



# **Manage cluster licensing**

## **ONTAP 9.12.1 REST API reference**

NetApp  
May 08, 2024

This PDF was generated from [https://docs.netapp.com/us-en/ontap-restapi-9121/ontap/cluster\\_licensing\\_licenses\\_endpoint\\_overview.html](https://docs.netapp.com/us-en/ontap-restapi-9121/ontap/cluster_licensing_licenses_endpoint_overview.html) on May 08, 2024. Always check docs.netapp.com for the latest.

# Table of Contents

- Manage cluster licensing ..... 1
  - Cluster licensing licenses endpoint overview ..... 1
  - Retrieve license packages ..... 20
  - Install one or more feature licenses ..... 28
  - Delete a license ..... 39
  - Retrieve a license package ..... 41

# Manage cluster licensing

## Cluster licensing licenses endpoint overview

### Overview

Licensing allows you to tailor a system to meet an organization's specific needs. You can enable new features by purchasing a license from a NetApp sales associate. After installation of the license, the new feature is available immediately.

This interface manages licenses according to their supported feature. By default, the interface displays packages with installed licenses, but you can also return unlicensed packages.

Each feature has a compliance state that is indicated at the package level. Individual licenses also contain a compliance state indicated in the "licenses" array. The state of the package is determined by analyzing the underlying licenses according to the following criteria:

- Licensing terms
- Cluster state

### Licensing terms

The licensing terms define the conditions under which a package is considered "compliant". Individual licenses are evaluated based on the following:

- Scope
- Time period
- Usage

### Scope

A package can be licensed under the following scopes:

- Site - Permits the feature to be used by any node in any cluster.
- Cluster - Permits the feature to be used by any node in a single specific cluster.
- Node - Permits the authorized node to use the feature. Within a cluster, if you don't supply every node with a valid license, the package state indicates "noncompliant". You must purchase a license for each node in a cluster for the package to be considered "compliant".

### Time period

Some package licenses are only valid for a limited period of time. After a license has expired, the package state changes to "noncompliant". You need to purchase a new license for the package to return to a "compliant" state.

### Usage

Some package licenses have additional terms that need to be maintained to keep a license in compliance. These conditions are defined by the individual license. For example, a license might define the maximum amount of storage that a node can allocate for the license to be "compliant".

## Cluster state

A cluster's state consists of the following:

- Node online status
- Node cluster membership

Some features require that a node be online to display a valid compliance state. If a node cannot be reached or is not known to the cluster, the individual license might indicate an "unknown" state.

## Licensing keys

A license is issued in one of the following three formats:

- 28-character key
- NetApp License File Version 1 (NLFv1)
- NetApp License File Version 2 (NLFv2)

### Overview of NLFv1 and NLFv2 License Formats

NLFv1 and NLFv2 licenses are both JSON based files that allow features to be enabled.

The difference between the two formats is that a NLFv2 license allows multiple features to be enabled with a single file. A NLFv1 license is capable of enabling a single feature.

These licenses are identified, in the various methods, as follows:

| Format           | Identifying Keys                           |
|------------------|--|
| 28 Character Key | name / serial_number                       |
| NLFv1            | name / serial_number                       |
| NLFv2            | licenses.installed_license / serial_number |

The following is an example of a 28-character key:

```
AMEPOSOIKLKGEEDGNDEKSJDEEE
```

The following is an example of an NLFv1 key:

```
{
  "statusResp": {
    "version": "1",
    "serialNumber": "123456789",
    "message": "Success",
    "licenses": {
      "capacity": "1",
      "type": "capacity",
      "licenseProtocol": "FABRICPOOL-TB",
      "package": "FabricPool",
      "licenseScope": "cluster"
    },
    "snStatus": "Active",
    "product": "fabricpool",
    "statusCode": "S007"
  },
  "Signature": "signatureABC"
}
```

The following is an example of an NLFv2 key:

```
{
  "statusResp": {
    "version": "2",
    "serialNumber": "123456789",
    "message": "Success",
    "product": "Sample NLFv2 License",
    "licenses": {
      "capacity": "1",
      "type": "capacity",
      "HostID": "5554444",
      "package": [ "NFS", "CIFS" ],
      "licenseScope": "node"
    },
    "snStatus": "Active",
    "statusCode": "S007"
  },
  "Signature": "signatureABC"
}
```

You can use this API to submit any format to enable features.

## Examples

### Retrieving a collection of licenses organized by package

This example retrieves a collection that contains one entry for each package (filtered to only the 'fabricpool' package).

```
# API
curl -X GET "https://<mgmt-
ip>/api/cluster/licensing/licenses?fields=*&name=fabricpool"

# Response
200 OK

# JSON Body
{
  "records": [
    {
      "name": "fabricpool",
      "scope": "cluster",
      "state": "compliant",
      "description": "FabricPool License",
      "licenses": [
        {
          "owner": "testcluster-1",
          "serial_number": "4149027342",
          "state": "compliant",
          "capacity": {
            "maximum_size": 1099511627776,
            "used_size": 0
          }
        }
      ],
      "_links": {
        "self": {
          "href": "/api/cluster/licensing/licenses/fabricpool"
        }
      }
    }
  ],
  "num_records": 1,
  "_links": {
    "self": {
      "href": "/api/cluster/licensing/licenses/?fields=*&name=fabricpool"
    }
  }
}
```

### Retrieving a collection of licenses organized by package - for package cloud

The following example retrieves a collection that contains one entry for each package (filtered to only the 'cloud' package). The cloud package, in this example, is in the enforcement period as the license has expired. The REST GET output displays an additional field 'shutdown\_imminent' to indicate that the system will

shutdown.

```
# API
curl -X GET "https://<mgmt-
ip>/api/cluster/licensing/licenses?fields=*&name=cloud"

# Response
200 OK

# JSON Body
{
  "records": [
    {
      "name": "cloud",
      "scope": "node",
      "state": "noncompliant",
      "description": "Cloud ONTAP License",
      "entitlement": {
        "action": "acquire_license",
        "risk": "unlicensed"
      },
      "licenses": [
        {
          "owner": "test-vsim1",
          "serial_number": "90120130000000000001",
          "active": false,
          "evaluation": true,
          "expiry_time": "2021-10-26T19:57:41Z",
          "shutdown_imminent": true,
          "compliance": {
            "state": "noncompliant"
          }
        }
      ],
      "_links": {
        "self": {
          "href": "/api/cluster/licensing/licenses/cloud"
        }
      }
    }
  ],
  "num_records": 1,
  "_links": {
    "self": {
      "href": "/api/cluster/licensing/licenses/?fields=*&name=cloud"
    }
  }
}
```



```
}  
}
```

## Retrieving a collection of licenses installed with NLFv2

This example retrieves a collection of licenses that were installed by a NLFv2 formatted license.



The license is referenced by the installed license "Core\*Bundle" and the license serial number "4212426890"

```
# API  
curl -X GET "https://<mgmt-  
ip>/api/cluster/licensing/licenses?fields=*&licenses.installed_license=Cor  
e*Bundle&serial_number=4212426890"  
  
# Response  
200 OK  
  
# JSON Body  
{  
  "records": [  
    {  
      "name": "nfs",  
      "scope": "node",  
      "state": "noncompliant",  
      "description": "NFS License",  
      "entitlement": {  
        "action": "acquire_license",  
        "risk": "medium"  
      },  
      "licenses": [  
        {  
          "owner": "test-vsim3",  
          "active": false,  
          "evaluation": false,  
          "compliance": {  
            "state": "unlicensed"  
          }  
        },  
        {  
          "owner": "test-vsim4",  
          "installed_license": "Core Bundle",  
          "host_id": "4212426890",  
          "serial_number": "4212426890",  
          "active": true,  
          "evaluation": false,
```

```

    "compliance": {
      "state": "compliant"
    },
    "capacity": {
      "maximum_size": 10995116277760
    }
  }
],
"_links": {
  "self": {
    "href":
"/api/cluster/licensing/licenses/nfs/?licenses.installed_license=Core*Bundle"
  }
},
{
  "name": "cifs",
  "scope": "node",
  "state": "noncompliant",
  "description": "CIFS License",
  "entitlement": {
    "action": "acquire_license",
    "risk": "medium"
  },
  "licenses": [
    {
      "owner": "test-vsim3",
      "active": false,
      "evaluation": false,
      "compliance": {
        "state": "unlicensed"
      }
    },
    {
      "owner": "test-vsim4",
      "installed_license": "Core Bundle",
      "host_id": "4212426890",
      "serial_number": "4212426890",
      "active": true,
      "evaluation": false,
      "compliance": {
        "state": "compliant"
      },
      "capacity": {
        "maximum_size": 10995116277760
      }
    }
  ]
}

```

```

    }
  }
],
"_links": {
  "self": {
    "href":
"/api/cluster/licensing/licenses/cifs/?licenses.installed_license=Core*Bundle"
  }
}
},
{
  "name": "iscsi",
  "scope": "node",
  "state": "noncompliant",
  "description": "iSCSI License",
  "entitlement": {
    "action": "acquire_license",
    "risk": "medium"
  },
  "licenses": [
    {
      "owner": "test-vsim3",
      "active": false,
      "evaluation": false,
      "compliance": {
        "state": "unlicensed"
      }
    },
    {
      "owner": "test-vsim4",
      "installed_license": "Core Bundle",
      "host_id": "4212426890",
      "serial_number": "4212426890",
      "active": true,
      "evaluation": false,
      "compliance": {
        "state": "compliant"
      },
      "capacity": {
        "maximum_size": 10995116277760
      }
    }
  ],
  "_links": {
    "self": {

```

```

      "href":
"/api/cluster/licensing/licenses/iscsi/?licenses.installed_license=Core*Bu
ndle"
    }
  },
  {
    "name": "fcp",
    "scope": "node",
    "state": "noncompliant",
    "description": "FCP License",
    "entitlement": {
      "action": "acquire_license",
      "risk": "medium"
    },
    "licenses": [
      {
        "owner": "test-vsim3",
        "active": false,
        "evaluation": false,
        "compliance": {
          "state": "unlicensed"
        }
      },
      {
        "owner": "test-vsim4",
        "installed_license": "Core Bundle",
        "host_id": "4212426890",
        "serial_number": "4212426890",
        "active": true,
        "evaluation": false,
        "compliance": {
          "state": "compliant"
        },
        "capacity": {
          "maximum_size": 10995116277760
        }
      }
    ],
    "_links": {
      "self": {
        "href":
"/api/cluster/licensing/licenses/fcp/?licenses.installed_license=Core*Bund
le"
      }
    }
  }
}

```

```

},
{
  "name": "snaprestore",
  "scope": "node",
  "state": "noncompliant",
  "description": "SnapRestore License",
  "entitlement": {
    "action": "acquire_license",
    "risk": "medium"
  },
  "licenses": [
    {
      "owner": "test-vsim3",
      "active": false,
      "evaluation": false,
      "compliance": {
        "state": "unlicensed"
      }
    },
    {
      "owner": "test-vsim4",
      "installed_license": "Core Bundle",
      "host_id": "4212426890",
      "serial_number": "4212426890",
      "active": true,
      "evaluation": false,
      "compliance": {
        "state": "compliant"
      },
      "capacity": {
        "maximum_size": 10995116277760
      }
    }
  ],
  "_links": {
    "self": {
      "href":
"/api/cluster/licensing/licenses/snaprestore/?licenses.installed_license=Core*Bundle"
    }
  }
},
{
  "name": "flexclone",
  "scope": "node",
  "state": "noncompliant",

```

```

    "description": "FlexClone License",
    "entitlement": {
      "action": "acquire_license",
      "risk": "medium"
    },
    "licenses": [
      {
        "owner": "test-vsim3",
        "active": false,
        "evaluation": false,
        "compliance": {
          "state": "unlicensed"
        }
      },
      {
        "owner": "test-vsim4",
        "installed_license": "Core Bundle",
        "host_id": "4212426890",
        "serial_number": "4212426890",
        "active": true,
        "evaluation": false,
        "compliance": {
          "state": "compliant"
        },
        "capacity": {
          "maximum_size": 10995116277760
        }
      }
    ],
    "_links": {
      "self": {
        "href":
"/api/cluster/licensing/licenses/flexclone/?licenses.installed_license=Core*Bundle"
      }
    }
  },
  {
    "name": "nvme_of",
    "scope": "node",
    "state": "noncompliant",
    "description": "NVMe-oF License",
    "licenses": [
      {
        "owner": "test-vsim3",
        "active": false,

```

```

    "evaluation": false,
    "compliance": {
      "state": "unlicensed"
    }
  },
  {
    "owner": "test-vsim4",
    "installed_license": "Core Bundle",
    "host_id": "4212426890",
    "serial_number": "4212426890",
    "active": true,
    "evaluation": false,
    "compliance": {
      "state": "compliant"
    },
    "capacity": {
      "maximum_size": 10995116277760
    }
  }
],
"_links": {
  "self": {
    "href":
"/api/cluster/licensing/licenses/nvme_of/?licenses.installed_license=Core*
Bundle"
  }
},
{
  "name": "s3",
  "scope": "node",
  "state": "noncompliant",
  "description": "S3 License",
  "entitlement": {
    "action": "acquire_license",
    "risk": "medium"
  }
},
"licenses": [
  {
    "owner": "test-vsim3",
    "active": false,
    "evaluation": false,
    "compliance": {
      "state": "unlicensed"
    }
  }
],

```

```

{
  "owner": "test-vsimg4",
  "installed_license": "Core Bundle",
  "host_id": "4212426890",
  "serial_number": "4212426890",
  "active": true,
  "evaluation": false,
  "compliance": {
    "state": "compliant"
  },
  "capacity": {
    "maximum_size": 10995116277760
  }
},
{
  "_links": {
    "self": {
      "href":
"/api/cluster/licensing/licenses/s3/?licenses.installed_license=Core*Bundle"
    }
  }
},
{
  "num_records": 8,
  "_links": {
    "self": {
      "href":
"/api/cluster/licensing/licenses?fields=*&licenses.installed_license=Core*Bundle&serial_number=4212426890"
    }
  }
}
}

```

### Retrieving a collection of installed licenses

This example retrieves a collection containing all packages (except base) that have installed licenses.

```

# API
curl -X GET "https://<mgmt-
ip>/api/cluster/licensing/licenses?fields=*&name=!base"

# Response
200 OK

# JSON Body

```



```

{
  "records": [
    {
      "name": "nfs",
      "scope": "node",
      "state": "compliant",
      "description": "NFS License",
      "entitlement": {
        "action": "none",
        "risk": "low"
      },
      "licenses": [
        {
          "owner": "testcluster-1",
          "serial_number": "1-81-0000000000000004149027492",
          "state": "compliant"
        }
      ],
      "_links": {
        "self": {
          "href": "/api/cluster/licensing/licenses/nfs"
        }
      }
    },
    {
      "name": "cifs",
      "scope": "node",
      "state": "compliant",
      "description": "CIFS License",
      "entitlement": {
        "action": "acquire_license",
        "risk": "medium"
      },
      "licenses": [
        {
          "owner": "testcluster-1",
          "serial_number": "1-81-0000000000000004149027492",
          "state": "compliant"
        }
      ],
      "_links": {
        "self": {
          "href": "/api/cluster/licensing/licenses/cifs"
        }
      }
    }
  ]
}

```

```

],
"num_records": 2,
"_links": {
"self": {
  "href": "/api/cluster/licensing/licenses/?fields=*&name=!base"
}
}
}

```

## Retrieving a collection of unlicensed packages

By default, unlicensed packages are filtered from the collection output. This example shows how to use a query to retrieve unlicensed packages.

```

# API
curl -X GET "https://<mgmt-
ip>/api/cluster/licensing/licenses?name=flexcache&state=unlicensed"

# Response
200 OK

# JSON Body
{
"records": [
  {
    "name": "flexcache",
    "_links": {
      "self": {
        "href": "/api/cluster/licensing/licenses/flexcache"
      }
    }
  }
],
"num_records": 1,
"_links": {
  "self": {
    "href":
"/api/cluster/licensing/licenses?name=flexcache&state=unlicensed"
  }
}
}

```

## Installing a NLF license

This example installs a single NLFv1 license. A NLFv2 license installs using the same procedure.



You must escape all the double quotes and backslash characters of the JSON license before it can be placed in the POST request.

```
# API
curl -X POST "https://<mgmt-ip>/api/cluster/licensing/licenses"

# JSON Body
{
  "keys" : [ "{\"statusResp\":{\"snStatus\": \"Active\", \"licenses\":
  {\"package\": \"FabricPool\", \"capacity\": \"1\", \"licenseProtocol\":
  \"FABRICPOOL-TB\", \"type\": \"capacity\", \"licenseScope\": \"cluster\"},
  \"message\": \"Success\", \"statusCode\": \"S007\", \"version\": \"1\",
  \"product\": \"fabricpool\", \"serialNumber\": \"4149027342\"},
  \"Signature\": \"SignatureABC\"}" ]
}

# Response
201 Created
```

### Installing a 28-character key

This example installs a single 28-character key formatted license.

```
# API
curl -X POST "https://<mgmt-ip>/api/cluster/licensing/licenses"

# JSON Body
{
  "keys" : [ "AAAAAAAAAAAAAAAAAAAAAAAAAAAA" ]
}

# Response
201 Created
```

### Installing multiple licenses with one API call

This example shows how multiple keys can be provided to install multiple features in a single API call.

```
# API
curl -X POST "https://<mgmt-ip>/api/cluster/licensing/licenses"

# JSON Body
{
  "keys" : [ "AAAAAAAAAAAAAAAAAAAAAAAAAAAA",
             "BBBBBBBBBBBBBBBBBBBBBBBBBBBB" ]
}

# Response
201 Created
```

### Promoting a capacity tier evaluation node

x-ntap-visibility: type: private This example promotes a capacity tier evaluation node to a capacity tier production node.



You must escape all double quotes and backslash characters in the JSON license before it is placed in the POST request.

```
# API
curl -X POST "https://<mgmt-ip>/api/cluster/licensing/licenses/?eval_node=nodeX"

# JSON Body
{
  "keys": [ "{\"statusResp\":{\"product\": \"FDvM300\", \"serialNumber\": \"230000002\", \"version\": \"1\", \"licenses\": {\"capacity\": \"100\", \"type\": \"capacity\", \"package\": \"ONTAP-SEL-M300_STD\"}, \"message\": \"Success\", \"snStatus\": \"Active\", \"statusCode\": \"S007\"}, \"Signature\": \"SignatureABC\"}" ]
}

# Response
201 Created
```

### Retrieving information for a specific license package

This example shows how to retrieve information about the specific feature package `fabricpool`.

```

# API
curl -X GET "https://<mgmt-ip>/api/cluster/licensing/licenses/fabricpool"

# Response
200 OK

# JSON Body
{
  "name": "fabricpool",
  "scope": "cluster",
  "state": "compliant",
  "description": "FabricPool License",
  "licenses": [
    {
      "owner": "testcluster-1",
      "serial_number": "123456789",
      "state": "compliant",
      "capacity": {
        "maximum_size": 109951162777600,
        "used_size": 0
      }
    }
  ],
  "_links": {
    "self": {
      "href": "/api/cluster/licensing/licenses/fabricpool/"
    }
  }
}

```

## Deleting a specific license

This example show how to delete a CIFS site license.

```

# API
curl -X DELETE "https://<mgmt-ip>/api/cluster/licensing/licenses/cifs?serial_number=1-80-000011"

# JSON Body
{}

# Response
200 OK

```

## Deleting with a query

The following example shows how to delete all NFS licenses specified with the '\*' query.

```
# API
curl -X DELETE "https://<mgmt-
ip>/api/cluster/licensing/licenses/nfs?serial_number=*"

# JSON Body
{}

# Response
200 OK
```

## Deleting all licenses installed with NLFv2

The following example shows how to delete all licenses installed by a NLFv2 formatted license.

```
# API
curl -X DELETE "https://<mgmt-
ip>/api/cluster/licensing/licenses?licenses.installed_license=Core*Bundle&
serial_number=4149026-97-8"

# JSON Body
{
  "num_records": 1,
  "_links": {
    "self": {
      "href":
"/api/cluster/licensing/licenses?licenses.installed_license=Core*Bundle&se
rial_number=4149026-97-8"
    }
  }
}

# Response
200 OK
```

## Retrieve license packages

GET /cluster/licensing/licenses

**Introduced In:** 9.6

Retrieves a collection of license packages.



By default, the GET method only returns licensed packages. You must provide the following query "state=unlicensed" to retrieve unlicensed packages. **Note:** Starting with ONTAP 9.11.1, the GET method no longer returns the Base license record.

## Related ONTAP commands

- `system license show-status`
- `system license show`

## Parameters

| Name                       | Type    | In    | Required | Description   |
|----------------------------|---------|-------|----------|---|
| state                      | string  | query | False    | Filter by state   |
| scope                      | string  | query | False    | Filter by scope   |
| name                       | string  | query | False    | Filter by name  |
| licenses.host_id           | string  | query | False    | Filter by licenses.host_id <ul style="list-style-type: none"><li>• Introduced in: 9.9</li></ul>           |
| licenses.expiry_time       | string  | query | False    | Filter by licenses.expiry_time  |
| licenses.evaluation        | boolean | query | False    | Filter by licenses.evaluation   |
| licenses.serial_number     | string  | query | False    | Filter by licenses.serial_number  |
| licenses.active            | boolean | query | False    | Filter by licenses.active   |
| licenses.start_time        | string  | query | False    | Filter by licenses.start_time   |
| licenses.installed_license | string  | query | False    | Filter by licenses.installed_license <ul style="list-style-type: none"><li>• Introduced in: 9.9</li></ul> |

| Name                           | Type          | In    | Required | Description   |
|--------------------------------|---------------|-------|----------|---|
| licenses.owner                 | string        | query | False    | Filter by licenses.owner  |
| licenses.shutdown_imminent     | boolean       | query | False    | Filter by licenses.shutdown_imminent<br><br>• Introduced in: 9.11 |
| licenses.compliance.state      | string        | query | False    | Filter by licenses.compliance.state                               |
| licenses.capacity.maximum_size | integer       | query | False    | Filter by licenses.capacity.maximum_size                          |
| licenses.capacity.used_size    | integer       | query | False    | Filter by licenses.capacity.used_size                             |
| entitlement.action             | string        | query | False    | Filter by entitlement.action<br><br>• Introduced in: 9.11         |
| entitlement.risk               | string        | query | False    | Filter by entitlement.risk<br><br>• Introduced in: 9.11           |
| description                    | string        | query | False    | Filter by description<br><br>• Introduced in: 9.11                |
| fields                         | array[string] | query | False    | Specify the fields to return.                                     |
| max_records                    | integer       | query | False    | Limit the number of records returned.                             |



| Name           | Type          | In    | Required | Description  |
|----------------|---------------|-------|----------|--|
| return_records | boolean       | query | False    | The default is true for GET calls. When set to false, only the number of records is returned.<br><br>• Default value: 1  |
| return_timeout | integer       | query | False    | 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.<br><br>• Default value: 1<br>• Max value: 120<br>• Min value: 0 |
| order_by       | array[string] | query | False    | Order results by specified fields and optional [asc  |

## Response

Status: 200, Ok

| Name        | Type                             | Description       |
|-------------|----------------------------------|-------------------|
| _links      | <a href="#">_links</a>           |                   |
| num_records | integer                          | Number of records |
| records     | array[ <a href="#">records</a> ] |                   |

## Example response

```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "description": "NFS License",
    "entitlement": {
      "action": "acquire_license",
      "risk": "high"
    },
    "keys": {
    },
    "licenses": {
      "capacity": {
        "maximum_size": 0,
        "used_size": 0
      },
      "compliance": {
        "state": "compliant"
      },
      "expiry_time": "2019-03-02T19:00:00Z",
      "host_id": "456-44-1234",
      "installed_license": "Core Bundle",
      "owner": "cluster1",
      "serial_number": "123456789",
      "start_time": "2019-02-02T19:00:00Z"
    },
    "name": "NFS",
    "scope": "not_available",
    "state": "compliant"
  }
}
```

## Error

Status: Default, Error

| Name  | Type  | Description |
|-------|-------|-------------|
| error | error |             |

### Example error

```
{
  "error": {
    "arguments": {
      "code": "string",
      "message": "string"
    },
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

## Definitions

## See Definitions

href

| Name | Type   | Description |
|------|--------|-------------|
| href | string |             |

\_links

| Name | Type                 | Description |
|------|----------------------|-------------|
| next | <a href="#">href</a> |             |
| self | <a href="#">href</a> |             |

\_links

| Name | Type                 | Description |
|------|----------------------|-------------|
| self | <a href="#">href</a> |             |

entitlement

| Name   | Type   | Description   |
|--------|--------|---|
| action | string | Entitlement action to be taken to mitigate the risk |
| risk   | string | Entitlement risk of the package                     |

capacity

| Name         | Type    | Description   |
|--------------|---------|---|
| maximum_size | integer | Licensed capacity size (in bytes) that can be used. |
| used_size    | integer | Capacity that is currently used (in bytes).         |

compliance

| Name  | Type   | Description                      |
|-------|--------|----------------------------------|
| state | string | Compliance state of the license. |

licenses

| Name              | Type                       | Description  |
|-------------------|----------------------------|--|
| active            | boolean                    | A flag indicating whether the license is currently being enforced.   |
| capacity          | <a href="#">capacity</a>   |  |
| compliance        | <a href="#">compliance</a> |  |
| evaluation        | boolean                    | A flag indicating whether the license is in evaluation mode.   |
| expiry_time       | string                     | Date and time when the license expires.  |
| host_id           | string                     | A string that associates the license with a node or cluster.   |
| installed_license | string                     | Name of license that enabled the feature.  |
| owner             | string                     | Cluster, node or license manager that owns the license.  |
| serial_number     | string                     | Serial number of the license.  |
| shutdown_imminent | boolean                    | <p>A flag indicating whether the Cloud ONTAP system is going to shutdown as the Cloud platform license has already expired.</p> <ul style="list-style-type: none"> <li>• readOnly: 1</li> <li>• Introduced in: 9.11</li> </ul> |
| start_time        | string                     | Date and time when the license starts.   |

#### records

| Name                   | Type                              | Description                        |
|------------------------|-----------------------------------|------------------------------------|
| <a href="#">_links</a> | <a href="#">_links</a>            |                                    |
| description            | string                            | License description                |
| entitlement            | <a href="#">entitlement</a>       |                                    |
| keys                   | array[string]                     |                                    |
| licenses               | array[ <a href="#">licenses</a> ] | Installed licenses of the package. |

| Name  | Type   | Description   |
|-------|--------|---|
| name  | string | Name of the license.                                      |
| scope | string | Scope of the license.                                     |
| state | string | Summary state of package based on all installed licenses. |

error\_arguments

| Name    | Type   | Description      |
|---------|--------|------------------|
| code    | string | Argument code    |
| message | string | Message argument |

error

| Name      | Type                                     | Description                                 |
|-----------|--|---|
| arguments | array[ <a href="#">error_arguments</a> ] | Message arguments                           |
| code      | string                                   | Error code                                  |
| message   | string                                   | Error message                               |
| target    | string                                   | The target parameter that caused the error. |

## Install one or more feature licenses

POST `/cluster/licensing/licenses`

**Introduced In:** 9.6

Installs one or more feature licenses.

### Required properties

- `keys` - Array containing a list of NLF or 28-character license keys.

### Related ONTAP commands

- `system license add`

## Parameters

| Name           | Type    | In    | Required | Description   |
|----------------|---------|-------|----------|---|
| return_records | boolean | query | False    | The default is false. If set to true, the records are returned.<br><br>• Default value: |

## Request Body

| Name        | Type                              | Description   |
|-------------|-----------------------------------|---|
| _links      | <a href="#">_links</a>            |   |
| description | string                            | License description                                       |
| entitlement | <a href="#">entitlement</a>       |   |
| keys        | array[string]                     |   |
| licenses    | array[ <a href="#">licenses</a> ] | Installed licenses of the package.                        |
| name        | string                            | Name of the license.                                      |
| scope       | string                            | Scope of the license.                                     |
| state       | string                            | Summary state of package based on all installed licenses. |

## Example request

```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "description": "NFS License",
  "entitlement": {
    "action": "acquire_license",
    "risk": "high"
  },
  "keys": {
  },
  "licenses": {
    "capacity": {
      "maximum_size": 0,
      "used_size": 0
    },
    "compliance": {
      "state": "compliant"
    },
    "expiry_time": "2019-03-02T19:00:00Z",
    "host_id": "456-44-1234",
    "installed_license": "Core Bundle",
    "owner": "cluster1",
    "serial_number": "123456789",
    "start_time": "2019-02-02T19:00:00Z"
  },
  "name": "NFS",
  "scope": "not_available",
  "state": "compliant"
}
```

## Response

Status: 201, Created

| Name   | Type                   | Description |
|--------|------------------------|-------------|
| _links | <a href="#">_links</a> |             |



| Name        | Type                             | Description       |
|-------------|----------------------------------|-------------------|
| num_records | integer                          | Number of records |
| records     | array[ <a href="#">records</a> ] |                   |

## Example response

```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "description": "NFS License",
    "entitlement": {
      "action": "acquire_license",
      "risk": "high"
    },
    "keys": {
    },
    "licenses": {
      "capacity": {
        "maximum_size": 0,
        "used_size": 0
      },
      "compliance": {
        "state": "compliant"
      },
      "expiry_time": "2019-03-02T19:00:00Z",
      "host_id": "456-44-1234",
      "installed_license": "Core Bundle",
      "owner": "cluster1",
      "serial_number": "123456789",
      "start_time": "2019-02-02T19:00:00Z"
    },
    "name": "NFS",
    "scope": "not_available",
    "state": "compliant"
  }
}
```

## Headers

| Name     | Description                               | Type   |
|----------|---|--------|
| Location | Useful for tracking the resource location | string |

## Error

Status: Default

### ONTAP Error Response Codes

| Error Code | Description   |
|------------|---|
| 1115117    | Generic licensing error   |
| 1115122    | No cluster serial number found  |
| 1115124    | No node serial number found   |
| 1115130    | No license code was provided  |
| 1115131    | Installation of the license failed  |
| 1115132    | License already exists on system  |
| 1115134    | Serial number does not belong to node   |
| 1115141    | License data is invalid   |
| 1115142    | License signature is invalid  |
| 1115143    | Internal error applying the requested license   |
| 1115152    | License does not apply to the platform  |
| 1115154    | Unable to retrieve cluster ID   |
| 1115155    | Invalid cluster ID found  |
| 1115159    | License is not in an acceptable format  |
| 1115160    | License has already expired   |
| 1115164    | Minimum ONTAP version requirements not met  |
| 1115165    | Minimum ONTAP version requirements are not met for license type enabled                 |
| 1115166    | Minimum ONTAP version requirements are not met for license protocol SEC-COMP-BNDL-ENBLD |
| 1115179    | FlexCache is not supported on this system   |
| 1115180    | FlexCache is not supported on cloud systems   |
| 1115407    | Capacity pool licenses cannot be installed directly                                     |
| 1115427    | License is incompatible with capacity pools licensing mode                              |

| Error Code | Description   |
|------------|---|
| 1115562    | One or more errors occurred when installing a NLFv2 license                     |
| 1115563    | Package details and serial number of license contained within the NLFv2 failure |
| 1115564    | Package cannot be deleted individually as it is part of a bundle                |
| 1115565    | NLFv2 install failed as the license serial number is already in use             |
| 1115616    | Package details and serial number of license included in the install conflict   |
| 1115617    | NLFv2 license install failed with summary of conflicting licenses               |
| 1115618    | NLFv2 license install failed as a license with newer timestamp already exists   |
| 5375355    | The cluster has more nodes than are supported by All SAN Array.                 |
| 5375366    | The cluster has one or more nodes that do not support All SAN Array.            |
| 66846818   | Failed to interpret FlexCache license information                               |
| 66846821   | FlexCache is not supported on cloud systems                                     |
| 66846822   | Invalid FlexCache capacity information provided                                 |
| 655294464  | Failed to extract license contents  |
| 655294465  | License key is invalid  |
| 655294466  | Serial number is invalid  |
| 655294467  | Version number is invalid   |
| 655294468  | Expired license   |
| 655294469  | License does not apply to the platform  |
| 655294470  | License does not apply to the product   |

| Name   | Type                           | Description |
|--------|--------------------------------|-------------|
| errors | array[ <a href="#">error</a> ] |             |

### Example error

```
{
  "errors": {
    "arguments": {
      "code": "string",
      "message": "string"
    },
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

### Definitions

## See Definitions

href

| Name | Type   | Description |
|------|--------|-------------|
| href | string |             |

\_links

| Name | Type                 | Description |
|------|----------------------|-------------|
| self | <a href="#">href</a> |             |

entitlement

| Name   | Type   | Description   |
|--------|--------|---|
| action | string | Entitlement action to be taken to mitigate the risk |
| risk   | string | Entitlement risk of the package                     |

capacity

| Name         | Type    | Description   |
|--------------|---------|---|
| maximum_size | integer | Licensed capacity size (in bytes) that can be used. |
| used_size    | integer | Capacity that is currently used (in bytes).         |

compliance

| Name  | Type   | Description                      |
|-------|--------|----------------------------------|
| state | string | Compliance state of the license. |

licenses

| Name       | Type                       | Description  |
|------------|----------------------------|--|
| active     | boolean                    | A flag indicating whether the license is currently being enforced. |
| capacity   | <a href="#">capacity</a>   |  |
| compliance | <a href="#">compliance</a> |  |

| Name              | Type    | Description  |
|-------------------|---------|--|
| evaluation        | boolean | A flag indicating whether the license is in evaluation mode.   |
| expiry_time       | string  | Date and time when the license expires.  |
| host_id           | string  | A string that associates the license with a node or cluster.   |
| installed_license | string  | Name of license that enabled the feature.  |
| owner             | string  | Cluster, node or license manager that owns the license.  |
| serial_number     | string  | Serial number of the license.  |
| shutdown_imminent | boolean | <p>A flag indicating whether the Cloud ONTAP system is going to shutdown as the Cloud platform license has already expired.</p> <ul style="list-style-type: none"> <li>• readOnly: 1</li> <li>• Introduced in: 9.11</li> </ul> |
| start_time        | string  | Date and time when the license starts.   |

#### license\_package

| Name        | Type                              | Description                        |
|-------------|-----------------------------------|------------------------------------|
| _links      | <a href="#">_links</a>            |                                    |
| description | string                            | License description                |
| entitlement | <a href="#">entitlement</a>       |                                    |
| keys        | array[string]                     |                                    |
| licenses    | array[ <a href="#">licenses</a> ] | Installed licenses of the package. |
| name        | string                            | Name of the license.               |
| scope       | string                            | Scope of the license.              |

| Name  | Type   | Description   |
|-------|--------|---|
| state | string | Summary state of package based on all installed licenses. |

[\\_links](#)

| Name | Type                 | Description |
|------|----------------------|-------------|
| next | <a href="#">href</a> |             |
| self | <a href="#">href</a> |             |

records

| Name                   | Type                              | Description   |
|------------------------|-----------------------------------|---|
| <a href="#">_links</a> | <a href="#">_links</a>            |   |
| description            | string                            | License description                                       |
| entitlement            | <a href="#">entitlement</a>       |   |
| keys                   | array[string]                     |   |
| licenses               | array[ <a href="#">licenses</a> ] | Installed licenses of the package.                        |
| name                   | string                            | Name of the license.                                      |
| scope                  | string                            | Scope of the license.                                     |
| state                  | string                            | Summary state of package based on all installed licenses. |

error\_arguments

| Name    | Type   | Description      |
|---------|--------|------------------|
| code    | string | Argument code    |
| message | string | Message argument |

error

| Name      | Type                                     | Description       |
|-----------|--|-------------------|
| arguments | array[ <a href="#">error_arguments</a> ] | Message arguments |
| code      | string                                   | Error code        |



| Name    | Type   | Description                                 |
|---------|--------|---|
| message | string | Error message                               |
| target  | string | The target parameter that caused the error. |

## Delete a license

DELETE /cluster/licensing/licenses/{name}

**Introduced In:** 9.6

Deletes a license.

### Related ONTAP commands

- `system license delete`

### Parameters

| Name          | Type   | In    | Required | Description                             |
|---------------|--------|-------|----------|---|
| name          | string | path  | True     | Name of the license package to delete.  |
| serial_number | string | query | True     | Serial number of the license to delete. |

### Response

Status: 200, Ok

### Error

Status: Default

#### ONTAP Error Response Codes

| Error Code | Description   |
|------------|---|
| 525028     | Error during volume limit check, cannot remove license      |
| 525029     | Current volume use will exceed limits if license is removed |

| Error Code | Description   |
|------------|---|
| 1115137    | Cluster license requires a base license to be installed               |
| 1115144    | Cloud licenses cannot be deleted                                      |
| 1115178    | A tier license that is still in use cannot be deleted                 |
| 1115213    | License is still in use and cannot be removed                         |
| 1115406    | Capacity pool licenses cannot be deleted                              |
| 1115564    | Package is part of a NLFv2 license and cannot be removed individually |
| 66846823   | A FlexCache license that is still in use cannot be deleted            |

| Name  | Type                  | Description |
|-------|-----------------------|-------------|
| error | <a href="#">error</a> |             |

### Example error

```
{
  "error": {
    "arguments": {
      "code": "string",
      "message": "string"
    },
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

## Definitions

## See Definitions

### error\_arguments

| Name    | Type   | Description      |
|---------|--------|------------------|
| code    | string | Argument code    |
| message | string | Message argument |

### error

| Name      | Type                                     | Description                                 |
|-----------|--|---|
| arguments | array[ <a href="#">error_arguments</a> ] | Message arguments                           |
| code      | string                                   | Error code                                  |
| message   | string                                   | Error message                               |
| target    | string                                   | The target parameter that caused the error. |

## Retrieve a license package

GET /cluster/licensing/licenses/{name}

Introduced In: 9.6

Retrieves a specific license package.



By default, the GET method only returns licensed packages. You must provide the following query "state=unlicensed" to retrieve unlicensed packages.

## Related ONTAP commands

- `system license show`
- `system license show-status`

## Parameters

| Name | Type   | In   | Required | Description                  |
|------|--------|------|----------|------------------------------|
| name | string | path | True     | Name of the license package. |

| Name                       | Type    | In    | Required | Description  |
|----------------------------|---------|-------|----------|--|
| state                      | string  | query | False    | Filter by state  |
| scope                      | string  | query | False    | Filter by scope  |
| licenses.host_id           | string  | query | False    | Filter by licenses.host_id <ul style="list-style-type: none"> <li>• Introduced in: 9.9</li> </ul>            |
| licenses.expiry_time       | string  | query | False    | Filter by licenses.expiry_time   |
| licenses.evaluation        | boolean | query | False    | Filter by licenses.evaluation  |
| licenses.serial_number     | string  | query | False    | Filter by licenses.serial_number   |
| licenses.active            | boolean | query | False    | Filter by licenses.active  |
| licenses.start_time        | string  | query | False    | Filter by licenses.start_time  |
| licenses.installed_license | string  | query | False    | Filter by licenses.installed_license <ul style="list-style-type: none"> <li>• Introduced in: 9.9</li> </ul>  |
| licenses.owner             | string  | query | False    | Filter by licenses.owner   |
| licenses.shutdown_imminent | boolean | query | False    | Filter by licenses.shutdown_imminent <ul style="list-style-type: none"> <li>• Introduced in: 9.11</li> </ul> |
| licenses.compliance.state  | string  | query | False    | Filter by licenses.compliance.state  |

| Name                           | Type          | In    | Required | Description  |
|--------------------------------|---------------|-------|----------|--|
| licenses.capacity.maximum_size | integer       | query | False    | Filter by licenses.capacity.maximum_size   |
| licenses.capacity.used_size    | integer       | query | False    | Filter by licenses.capacity.used_size  |
| entitlement.action             | string        | query | False    | Filter by entitlement.action <ul style="list-style-type: none"> <li>Introduced in: 9.11</li> </ul> |
| entitlement.risk               | string        | query | False    | Filter by entitlement.risk <ul style="list-style-type: none"> <li>Introduced in: 9.11</li> </ul>   |
| description                    | string        | query | False    | Filter by description <ul style="list-style-type: none"> <li>Introduced in: 9.11</li> </ul>        |
| fields                         | array[string] | query | False    | Specify the fields to return.  |

## Response

Status: 200, Ok

| Name        | Type                              | Description                        |
|-------------|-----------------------------------|------------------------------------|
| _links      | <a href="#">_links</a>            |                                    |
| description | string                            | License description                |
| entitlement | <a href="#">entitlement</a>       |                                    |
| keys        | array[string]                     |                                    |
| licenses    | array[ <a href="#">licenses</a> ] | Installed licenses of the package. |
| name        | string                            | Name of the license.               |
| scope       | string                            | Scope of the license.              |

| Name  | Type   | Description   |
|-------|--------|---|
| state | string | Summary state of package based on all installed licenses. |

### Example response

```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "description": "NFS License",
  "entitlement": {
    "action": "acquire_license",
    "risk": "high"
  },
  "keys": {
  },
  "licenses": {
    "capacity": {
      "maximum_size": 0,
      "used_size": 0
    },
    "compliance": {
      "state": "compliant"
    },
    "expiry_time": "2019-03-02T19:00:00Z",
    "host_id": "456-44-1234",
    "installed_license": "Core Bundle",
    "owner": "cluster1",
    "serial_number": "123456789",
    "start_time": "2019-02-02T19:00:00Z"
  },
  "name": "NFS",
  "scope": "not_available",
  "state": "compliant"
}
```

### Error

Status: Default, Error

| Name  | Type  | Description |
|-------|-------|-------------|
| error | error |             |

### Example error

```
{
  "error": {
    "arguments": {
      "code": "string",
      "message": "string"
    },
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

### Definitions

## See Definitions

href

| Name | Type   | Description |
|------|--------|-------------|
| href | string |             |

\_links

| Name | Type                 | Description |
|------|----------------------|-------------|
| self | <a href="#">href</a> |             |

entitlement

| Name   | Type   | Description   |
|--------|--------|---|
| action | string | Entitlement action to be taken to mitigate the risk |
| risk   | string | Entitlement risk of the package                     |

capacity

| Name         | Type    | Description   |
|--------------|---------|---|
| maximum_size | integer | Licensed capacity size (in bytes) that can be used. |
| used_size    | integer | Capacity that is currently used (in bytes).         |

compliance

| Name  | Type   | Description                      |
|-------|--------|----------------------------------|
| state | string | Compliance state of the license. |

licenses

| Name       | Type                       | Description  |
|------------|----------------------------|--|
| active     | boolean                    | A flag indicating whether the license is currently being enforced. |
| capacity   | <a href="#">capacity</a>   |  |
| compliance | <a href="#">compliance</a> |  |



| Name              | Type    | Description  |
|-------------------|---------|--|
| evaluation        | boolean | A flag indicating whether the license is in evaluation mode.   |
| expiry_time       | string  | Date and time when the license expires.  |
| host_id           | string  | A string that associates the license with a node or cluster.   |
| installed_license | string  | Name of license that enabled the feature.  |
| owner             | string  | Cluster, node or license manager that owns the license.  |
| serial_number     | string  | Serial number of the license.  |
| shutdown_imminent | boolean | <p>A flag indicating whether the Cloud ONTAP system is going to shutdown as the Cloud platform license has already expired.</p> <ul style="list-style-type: none"> <li>• readOnly: 1</li> <li>• Introduced in: 9.11</li> </ul> |
| start_time        | string  | Date and time when the license starts.   |

#### error\_arguments

| Name    | Type   | Description      |
|---------|--------|------------------|
| code    | string | Argument code    |
| message | string | Message argument |

#### error

| Name      | Type                                     | Description       |
|-----------|--|-------------------|
| arguments | array[ <a href="#">error_arguments</a> ] | Message arguments |
| code      | string                                   | Error code        |
| message   | string                                   | Error message     |

| Name   | Type   | Description                                 |
|--------|--------|---|
| target | string | The target parameter that caused the error. |

## Copyright information

Copyright © 2024 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. Government has a non-exclusive, non-transferrable, nonsublicensable, worldwide, limited irrevocable license to use the Data only in connection with and in support of the U.S. Government contract under which the Data was delivered. Except as provided herein, the Data may not be used, disclosed, reproduced, modified, performed, or displayed without the prior written approval of NetApp, Inc. United States Government license rights for the Department of Defense are limited to those rights identified in DFARS clause 252.227-7015(b) (FEB 2014).

## Trademark information

NETAPP, the NETAPP logo, and the marks listed at <http://www.netapp.com/TM> are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.