Name (ARN) does not refer to an existing resource.

HTTP Status Code: 400

ResourceInUseException

The certificate is in use by another AWS service in the caller's account. Remove the association and try again.

HTTP Status Code: 400

ResourceNotFoundException

The specified certificate cannot be found in the caller's account or the caller's account cannot be found.

HTTP Status Code: 400

Example

Delete an ACM certificate

Sample Request

```
POST / HTTP/1.1
Host: acm.us-east-1.amazonaws.com
X-Amz-Target: CertificateManager.DeleteCertificate
X-Amz-Date: 20151222T164207Z
User-Agent: aws-cli/1.9.7 Python/2.7.3 Linux/3.13.0-73-generic botocore/1.3.7
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20151222/us-east-1/acm/
aws4_request,
SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-target,
Signature=0b29b04bb5f1ebb5fe9e6b1cbcdeda903b4ed2e06f3abe8a092c0ed1193b4dfc

{
  "CertificateArn": "arn:aws:acm:us-
east-1:111122223333:certificate/12345678-1234-1234-1234-123456789012"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: ee2db085-a8ca-11e5-9561-b3f6248b5775
Content-Type: application/x-amz-json-1.1
Content-Length: 0
Date: Tue, 22 Dec 2015 16:42:03 GMT
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript

- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

DescribeCertificate

Returns detailed metadata about the specified ACM certificate.

Request Syntax

```
{
   "CertificateArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 64).

The request accepts the following data in JSON format.

Note
In the following list, the required parameters are described first.

CertificateArn (p. 9)

The Amazon Resource Name (ARN) of the ACM certificate. The ARN must have the following form:

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

For more information about ARNs, see Amazon Resource Names (ARNs) and AWS Service Namespaces.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=,.@-]+(/[\w+=,.@-]+)*`

Required: Yes

Response Syntax

```
{
   "Certificate": {
      "CertificateArn": "string",
      "CertificateAuthorityArn": "string",
      "CreatedAt": number,
      "DomainName": "string",
      "DomainValidationOptions": [
         {
            "DomainName": "string",
            "ResourceRecord": {
               "Name": "string",
               "Type": "string",
               "Value": "string"
            },
            "ValidationDomain": "string",
            "ValidationEmails": [ "string" ],
```

```
            "ValidationMethod": "string",
            "ValidationStatus": "string"
         }
      ],
      "ExtendedKeyUsages": [
         {
            "Name": "string",
            "OID": "string"
         }
      ],
      "FailureReason": "string",
      "ImportedAt": number,
      "InUseBy": [ "string" ],
      "IssuedAt": number,
      "Issuer": "string",
      "KeyAlgorithm": "string",
      "KeyUsages": [
         {
            "Name": "string"
         }
      ],
      "NotAfter": number,
      "NotBefore": number,
      "Options": {
         "CertificateTransparencyLoggingPreference": "string"
      },
      "RenewalEligibility": "string",
      "RenewalSummary": {
         "DomainValidationOptions": [
            {
               "DomainName": "string",
               "ResourceRecord": {
                  "Name": "string",
                  "Type": "string",
                  "Value": "string"
               },
               "ValidationDomain": "string",
               "ValidationEmails": [ "string" ],
               "ValidationMethod": "string",
               "ValidationStatus": "string"
            }
         ],
         "RenewalStatus": "string"
      },
      "RevocationReason": "string",
      "RevokedAt": number,
      "Serial": "string",
      "SignatureAlgorithm": "string",
      "Status": "string",
      "Subject": "string",
      "SubjectAlternativeNames": [ "string" ],
      "Type": "string"
   }
}
```


Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

Certificate (p. 9)

Metadata about an ACM certificate.

Type: CertificateDetail (p. 48) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 66).

InvalidArnException

The requested Amazon Resource Name (ARN) does not refer to an existing resource.

HTTP Status Code: 400

ResourceNotFoundException

The specified certificate cannot be found in the caller's account or the caller's account cannot be found.

HTTP Status Code: 400

Example

Describe an ACM Certificate

Sample Request

```
POST / HTTP/1.1
Host: acm.us-east-1.amazonaws.com
X-Amz-Target: CertificateManager.DescribeCertificate
X-Amz-Date: 20151221T203246Z
User-Agent: aws-cli/1.9.7 Python/2.7.3 Linux/3.13.0-71-generic botocore/1.3.7
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151221/us-east-1/acm/
aws4_request,
SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-target,
Signature=76913a7d6013d34afbdc1bbd6c3e77d5edd3fa2d9883a94d946c6eeea5908d9e

{
  "CertificateArn": "arn:aws:acm:us-
east-1:111122223333:certificate/12345678-1234-1234-1234-123456789012"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: fd1e5a07-a821-11e5-845d-95c070464235
Content-Type: application/x-amz-json-1.1
Content-Length: 1035
Date: Mon, 21 Dec 2015 20:32:43 GMT

{
  "Certificate": {
    "CertificateArn": "arn:aws:acm:us-
east-1:111122223333:certificate/12345678-1234-1234-1234-123456789012",
    "CreatedAt": 1450212224.0,
    "DomainName": "example.com",
    "DomainValidationOptions": [
      {
```

```
        "DomainName": "example.com",
        "ValidationDomain": "example.com",
        "ValidationEmails": [
          "hostmaster@example.com",
          "admin@example.com",
          "admin@example.com.whoisprivacyservice.org",
          "tech@example.com.whoisprivacyservice.org",
          "owner@example.com.whoisprivacyservice.org",
          "postmaster@example.com",
          "webmaster@example.com",
          "administrator@example.com"
        ]
      },
      {
        "DomainName": "www.example.com",
        "ValidationDomain": "www.example.com",
        "ValidationEmails": [
          "hostmaster@example.com",
          "admin@example.com",
          "admin@example.com.whoisprivacyservice.org",
          "tech@example.com.whoisprivacyservice.org",
          "owner@example.com.whoisprivacyservice.org",
          "postmaster@example.com",
          "webmaster@example.com",
          "administrator@example.com"
        ]
      }
    ],
    "InUseBy": [
      "arn:aws:cloudfront::111122223333:distribution/E12KXPQHVLSYVC"
    ],
    "IssuedAt": 1450212292.0,
    "Issuer": "Amazon",
    "KeyAlgorithm": "RSA-2048",
    "NotAfter": 1484481600.0,
    "NotBefore": 1450137600.0,
    "Serial": "07:71:71:f4:6b:e7:bf:63:87:e6:ad:3c:b2:0f:d0:5b",
    "SignatureAlgorithm": "SHA256WITHRSA",
    "Status": "ISSUED",
    "Subject": "CN=example.com",
    "SubjectAlternativeNames": [
      "example.com",
      "www.example.com"
    ]
  }
}
```


See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

ExportCertificate

Exports a private certificate issued by a private certificate authority (CA) for use anywhere. You can export the certificate, the certificate chain, and the encrypted private key associated with the public key embedded in the certificate. You must store the private key securely. The private key is a 2048 bit RSA key. You must provide a passphrase for the private key when exporting it. You can use the following OpenSSL command to decrypt it later. Provide the passphrase when prompted.

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

Request Syntax

```
{
   "CertificateArn": "string",
   "Passphrase": blob
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 64).

The request accepts the following data in JSON format.

Note
In the following list, the required parameters are described first.

CertificateArn (p. 14)

An Amazon Resource Name (ARN) of the issued certificate. This must be of the form:

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

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=,.@-]+(/[\w+=,.@-]+)*`

Required: Yes

Passphrase (p. 14)

Passphrase to associate with the encrypted exported private key. If you want to later decrypt the private key, you must have the passphrase. You can use the following OpenSSL command to decrypt a private key:

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

Type: Base64-encoded binary data object

Length Constraints: Minimum length of 4. Maximum length of 128.

Required: Yes

Response Syntax

```
{
```

```
   "Certificate": "string",
   "CertificateChain": "string",
   "PrivateKey": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

Certificate (p. 14)

The base64 PEM-encoded certificate.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 32768.

Pattern: `-{5}BEGIN CERTIFICATE-{5}\u000D?\u000A([A-Za-z0-9/+]{64}\u000D?\u000A)*[A-Za-z0-9/+]{1,64}={0,2}\u000D?\u000A-{5}END CERTIFICATE-{5}(\u000D?\u000A)?`

CertificateChain (p. 14)

The base64 PEM-encoded certificate chain. This does not include the certificate that you are exporting.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2097152.

Pattern: `(-{5}BEGIN CERTIFICATE-{5}\u000D?\u000A([A-Za-z0-9/+]{64}\u000D?\u000A)*[A-Za-z0-9/+]{1,64}={0,2}\u000D?\u000A-{5}END CERTIFICATE-{5}\u000D?\u000A)*-{5}BEGIN CERTIFICATE-{5}\u000D?\u000A([A-Za-z0-9/+]{64}\u000D?\u000A)*[A-Za-z0-9/+]{1,64}={0,2}\u000D?\u000A-{5}END CERTIFICATE-{5}(\u000D?\u000A)?`

PrivateKey (p. 14)

The encrypted private key associated with the public key in the certificate. The key is output in PKCS #8 format and is base64 PEM-encoded.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 524288.

Pattern: `-{5}BEGIN PRIVATE KEY-{5}\u000D?\u000A([A-Za-z0-9/+]{64}\u000D?\u000A)*[A-Za-z0-9/+]{1,64}={0,2}\u000D?\u000A-{5}END PRIVATE KEY-{5}(\u000D?\u000A)?`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 66).

InvalidArnException

The requested Amazon Resource Name (ARN) does not refer to an existing resource.

HTTP Status Code: 400

RequestInProgressException

The certificate request is in process and the certificate in your account has not yet been issued.

HTTP Status Code: 400

ResourceNotFoundException

The specified certificate cannot be found in the caller's account or the caller's account cannot be found.

HTTP Status Code: 400

Example

Sample Request

```
POST / HTTP/1.1
Host: acm.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 135
X-Amz-Target: CertificateManager.ExportCertificate
X-Amz-Date: 20180331T175638Z
User-Agent: aws-cli/1.14.28 Python/2.7.9 Windows/8 botocore/1.8.32
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=key_ID/20180331/us-east-1/acm/aws4_request,
SignedHeaders=content-type;host;x-amz-date;x-amz-target,
Signature=7b3f783da1b701aea1b6b49dea7d5194d7e2b253f152cfb939459ba3b0ba2c1d

{
  "CertificateArn": "arn:aws:acm:us-
east-1:account:certificate/12345678-1234-1234-1234-1234556789012",
  "Passphrase": "cGFzc3dvcmQ="
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: dd520651-350c-11e8-a99a-c76ec78904bf
Content-Type: application/x-amz-json-1.1
Content-Length: 5860
Date: Sat, 31 Mar 2018 17:56:41 GMT
Connection: Keep-alive

{
  "Certificate":
    "-----BEGIN CERTIFICATE-----Base64-encodedEND CERTIFICATE-----",
  "CertificateChain":
    "-----BEGIN CERTIFICATE-----Base64-encodedEND CERTIFICATE-----
     -----BEGIN CERTIFICATE-----Base64-encodedEND CERTIFICATE-----",
  "PrivateKey":
    "-----BEGIN ENCRYPTED PRIVATE KEYBase64-encoded-----END ENCRYPTED PRIVATE KEY-----"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

GetCertificate

Retrieves a certificate specified by an ARN and its certificate chain . The chain is an ordered list of certificates that contains the end entity certificate, intermediate certificates of subordinate CAs, and the root certificate in that order. The certificate and certificate chain are base64 encoded. If you want to decode the certificate to see the individual fields, you can use OpenSSL.

Request Syntax

```
{
   "CertificateArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 64).

The request accepts the following data in JSON format.

Note
In the following list, the required parameters are described first.

CertificateArn (p. 18)

String that contains a certificate ARN in the following format:

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

For more information about ARNs, see Amazon Resource Names (ARNs) and AWS Service Namespaces.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=,.@-]+(/[\w+=,.@-]+)*`

Required: Yes

Response Syntax

```
{
   "Certificate": "string",
   "CertificateChain": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

Certificate (p. 18)

String that contains the ACM certificate represented by the ARN specified at input.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 32768.

Pattern: `-{5}BEGIN CERTIFICATE-{5}\u000D?\u000A([A-Za-z0-9/+]{64}\u000D?\u000A)*[A-Za-z0-9/+]{1,64}={0,2}\u000D?\u000A-{5}END CERTIFICATE-{5}(\u000D?\u000A)?`

CertificateChain (p. 18)

The certificate chain that contains the root certificate issued by the certificate authority (CA).

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2097152.

Pattern: `(-{5}BEGIN CERTIFICATE-{5}\u000D?\u000A([A-Za-z0-9/+]{64}\u000D?\u000A)*[A-Za-z0-9/+]{1,64}={0,2}\u000D?\u000A-{5}END CERTIFICATE-{5}\u000D?\u000A)*-{5}BEGIN CERTIFICATE-{5}\u000D?\u000A([A-Za-z0-9/+]{64}\u000D?\u000A)*[A-Za-z0-9/+]{1,64}={0,2}\u000D?\u000A-{5}END CERTIFICATE-{5}(\u000D?\u000A)?`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 66).

InvalidArnException

The requested Amazon Resource Name (ARN) does not refer to an existing resource.

HTTP Status Code: 400

RequestInProgressException

The certificate request is in process and the certificate in your account has not yet been issued.

HTTP Status Code: 400

ResourceNotFoundException

The specified certificate cannot be found in the caller's account or the caller's account cannot be found.

HTTP Status Code: 400

Example

Get an ACM Certificate

Sample Request

```
POST / HTTP/1.1
Host: acm.us-east-1.amazonaws.com
X-Amz-Target: CertificateManager.GetCertificate
X-Amz-Date: 20151221T210018Z
```

```
User-Agent: aws-cli/1.9.7 Python/2.7.3 Linux/3.13.0-71-generic botocore/1.3.7
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151221/us-east-1/acm/
aws4_request,
SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-target,
Signature=b51b4c2d5518473a8552fdab8e313c76254e9ca64e4d8ab69c2ebef83dbd459b

{
  "CertificateArn": "arn:aws:acm:us-
east-1:111122223333:certificate/12345678-1234-1234-1234-123456789012"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: d5300b5a-a825-11e5-9141-fbb8a078e3eb
Content-Type: application/x-amz-json-1.1
Content-Length: 6506
Date: Mon, 21 Dec 2015 21:00:15 GMT

{
  "Certificate":
    "------BEGIN CERTIFICATE-----Base64-encoded-----END CERTIFICATE-----",
  "CertificateChain":
    "------BEGIN CERTIFICATE-----Base64-encoded-----END CERTIFICATE-----"
    "------BEGIN CERTIFICATE-----Base64-encoded-----END CERTIFICATE-----"
    "------BEGIN CERTIFICATE-----Base64-encoded-----END CERTIFICATE-----"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

ImportCertificate

Imports a certificate into AWS Certificate Manager (ACM) to use with services that are integrated with ACM. Note that integrated services allow only certificate types and keys they support to be associated with their resources. Further, their support differs depending on whether the certificate is imported into IAM or into ACM. For more information, see the documentation for each service. For more information about importing certificates into ACM, see Importing Certificates in the *AWS Certificate Manager User Guide*.

> **Note**
> ACM does not provide managed renewal for certificates that you import.

Note the following guidelines when importing third party certificates:

- You must enter the private key that matches the certificate you are importing.
- The private key must be unencrypted. You cannot import a private key that is protected by a password or a passphrase.
- If the certificate you are importing is not self-signed, you must enter its certificate chain.
- If a certificate chain is included, the issuer must be the subject of one of the certificates in the chain.
- The certificate, private key, and certificate chain must be PEM-encoded.
- The current time must be between the `Not Before` and `Not After` certificate fields.
- The `Issuer` field must not be empty.
- The OCSP authority URL, if present, must not exceed 1000 characters.
- To import a new certificate, omit the `CertificateArn` argument. Include this argument only when you want to replace a previously imported certificate.
- When you import a certificate by using the CLI, you must specify the certificate, the certificate chain, and the private key by their file names preceded by `file://`. For example, you can specify a certificate saved in the `C:\temp` folder as `file://C:\temp\certificate_to_import.pem`. If you are making an HTTP or HTTPS Query request, include these arguments as BLOBs.
- When you import a certificate by using an SDK, you must specify the certificate, the certificate chain, and the private key files in the manner required by the programming language you're using.

This operation returns the Amazon Resource Name (ARN) of the imported certificate.

Request Syntax

```
{
   "Certificate": blob,
   "CertificateArn": "string",
   "CertificateChain": blob,
   "PrivateKey": blob
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 64).

The request accepts the following data in JSON format.

> **Note**
> In the following list, the required parameters are described first.

Certificate (p. 21)

The certificate to import.

Type: Base64-encoded binary data object

Length Constraints: Minimum length of 1. Maximum length of 32768.

Required: Yes

PrivateKey (p. 21)

The private key that matches the public key in the certificate.

Type: Base64-encoded binary data object

Length Constraints: Minimum length of 1. Maximum length of 524288.

Required: Yes

CertificateArn (p. 21)

The Amazon Resource Name (ARN) of an imported certificate to replace. To import a new certificate, omit this field.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=,.@-]+(/[\w+=,.@-]+)*`

Required: No

CertificateChain (p. 21)

The PEM encoded certificate chain.

Type: Base64-encoded binary data object

Length Constraints: Minimum length of 1. Maximum length of 2097152.

Required: No

Response Syntax

```
{
   "CertificateArn": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

CertificateArn (p. 22)

The Amazon Resource Name (ARN) of the imported certificate.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=,.@-]+(/[\w+=,.@-]+)*`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 66).

LimitExceededException

An ACM limit has been exceeded.

HTTP Status Code: 400

ResourceNotFoundException

The specified certificate cannot be found in the caller's account or the caller's account cannot be found.

HTTP Status Code: 400

Example

Import a certificate

Sample Request

```
POST / HTTP/1.1
Host: acm.us-east-1.amazonaws.com
X-Amz-Target: CertificateManager.ImportCertificate
X-Amz-Date: 20161011T184744Z
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=key_ID/20161011/us-east-1/acm/aws4_request,
SignedHeaders=content-type;host;x-amz-date;x-amz-target,
Signature=60f965247476c4672c498c24ba255e52a62a7e4bd8678d8ee788af5ffe42f377

{
    "CertificateChain": "Base64-encoded blob",
    "PrivateKey": "Base64-encoded blob",
    "Certificate": "Base64-encoded blob"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 32f9ab0a-8fe3-11e6-8d69-c91606b24a3f
Content-Type: application/x-amz-json-1.1
Content-Length: 104
Date: Tue, 11 Oct 2016 18:47:46 GMT

{"CertificateArn":"arn:aws:acm:us-east-1:111122223333:certificate/91228a40-
ad89-4ce0-9f6c-07009fc8fdfb"}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

ListCertificates

Retrieves a list of certificate ARNs and domain names. You can request that only certificates that match a specific status be listed. You can also filter by specific attributes of the certificate.

Request Syntax

```
{
   "CertificateStatuses": [ "string" ],
   "Includes": {
      "extendedKeyUsage": [ "string" ],
      "keyTypes": [ "string" ],
      "keyUsage": [ "string" ]
   },
   "MaxItems": number,
   "NextToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 64).

The request accepts the following data in JSON format.

Note
In the following list, the required parameters are described first.

CertificateStatuses (p. 25)

Filter the certificate list by status value.

Type: Array of strings

Valid Values: `PENDING_VALIDATION | ISSUED | INACTIVE | EXPIRED | VALIDATION_TIMED_OUT | REVOKED | FAILED`

Required: No

Includes (p. 25)

Filter the certificate list. For more information, see the Filters (p. 59) structure.

Type: Filters (p. 59) object

Required: No

MaxItems (p. 25)

Use this parameter when paginating results to specify the maximum number of items to return in the response. If additional items exist beyond the number you specify, the `NextToken` element is sent in the response. Use this `NextToken` value in a subsequent request to retrieve additional items.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 1000.

Required: No

NextToken (p. 25)

Use this parameter only when paginating results and only in a subsequent request after you receive a response with truncated results. Set it to the value of `NextToken` from the response you just received.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 320.

Pattern: `[\u0009\u000A\u000D\u0020-\u00FF]*`

Required: No

Response Syntax

```
{
   "CertificateSummaryList": [
      {
         "CertificateArn": "string",
         "DomainName": "string"
      }
   ],
   "NextToken": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

CertificateSummaryList (p. 26)

A list of ACM certificates.

Type: Array of CertificateSummary (p. 54) objects

NextToken (p. 26)

When the list is truncated, this value is present and contains the value to use for the `NextToken` parameter in a subsequent pagination request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 320.

Pattern: `[\u0009\u000A\u000D\u0020-\u00FF]*`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 66).

Example

List Certificates

The following example lists certificates that you can use to create digital signatures and to sign code.

Sample Request

```
POST / HTTP/1.1
Host: acm.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 129
X-Amz-Target: CertificateManager.ListCertificates
X-Amz-Date: 20171118T204928Z
User-Agent: aws-cli/1.11.132 Python/2.7.9 Windows/8 botocore/1.5.95
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=key_ID/20171118/us-east-1/acm/aws4_request,
SignedHeaders=content-type;host;x-amz-date;x-amz-target,
Signature=49a54...

{
 "MaxItems": 10,
 "Includes": {
  "keyUsage": ["DIGITAL_SIGNATURE"],
  "keyTypes": ["RSA_2048"],
  "extendedKeyUsage": ["CODE_SIGNING"]
 }
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: fa8ffa7f-cca1-11e7-80db-736b2201613a
Content-Type: application/x-amz-json-1.1
Content-Length: 164
Date: Sat, 18 Nov 2017 20:49:32 GMT
Connection: Keep-alive

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

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript

- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

ListTagsForCertificate

Lists the tags that have been applied to the ACM certificate. Use the certificate's Amazon Resource Name (ARN) to specify the certificate. To add a tag to an ACM certificate, use the AddTagsToCertificate (p. 3) action. To delete a tag, use the RemoveTagsFromCertificate (p. 32) action.

Request Syntax

```
{
   "CertificateArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 64).

The request accepts the following data in JSON format.

Note
In the following list, the required parameters are described first.

CertificateArn (p. 29)

String that contains the ARN of the ACM certificate for which you want to list the tags. This must have the following form:

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

For more information about ARNs, see Amazon Resource Names (ARNs) and AWS Service Namespaces.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=,.@-]+(/[\w+=,.@-]+)*`

Required: Yes

Response Syntax

```
{
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

Tags (p. 29)

The key-value pairs that define the applied tags.

Type: Array of Tag (p. 63) objects

Array Members: Minimum number of 1 item. Maximum number of 50 items.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 66).

InvalidArnException

The requested Amazon Resource Name (ARN) does not refer to an existing resource.

HTTP Status Code: 400

ResourceNotFoundException

The specified certificate cannot be found in the caller's account or the caller's account cannot be found.

HTTP Status Code: 400

Example

List tags for an ACM Certificate

Sample Request

```
POST / HTTP/1.1
Host: acm.us-east-1.amazonaws.com
X-Amz-Target: CertificateManager.ListTagsForCertificate
X-Amz-Date: 20160414T162913Z
User-Agent: aws-cli/1.10.20 Python/2.7.3 Linux/3.13.0-83-generic botocore/1.4.11
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=key_ID/20160414/us-east-1/acm/aws4_request,
SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-target,
Signature=c1b80f2b1b6c73c39e1a9594e621648e673b1419101809239b9a5dd8c397953a

{"CertificateArn": "arn:aws:acm:us-
east-1:111122223333:certificate/12345678-1234-1234-1234-123456789012"}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 07c10419-025e-11e6-baa2-cd9f4ef8cda6
Content-Type: application/x-amz-json-1.1
Content-Length: 87
Date: Thu, 14 Apr 2016 16:29:16 GMT

{
  "Tags": [{
    "Key": "stack",
```

```
    "Value": "production"
  },
  {
    "Key": "website",
    "Value": "example.com"
  }]
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

RemoveTagsFromCertificate

Remove one or more tags from an ACM certificate. A tag consists of a key-value pair. If you do not specify the value portion of the tag when calling this function, the tag will be removed regardless of value. If you specify a value, the tag is removed only if it is associated with the specified value.

To add tags to a certificate, use the AddTagsToCertificate (p. 3) action. To view all of the tags that have been applied to a specific ACM certificate, use the ListTagsForCertificate (p. 29) action.

Request Syntax

```
{
   "CertificateArn": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 64).

The request accepts the following data in JSON format.

Note
In the following list, the required parameters are described first.

CertificateArn (p. 32)

String that contains the ARN of the ACM Certificate with one or more tags that you want to remove. This must be of the form:

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

For more information about ARNs, see Amazon Resource Names (ARNs) and AWS Service Namespaces.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=,.@-]+(/[\w+=,.@-]+)*`

Required: Yes

Tags (p. 32)

The key-value pair that defines the tag to remove.

Type: Array of Tag (p. 63) objects

Array Members: Minimum number of 1 item. Maximum number of 50 items.

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 66).

InvalidArnException

The requested Amazon Resource Name (ARN) does not refer to an existing resource.

HTTP Status Code: 400

InvalidTagException

One or both of the values that make up the key-value pair is not valid. For example, you cannot specify a tag value that begins with `aws:`.

HTTP Status Code: 400

ResourceNotFoundException

The specified certificate cannot be found in the caller's account or the caller's account cannot be found.

HTTP Status Code: 400

Example

Remove two tags from an ACM certificate

Sample Request

```
POST / HTTP/1.1
Host: acm.us-east-1.amazonaws.com
X-Amz-Target: CertificateManager.RemoveTagsFromCertificate
X-Amz-Date: 20160414T163042Z
User-Agent: aws-cli/1.10.20 Python/2.7.3 Linux/3.13.0-83-generic botocore/1.4.11
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20160414/us-east-1/acm/
aws4_request,
SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-target,
Signature=379429306c5e89b9b4be5b35e29c26cc1da38215d8055a5ed0bdda57bcc881cc

{
  "CertificateArn": "arn:aws:acm:us-
east-1:111122223333:certificate/12345678-1234-1234-1234-123456789012",
  "Tags": [{
    "Key": "website",
    "Value": "example.com"
  },
  {
    "Key": "stack",
    "Value": "production"
  }]
```

```
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 3c8d676d-025e-11e6-8823-93164b47113c
Content-Type: application/x-amz-json-1.1
Content-Length: 0
Date: Thu, 14 Apr 2016 16:30:44 GMT
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

RequestCertificate

Requests an ACM certificate for use with other AWS services. To request an ACM certificate, you must specify a fully qualified domain name (FQDN) in the `DomainName` parameter. You can also specify additional FQDNs in the `SubjectAlternativeNames` parameter.

If you are requesting a private certificate, domain validation is not required. If you are requesting a public certificate, each domain name that you specify must be validated to verify that you own or control the domain. You can use DNS validation or email validation. We recommend that you use DNS validation. ACM issues public certificates after receiving approval from the domain owner.

Request Syntax

```
{
   "CertificateAuthorityArn": "string",
   "DomainName": "string",
   "DomainValidationOptions": [
      {
         "DomainName": "string",
         "ValidationDomain": "string"
      }
   ],
   "IdempotencyToken": "string",
   "Options": {
      "CertificateTransparencyLoggingPreference": "string"
   },
   "SubjectAlternativeNames": [ "string" ],
   "ValidationMethod": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 64).

The request accepts the following data in JSON format.

Note
In the following list, the required parameters are described first.

DomainName (p. 35)

Fully qualified domain name (FQDN), such as www.example.com, that you want to secure with an ACM certificate. Use an asterisk (*) to create a wildcard certificate that protects several sites in the same domain. For example, *.example.com protects www.example.com, site.example.com, and images.example.com.

The first domain name you enter cannot exceed 63 octets, including periods. Each subsequent Subject Alternative Name (SAN), however, can be up to 253 octets in length.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 253.

Pattern: `^(\*\.)?(((?!-)[A-Za-z0-9-]{0,62}[A-Za-z0-9])\.)+((?!-)[A-Za-z0-9-]{1,62}[A-Za-z0-9])$`

Required: Yes

CertificateAuthorityArn (p. 35)

The Amazon Resource Name (ARN) of the private certificate authority (CA) that will be used to issue the certificate. If you do not provide an ARN and you are trying to request a private certificate, ACM will attempt to issue a public certificate. For more information about private CAs, see the AWS Certificate Manager Private Certificate Authority (PCA) user guide. The ARN must have the following form:

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

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=,.@-]+(/[\w+=,.@-]+)*`

Required: No

DomainValidationOptions (p. 35)

The domain name that you want ACM to use to send you emails so that you can validate domain ownership.

Type: Array of DomainValidationOption (p. 57) objects

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Required: No

IdempotencyToken (p. 35)

Customer chosen string that can be used to distinguish between calls to `RequestCertificate`. Idempotency tokens time out after one hour. Therefore, if you call `RequestCertificate` multiple times with the same idempotency token within one hour, ACM recognizes that you are requesting only one certificate and will issue only one. If you change the idempotency token for each call, ACM recognizes that you are requesting multiple certificates.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 32.

Pattern: `\w+`

Required: No

Options (p. 35)

Currently, you can use this parameter to specify whether to add the certificate to a certificate transparency log. Certificate transparency makes it possible to detect SSL/TLS certificates that have been mistakenly or maliciously issued. Certificates that have not been logged typically produce an error message in a browser. For more information, see Opting Out of Certificate Transparency Logging.

Type: CertificateOptions (p. 53) object

Required: No

SubjectAlternativeNames (p. 35)

Additional FQDNs to be included in the Subject Alternative Name extension of the ACM certificate. For example, add the name www.example.net to a certificate for which the `DomainName` field is

www.example.com if users can reach your site by using either name. The maximum number of domain names that you can add to an ACM certificate is 100. However, the initial limit is 10 domain names. If you need more than 10 names, you must request a limit increase. For more information, see Limits.

The maximum length of a SAN DNS name is 253 octets. The name is made up of multiple labels separated by periods. No label can be longer than 63 octets. Consider the following examples:

- `(63 octets).(63 octets).(63 octets).(61 octets)` is legal because the total length is 253 octets (63+1+63+1+63+1+61) and no label exceeds 63 octets.
- `(64 octets).(63 octets).(63 octets).(61 octets)` is not legal because the total length exceeds 253 octets (64+1+63+1+63+1+61) and the first label exceeds 63 octets.
- `(63 octets).(63 octets).(63 octets).(62 octets)` is not legal because the total length of the DNS name (63+1+63+1+63+1+62) exceeds 253 octets.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Length Constraints: Minimum length of 1. Maximum length of 253.

Pattern: `^(\*\.)?(((?!-)[A-Za-z0-9-]{0,62}[A-Za-z0-9])\.)+((?!-)[A-Za-z0-9-]{1,62}[A-Za-z0-9])$`

Required: No

ValidationMethod (p. 35)

The method you want to use if you are requesting a public certificate to validate that you own or control domain. You can validate with DNS or validate with email. We recommend that you use DNS validation.

Type: String

Valid Values: `EMAIL | DNS`

Required: No

Response Syntax

```
{
   "CertificateArn": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

CertificateArn (p. 37)

String that contains the ARN of the issued certificate. This must be of the form:

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

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=,.@-]+(/[\w+=,.@-]+)*`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 66).

InvalidArnException

The requested Amazon Resource Name (ARN) does not refer to an existing resource.

HTTP Status Code: 400

InvalidDomainValidationOptionsException

One or more values in the DomainValidationOption (p. 57) structure is incorrect.

HTTP Status Code: 400

LimitExceededException

An ACM limit has been exceeded.

HTTP Status Code: 400

Examples

Request a public ACM certificate

Sample Request

```
POST / HTTP/1.1
Host: acm.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 171
X-Amz-Target: CertificateManager.RequestCertificate
X-Amz-Date: 20180326T215401Z
User-Agent: aws-cli/1.14.28 Python/2.7.9 Windows/8 botocore/1.8.32
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20151222/us-east-1/acm/
aws4_request,
SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-target,
Signature=dbba4b1fa1199c011c0b781b94c97b14cbe75fa64dc6424232c903798d2a83b5

{
  "IdempotencyToken": "184627",
  "CertificateOptions": {
    "CertificateTransparencyLoggingPreference": "DISABLED"
  },
  "ValidationMethod": "DNS",
  "DomainName": "www.example.com"
}
```

Sample Response

```
HTTP/1.1 200 OK
```

```
x-amzn-RequestId: 32c3ca21-3140-11e8-8ba0-f79627c5200e
Content-Type: application/x-amz-json-1.1
Content-Length: 104
Date: Mon, 26 Mar 2018 21:54:03 GMT

{
  "CertificateArn":"arn:aws:acm:us-east-1:123456789012:certificate/1ad574bd-eeb0-466e-
b961-74ec8b405093"
}
```

Request a private certificate

Sample Request

```
POST / HTTP/1.1
Host: acm.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 305
X-Amz-Target: CertificateManager.RequestCertificate
X-Amz-Date: 20180331T173532Z
User-Agent: aws-cli/1.14.28 Python/2.7.9 Windows/8 botocore/1.8.32
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=key_ID/20180331/us-east-1/acm/aws4_request,
SignedHeaders=content-type;host;x-amz-date;x-amz-target,
Signature=11be86a0995ac158327fe8ccf6f44c19af7e6768fbafe0ec10e74436770272fa

{
  "IdempotencyToken": "12563",
  "CertificateAuthorityArn": "arn:aws:acm-pca:us-east-1:account:certificate-
authority/12345678-1234-1234-1234-123456789012",
  "DomainName": "www.example.com"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: eaedc93a-3509-11e8-a99a-c76ec78904bf
Content-Type: application/x-amz-json-1.1
Content-Length: 104
Date: Sat, 31 Mar 2018 17:35:34 GMT
Connection: Keep-alive

{
  "CertificateArn": "arn:aws:acm:us-
east-1:account:certificate/88888888-4444-4444-4444-111111111111"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript

- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

ResendValidationEmail

Resends the email that requests domain ownership validation. The domain owner or an authorized representative must approve the ACM certificate before it can be issued. The certificate can be approved by clicking a link in the mail to navigate to the Amazon certificate approval website and then clicking **I Approve**. However, the validation email can be blocked by spam filters. Therefore, if you do not receive the original mail, you can request that the mail be resent within 72 hours of requesting the ACM certificate. If more than 72 hours have elapsed since your original request or since your last attempt to resend validation mail, you must request a new certificate. For more information about setting up your contact email addresses, see Configure Email for your Domain.

Request Syntax

```
{
   "CertificateArn": "string",
   "Domain": "string",
   "ValidationDomain": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 64).

The request accepts the following data in JSON format.

Note
In the following list, the required parameters are described first.

CertificateArn (p. 41)

String that contains the ARN of the requested certificate. The certificate ARN is generated and returned by the RequestCertificate (p. 35) action as soon as the request is made. By default, using this parameter causes email to be sent to all top-level domains you specified in the certificate request. The ARN must be of the form:

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

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=,.@-]+(/[\w+=,.@-]+)*`

Required: Yes

Domain (p. 41)

The fully qualified domain name (FQDN) of the certificate that needs to be validated.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 253.

Pattern: `^(\*\.)?(((?!-)[A-Za-z0-9-]{0,62}[A-Za-z0-9])\.)+((?!-)[A-Za-z0-9-]{1,62}[A-Za-z0-9])$`

Required: Yes

ValidationDomain (p. 41)

The base validation domain that will act as the suffix of the email addresses that are used to send the emails. This must be the same as the `Domain` value or a superdomain of the `Domain` value. For example, if you requested a certificate for `site.subdomain.example.com` and specify a **ValidationDomain** of `subdomain.example.com`, ACM sends email to the domain registrant, technical contact, and administrative contact in WHOIS and the following five addresses:

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

Type: String

Length Constraints: Minimum length of 1. Maximum length of 253.

Pattern: `^(\*\.)?(((?!-)[A-Za-z0-9-]{0,62}[A-Za-z0-9])\.)+((?!-)[A-Za-z0-9-]{1,62}[A-Za-z0-9])$`

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 66).

InvalidArnException

The requested Amazon Resource Name (ARN) does not refer to an existing resource.

HTTP Status Code: 400

InvalidDomainValidationOptionsException

One or more values in the DomainValidationOption (p. 57) structure is incorrect.

HTTP Status Code: 400

InvalidStateException

Processing has reached an invalid state.

HTTP Status Code: 400

ResourceNotFoundException

The specified certificate cannot be found in the caller's account or the caller's account cannot be found.

HTTP Status Code: 400

Example

Resend Validation Email

Sample Request

```
POST / HTTP/1.1
Host: acm.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 167
X-Amz-Target: CertificateManager.ResendValidationEmail
X-Amz-Date: 20151222T170722Z
User-Agent: aws-cli/1.9.7 Python/2.7.3 Linux/3.13.0-73-generic botocore/1.3.7
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=key_ID/20151222/us-east-1/acm/aws4_request,
SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-target,
Signature=7ec7e70cd614724945545b22bc28296f77803d0c2524573d41c994668f07f435

{
  "CertificateArn": "arn:aws:acm:us-
east-1:111122223333 :certificate/12345678-1234-1234-1234-1234567890912",
  "Domain": "www.example.com",
  "ValidationDomain": "example.com"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 74bada6d-a8ce-11e5-82ad-d565a2aaa0b3
Content-Type: application/x-amz-json-1.1
Content-Length: 0
Date: Tue, 22 Dec 2015 17:07:18 GMT
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

UpdateCertificateOptions

Updates a certificate. Currently, you can use this function to specify whether to opt in to or out of recording your certificate in a certificate transparency log. For more information, see Opting Out of Certificate Transparency Logging.

Request Syntax

```
{
   "CertificateArn": "string",
   "Options": {
      "CertificateTransparencyLoggingPreference": "string"
   }
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 64).

The request accepts the following data in JSON format.

Note
In the following list, the required parameters are described first.

CertificateArn (p. 44)

ARN of the requested certificate to update. This must be of the form:

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

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=,.@-]+(/[\w+=,.@-]+)*`

Required: Yes

Options (p. 44)

Use to update the options for your certificate. Currently, you can specify whether to add your certificate to a transparency log. Certificate transparency makes it possible to detect SSL/TLS certificates that have been mistakenly or maliciously issued. Certificates that have not been logged typically produce an error message in a browser.

Type: CertificateOptions (p. 53) object

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 66).

InvalidArnException

The requested Amazon Resource Name (ARN) does not refer to an existing resource.

HTTP Status Code: 400

InvalidStateException

Processing has reached an invalid state.

HTTP Status Code: 400

LimitExceededException

An ACM limit has been exceeded.

HTTP Status Code: 400

ResourceNotFoundException

The specified certificate cannot be found in the caller's account or the caller's account cannot be found.

HTTP Status Code: 400

Examples

UpdateCertificateOptions

Sample Request

```
POST / HTTP/1.1
acm.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 185
X-Amz-Target: CertificateManager.UpdateCertificateOptions
X-Amz-Date: 20180326T222032Z
User-Agent: aws-cli/1.14.28 Python/2.7.9 Windows/8 botocore/1.8.32
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=key_ID/20151222/us-east-1/acm/aws4_request,
SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-target,
Signature=7ec7e70cd614724945545b22bc28296f77803d0c2524573d41c994668f07f435

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

Example

Sample Response

```
HTTP/1.1 200 OK
```

```
x-amzn-RequestId: e6f55ecb-3143-11e8-af72-0bd5049841d5
Content-Type: application/x-amz-json-1.1
Content-Length: 0
Date: Tue, 22 Dec 2015 17:07:18 GMT
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

Data Types

The AWS Certificate Manager API contains several data types that various actions use. This section describes each data type in detail.

> **Note**
> The order of each element in a data type structure is not guaranteed. Applications should not assume a particular order.

The following data types are supported:

CertificateDetail

Contains metadata about an ACM certificate. This structure is returned in the response to a DescribeCertificate (p. 9) request.

Contents

Note
In the following list, the required parameters are described first.

CertificateArn

The Amazon Resource Name (ARN) of the certificate. For more information about ARNs, see Amazon Resource Names (ARNs) and AWS Service Namespaces in the *AWS General Reference*.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=,.@-]+(/[\w+=,.@-]+)*`

Required: No

CertificateAuthorityArn

The Amazon Resource Name (ARN) of the ACM PCA private certificate authority (CA) that issued the certificate. This has the following format:

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

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=,.@-]+(/[\w+=,.@-]+)*`

Required: No

CreatedAt

The time at which the certificate was requested. This value exists only when the certificate type is `AMAZON_ISSUED`.

Type: Timestamp

Required: No

DomainName

The fully qualified domain name for the certificate, such as www.example.com or example.com.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 253.

Pattern: `^(\*\.)?(((?!-)[A-Za-z0-9-]{0,62}[A-Za-z0-9])\.)+((?!-)[A-Za-z0-9-]{1,62}[A-Za-z0-9])$`

Required: No

DomainValidationOptions

Contains information about the initial validation of each domain name that occurs as a result of the RequestCertificate (p. 35) request. This field exists only when the certificate type is `AMAZON_ISSUED`.

Type: Array of DomainValidation (p. 55) objects

Array Members: Minimum number of 1 item. Maximum number of 1000 items.

Required: No

ExtendedKeyUsages

Contains a list of Extended Key Usage X.509 v3 extension objects. Each object specifies a purpose for which the certificate public key can be used and consists of a name and an object identifier (OID).

Type: Array of ExtendedKeyUsage (p. 58) objects

Required: No

FailureReason

The reason the certificate request failed. This value exists only when the certificate status is `FAILED`. For more information, see Certificate Request Failed in the *AWS Certificate Manager User Guide*.

Type: String

Valid Values: `NO_AVAILABLE_CONTACTS | ADDITIONAL_VERIFICATION_REQUIRED | DOMAIN_NOT_ALLOWED | INVALID_PUBLIC_DOMAIN | CAA_ERROR | PCA_LIMIT_EXCEEDED | PCA_INVALID_ARN | PCA_INVALID_STATE | PCA_REQUEST_FAILED | PCA_RESOURCE_NOT_FOUND | PCA_INVALID_ARGS | PCA_ACCESS_DENIED | OTHER`

Required: No

ImportedAt

The date and time at which the certificate was imported. This value exists only when the certificate type is `IMPORTED`.

Type: Timestamp

Required: No

InUseBy

A list of ARNs for the AWS resources that are using the certificate. A certificate can be used by multiple AWS resources.

Type: Array of strings

Required: No

IssuedAt

The time at which the certificate was issued. This value exists only when the certificate type is `AMAZON_ISSUED`.

Type: Timestamp

Required: No

Issuer

The name of the certificate authority that issued and signed the certificate.

Type: String

Required: No

KeyAlgorithm

The algorithm that was used to generate the public-private key pair.

Type: String

Valid Values: `RSA_2048 | RSA_1024 | RSA_4096 | EC_prime256v1 | EC_secp384r1 | EC_secp521r1`

Required: No

KeyUsages

A list of Key Usage X.509 v3 extension objects. Each object is a string value that identifies the purpose of the public key contained in the certificate. Possible extension values include DIGITAL_SIGNATURE, KEY_ENCHIPHERMENT, NON_REPUDIATION, and more.

Type: Array of KeyUsage (p. 60) objects

Required: No

NotAfter

The time after which the certificate is not valid.

Type: Timestamp

Required: No

NotBefore

The time before which the certificate is not valid.

Type: Timestamp

Required: No

Options

Value that specifies whether to add the certificate to a transparency log. Certificate transparency makes it possible to detect SSL certificates that have been mistakenly or maliciously issued. A browser might respond to certificate that has not been logged by showing an error message. The logs are cryptographically secure.

Type: CertificateOptions (p. 53) object

Required: No

RenewalEligibility

Specifies whether the certificate is eligible for renewal.

Type: String

Valid Values: `ELIGIBLE | INELIGIBLE`

Required: No

RenewalSummary

Contains information about the status of ACM's managed renewal for the certificate. This field exists only when the certificate type is `AMAZON_ISSUED`.

Type: RenewalSummary (p. 61) object

Required: No

RevocationReason

The reason the certificate was revoked. This value exists only when the certificate status is `REVOKED`.

Type: String

Valid Values: `UNSPECIFIED | KEY_COMPROMISE | CA_COMPROMISE | AFFILIATION_CHANGED | SUPERCEDED | CESSATION_OF_OPERATION | CERTIFICATE_HOLD | REMOVE_FROM_CRL | PRIVILEGE_WITHDRAWN | A_A_COMPROMISE`

Required: No

RevokedAt

The time at which the certificate was revoked. This value exists only when the certificate status is `REVOKED`.

Type: Timestamp

Required: No

Serial

The serial number of the certificate.

Type: String

Required: No

SignatureAlgorithm

The algorithm that was used to sign the certificate.

Type: String

Required: No

Status

The status of the certificate.

Type: String

Valid Values: `PENDING_VALIDATION | ISSUED | INACTIVE | EXPIRED | VALIDATION_TIMED_OUT | REVOKED | FAILED`

Required: No

Subject

The name of the entity that is associated with the public key contained in the certificate.

Type: String

Required: No

SubjectAlternativeNames

One or more domain names (subject alternative names) included in the certificate. This list contains the domain names that are bound to the public key that is contained in the certificate. The subject alternative names include the canonical domain name (CN) of the certificate and additional domain names that can be used to connect to the website.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Length Constraints: Minimum length of 1. Maximum length of 253.

Pattern: `^(\*\.)?(((?!-)[A-Za-z0-9-]{0,62}[A-Za-z0-9])\.)+((?!-)[A-Za-z0-9-]{1,62}[A-Za-z0-9])$`

Required: No

Type

The source of the certificate. For certificates provided by ACM, this value is `AMAZON_ISSUED`. For certificates that you imported with ImportCertificate (p. 21), this value is `IMPORTED`. ACM does not provide managed renewal for imported certificates. For more information about the differences between certificates that you import and those that ACM provides, see Importing Certificates in the *AWS Certificate Manager User Guide*.

Type: String

Valid Values: `IMPORTED | AMAZON_ISSUED | PRIVATE`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2

CertificateOptions

Structure that contains options for your certificate. Currently, you can use this only to specify whether to opt in to or out of certificate transparency logging. Some browsers require that public certificates issued for your domain be recorded in a log. Certificates that are not logged typically generate a browser error. Transparency makes it possible for you to detect SSL/TLS certificates that have been mistakenly or maliciously issued for your domain. For general information, see Certificate Transparency Logging.

Contents

Note
In the following list, the required parameters are described first.

CertificateTransparencyLoggingPreference

You can opt out of certificate transparency logging by specifying the `DISABLED` option. Opt in by specifying `ENABLED`.

Type: String

Valid Values: `ENABLED | DISABLED`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2

CertificateSummary

This structure is returned in the response object of ListCertificates (p. 25) action.

Contents

Note
In the following list, the required parameters are described first.

CertificateArn

Amazon Resource Name (ARN) of the certificate. This is of the form:

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

For more information about ARNs, see Amazon Resource Names (ARNs) and AWS Service Namespaces.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=,.@-]+(/[\w+=,.@-]+)*`

Required: No

DomainName

Fully qualified domain name (FQDN), such as www.example.com or example.com, for the certificate.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 253.

Pattern: `^(\*\.)?(((?!-)[A-Za-z0-9-]{0,62}[A-Za-z0-9])\.)+((?!-)[A-Za-z0-9-]{1,62}[A-Za-z0-9])$`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2

DomainValidation

Contains information about the validation of each domain name in the certificate.

Contents

Note
In the following list, the required parameters are described first.

DomainName

A fully qualified domain name (FQDN) in the certificate. For example, `www.example.com` or `example.com`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 253.

Pattern: `^(\*\.)?(((?!-)[A-Za-z0-9-]{0,62}[A-Za-z0-9])\.)+((?!-)[A-Za-z0-9-]{1,62}[A-Za-z0-9])$`

Required: Yes

ResourceRecord

Contains the CNAME record that you add to your DNS database for domain validation. For more information, see Use DNS to Validate Domain Ownership.

Type: ResourceRecord (p. 62) object

Required: No

ValidationDomain

The domain name that ACM used to send domain validation emails.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 253.

Pattern: `^(\*\.)?(((?!-)[A-Za-z0-9-]{0,62}[A-Za-z0-9])\.)+((?!-)[A-Za-z0-9-]{1,62}[A-Za-z0-9])$`

Required: No

ValidationEmails

A list of email addresses that ACM used to send domain validation emails.

Type: Array of strings

Required: No

ValidationMethod

Specifies the domain validation method.

Type: String

Valid Values: `EMAIL | DNS`

Required: No

ValidationStatus

The validation status of the domain name. This can be one of the following values:

- `PENDING_VALIDATION`
- SUCCESS
- FAILED

Type: String

Valid Values: `PENDING_VALIDATION | SUCCESS | FAILED`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2

DomainValidationOption

Contains information about the domain names that you want ACM to use to send you emails that enable you to validate domain ownership.

Contents

Note
In the following list, the required parameters are described first.

DomainName

A fully qualified domain name (FQDN) in the certificate request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 253.

Pattern: `^(\*\.)?(((?!-)[A-Za-z0-9-]{0,62}[A-Za-z0-9])\.)+((?!-)[A-Za-z0-9-]{1,62}[A-Za-z0-9])$`

Required: Yes

ValidationDomain

The domain name that you want ACM to use to send you validation emails. This domain name is the suffix of the email addresses that you want ACM to use. This must be the same as the `DomainName` value or a superdomain of the `DomainName` value. For example, if you request a certificate for `testing.example.com`, you can specify `example.com` for this value. In that case, ACM sends domain validation emails to the following five addresses:

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

Type: String

Length Constraints: Minimum length of 1. Maximum length of 253.

Pattern: `^(\*\.)?(((?!-)[A-Za-z0-9-]{0,62}[A-Za-z0-9])\.)+((?!-)[A-Za-z0-9-]{1,62}[A-Za-z0-9])$`

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2

ExtendedKeyUsage

The Extended Key Usage X.509 v3 extension defines one or more purposes for which the public key can be used. This is in addition to or in place of the basic purposes specified by the Key Usage extension.

Contents

Note
In the following list, the required parameters are described first.

Name

The name of an Extended Key Usage value.

Type: String

Valid Values: `TLS_WEB_SERVER_AUTHENTICATION | TLS_WEB_CLIENT_AUTHENTICATION | CODE_SIGNING | EMAIL_PROTECTION | TIME_STAMPING | OCSP_SIGNING | IPSEC_END_SYSTEM | IPSEC_TUNNEL | IPSEC_USER | ANY | NONE | CUSTOM`

Required: No

OID

An object identifier (OID) for the extension value. OIDs are strings of numbers separated by periods. The following OIDs are defined in RFC 3280 and RFC 5280.

- `1.3.6.1.5.5.7.3.1 (TLS_WEB_SERVER_AUTHENTICATION)`
- `1.3.6.1.5.5.7.3.2 (TLS_WEB_CLIENT_AUTHENTICATION)`
- `1.3.6.1.5.5.7.3.3 (CODE_SIGNING)`
- `1.3.6.1.5.5.7.3.4 (EMAIL_PROTECTION)`
- `1.3.6.1.5.5.7.3.8 (TIME_STAMPING)`
- `1.3.6.1.5.5.7.3.9 (OCSP_SIGNING)`
- `1.3.6.1.5.5.7.3.5 (IPSEC_END_SYSTEM)`
- `1.3.6.1.5.5.7.3.6 (IPSEC_TUNNEL)`
- `1.3.6.1.5.5.7.3.7 (IPSEC_USER)`

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2

Filters

This structure can be used in the ListCertificates (p. 25) action to filter the output of the certificate list.

Contents

Note
In the following list, the required parameters are described first.

extendedKeyUsage

Specify one or more ExtendedKeyUsage (p. 58) extension values.

Type: Array of strings

Valid Values: `TLS_WEB_SERVER_AUTHENTICATION | TLS_WEB_CLIENT_AUTHENTICATION | CODE_SIGNING | EMAIL_PROTECTION | TIME_STAMPING | OCSP_SIGNING | IPSEC_END_SYSTEM | IPSEC_TUNNEL | IPSEC_USER | ANY | NONE | CUSTOM`

Required: No

keyTypes

Specify one or more algorithms that can be used to generate key pairs.

Type: Array of strings

Valid Values: `RSA_2048 | RSA_1024 | RSA_4096 | EC_prime256v1 | EC_secp384r1 | EC_secp521r1`

Required: No

keyUsage

Specify one or more KeyUsage (p. 60) extension values.

Type: Array of strings

Valid Values: `DIGITAL_SIGNATURE | NON_REPUDIATION | KEY_ENCIPHERMENT | DATA_ENCIPHERMENT | KEY_AGREEMENT | CERTIFICATE_SIGNING | CRL_SIGNING | ENCIPHER_ONLY | DECIPHER_ONLY | ANY | CUSTOM`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2

KeyUsage

The Key Usage X.509 v3 extension defines the purpose of the public key contained in the certificate.

Contents

Note
In the following list, the required parameters are described first.

Name

A string value that contains a Key Usage extension name.

Type: String

Valid Values: `DIGITAL_SIGNATURE | NON_REPUDIATION | KEY_ENCIPHERMENT | DATA_ENCIPHERMENT | KEY_AGREEMENT | CERTIFICATE_SIGNING | CRL_SIGNING | ENCIPHER_ONLY | DECIPHER_ONLY | ANY | CUSTOM`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2

RenewalSummary

Contains information about the status of ACM's managed renewal for the certificate. This structure exists only when the certificate type is `AMAZON_ISSUED`.

Contents

Note
In the following list, the required parameters are described first.

DomainValidationOptions

Contains information about the validation of each domain name in the certificate, as it pertains to ACM's managed renewal. This is different from the initial validation that occurs as a result of the RequestCertificate (p. 35) request. This field exists only when the certificate type is `AMAZON_ISSUED`.

Type: Array of DomainValidation (p. 55) objects

Array Members: Minimum number of 1 item. Maximum number of 1000 items.

Required: Yes

RenewalStatus

The status of ACM's managed renewal of the certificate.

Type: String

Valid Values: `PENDING_AUTO_RENEWAL | PENDING_VALIDATION | SUCCESS | FAILED`

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2

ResourceRecord

Contains a DNS record value that you can use to can use to validate ownership or control of a domain. This is used by the DescribeCertificate (p. 9) action.

Contents

Note
In the following list, the required parameters are described first.

Name

The name of the DNS record to create in your domain. This is supplied by ACM.

Type: String

Required: Yes

Type

The type of DNS record. Currently this can be `CNAME`.

Type: String

Valid Values: `CNAME`

Required: Yes

Value

The value of the CNAME record to add to your DNS database. This is supplied by ACM.

Type: String

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2

Tag

A key-value pair that identifies or specifies metadata about an ACM resource.

Contents

Note
In the following list, the required parameters are described first.

Key

The key of the tag.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `[\p{L}\p{Z}\p{N}_.:\/=+\-@]*`

Required: Yes

Value

The value of the tag.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 256.

Pattern: `[\p{L}\p{Z}\p{N}_.:\/=+\-@]*`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2

Common Parameters

The following list contains the parameters that all actions use for signing Signature Version 4 requests with a query string. Any action-specific parameters are listed in the topic for that action. For more information about Signature Version 4, see Signature Version 4 Signing Process in the *Amazon Web Services General Reference*.

Action

The action to be performed.

Type: string

Required: Yes

Version

The API version that the request is written for, expressed in the format YYYY-MM-DD.

Type: string

Required: Yes

X-Amz-Algorithm

The hash algorithm that you used to create the request signature.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Valid Values: `AWS4-HMAC-SHA256`

Required: Conditional

X-Amz-Credential

The credential scope value, which is a string that includes your access key, the date, the region you are targeting, the service you are requesting, and a termination string ("aws4_request"). The value is expressed in the following format: *access_key*/*YYYYMMDD*/*region*/*service*/aws4_request.

For more information, see Task 2: Create a String to Sign for Signature Version 4 in the *Amazon Web Services General Reference*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

X-Amz-Date

The date that is used to create the signature. The format must be ISO 8601 basic format (YYYYMMDD'T'HHMMSS'Z'). For example, the following date time is a valid X-Amz-Date value: `20120325T120000Z`.

Condition: X-Amz-Date is optional for all requests; it can be used to override the date used for signing requests. If the Date header is specified in the ISO 8601 basic format, X-Amz-Date is

not required. When X-Amz-Date is used, it always overrides the value of the Date header. For more information, see Handling Dates in Signature Version 4 in the *Amazon Web Services General Reference*.

Type: string

Required: Conditional

X-Amz-Security-Token

The temporary security token that was obtained through a call to AWS Security Token Service (AWS STS). For a list of services that support temporary security credentials from AWS Security Token Service, go to AWS Services That Work with IAM in the *IAM User Guide*.

Condition: If you're using temporary security credentials from the AWS Security Token Service, you must include the security token.

Type: string

Required: Conditional

X-Amz-Signature

Specifies the hex-encoded signature that was calculated from the string to sign and the derived signing key.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

X-Amz-SignedHeaders

Specifies all the HTTP headers that were included as part of the canonical request. For more information about specifying signed headers, see Task 1: Create a Canonical Request For Signature Version 4 in the *Amazon Web Services General Reference*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

Common Errors

This section lists the errors common to the API actions of all AWS services. For errors specific to an API action for this service, see the topic for that API action.

AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 400

IncompleteSignature

The request signature does not conform to AWS standards.

HTTP Status Code: 400

InternalFailure

The request processing has failed because of an unknown error, exception or failure.

HTTP Status Code: 500

InvalidAction

The action or operation requested is invalid. Verify that the action is typed correctly.

HTTP Status Code: 400

InvalidClientTokenId

The X.509 certificate or AWS access key ID provided does not exist in our records.

HTTP Status Code: 403

InvalidParameterCombination

Parameters that must not be used together were used together.

HTTP Status Code: 400

InvalidParameterValue

An invalid or out-of-range value was supplied for the input parameter.

HTTP Status Code: 400

InvalidQueryParameter

The AWS query string is malformed or does not adhere to AWS standards.

HTTP Status Code: 400

MalformedQueryString

The query string contains a syntax error.

HTTP Status Code: 404

MissingAction

The request is missing an action or a required parameter.

HTTP Status Code: 400

MissingAuthenticationToken

The request must contain either a valid (registered) AWS access key ID or X.509 certificate.

HTTP Status Code: 403

MissingParameter

A required parameter for the specified action is not supplied.

HTTP Status Code: 400

OptInRequired

The AWS access key ID needs a subscription for the service.

HTTP Status Code: 403

RequestExpired

The request reached the service more than 15 minutes after the date stamp on the request or more than 15 minutes after the request expiration date (such as for pre-signed URLs), or the date stamp on the request is more than 15 minutes in the future.

HTTP Status Code: 400

ServiceUnavailable

The request has failed due to a temporary failure of the server.

HTTP Status Code: 503

ThrottlingException

The request was denied due to request throttling.

HTTP Status Code: 400

ValidationError

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400