Manage security certificates

REST API reference

NetApp
February 07, 2026

Table of Contents

Manage security certificates

Manage security certificates

Overview

This API displays security certificate information and manages the certificates in ONTAP.

Installing certificates in ONTAP

The security certificates GET request retrieves all of the certificates in the cluster.

Examples

Retrieving all certificates installed in the cluster with their common-names

```
# The API:
/api/security/certificates

# The call:
curl -X GET "https://<mgmt-
ip>/api/security/certificates?fields=common_name" -H "accept:
application/hal+json"

# The response:
{
"records": [
  {
    "svm": {
      "name": "vs0"
    },
    "uuid": "dad2363b-8ac0-11e8-9058-005056b482fc",
    "common_name": "vs0",
    "_links": {
      "self": {
        "href": "/api/security/certificates/dad2363b-8ac0-11e8-9058-
005056b482fc"
      }
    }
  },
  {
    "uuid": "1941e048-8ac1-11e8-9058-005056b482fc",
    "common_name": "ROOT",
    "_links": {
      "self": {
        "href": "/api/security/certificates/1941e048-8ac1-11e8-9058-
```

```
005056b482fc"
      }
    }
  },
  {
    "uuid": "5a3a77a8-892d-11e8-b7da-005056b482fc",
    "common_name": "cert_name",
    "_links": {
      "self": {
        "href": "/api/security/certificates/5a3a77a8-892d-11e8-b7da-
005056b482fc"
      }
    }
  }
],
"num_records": 3,
"_links": {
  "self": {
    "href": "/api/security/certificates?fields=common_name"
  }
}
}
```

Retrieving all certificates installed at cluster-scope with their common-names

```
# The API:
/api/security/certificates

# The call:
curl -X GET "https://<mgmt-
ip>/api/security/certificates?scope=cluster&fields=common_name" -H
"accept: application/hal+json"

# The response:
{
"records": [
  {
    "uuid": "1941e048-8ac1-11e8-9058-005056b482fc",
    "scope": "cluster",
    "common_name": "ROOT",
    "_links": {
      "self": {
        "href": "/api/security/certificates/1941e048-8ac1-11e8-9058-
005056b482fc"
      }
    }
  },
  {
    "uuid": "5a3a77a8-892d-11e8-b7da-005056b482fc",
    "scope": "cluster",
    "common_name": "cert_name",
    "_links": {
      "self": {
        "href": "/api/security/certificates/5a3a77a8-892d-11e8-b7da-
005056b482fc"
      }
    }
  }
],
"num_records": 2,
"_links": {
  "self": {
    "href": "/api/security/certificates?scope=cluster&fields=common_name"
  }
}
}
```

Retrieving all certificates installed on a specific SVM with their common-names

```
# The API:
/api/security/certificates

# The call:
curl -X GET "https://<mgmt-
ip>/api/security/certificates?svm.name=vs0&fields=common_name" -H "accept:
application/hal+json"

# The response:
{
"records": [
  {
    "svm": {
      "name": "vs0"
    },
    "uuid": "dad2363b-8ac0-11e8-9058-005056b482fc",
    "common_name": "vs0",
    "_links": {
      "self": {
        "href": "/api/security/certificates/dad2363b-8ac0-11e8-9058-
005056b482fc"
      }
    }
  }
],
"num_records": 1,
"_links": {
  "self": {
    "href": "/api/security/certificates?svm.name=vs0&fields=common_name"
  }
}
}
```

Retrieving a certificate using its UUID for all fields

```
# The API:
/api/security/certificates/{uuid}

# The call:
curl -X GET "https://<mgmt-ip>/api/security/certificates/dad2363b-8ac0-
11e8-9058-005056b482fc?fields=*" -H "accept: application/hal+json"

# The response:
{
"svm": {
  "uuid": "d817293c-8ac0-11e8-9058-005056b482fc",
  "name": "vs0"
},
"uuid": "dad2363b-8ac0-11e8-9058-005056b482fc",
"scope": "svm",
"type": "server",
"common_name": "vs0",
"serial_number": "15428D45CF81CF56",
"ca": "vs0",
"hash_function": "sha256",
"key_size": 2048,
"expiry_time": "2019-07-18T15:29:14-04:00",
"public_certificate": "<CERTIFICATE-CONTENT>",
"_links": {
  "self": {
    "href": "/api/security/certificates/dad2363b-8ac0-11e8-9058-
005056b482fc"
  }
}
}
```

Creating a certificate in a cluster

These certificates can be used to help administrators enable certificate-based authentication and to enable SSL-based communication to the cluster.

```
# The API:
/api/security/certificates

# The call:
curl -X POST "https://<mgmt-ip>/api/security/certificates" -H "accept:
application/hal+json" -H "Content-Type: application/json" -d "{
\"common_name\": \"TEST-SERVER\",  \"type\": \"server\"  }"
```

Installing a certificate in a cluster

These certificates can be used to help administrators enable certificate-based authentication and to enable-SSL based communication to the cluster.

```
# The API:
/api/security/certificates

# The call:
curl -X POST "https://<mgmt-ip>/api/security/certificates" -H  "accept:
application/json" -H  "Content-Type: application/json" -d "{ \"type\":
\"server_ca\", \"public_certificate\": \"<CERTIFICATE-CONTENT>\"  }"
```

Installing a certificate on a specific SVM

```
# The API:
/api/security/certificates

# The call:
curl -X POST "https://<mgmt-ip>/api/security/certificates" -H  "accept:
application/json" -H  "Content-Type: application/json" -d "{   \"svm\" : {
\"name\" : \"vs0\" }, \"type\": \"server_ca\", \"public_certificate\":
\"<CERTIFICATE-CONTENT>\"  }"
```

Installing a CA-signed certificate on a specific SVM

```
# The API:
/api/security/certificates

# The call:
curl -X POST "https://<mgmt-ip>/api/security/certificates" -H  "accept:
application/json" -H  "Content-Type: application/json" -d "{   \"svm\" : {
\"name\" : \"vs0\" }, \"type\": \"server\", \"public_certificate\":
\"<CERTIFICATE-CONTENT>\", \"intermediate_certificates\": [\"<CERTIFICATE-
CONTENT>\",\"<CERTIFICATE-CONTENT>\"]}"
```

Deleting a certificate using its UUID

```
# The API:
/api/security/certificates/{uuid}

# The call:
curl -X DELETE "https://<mgmt-ip>/api/security/certificates/dad2363b-8ac0-
11e8-9058-005056b482fc?fields=*" -H "accept: application/hal+json"
```

Signing a new certificate signing request using an existing CA certificate UUID

Once you have created a certificate of type "root_ca", you can use that certificate to act as a local Certificate Authority to sign new certificate signing requests. The following example signs a new certificate signing request using an existing CA certificate UUID. If successful, the API returns a signed certificate.

```
# The API:
/api/security/certificates/{ca.uuid}/sign

# The call:
curl -X POST "https://<mgmt-ip>/api/security/certificates/253add53-8ac9-
11e8-9058-005056b482fc/sign" -H  "accept: application/json" -H  "Content-
Type: application/json" -d "{ \"signing_request\": \"<CERTIFICATE-
CONTENT>\",  \"hash_function\": \"sha256\"}"

# The response:
{
"public_certificate": "<CERTIFICATE-CONTENT>"
}
```

Generate a new Certificate Signing Request (CSR)

```
# The API:
/api/security/certificate-signing-request

# The call:
curl -X POST "https://<mgmt-ip>/api/security/certificate-signing-request"
-H  "accept: application/json" -H  "Content-Type: application/json" -d "{
\"algorithm\": \"rsa\", \"extended_key_usage\": [\"serverauth\"],
\"hash_function\": \"sha256\",  \"key_usage\": [\"digitalsignature\"],
\"security_strength\": \"112\", \"subject_alternatives\": { \"dns\": [
\"*.example.com\", \"*.example1.com\" ], \"email\": [\"abc@example.com\",
\"abc@example1.com\"], \"ip\": [\"10.225.34.223\", \"10.225.34.224\"],
\"uri\": [\"http://example.com\", \"http://example1.com\"] },
\"subject_name\": \"C=US,O=NTAP,CN=test.domain.com\"}"
{
"csr": "-----BEGIN CERTIFICATE REQUEST-----\n-----END CERTIFICATE
REQUEST-----\n",
"generated_private_key": "-----BEGIN PRIVATE KEY-----\n-----END PRIVATE
KEY-----\n"
}
```

```
### Download and install a certificate from the Azure Key Vault.
```

The API:

/api/security/certificates

The call:

curl -X POST "https://<mgmt-ip>/api/security/certificates" -H "accept: application/json" -H "Content-Type: application/json" -d "{\"svm\": {\"name\" : \"vs0\"}, \"name\" : \"vs0-client-cert\", \"type\" : \"client\", \"azure\": { \"key_vault\" : \"https://example.vault.azure.net\", \"client_id\" : \"12345678-abcd-1234-12ad-dfasdffgfdaaa\", \"tenant_id\" : \"12345678-abcd-abcd-test-720ef604b100\", \"client_secret\" : \"clientSecretString\", \"verify_host\": false }}" { "job": { "uuid": "be8d45cb-1d41-11ee-9725-005056ae0f31", "_links": { "self": { "href": "/api/cluster/jobs/be8d45cb-1d41-11ee-9725-005056ae0f31" } } } }</mgmt-ip>

```
'''




[[IDaf6e20bd74da539a1c63bfe99df64957]]
```

```
= Retrieve security certificates

[.api-doc-operation .api-doc-operation-get]#GET# [.api-doc-code-
block]#`/security/certificates`#

*Introduced In:* 9.6

Retrieves security certificates.

== Related ONTAP commands

* `security certificate show`


== Parameters

[cols=5*,options=header]
|===

|Name
|Type
|In
|Required
|Description

|svm.name
|string
|query
|False
a|Filter by svm.name


|svm.uuid
|string
|query
|False
a|Filter by svm.uuid


|subject_key_identifier
|string
|query
|False
a|Filter by subject_key_identifier

* Introduced in: 9.8
```

```
|ca
|string
|query
|False
a|Filter by ca

* maxLength: 256
* minLength: 1


|public_certificate
|string
|query
|False
a|Filter by public_certificate

* Introduced in: 9.8


|hash_function
|string
|query
|False
a|Filter by hash_function


|scope
|string
|query
|False
a|Filter by scope


|subject_alternatives.dns
|string
|query
|False
a|Filter by subject_alternatives.dns

* Introduced in: 9.15


|subject_alternatives.uri
|string
|query
|False
```

```
a|Filter by subject_alternatives.uri

* Introduced in: 9.15


|subject_alternatives.email
|string
|query
|False
a|Filter by subject_alternatives.email

* Introduced in: 9.15


|subject_alternatives.ip
|string
|query
|False
a|Filter by subject_alternatives.ip

* Introduced in: 9.15


|serial_number
|string
|query
|False
a|Filter by serial_number

* maxLength: 40
* minLength: 1


|uuid
|string
|query
|False
a|Filter by uuid

* Introduced in: 9.8


|key_size
|integer
|query
|False
a|Filter by key_size
```

```
|authority_key_identifier
|string
|query
|False
a|Filter by authority_key_identifier

* Introduced in: 9.8


|common_name
|string
|query
|False
a|Filter by common_name


|type
|string
|query
|False
a|Filter by type


|expiry_time
|string
|query
|False
a|Filter by expiry_time


|name
|string
|query
|False
a|Filter by name

* Introduced in: 9.8


|fields
|array[string]
|query
|False
a|Specify the fields to return.
```

```
|max_records
|integer
|query
|False
a|Limit the number of records returned.


|return_timeout
|integer
|query
|False
a|The number of seconds to allow the call to execute before returning.
When iterating over a collection, the default is 15 seconds.  ONTAP
returns earlier if either max records or the end of the collection is
reached.

* Default value: 15
* Max value: 120
* Min value: 0


|return_records
|boolean
|query
|False
a|The default is true for GET calls.  When set to false, only the number
of records is returned.

* Default value: 1


|order_by
|array[string]
|query
|False
a|Order results by specified fields and optional [asc|desc] direction.
Default direction is 'asc' for ascending.

|===

== Response
```

Status: 200, Ok

```
[cols=3*,options=header]
```

```
|===
|Name
|Type
|Description

|_links
|link:#_links[_links]
a|

|num_records
|integer
a|Number of records


|records
|array[link:#security_certificate[security_certificate]]
a|

|===


.Example response
[%collapsible%closed]
====
[source,json,subs=+macros]
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "authority_key_identifier":
"26:1F:C5:53:5B:D7:9E:E2:37:74:F4:F4:06:09:03:3D:EB:41:75:D7",
      "azure": {
        "verify_host": true
```

```
      },
      "ca": "string",
      "common_name": "test.domain.com",
      "expiry_time": "2030-01-25 06:20:13 -0500",
      "hash_function": "string",
      "key_size": 512,
      "name": "string",
      "public_certificate": "<CERTIFICATE-CONTENT>",
      "scope": "string",
      "serial_number": "string",
      "subject_alternatives": {
        "dns": [
          "*.example.com"
        ],
        "email": [
          "abc@example.com"
        ],
        "ip": [
          "10.225.34.10"
        ],
        "uri": [
          "http://example.com"
        ]
      },
      "subject_key_identifier":
"26:1F:C5:53:5B:D7:9E:E2:37:74:F4:F4:06:09:03:3D:EB:41:75:D8",
      "svm": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      },
      "type": "string",
      "uuid": "string"
    }
  ]
}
====

== Error
```

Status: Default, Error

```
[cols=3*,options=header]
|===
|Name
|Type
|Description

|error
|link:#returned_error[returned_error]
a|

|===


.Example error
[%collapsible%closed]
====
[source,json,subs=+macros]
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
====

== Definitions

[.api-def-first-level]
.See Definitions
[%collapsible%closed]
//Start collapsible Definitions block
====
[#href]
[.api-collapsible-fifth-title]
href

[cols=3*,options=header]
|===
|Name
```

```
|Type
|Description

|href
|string
a|

|===


[#_links]
[.api-collapsible-fifth-title]
_links

[cols=3*,options=header]
|===
|Name
|Type
|Description

|next
|link:#href[href]
a|

|self
|link:#href[href]
a|

|===


[#_links]
[.api-collapsible-fifth-title]
_links

[cols=3*,options=header]
|===
|Name
|Type
|Description

|self
|link:#href[href]
a|

|===
```

```
[#proxy]
[.api-collapsible-fifth-title]
proxy
[#azure]
[.api-collapsible-fifth-title]
azure

[cols=3*,options=header]
|===
|Name
|Type
|Description

|proxy
|link:#proxy[proxy]
a|

|===


[#subject_alternatives]
[.api-collapsible-fifth-title]
subject_alternatives

[cols=3*,options=header]
|===
|Name
|Type
|Description

|dns
|array[string]
a|A list of DNS names for Subject Alternate name extension.


|email
|array[string]
a|A list of email addresses for Subject Alternate name extension


|ip
|array[string]
a|A list of IP addresses for Subject Alternate name extension.


|uri
```

```
|array[string]
a|A list of URIs for Subject Alternate name extension.


|===


[#svm]
[.api-collapsible-fifth-title]
svm

SVM, applies only to SVM-scoped objects.


[cols=3*,options=header]
|===
|Name
|Type
|Description

|_links
|link:#_links[_links]
a|

|name
|string
a|The name of the SVM. This field cannot be specified in a PATCH method.


|uuid
|string
a|The unique identifier of the SVM. This field cannot be specified in a
PATCH method.


|===


[#security_certificate]
[.api-collapsible-fifth-title]
security_certificate

[cols=3*,options=header]
|===
|Name
|Type
|Description
```

```
|_links
|link:#_links[_links]
a|

|authority_key_identifier
|string
a|Provides the key identifier of the issuing CA certificate that signed
the SSL certificate.


|azure
|link:#azure[azure]
a|

|ca
|string
a|Certificate authority


|common_name
|string
a|FQDN or custom common name. Provide on POST when creating a self-signed
certificate.


|expiry_time
|string
a|Certificate expiration time, in ISO 8601 duration format or date and
time format. Can be provided on POST if creating self-signed certificate.
The expiration time range is between 1 day to 10 years.


|hash_function
|string
a|Hashing function. Can be provided on POST when creating a self-signed
certificate. Hash functions md5 and sha1 are not allowed on POST.


|key_size
|integer
a|Key size of requested Certificate in bits. One of 512, 1024, 1536, 2048,
3072. Can be provided on POST if creating self-signed certificate with a
minimum permissible value of 2048.


|name
```

```
|string
a|Certificate name or name of the certificate to be downloaded from the
Azure Key Vault (AKV). If not provided in POST, a unique name specific to
the SVM is automatically generated.


|public_certificate
|string
a|Public key Certificate in PEM format. If this is not provided in POST, a
self-signed certificate is created.


|scope
|string
a|Set to "svm" for interfaces owned by an SVM. Otherwise, set to
"cluster".


|serial_number
|string
a|Serial number of certificate.


|subject_alternatives
|link:#subject_alternatives[subject_alternatives]
a|

|subject_key_identifier
|string
a|Provides the key identifier used to identify the public key in the SSL
certificate.


|svm
|link:#svm[svm]
a|SVM, applies only to SVM-scoped objects.


|type
|string
a|Type of Certificate. The following types are supported:

* client - a certificate and its private key used by an SSL client in
ONTAP.
* server - a certificate and its private key used by an SSL server in
ONTAP.
* client_ca - a Certificate Authority certificate used by an SSL server in
```

```
ONTAP to verify an SSL client certificate.
* server_ca - a Certificate Authority certificate used by an SSL client in
ONTAP to verify an SSL server certificate.
* root_ca - a self-signed certificate used by ONTAP to sign other
certificates by acting as a Certificate Authority.
* enum: ["client", "server", "client_ca", "server_ca", "root_ca"]
* Introduced in: 9.6
* x-nullable: true


|uuid
|string
a|Unique ID that identifies a certificate.


|===


[#error_arguments]
[.api-collapsible-fifth-title]
error_arguments

[cols=3*,options=header]
|===
|Name
|Type
|Description

|code
|string
a|Argument code


|message
|string
a|Message argument


|===


[#returned_error]
[.api-collapsible-fifth-title]
returned_error

[cols=3*,options=header]
|===
```

```
|Name
|Type
|Description

|arguments
|array[link:#error_arguments[error_arguments]]
a|Message arguments


|code
|string
a|Error code


|message
|string
a|Error message


|target
|string
a|The target parameter that caused the error.


|===


//end collapsible .Definitions block
====



[[IDff99d27e7244836f72783fceb669bfdb]]
= Create or install security certificates

[.api-doc-operation .api-doc-operation-post]#POST# [.api-doc-code-
block]#`/security/certificates`#

*Introduced In:* 9.6

Creates or installs a certificate or downloads a certificate from Azure
Key Vault (AKV) and installs it on the ONTAP cluster.

== Required properties

* `svm.uuid` or `svm.name` - Existing SVM in which to create or install
the certificate.
```

```
* `common_name` - Common name of the certificate. Required when creating a
certificate.
* `type` - Type of certificate.
* `public_certificate` - Public key certificate in PEM format. Required
when installing a certificate.
* `private_key` - Private key certificate in PEM format. Required when
installing a CA-signed certificate.

== Recommended optional properties

* `expiry_time` - Certificate expiration time. Specifying an expiration
time is recommended when creating a certificate.
* `key_size` - Key size of the certificate in bits. Specifying a strong
key size is recommended when creating a certificate.
* `name` - Unique certificate name per SVM or the name of the certificate
in AKV, required for downloading AKV certificates. If one is not provided,
it is automatically generated.

== AKV required properties for downloading a certificate

* `azure.key_vault` - URI of the Azure Key Vault.
* `azure.client_id` - Application (client) ID of the deployed Azure
application with appropriate access to an AKV.
* `azure.tenant_id` - Directory (tenant) ID of the deployed Azure
application with appropriate access to an AKV.
* `azure.client_secret` - Secret used by the application to prove its
identity to AKV.
* `azure.client_certificate` - PKCS12 certificate used by the application
to prove its identity to AKV.

== AKV optional properties for downloading a certificate

* `azure.oauth_host` - Open authorization server host name.
* `azure.proxy.type` - Type of proxy (http, https etc.) if proxy
configuration is used.
* `azure.proxy.host` - Proxy hostname if proxy configuration is used.
* `azure.proxy.port` - Proxy port number if proxy configuration is used.
* `azure.proxy.username` - Proxy username if proxy configuration is used.
* `azure.proxy.password` - Proxy password if proxy configuration is used.
* `azure.timeout` - AKV connection timeout in seconds.
* `azure.verify_host` - Verify the identity of the AKV host name.

== Default property values

If not specified in POST, the following default property values are
assigned:
```

```
* `key_size` - _2048_
* `expiry_time` - _P365DT_
* `hash_function` - _sha256_

== Related ONTAP commands

* `security certificate create`
* `security certificate install`
* `security certificate azure-install`


== Parameters

[cols=5*,options=header]
|===

|Name
|Type
|In
|Required
|Description

|return_records
|boolean
|query
|False
a|The default is false.  If set to true, the records are returned.

* Default value:

|===

== Request Body



[cols=3*,options=header]
|===
|Name
|Type
|Description

|authority_key_identifier
|string
a|Provides the key identifier of the issuing CA certificate that signed
the SSL certificate.
```

```
|azure
|link:#azure[azure]
a|

|ca
|string
a|Certificate authority


|common_name
|string
a|FQDN or custom common name. Provide on POST when creating a self-signed
certificate.


|expiry_time
|string
a|Certificate expiration time, in ISO 8601 duration format or date and
time format. Can be provided on POST if creating self-signed certificate.
The expiration time range is between 1 day to 10 years.


|hash_function
|string
a|Hashing function. Can be provided on POST when creating a self-signed
certificate. Hash functions md5 and sha1 are not allowed on POST.


|intermediate_certificates
|array[string]
a|Chain of intermediate Certificates in PEM format. Only valid in POST
when installing a certificate.


|key_size
|integer
a|Key size of requested Certificate in bits. One of 512, 1024, 1536, 2048,
3072. Can be provided on POST if creating self-signed certificate with a
minimum permissible value of 2048.


|name
|string
a|Certificate name or name of the certificate to be downloaded from the
Azure Key Vault (AKV). If not provided in POST, a unique name specific to
the SVM is automatically generated.
```

```
|private_key
|string
a|Private key Certificate in PEM format. Only valid for create when
installing a CA-signed certificate. This is not audited.


|public_certificate
|string
a|Public key Certificate in PEM format. If this is not provided in POST, a
self-signed certificate is created.


|serial_number
|string
a|Serial number of certificate.


|subject_alternatives
|link:#subject_alternatives[subject_alternatives]
a|

|subject_key_identifier
|string
a|Provides the key identifier used to identify the public key in the SSL
certificate.


|svm
|link:#svm[svm]
a|SVM, applies only to SVM-scoped objects.


|type
|string
a|Type of Certificate. The following types are supported:

* client - a certificate and its private key used by an SSL client in
ONTAP.
* server - a certificate and its private key used by an SSL server in
ONTAP.
* client_ca - a Certificate Authority certificate used by an SSL server in
ONTAP to verify an SSL client certificate.
* server_ca - a Certificate Authority certificate used by an SSL client in
ONTAP to verify an SSL server certificate.
* root_ca - a self-signed certificate used by ONTAP to sign other
```

```
certificates by acting as a Certificate Authority.
* enum: ["client", "server", "client_ca", "server_ca", "root_ca"]
* Introduced in: 9.6
* x-nullable: true


|uuid
|string
a|Unique ID that identifies a certificate.


|===


.Example request
[%collapsible%closed]
====
[source,json,subs=+macros]
{
  "authority_key_identifier":
"26:1F:C5:53:5B:D7:9E:E2:37:74:F4:F4:06:09:03:3D:EB:41:75:D7",
  "azure": {
    "client_certificate": "PEM Cert",
    "client_id": "aaaaaaaa-bbbb-aaaa-bbbb-aaaaaaaaaaaa",
    "client_secret": "abcdef",
    "key_vault": "https://kmip-akv-keyvault.vault.azure.net/",
    "oauth_host": "login.microsoftonline.com",
    "proxy": {
      "host": "proxy.eng.com",
      "password": "proxypassword",
      "port": 1234,
      "type": "string",
      "username": "proxyuser"
    },
    "tenant_id": "zzzzzzzz-yyyy-zzzz-yyyy-zzzzzzzzzzzz",
    "timeout": 25
  },
  "ca": "string",
  "common_name": "test.domain.com",
  "expiry_time": "2030-01-25 06:20:13 -0500",
  "hash_function": "string",
  "intermediate_certificates": [
    "<CERTIFICATE-CONTENT>"
  ],
  "key_size": 512,
  "name": "string",
```

```
  "private_key": "-----BEGIN PRIVATE KEY-----\\nprivate-key\\n-----END
PRIVATE KEY-----\\n",
  "public_certificate": "<CERTIFICATE-CONTENT>",
  "serial_number": "string",
  "subject_alternatives": {
    "dns": [
      "*.example.com"
    ],
    "email": [
      "abc@example.com"
    ],
    "ip": [
      "10.225.34.10"
    ],
    "uri": [
      "http://example.com"
    ]
  },
  "subject_key_identifier":
"26:1F:C5:53:5B:D7:9E:E2:37:74:F4:F4:06:09:03:3D:EB:41:75:D8",
  "svm": {
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "type": "string",
  "uuid": "string"
}
====

== Response
```

Status: 201, Created

```
[cols=3*,options=header]
|===
|Name
|Type
|Description

|num_records
|integer
a|Number of records


|records
```

```
|array[link:#security_certificate[security_certificate]]
a|

|===


.Example response
[%collapsible%closed]
====
[source,json,subs=+macros]
{
  "num_records": 1,
  "records": [
    {
      "authority_key_identifier":
"26:1F:C5:53:5B:D7:9E:E2:37:74:F4:F4:06:09:03:3D:EB:41:75:D7",
      "azure": {
        "client_certificate": "PEM Cert",
        "client_id": "aaaaaaaa-bbbb-aaaa-bbbb-aaaaaaaaaaaa",
        "client_secret": "abcdef",
        "key_vault": "https://kmip-akv-keyvault.vault.azure.net/",
        "oauth_host": "login.microsoftonline.com",
        "proxy": {
          "host": "proxy.eng.com",
          "password": "proxypassword",
          "port": 1234,
          "type": "string",
          "username": "proxyuser"
        },
        "tenant_id": "zzzzzzzz-yyyy-zzzz-yyyy-zzzzzzzzzzzz",
        "timeout": 25
      },
      "ca": "string",
      "common_name": "test.domain.com",
      "expiry_time": "2030-01-25 06:20:13 -0500",
      "hash_function": "string",
      "intermediate_certificates": [
        "<CERTIFICATE-CONTENT>"
      ],
      "key_size": 512,
      "name": "string",
      "private_key": "-----BEGIN PRIVATE KEY-----\\nprivate-key\\n-----END
PRIVATE KEY-----\\n",
      "public_certificate": "<CERTIFICATE-CONTENT>",
      "serial_number": "string",
      "subject_alternatives": {
```

```
        "dns": [
          "*.example.com"
        ],
        "email": [
          "abc@example.com"
        ],
        "ip": [
          "10.225.34.10"
        ],
        "uri": [
          "http://example.com"
        ]
      },
      "subject_key_identifier":
"26:1F:C5:53:5B:D7:9E:E2:37:74:F4:F4:06:09:03:3D:EB:41:75:D8",
      "svm": {
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      },
      "type": "string",
      "uuid": "string"
    }
  ]
}
====

=== Headers

[cols=3*,options=header]
|===
//header
|Name
|Description
|Type
//end header

//start row
|Location
|Useful for tracking the resource location
|string
//end row
//end table
|===

== Error
```

Status: Default

```
ONTAP Error Response Codes
//start table
[cols=2*,options=header]
|===
//header
| Error Code | Description
//end header
//end row
//start row
|3735645  +
//end row
//start row
|Cannot specify a value for serial. It is generated automatically.
//end row
//start row
|
//end row
//start row
|3735622  +
//end row
//start row
|The certificate type is not supported.
//end row
//start row
|
//end row
//start row
|3735664  +
//end row
//start row
|The specified key size is not supported in FIPS mode.
//end row
//start row
|
//end row
//start row
|3735665  +
//end row
//start row
|The specified hash function is not supported in FIPS mode.
//end row
//start row
|
//end row
```

```
//start row
|3735553  +
//end row
//start row
|Failed to create self-signed Certificate.
//end row
//start row
|
//end row
//start row
|3735646  +
//end row
//start row
|Failed to store the certificates.
//end row
//start row
|
//end row
//start row
|3735693  +
//end row
//start row
|The certificate installation failed as private key was empty.
//end row
//start row
|
//end row
//start row
|3735618  +
//end row
//start row
|Cannot accept private key for server_ca or client_ca.
//end row
//start row
|
//end row
//start row
|52363365 +
//end row
//start row
|Failed to allocate memory.
//end row
//start row
|
//end row
//start row
```

```
|52559975 +
//end row
//start row
|Failed to read the certificate due to incorrect formatting.
//end row
//start row
|
//end row
//start row
|52363366 +
//end row
//start row
|Unsupported key type.
//end row
//start row
|
//end row
//start row
|52560123 +
//end row
//start row
|Failed to read the key due to incorrect formatting.
//end row
//start row
|
//end row
//start row
|52559972 +
//end row
//start row
|The certificates start date is later than the current date.
//end row
//start row
|
//end row
//start row
|52559976 +
//end row
//start row
|The certificate and private key do not match.
//end row
//start row
|
//end row
//start row
|52559973 +
```

```
//end row
//start row
|The certificate has expired.
//end row
//start row
|
//end row
//start row
|52363366 +
//end row
//start row
|Logic error: use of a dead object.
//end row
//start row
|
//end row
//start row
|3735696  +
//end row
//start row
|Intermediate certificates are not supported with client_ca and server_ca
type certificates.
//end row
//start row
|
//end row
//start row
|52559974 +
//end row
//start row
|The certificate is not supported in FIPS mode.
//end row
//start row
|
//end row
//start row
|3735676  +
//end row
//start row
|Cannot continue the installation without a value for the common name.
Since the subject field in the certificate is empty, the field
"common_name" must have a value to continue with the installation.
//end row
//start row
|
//end row
```

```
//start row
|3735558  +
//end row
//start row
|Failed to extract information about Common Name from the certificate.
//end row
//start row
|
//end row
//start row
|3735588  +
//end row
//start row
|The common name (CN) extracted from the certificate is not valid.
//end row
//start row
|
//end row
//start row
|3735632  +
//end row
//start row
|Failed to extract Certificate Authority Information from the certificate.
//end row
//start row
|
//end row
//start row
|3735700  +
//end row
//start row
|The specified key size is not supported.
//end row
//start row
|
//end row
//start row
|52560173 +
//end row
//start row
|The hash function is not supported for digital signatures.
//end row
//start row
|
//end row
//start row
```

```
|3735751  +
//end row
//start row
|Failed to authenticate and fetch the access token from Azure OAuth host.
//end row
//start row
|
//end row
//start row
|3735752  +
//end row
//start row
|Failed to extract the private key from the Azure Key Vault certificate.
//end row
//start row
|3735753  +
//end row
//start row
|Unsupported content_type in the Azure secrets response.
//end row
//start row
|3735754  +
//end row
//start row
|Internal error. Failed to parse the JSON response from Azure Key Vault.
//end row
//start row
|3735755  +
//end row
//start row
|REST call to Azure failed.
//end row
//start row
|3735756  +
//end row
//start row
|Invalid client certificate.
//end row
//start row
|3735757  +
//end row
//start row
|Internal error. Failed to generate client assertion.
//end row
//start row
|3735762  +
```

```
//end row
//start row
|Provided Azure Key Vault configuration is incorrect.
//end row
//start row
|3735763  +
//end row
//start row
|Provided Azure Key Vault configuration is incomplete.
//end row
//start row
|3735764  +
//end row
//start row
|Request to Azure failed. Reason - Azure error code and Azure error
message.
//end row
|===
//end table


== Definitions

[.api-def-first-level]
.See Definitions
[%collapsible%closed]
//Start collapsible Definitions block
====
[#href]
[.api-collapsible-fifth-title]
href

[cols=3*,options=header]
|===
|Name
|Type
|Description

|href
|string
a|

|===


[#_links]
```

```
[.api-collapsible-fifth-title]
_links
[#proxy]
[.api-collapsible-fifth-title]
proxy

[cols=3*,options=header]
|===
|Name
|Type
|Description

|host
|string
a|Proxy host.


|password
|string
a|Proxy password. Password is not audited.


|port
|integer
a|Proxy port.


|type
|string
a|Proxy type.


|username
|string
a|Proxy username.


|===


[#azure]
[.api-collapsible-fifth-title]
azure

[cols=3*,options=header]
|===
|Name
```

```
|Type
|Description

|client_certificate
|string
a|PKCS12 certificate used by the application to prove its identity to AKV.


|client_id
|string
a|Application client ID of the deployed Azure application with appropriate
access to an AKV.


|client_secret
|string
a|Secret used by the application to prove its identity to AKV.


|key_vault
|string
a|URI of the deployed AKV that is used by ONTAP for storing keys.

* example: https://kmip-akv-keyvault.vault.azure.net/
* format: uri
* x-ntap-createOnly: true
* Introduced in: 9.14
* x-nullable: true


|oauth_host
|string
a|Open authorization server host name.


|proxy
|link:#proxy[proxy]
a|

|tenant_id
|string
a|Directory (tenant) ID of the deployed Azure application with appropriate
access to an AKV.


|timeout
|integer
```

```
a|AKV connection timeout, in seconds. The allowed range is between 0 to 30
seconds.


|verify_host
|boolean
a|Verify the identity of the AKV host name. By default, verify_host is set
to true.


|===


[#subject_alternatives]
[.api-collapsible-fifth-title]
subject_alternatives

[cols=3*,options=header]
|===
|Name
|Type
|Description

|dns
|array[string]
a|A list of DNS names for Subject Alternate name extension.


|email
|array[string]
a|A list of email addresses for Subject Alternate name extension


|ip
|array[string]
a|A list of IP addresses for Subject Alternate name extension.


|uri
|array[string]
a|A list of URIs for Subject Alternate name extension.


|===


[#svm]
```

```
[.api-collapsible-fifth-title]
svm

SVM, applies only to SVM-scoped objects.


[cols=3*,options=header]
|===
|Name
|Type
|Description

|name
|string
a|The name of the SVM. This field cannot be specified in a PATCH method.


|uuid
|string
a|The unique identifier of the SVM. This field cannot be specified in a
PATCH method.


|===


[#security_certificate]
[.api-collapsible-fifth-title]
security_certificate

[cols=3*,options=header]
|===
|Name
|Type
|Description

|authority_key_identifier
|string
a|Provides the key identifier of the issuing CA certificate that signed
the SSL certificate.


|azure
|link:#azure[azure]
a|

|ca
```

```
|string
a|Certificate authority


|common_name
|string
a|FQDN or custom common name. Provide on POST when creating a self-signed
certificate.


|expiry_time
|string
a|Certificate expiration time, in ISO 8601 duration format or date and
time format. Can be provided on POST if creating self-signed certificate.
The expiration time range is between 1 day to 10 years.


|hash_function
|string
a|Hashing function. Can be provided on POST when creating a self-signed
certificate. Hash functions md5 and sha1 are not allowed on POST.


|intermediate_certificates
|array[string]
a|Chain of intermediate Certificates in PEM format. Only valid in POST
when installing a certificate.


|key_size
|integer
a|Key size of requested Certificate in bits. One of 512, 1024, 1536, 2048,
3072. Can be provided on POST if creating self-signed certificate with a
minimum permissible value of 2048.


|name
|string
a|Certificate name or name of the certificate to be downloaded from the
Azure Key Vault (AKV). If not provided in POST, a unique name specific to
the SVM is automatically generated.


|private_key
|string
a|Private key Certificate in PEM format. Only valid for create when
installing a CA-signed certificate. This is not audited.
```

```
|public_certificate
|string
a|Public key Certificate in PEM format. If this is not provided in POST, a
self-signed certificate is created.


|serial_number
|string
a|Serial number of certificate.


|subject_alternatives
|link:#subject_alternatives[subject_alternatives]
a|

|subject_key_identifier
|string
a|Provides the key identifier used to identify the public key in the SSL
certificate.


|svm
|link:#svm[svm]
a|SVM, applies only to SVM-scoped objects.


|type
|string
a|Type of Certificate. The following types are supported:

* client - a certificate and its private key used by an SSL client in
ONTAP.
* server - a certificate and its private key used by an SSL server in
ONTAP.
* client_ca - a Certificate Authority certificate used by an SSL server in
ONTAP to verify an SSL client certificate.
* server_ca - a Certificate Authority certificate used by an SSL client in
ONTAP to verify an SSL server certificate.
* root_ca - a self-signed certificate used by ONTAP to sign other
certificates by acting as a Certificate Authority.
* enum: ["client", "server", "client_ca", "server_ca", "root_ca"]
* Introduced in: 9.6
* x-nullable: true
```

```
|uuid
|string
a|Unique ID that identifies a certificate.


|===


[#error_arguments]
[.api-collapsible-fifth-title]
error_arguments

[cols=3*,options=header]
|===
|Name
|Type
|Description

|code
|string
a|Argument code


|message
|string
a|Message argument


|===


[#returned_error]
[.api-collapsible-fifth-title]
returned_error

[cols=3*,options=header]
|===
|Name
|Type
|Description

|arguments
|array[link:#error_arguments[error_arguments]]
a|Message arguments


|code
```

```
|string
a|Error code


|message
|string
a|Error message


|target
|string
a|The target parameter that caused the error.


|===


//end collapsible .Definitions block
====



[[IDd2bce8f0002ee83635e2bb0705d49029]]
= Sign security certificates

[.api-doc-operation .api-doc-operation-post]#POST# [.api-doc-code-
block]#`/security/certificates/{ca.uuid}/sign`#

*Introduced In:* 9.6

Signs a certificate.

== Required properties

* `signing_request` - Certificate signing request to be signed by the
given certificate authority.

== Recommended optional properties

* `expiry_time` - Certificate expiration time. Specifying an expiration
time for a signed certificate is recommended.
* `hash_function` - Hashing function. Specifying a strong hashing function
is recommended when signing a certificate.

== Default property values

If not specified in POST, the following default property values are
```

```
assigned:

* `expiry_time` - _P365DT_
* `hash_function` - _sha256_

== Related ONTAP commands

* `security certificate sign`
This API is used to sign a certificate request using a pre-existing self-signed root certificate. The self-signed root certificate acts as a certificate authority within its scope and maintains the records of its signed certificates.

The root certificate can be created for a given SVM or for the cluster using [`POST security/certificates`].


== Parameters

[cols=5*,options=header]
|===

|Name
|Type
|In
|Required
|Description

|ca.uuid
|string
|path
|True
a|UUID of the existing certificate authority certificate


|return_records
|boolean
|query
|False
a|The default is false.  If set to true, the records are returned.

* Default value:

|===

== Request Body
```

```
[cols=3*,options=header]
|===
|Name
|Type
|Description

|expiry_time
|string
a|Certificate expiration time, in ISO 8601 duration format or date and
time format. The allowed expiration time range is between 1 day to 10
years.


|hash_function
|string
a|Hashing function


|signing_request
|string
a|Certificate signing request to be signed by the given certificate
authority. Request should be in X509 PEM format.


|===


.Example request
[%collapsible%closed]
====
[source,json,subs=+macros]
{
  "expiry_time": "P1DT2H3M4S or '2030-01-25T11:20:13Z'",
  "hash_function": "string",
  "signing_request": "<CERTIFICATE-CONTENT>"
}
====

== Response
```

Status: 200, Ok

```
[cols=3*,options=header]
|===
|Name
|Type
|Description

|public_certificate
|string
a|CA signed public key Certificate


|===


.Example response
[%collapsible%closed]
====
[source,json,subs=+macros]
{
  "public_certificate": "string"
}
====

== Error
```

Status: Default

```
ONTAP Error Response Codes

|===
| Error Code | Description

| 3735628
| Failed to use CA certificate for signing.

| 3735665
| The specified hash function is not supported in FIPS mode.

| 52559974
| The certificate is not supported in FIPS mode.

| 3735626
| Failed to generate signed Certificate.

| 3735558
```

```
| Failed to extract information about Common Name from the certificate.

| 3735588
| The common name (CN) extracted from the certificate is not valid.

| 3735632
| Failed to extract Certificate Authority Information from the
certificate.

| 3735629
| Failed to sign the certificate because Common Name of signing
certificate and Common Name of CA certificate are same.

| 3735630
| Failed to sign the certificate because expiry date of signing
certificate exceeds the expiry date of CA certificate.

| 3735701
| Invalid expiration period. The allowed range for expiration time is
between 1 and 3652 days.
|===


== Definitions

[.api-def-first-level]
.See Definitions
[%collapsible%closed]
//Start collapsible Definitions block
====
[#security_certificate_sign]
[.api-collapsible-fifth-title]
security_certificate_sign

[cols=3*,options=header]
|===
|Name
|Type
|Description

|expiry_time
|string
a|Certificate expiration time, in ISO 8601 duration format or date and
time format. The allowed expiration time range is between 1 day to 10
years.
```

```
|hash_function
|string
a|Hashing function


|signing_request
|string
a|Certificate signing request to be signed by the given certificate
authority. Request should be in X509 PEM format.


|===


[#error_arguments]
[.api-collapsible-fifth-title]
error_arguments

[cols=3*,options=header]
|===
|Name
|Type
|Description

|code
|string
a|Argument code


|message
|string
a|Message argument


|===


[#returned_error]
[.api-collapsible-fifth-title]
returned_error

[cols=3*,options=header]
|===
|Name
|Type
|Description
```

```
|arguments
|array[link:#error_arguments[error_arguments]]
a|Message arguments


|code
|string
a|Error code


|message
|string
a|Error message


|target
|string
a|The target parameter that caused the error.


|===


//end collapsible .Definitions block
====



[[IDa6bb13f96cf93162fa4abab1433eef66]]
= Delete security certificates

[.api-doc-operation .api-doc-operation-delete]#DELETE# [.api-doc-code-
block]#`/security/certificates/{uuid}`#

*Introduced In:* 9.6

Deletes a security certificate.

== Related ONTAP commands

* `security certificate delete`


== Parameters

[cols=5*,options=header]
|===
```

```
|Name
|Type
|In
|Required
|Description

|uuid
|string
|path
|True
a|Certificate UUID

|===

== Response
```

Status: 200, Ok

```
== Error
```

Status: Default

```
ONTAP Error Response Codes

|===
| Error Code | Description

| 3735644
| Cannot delete server-chain certificate. Reason: There is a corresponding
server certificate for it.

| 3735679
| Cannot delete pre-installed server_ca certificates through REST. Use CLI
or ZAPI.

| 3735650
| Deleting this client_ca certificate directly is not supported. Delete
the corresponding root-ca certificate using type `root_ca` to delete the
root, client, and server certificates.

| 3735627
| Deleting this server_ca certificate directly is not supported. Delete
the corresponding root-ca certificate using type `root_ca` to delete the
root, client, and server certificates.
```

```
| 3735589
| Cannot delete certificate.

| 3735590
| Cannot delete certificate. Failed to remove SSL configuration for the
certificate.

| 3735683
| Cannot remove this certificate while external key manager is configured.

| 3735681
| Cannot delete preinstalled `server-ca` certificates. Use the CLI to
complete the operation.

| 52560272
| The certificate could not be removed due to being in use by one or more
subsystems.
|===



[cols=3*,options=header]
|===
|Name
|Type
|Description

|error
|link:#returned_error[returned_error]
a|

|===


.Example error
[%collapsible%closed]
====
[source,json,subs=+macros]
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
```

```
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
====

== Definitions

[.api-def-first-level]
.See Definitions
[%collapsible%closed]
//Start collapsible Definitions block
====
[#error_arguments]
[.api-collapsible-fifth-title]
error_arguments

[cols=3*,options=header]
|===
|Name
|Type
|Description

|code
|string
a|Argument code


|message
|string
a|Message argument


|===


[#returned_error]
[.api-collapsible-fifth-title]
returned_error

[cols=3*,options=header]
|===
|Name
|Type
|Description
```

```
|arguments
|array[link:#error_arguments[error_arguments]]
a|Message arguments


|code
|string
a|Error code


|message
|string
a|Error message


|target
|string
a|The target parameter that caused the error.


|===


//end collapsible .Definitions block
====



[[IDe25c8d469dc87afcbc59048dc7f2cfc4]]
= Retrieve security certificates

[.api-doc-operation .api-doc-operation-get]#GET# [.api-doc-code-
block]#`/security/certificates/{uuid}`#

*Introduced In:* 9.6

Retrieves security certificates.

== Related ONTAP commands

* `security certificate show`


== Parameters

[cols=5*,options=header]
|===
```

```
|Name
|Type
|In
|Required
|Description

|uuid
|string
|path
|True
a|Certificate UUID


|fields
|array[string]
|query
|False
a|Specify the fields to return.

|===

== Response
```

Status: 200, Ok

```
[cols=3*,options=header]
|===
|Name
|Type
|Description

|_links
|link:#_links[_links]
a|

|authority_key_identifier
|string
a|Provides the key identifier of the issuing CA certificate that signed
the SSL certificate.


|azure
|link:#azure[azure]
a|
```

```
|ca
|string
a|Certificate authority


|common_name
|string
a|FQDN or custom common name. Provide on POST when creating a self-signed
certificate.


|expiry_time
|string
a|Certificate expiration time, in ISO 8601 duration format or date and
time format. Can be provided on POST if creating self-signed certificate.
The expiration time range is between 1 day to 10 years.


|hash_function
|string
a|Hashing function. Can be provided on POST when creating a self-signed
certificate. Hash functions md5 and sha1 are not allowed on POST.


|key_size
|integer
a|Key size of requested Certificate in bits. One of 512, 1024, 1536, 2048,
3072. Can be provided on POST if creating self-signed certificate with a
minimum permissible value of 2048.


|name
|string
a|Certificate name or name of the certificate to be downloaded from the
Azure Key Vault (AKV). If not provided in POST, a unique name specific to
the SVM is automatically generated.


|public_certificate
|string
a|Public key Certificate in PEM format. If this is not provided in POST, a
self-signed certificate is created.


|scope
|string
a|Set to "svm" for interfaces owned by an SVM. Otherwise, set to
```

```
"cluster".


|serial_number
|string
a|Serial number of certificate.


|subject_alternatives
|link:#subject_alternatives[subject_alternatives]
a|

|subject_key_identifier
|string
a|Provides the key identifier used to identify the public key in the SSL
certificate.


|svm
|link:#svm[svm]
a|SVM, applies only to SVM-scoped objects.


|type
|string
a|Type of Certificate. The following types are supported:

* client - a certificate and its private key used by an SSL client in
ONTAP.
* server - a certificate and its private key used by an SSL server in
ONTAP.
* client_ca - a Certificate Authority certificate used by an SSL server in
ONTAP to verify an SSL client certificate.
* server_ca - a Certificate Authority certificate used by an SSL client in
ONTAP to verify an SSL server certificate.
* root_ca - a self-signed certificate used by ONTAP to sign other
certificates by acting as a Certificate Authority.
* enum: ["client", "server", "client_ca", "server_ca", "root_ca"]
* Introduced in: 9.6
* x-nullable: true


|uuid
|string
a|Unique ID that identifies a certificate.
```

```
|===


.Example response
[%collapsible%closed]
====
[source,json,subs=+macros]
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "authority_key_identifier":
"26:1F:C5:53:5B:D7:9E:E2:37:74:F4:F4:06:09:03:3D:EB:41:75:D7",
  "azure": {
    "verify_host": true
  },
  "ca": "string",
  "common_name": "test.domain.com",
  "expiry_time": "2030-01-25 06:20:13 -0500",
  "hash_function": "string",
  "key_size": 512,
  "name": "string",
  "public_certificate": "<CERTIFICATE-CONTENT>",
  "scope": "string",
  "serial_number": "string",
  "subject_alternatives": {
    "dns": [
      "*.example.com"
    ],
    "email": [
      "abc@example.com"
    ],
    "ip": [
      "10.225.34.10"
    ],
    "uri": [
      "http://example.com"
    ]
  },
  "subject_key_identifier":
"26:1F:C5:53:5B:D7:9E:E2:37:74:F4:F4:06:09:03:3D:EB:41:75:D8",
  "svm": {
    "_links": {
      "self": {
```

```
        "href": "/api/resourcelink"
      }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "type": "string",
  "uuid": "string"
}
====

== Error
```

Status: Default, Error

```
[cols=3*,options=header]
|===
|Name
|Type
|Description

|error
|link:#returned_error[returned_error]
a|

|===


.Example error
[%collapsible%closed]
====
[source,json,subs=+macros]
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

```
====

== Definitions

[.api-def-first-level]
.See Definitions
[%collapsible%closed]
//Start collapsible Definitions block
====
[#href]
[.api-collapsible-fifth-title]
href

[cols=3*,options=header]
|===
|Name
|Type
|Description

|href
|string
a|

|===


[#_links]
[.api-collapsible-fifth-title]
_links

[cols=3*,options=header]
|===
|Name
|Type
|Description

|self
|link:#href[href]
a|

|===


[#proxy]
[.api-collapsible-fifth-title]
proxy
[#azure]
```

```
[.api-collapsible-fifth-title]
azure

[cols=3*,options=header]
|===
|Name
|Type
|Description

|proxy
|link:#proxy[proxy]
a|

|===


[#subject_alternatives]
[.api-collapsible-fifth-title]
subject_alternatives

[cols=3*,options=header]
|===
|Name
|Type
|Description

|dns
|array[string]
a|A list of DNS names for Subject Alternate name extension.


|email
|array[string]
a|A list of email addresses for Subject Alternate name extension


|ip
|array[string]
a|A list of IP addresses for Subject Alternate name extension.


|uri
|array[string]
a|A list of URIs for Subject Alternate name extension.


|===
```

```
[#svm]
[.api-collapsible-fifth-title]
svm

SVM, applies only to SVM-scoped objects.


[cols=3*,options=header]
|===
|Name
|Type
|Description

|_links
|link:#_links[_links]
a|

|name
|string
a|The name of the SVM. This field cannot be specified in a PATCH method.


|uuid
|string
a|The unique identifier of the SVM. This field cannot be specified in a
PATCH method.


|===


[#error_arguments]
[.api-collapsible-fifth-title]
error_arguments

[cols=3*,options=header]
|===
|Name
|Type
|Description

|code
|string
a|Argument code
```

```
|message
|string
a|Message argument


|===


[#returned_error]
[.api-collapsible-fifth-title]
returned_error

[cols=3*,options=header]
|===
|Name
|Type
|Description

|arguments
|array[link:#error_arguments[error_arguments]]
a|Message arguments


|code
|string
a|Error code


|message
|string
a|Error message


|target
|string
a|The target parameter that caused the error.


|===


//end collapsible .Definitions block
====
```

```
:leveloffset: -1


:leveloffset: -1

<<<
*Copyright information*

Copyright © 2026 NetApp, Inc. All Rights Reserved. Printed in the U.S. No
part of this document covered by copyright may be reproduced in any form
or by any means—graphic, electronic, or mechanical, including
photocopying, recording, taping, or storage in an electronic retrieval
system—without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the
following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP “AS IS” AND WITHOUT ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY
DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any
time, and without notice. NetApp assumes no responsibility or liability
arising from the use of products described herein, except as expressly
agreed to in writing by NetApp. The use or purchase of this product does
not convey a license under any patent rights, trademark rights, or any
other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S.
patents, foreign patents, or pending applications.

LIMITED RIGHTS LEGEND: Use, duplication, or disclosure by the government
is subject to restrictions as set forth in subparagraph (b)(3) of the
Rights in Technical Data -Noncommercial Items at DFARS 252.227-7013 (FEB
2014) and FAR 52.227-19 (DEC 2007).

Data contained herein pertains to a commercial product and/or commercial
service (as defined in FAR 2.101) and is proprietary to NetApp, Inc. All
NetApp technical data and computer software provided under this Agreement
is commercial in nature and developed solely at private expense.  The U.S.
```