Common workflows for storage management

Active IQ Unified Manager

NetApp
May 24, 2022

Table of Contents

Common API workflows for storage management

The common workflows provide client application developers with examples of how Active IQ Unified Manager APIs can be called by a client application to execute common storage management functions. This section contains some of these sample workflows.

The workflows describe some of the commonly used storage management use cases along with sample codes for you to use. Each of the tasks is described using a workflow process consisting of one or more API calls.

Understanding the API calls used in the workflows

You can view the online documentation page from your Unified Manager instance that includes the details of every REST API call. This document does not repeat the details of the online documentation. Each API call used in the workflow samples in this document includes only the information you need to locate the call on the documentation page. After locating a specific API call, you can review the complete details of the call, including the input parameters, output formats, HTTP status codes, and request processing type.

The following information is included for each API call within a workflow to help locate the call on the documentation page:

- Category: The API calls are organized on the documentation page into functionally related areas or categories. To locate a specific API call, scroll to the bottom of the page and click the applicable API category.
- HTTP verb (call): The HTTP verb identifies the action performed on a resource. Each API call is executed through a single HTTP verb.
- Path: The path determines the specific resource which the action applies to as part of performing a call. The path string is appended to the core URL to form the complete URL identifying the resource.

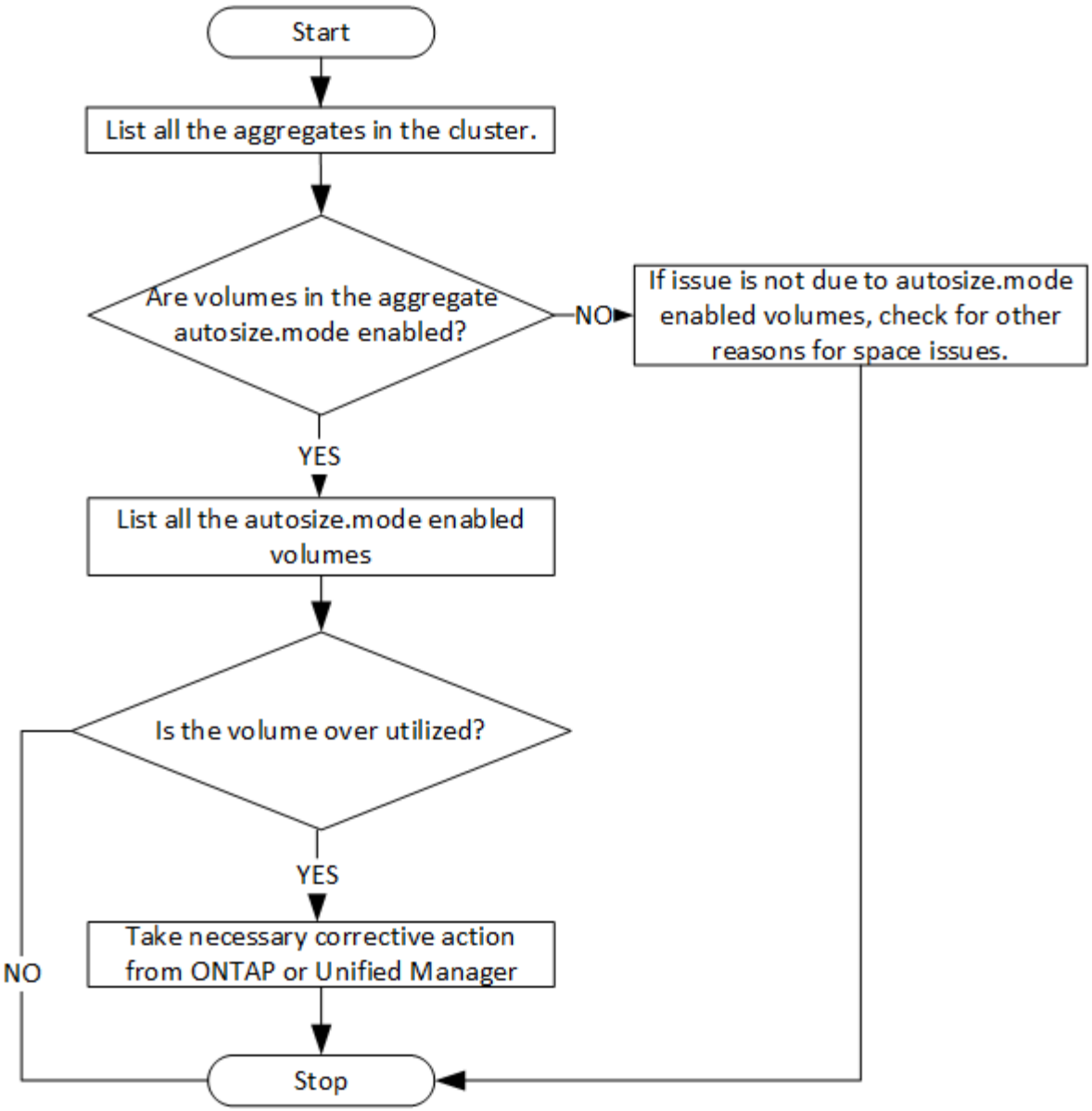
Determining space issues in aggregates by using APIs

You can use the data center APIs in Active IQ Unified Manager to monitor the availability and utilization of space in your volumes. You can determine space issues in your volume and identify storage resources that are overutilized or underutilized.

The data center APIs for aggregates retrieve the relevant information about available and used space, and space saving efficiency settings. You can also filter the retrieved information based on specified attributes.

One method to determine any lack of space in your aggregates is to verify whether there are volumes in your environment with autosize-mode enabled. You should then identify which volumes are being over-utilized and perform any corrective actions.

The following flowchart illustrates the process of retrieving information about volumes with autosize-mode enabled:



This flow assumes that the clusters have already been created in ONTAP and added to Unified Manager.

1. Obtain the cluster key, unless you know the value:

Category	HTTP verb	Path
datacenter	GET	`/datacenter/cluster/clusters`

2. Using the cluster key as the filter parameter, query the aggregates on that cluster.

Category	HTTP verb	Path
datacenter	GET	`/datacenter/storage/aggregates`

3. From the response, analyze the space usage of the aggregates and determine which aggregates have space issues. For each aggregate with space issue, obtain the aggregate key from the same JSON output.
4. Using each aggregate key, filter all the volumes that have the value for the autosize.mode parameter as `grow`.

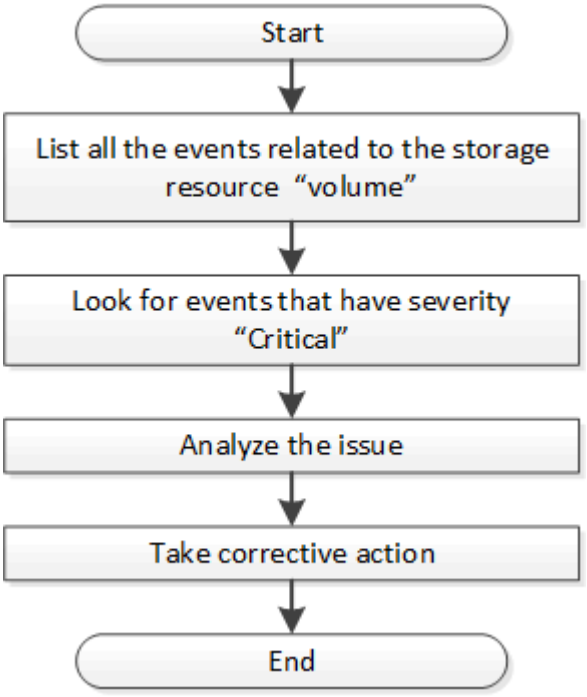
Category	HTTP verb	Path
datacenter	GET	`/datacenter/storage/volumes`

5. Analyze which volumes are being over-utilized.
6. Perform any necessary corrective action, such as moving the volume across aggregates, to address the space issues in your volume. You can perform these actions from ONTAP or Unified Manager web UI.

Determining issues in storage objects using event APIs

When a storage object in your data center crosses a threshold, you get a notification about that event. Using this notification, you can analyze the issue and take corrective action by using the `events` APIs.

This workflow takes the example of a volume as the resource object. You can use the `events` APIs to retrieve the list of events related to a volume, analyze the critical issues for that volume, and then take corrective actions to rectify the issue.



Follow these steps to determine the issues in your volume before taking remedial steps.

Steps

1. Analyze the critical Active IQ Unified Manager events notifications for the volumes in your data center.
2. Query all the events for the volumes by using the following parameters in the /management-server/events

API: `"resource_type": "volume" "severity": "critical"`

Category	HTTP verb	Path
management-server	GET	/management-server/events

3. View the output and analyze the issues in the specific volumes.
4. Perform the necessary actions by using the Unified Manager REST APIs or web UI to resolve the issues.

Troubleshooting ONTAP volumes by using gateway APIs

The gateway APIs act as a gateway to invoke ONTAP APIs to query information about your ONTAP storage objects and take remedial measures to address the reported issues.

This workflow takes up a sample use case in which an event is raised when an ONTAP volume almost reaches its capacity. The workflow also demonstrates how to address this issue by invoking a combination of Active IQ Unified Manager and ONTAP REST APIs.

Before running the workflow steps, ensure the following:

- You are aware of the gateway APIs and how they are used. For information, see the “Gateway APIs” section.

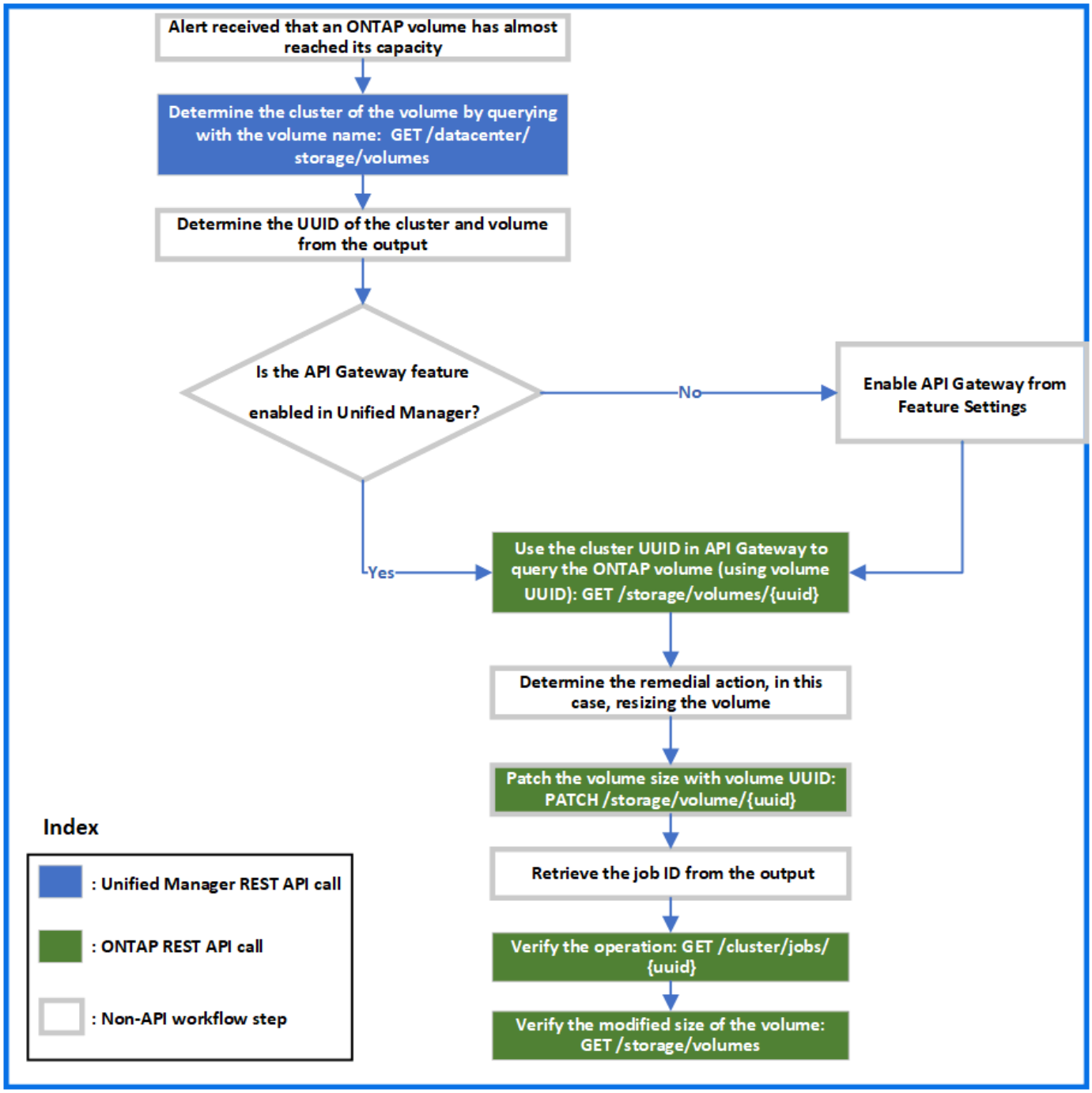
  Accessing ONTAP APIs through proxy access

- You are aware of the usage of ONTAP REST APIs. For information about using ONTAP REST APIs, see the *ONTAP REST API Developers guide*.

  ONTAP REST API Developers guide

- You are an Application Administrator.
- The cluster on which you want to run the REST API operations is supported by ONTAP 9.5 or later, and the cluster is added to Unified Manager over HTTPS.

The following diagram illustrates each step in the workflow for troubleshooting the issue of ONTAP volume capacity use.



The workflow covers the invocation points of both the Unified Manager and ONTAP REST APIs.

1. Note the volume name from the event notifying the volume capacity utilization.
2. By using the volume name as the value in the name parameter, query the volume by running the following Unified Manager API.

Category	HTTP verb	Path
datacenter	GET	`/datacenter/storage/volumes`

3. Retrieve the cluster UUID and volume UUID from the output.

4. On the Unified Manager web UI, navigate to **General** > **Feature Settings** > **API Gateway** to verify whether the API Gateway feature is enabled. Unless it is enabled, the APIs under the gateway category are not available for you to invoke. Enable the feature if it is disabled.
5. Use the cluster UUID to run the ONTAP API `/storage/volumes/{uuid}` through API gateway. The query returns the volume details when the volume UUID is passed as the API parameter.

   For running the ONTAP APIs through API gateway, the Unified Manager credentials are passed internally for authentication, and you do not need to run an additional authentication step for individual cluster access.

Category	HTTP verb	Path
Unified Manager: gateway ONTAP: storage	GET	Gateway API: `/gateways/{uuid}/{path}` ONTAP API: `/storage/volumes/{uuid}`

   In /gateways/{uuid}/{path}, the value for {uuid} must be replaced with the cluster UUID on which the REST operation is to be performed. {path} must be replaced by the ONTAP REST URL /storage/volumes/{uuid}.

   The appended URL is: `/gateways/{cluster_uuid}/storage/volumes/{volume_uuid}`

   On running the GET operation, the generated URL is: `GEThttps://<hostname\>/api/gateways/<cluster_UUID\>/storage/volumes/{volume_uuid}`

   Sample cURL command

```
curl -X GET "https://<hostname>/api/gateways/1cd8a442-86d1-11e0-ae1c-
9876567890123/storage/volumes/028baa66-41bd-11e9-81d5-00a0986138f7"
-H "accept: application/hal+json" -H "Authorization: Basic
<Base64EncodedCredentials>"
```

6. From the output, determine the size, usage, and remedial measure to be taken. In this workflow, the remedial measure taken is resizing the volume.
7. Use the cluster UUID and run the following ONTAP API through the API gateway to resize the volume. For information about the input parameters for the gateway and ONTAP APIs, see step 5.

Category	HTTP verb	Path
Unified Manager: gateway ONTAP: storage	PATCH	Gateway API: `/gateways/{uuid}/{path}` ONTAP API: `/storage/volumes/{uuid}`

Along with the cluster UUID and volume UUID, you must enter a value for the size parameter for resizing the volume. Ensure to enter the value *in bytes*. For example, if you want to increase the size of a volume from 100 GB to 120 GB, enter the value for parameter size at the end of the query: `-d {\"size\": 128849018880}"`

Sample cURL command

```
curl -X PATCH "https://<hostname>/api/gateways/1cd8a442-86d1-11e0-ae1c-
9876567890123/storage/volumes/028baa66-41bd-11e9-81d5-00a0986138f7" -H
    "accept: application/hal+json" -H "Authorization: Basic
<Base64EncodedCredentials>" -d
    {\"size\": 128849018880}"
```

The JSON output returns a Job UUID.

8. Verify whether the job ran successfully by using the Job UUID. Use the cluster UUID and Job UUID to run the following ONTAP API through the API gateway. For information about the input parameters for the gateway and ONTAP APIs, see step 5.

Category	HTTP verb	Path

Unified Manager: gateway
 	
ONTAP: cluster
	GET	Gateway API:
		`/gateways/{uuid}/{path}`
		
		ONTAP API:
		`/cluster/jobs/{uuid}`

The HTTP codes returned are the same as the ONTAP REST API HTTP status codes.

9. Run the following ONTAP API to query the details of the resized volume. For information about the input parameters for the gateway and ONTAP APIs, see step 5.

Category	HTTP verb	Path

Unified Manager: gateway
 	
ONTAP: storage
	GET	Gateway API:
		`/gateways/{uuid}/{path}`
		
		ONTAP API:
		`/storage/volumes/{uuid}`

The output displays an increased volume size of 120 GB.

API workflows for workload management

Using Active IQ Unified Manager, you can provision and modify storage workloads (LUNs, NFS file shares, and CIFS shares). Provisioning consists of multiple steps, from the creation of the Storage Virtual Machine (SVM) to applying Performance Service Level

and Storage Efficiency Policies on the storage workloads. Modifying workloads consist of the steps for modifying specific parameters and enabling additional features on them.

The following workflows are described:

- Workflow for provisioning Storage Virtual Machines (SVMs) on Unified Manager.

  this workflow is required to be performed before provisioning LUNs or file shares on Unified Manager.

- Provisioning file shares.
- Provisioning LUNs.
- Modifying LUNs and file shares (by using the example for updating the Performance Service Level parameter for the storage workloads).
- Modifying an NFS file share to support CIFS protocol
- Modifying workloads to upgrade QoS to AQoS

For each provisioning workflow (LUN and file shares), ensure you must have completed the workflow for verifying the SVMs on the clusters.

You must also read the recommendations and limitations before using each API in the workflows. The relevant details of the APIs are available in their individual sections listed in the related concepts and references.

Verifying SVMs on clusters by using APIs

Before provisioning file shares or LUNs, you must verify whether the clusters have Storage Virtual Machines (SVMs) created on them.

The workflow assumes that ONTAP clusters to have been added to Unified Manager, and the cluster key has been obtained. Clusters should have the required licenses for provisioning LUNs and file shares on them.

1. Verify whether the cluster has an SVM created.

Category	HTTP verb	Path
datacenter	GET	`/datacenter/svm/svms` `/datacenter/svm/svms/{key}`

Sample cURL

```
curl -X GET "https://<hostname>/api/datacenter/svm/svms" -H "accept:
application/json" -H "Authorization: Basic <Base64EncodedCredentials>"
```

2. If the SVM key is not returned, then create the SVM. For creating the SVMs, you require the cluster key on which you provision the SVM. You also need to specify the SVM name. Follow these steps.

Category	HTTP verb	Path
datacenter	GET	`/datacenter/cluster/clusters` `/datacenter/cluster/clusters/{key}`

Get the cluster key.

Sample cURL

```
curl -X GET "https://<hostname>/api/datacenter/cluster/clusters" -H
"accept: application/json" -H "Authorization: Basic
<Base64EncodedCredentials>"
```

3. From the output, get the cluster key, and then use it as an input for creating the SVM.

> While creating the SVM, ensure that it supports all the protocols required for provisioning LUNs and file shares on them, for example, CIFS, NFS, FCP, and iSCSI. The provisioning workflows might fail if the SVM does not support the required services. It is recommended that the services for the respective types of workloads are also enabled on the SVM.

Category	HTTP verb	Path
datacenter	POST	`/datacenter/svm/svms`

Sample cURL

Enter the SVM object details as input parameters.

```
curl -X POST "https://<hostname>/api/datacenter/svm/svms" -H "accept:
application/json" -H "Content-Type: application/json" -H "Authorization:
Basic <Base64EncodedCredentials>" "{ \"aggregates\": [ { \"_links\": {},
\"key\": \"1cd8a442-86d1,type=objecttype,uuid=1cd8a442-86d1-11e0-ae1c-
9876567890123\",
\"name\": \"cluster2\", \"uuid\": \"02c9e252-41be-11e9-81d5-
00a0986138f7\" } ],
\"cifs\": { \"ad_domain\": { \"fqdn\": \"string\", \"password\":
\"string\",
\"user\": \"string\" }, \"enabled\": true, \"name\": \"CIFS1\" },
\"cluster\": { \"key\": \"1cd8a442-86d1-11e0-ae1c-
123478563412,type=object type,uuid=1cd8a442-86d1-11e0-ae1c-
9876567890123\" },
\"dns\": { \"domains\": [ \"example.com\", \"example2.example3.com\" ],
\"servers\": [ \"10.224.65.20\", \"2001:db08:a0b:12f0::1\" ] },
\"fcp\": { \"enabled\": true }, \"ip_interface\": [ { \"enabled\": true,
\"ip\": { \"address\": \"10.10.10.7\", \"netmask\": \"24\" },
\"location\": { \"home_node\": { \"name\": \"node1\" } }, \"name\":
\"dataLif1\" } ], \"ipspace\": { \"name\": \"exchange\" },
\"iscsi\": { \"enabled\": true }, \"language\": \"c.utf_8\",
\"ldap\": { \"ad_domain\": \"string\", \"base_dn\": \"string\",
\"bind_dn\": \"string\", \"enabled\": true, \"servers\": [ \"string\" ]
},
\"name\": \"svm1\", \"nfs\": { \"enabled\": true },
\"nis\": { \"domain\": \"string\", \"enabled\": true,
\"servers\": [ \"string\" ] }, \"nvme\": { \"enabled\": true },
\"routes\": [ { \"destination\": { \"address\": \"10.10.10.7\",
\"netmask\": \"24\" }, \"gateway\": \"string\" } ],
\"snapshot_policy\": { \"name\": \"default\" },
\"state\": \"running\", \"subtype\": \"default\"}"
```

The JSON output displays a Job object key that you can use to verify the SVM that you created.

4. Verify the SVM creation by using the job object key for query. If the SVM is created successfully, the SVM key is returned in the response.

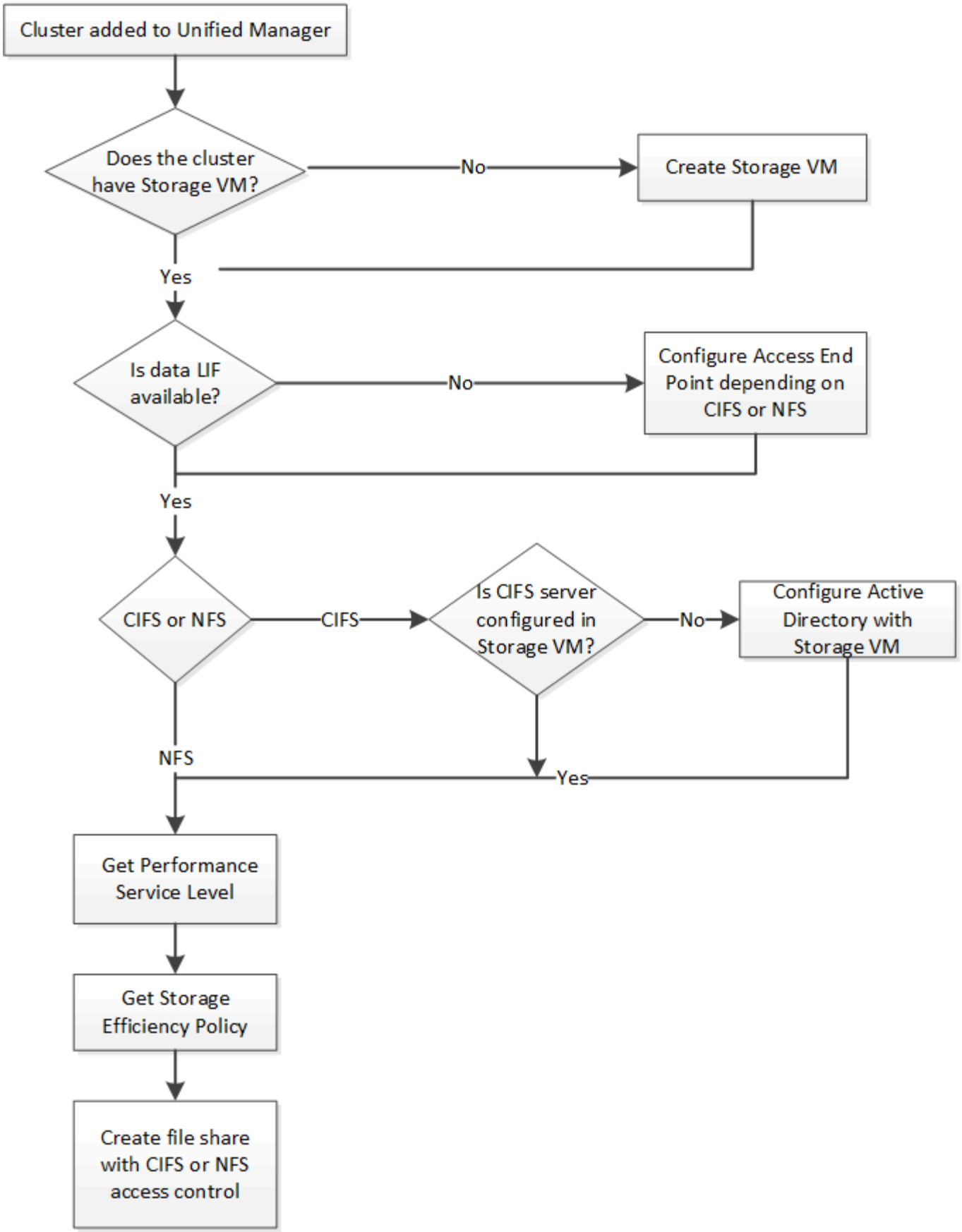
Category	HTTP verb	Path
management-server	GET	`/management-server/jobs/{key}`

Provisioning CIFS and NFS file shares by using APIs

You can provision CIFS shares and NFS file shares on your Storage Virtual Machines (SVMs) by using the provisioning APIs provided as a part of Active IQ Unified Manager.

This provisioning workflow details the steps for retrieving the keys of the SVMs, Performance Service Levels, and Storage Efficiency Policies before creating the file shares.

The following diagram illustrates each step in a file share provisioning workflow. It includes provisioning both CIFS shares and NFS file shares.


Cluster added to Unified Manager
Does the cluster have Storage VM?
No
Create Storage VM
Yes
Is data LIF available?
No
Configure Access End Point depending on CIFS or NFS
Yes
CIFS or NFS
CIFS
Is CIFS server configured in Storage VM?
No
Configure Active Directory with Storage VM
NFS
Yes
Get Performance Service Level
Get Storage Efficiency Policy
Create file share with CIFS or NFS access control

Ensure the following:

- ONTAP clusters have been added to Unified Manager, and the cluster key has been obtained.
- SVMs have been created on the clusters.
- The SVMs support CIFS and NFS services. Provisioning file shares might fail if the SVMs do not support the required services.
- The FCP port is online for port provisioning.

1. Determine whether Data LIFs or access endpoints are available on the SVM on which you want to create the CIFS share. Get the list of available access endpoints on the SVM:

Category	HTTP verb	Path
storage-provider	GET	`/storage-provider/access-endpoints /storage-provider/access-endpoints/{key}`

Sample cURL

```
curl -X GET "https://<hostname>/api/storage-provider/access-endpoints?resource.key=7d5a59b3-953a-11e8-8857-00a098dcc959" -H "accept: application/json" -H "Authorization: Basic <Base64EncodedCredentials>"
```

2. If your access endpoint is available on the list, obtain the access endpoint key, else create the access endpoint.

Ensure that you create access endpoints that have the CIFS protocol enabled on them. Provisioning CIFS shares fails unless you have created an access endpoint with the CIFS protocol enabled on it.

Category	HTTP verb	Path
storage-provider	POST	`/storage-provider/access-endpoints`

Sample cURL

You must enter the details of the access endpoint that you want to create, as input parameters.

```
curl -X POST "https://<hostname>/api/storage-provider/access-endpoints"
-H "accept: application/json" -H "Content-Type: application/json" -H
"Authorization: Basic <Base64EncodedCredentials>"
{ \"data_protocols\": \"nfs\",
\"fileshare\": { \"key\": \"cbd1757b-0580-11e8-bd9d-
00a098d39e12:type=volume,uuid=f3063d27-2c71-44e5-9a69-a3927c19c8fc\" },
\"gateway\": \"10.132.72.12\",
\"ip\": { \"address\": \"10.162.83.26\",
\"ha_address\": \"10.142.83.26\",
\"netmask\": \"255.255.0.0\" },
\"lun\": { \"key\": \"cbd1757b-0580-11e8-bd9d-
00a098d39e12:type=lun,uuid=d208cc7d-80a3-4755-93d4-5db2c38f55a6\" },
\"mtu\": 15000, \"name\": \"aep1\",
\"svm\": { \"key\": \"cbd1757b-0580-11e8-bd9d-
00a178d39e12:type=vserver,uuid=1d1c3198-fc57-11e8-99ca-00a098d38e12\" },
\"vlan\": 10}"
```

The JSON output displays a Job object key that you can use to verify the access endpoint that you created.

3. Verify the access endpoint:

Category	HTTP verb	Path
management-server	GET	`/management-server/jobs/{key}`

4. Determine whether you have to create a CIFS share or an NFS file share. For creating CIFS shares, follow these substeps:
   a. Determine whether the CIFS server is configured on your SVM, that is determine whether an Active Directory mapping is created on the SVM.

Category	HTTP verb	Path
storage-provider	GET	`/storage-provider/active-directories-mappings`

   b. If the Active Directory mapping is created, take the key, else create the Active Directory mapping on the SVM.

Category	HTTP verb	Path
storage-provider	POST	`/storage-provider/active-directories-mappings`

Sample cURL

You must enter the details for creating the Active Directory mapping, as the input parameters.

```
curl -X POST "https://<hostname>/api/storage-provider/active-
directories-mappings" -H "accept: application/json" -H "Content-Type:
application/json" -H "Authorization: Basic
<Base64EncodedCredentials>"
{ \"_links\": {},
\"dns\": \"10.000.000.000\",
\"domain\": \"example.com\",
\"password\": \"string\",
\"svm\": { \"key\": \"9f4ddea-e395-11e9-b660-
005056a71be9:type=vserver,uuid=191a554a-f0ce-11e9-b660-005056a71be9\"
},
\"username\": \"string\"}"
```

This is a synchronous call and you can verify the creation of the Active Directory mapping in the output. In case of an error, the error message is displayed for you to troubleshoot and rerun the request.

5. Obtain the SVM key for the SVM on which you want to create the CIFS share or the NFS file share, as described in the *Verifying SVMs on clusters* workflow topic.
6. Obtain the key for the Performance Service Level by running the following API and retrieving the key from the response.

Category	HTTP verb	Path
storage-provider	GET	`/storage-provider/performance-service-levels`

You can retrieve the details of the system-defined Performance Service Levels by setting the `system_defined` input parameter to `true`. From the output, obtain the key of the Performance Service Level that you want to apply on the file share.

7. Optionally, obtain the Storage Efficiency Policy key for the Storage Efficiency Policy that you want to apply on the file share by running the following API and retrieving the key from the response.

Category	HTTP verb	Path
storage-provider	GET	`/storage-provider/storage-efficiency-policies`

8. Create the file share. You can create a file share that supports both CIFS and NFS by specifying the access control list and export policy. The following substeps provide information if you want to create a file share for supporting only one of the protocols on the volume. You can also update an NFS file share to include the access control list after you have created the NFS share. For information, see the *Modifying*

storage workloads topic.

a. For creating only a CIFS share, gather the information about access control list (ACL). For creating the CIFS share, provide valid values for the following input parameters. For each user group that you assign, an ACL is created when a CIFS/SMB share is provisioned. Based on the values you enter for ACL and Active Directory mapping, the access control and mapping are determined for the CIFS share when it is created.

A cURL command with sample values

```
{
  "access_control": {
    "acl": [
      {
        "permission": "read",
        "user_or_group": "everyone"
      }
    ],
    "active_directory_mapping": {
      "key": "3b648c1b-d965-03b7-20da-61b791a6263c"
    },
```

b. For creating only an NFS file share, gather the information about the export policy. For creating the NFS file share, provide valid values for the following input parameters. Based on your values, the export policy is attached with the NFS file share when it is created.

> While provisioning the NFS share, you can either create an export policy by providing all the required values or provide the export policy key and reuse an existing export policy. If you want to reuse an export policy for the storage VM, you need to add the export policy key. Unless you know the key, you can retrieve the export policy key by using the `/datacenter/protocols/nfs/export-policies` API. For creating a new policy, you must enter the rules as displayed in the following sample. For the entered rules, the API tries to search for an existing export policy by matching the host, storage VM, and rules. If there is an existing export policy, it is used. Otherwise a new export policy is created.

A cURL command with sample values

```
"export_policy": {
      "key": "7d5a59b3-953a-11e8-8857-
00a098dcc959:type=export_policy,uuid=1460288880641",
      "name_tag": "ExportPolicyNameTag",
      "rules": [
        {
          "clients": [
            {
              "match": "0.0.0.0/0"
            }
```

After configuring access control list and export policy, provide the valid values for the mandatory input parameters for both CIFS and NFS file shares:

Storage Efficiency Policy is an optional parameter for creating file shares.

Category	HTTP verb	Path
storage-provider	POST	`/storage-provider/file-shares`

The JSON output displays a Job object key that you can use to verify the file share that you created. . Verify the file share creation by using the Job object key returned in querying the job:

Category	HTTP verb	Path
management-server	GET	`/management-server/jobs/{key}`

At the end of the response, you see the key of the file share created.

```
    ],
    "job_results": [
        {
            "name": "fileshareKey",
            "value": "7d5a59b3-953a-11e8-8857-
00a098dcc959:type=volume,uuid=e581c23a-1037-11ea-ac5a-00a098dcc6b6"
        }
    ],
    "_links": {
        "self": {
            "href": "/api/management-server/jobs/06a6148bf9e862df:-
2611856e:16e8d47e722:-7f87"
        }
    }
}
```

1. Verify the creation of the file share by running the following API with the returned key:

Category	HTTP verb	Path
storage-provider	GET	`/storage-provider/file-shares/{key}`

Sample JSON output

You can see that the POST method of `/storage-provider/file-shares` internally invokes all the APIs required for each of the functions and creates the object. For example, it invokes the `/storage-provider/performance-service-levels/` API for assigning the Performance Service Level on the file share.

```
{
    "key": "7d5a59b3-953a-11e8-8857-
00a098dcc959:type=volume,uuid=e581c23a-1037-11ea-ac5a-00a098dcc6b6",
    "name": "FileShare_377",
    "cluster": {
        "uuid": "7d5a59b3-953a-11e8-8857-00a098dcc959",
        "key": "7d5a59b3-953a-11e8-8857-
00a098dcc959:type=cluster,uuid=7d5a59b3-953a-11e8-8857-00a098dcc959",
        "name": "AFFA300-206-68-70-72-74",
        "_links": {
            "self": {
                "href": "/api/datacenter/cluster/clusters/7d5a59b3-953a-
11e8-8857-00a098dcc959:type=cluster,uuid=7d5a59b3-953a-11e8-8857-
00a098dcc959"
            }
```

```
        }
    },
    "svm": {
        "uuid": "b106d7b1-51e9-11e9-8857-00a098dcc959",
        "key": "7d5a59b3-953a-11e8-8857-
00a098dcc959:type=vserver,uuid=b106d7b1-51e9-11e9-8857-00a098dcc959",
        "name": "RRT_ritu_vs1",
        "_links": {
            "self": {
                "href": "/api/datacenter/svm/svms/7d5a59b3-953a-11e8-
8857-00a098dcc959:type=vserver,uuid=b106d7b1-51e9-11e9-8857-
00a098dcc959"
            }
        }
    },
    "assigned_performance_service_level": {
        "key": "1251e51b-069f-11ea-980d-fa163e82bbf2",
        "name": "Value",
        "peak_iops": 75,
        "expected_iops": 75,
        "_links": {
            "self": {
                "href": "/api/storage-provider/performance-service-
levels/1251e51b-069f-11ea-980d-fa163e82bbf2"
            }
        }
    },
    "recommended_performance_service_level": {
        "key": null,
        "name": "Idle",
        "peak_iops": null,
        "expected_iops": null,
        "_links": {}
    },
    "space": {
        "size": 104857600
    },
    "assigned_storage_efficiency_policy": {
        "key": null,
        "name": "Unassigned",
        "_links": {}
    },
    "access_control": {
        "acl": [
            {
                "user_or_group": "everyone",
```

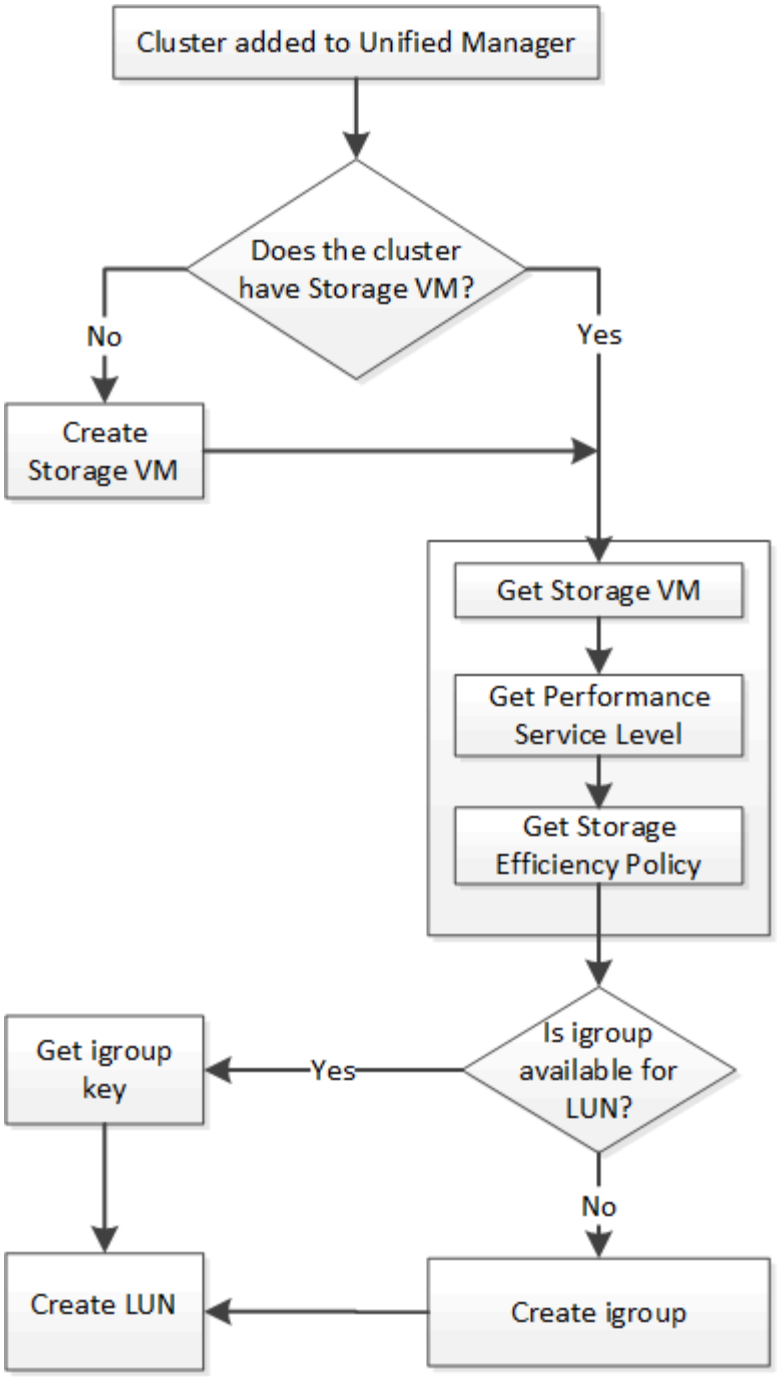
```
                "permission": "read"
            }
        ],
        "export_policy": {
            "id": 1460288880641,
            "key": "7d5a59b3-953a-11e8-8857-
00a098dcc959:type=export_policy,uuid=1460288880641",
            "name": "default",
            "rules": [
                {
                    "anonymous_user": "65534",
                    "clients": [
                        {
                            "match": "0.0.0.0/0"
                        }
                    ],
                    "index": 1,
                    "protocols": [
                        "nfs3",
                        "nfs4"
                    ],
                    "ro_rule": [
                        "sys"
                    ],
                    "rw_rule": [
                        "sys"
                    ],
                    "superuser": [
                        "none"
                    ]
                },
                {
                    "anonymous_user": "65534",
                    "clients": [
                        {
                            "match": "0.0.0.0/0"
                        }
                    ],
                    "index": 2,
                    "protocols": [
                        "cifs"
                    ],
                    "ro_rule": [
                        "ntlm"
                    ],
                    "rw_rule": [
```

```
                        "ntlm"
                    ],
                    "superuser": [
                        "none"
                    ]
                }
            ],
            "_links": {
                "self": {
                    "href": "/api/datacenter/protocols/nfs/export-
policies/7d5a59b3-953a-11e8-8857-
00a098dcc959:type=export_policy,uuid=1460288880641"
                }
            }
        }
    },
    "_links": {
        "self": {
            "href": "/api/storage-provider/file-shares/7d5a59b3-953a-
11e8-8857-00a098dcc959:type=volume,uuid=e581c23a-1037-11ea-ac5a-
00a098dcc6b6"
        }
    }
}
```

Provisioning LUNs by using APIs

You can provision LUNs on your Storage Virtual Machines (SVMs) by using the provisioning APIs provided as a part of Active IQ Unified Manager. This provisioning workflow details the steps for retrieving the keys of the SVMs, Performance Service Levels, and Storage Efficiency Policies before creating the LUN.

The following diagram illustrates the steps in a LUN provisioning workflow.



This workflow assumes that the ONTAP clusters have been added to Unified Manager, and the cluster key has been obtained. The workflow also assumes that the SVMs have already been created on the clusters.

1. Obtain the SVM key for the SVM on which you want to create the LUN, as described in the *Verifying SVMs on clusters* workflow topic.
2. Obtain the key for the Performance Service Level by running the following API and retrieving the key from the response.

Category	HTTP verb	Path
storage-provider	GET	`/storage-provider/performance-service-levels`

You can retrieve the details of the system-defined Performance Service Levels by setting the `system_defined` input parameter to `true`. From the output, obtain the key of the Performance Service Level that you want to apply on the LUN.

3. Optionally, obtain the Storage Efficiency Policy key for the Storage Efficiency Policy that you want to apply on the LUN by running the following API and retrieving the key from the response.

Category	HTTP verb	Path
storage-provider	GET	`/storage-provider/storage-efficiency-policies`

4. Determine if initiator groups (igroups) have been created to grant access to the LUN target that you want to create.

Category	HTTP verb	Path
datacenter	GET	`/datacenter/protocols/san/igroups` `/datacenter/protocols/san/igroups/{key}`

You must enter the parameter value for indicating the SVM for which the igroup has authorized access. Additionally, if you want to query a particular igroup, enter the igroup name (key) as an input parameter.

5. In the output, if you can find the igroup that you want to grant access to, obtain the key. Otherwise create the igroup.

Category	HTTP verb	Path
datacenter	POST	`/datacenter/protocols/san/igroups`

You must enter the details of the igroup that you want to create, as the input parameters. This is a synchronous call and you can verify the igroup creation in the output. In case of an error, a message is displayed for you to troubleshoot and rerun the API.

6. Create the LUN.

Category	HTTP verb	Path
storage-provider	POST	`/storage-provider/luns`

For creating the LUN, ensure that you have added the retrieved values as mandatory input parameters.

Storage Efficiency Policy is an optional parameter for creating LUNs.

Sample cURL

You must enter all the details of the LUN that you want to create, as the input parameters.

The JSON output displays a Job object key that you can use to verify the LUN that you created.

7. Verify the LUN creation by using the Job object key returned in querying the Job:

Category	HTTP verb	Path
management-server	GET	`/management-server/jobs/{key}`

At the end of the response, you see the key of the LUN created.

8. Verify the creation of the LUN by running the following API with the returned key:

Category	HTTP verb	Path
storage-provider	GET	`/storage-provider/luns/{key}`

Sample JSON output

You can see that the POST method of `/storage-provider/luns` internally invokes all the APIs required for each of the functions and creates the object. For example, it invokes the `/storage-provider/performance-service-levels/` API for assigning the Performance Service Level on the LUN.

== Troubleshooting steps for failure in LUN creation or mapping

On completing this workflow, you might still see a failure in your LUN creation. Even if the LUN is created successfully, the LUN mapping with the igroup might fail due to an unavailability of a SAN LIF or access endpoint on the node on which you create the LUN. In case of a failure, you can see the following message:

```
The nodes <node_name> and <partner_node_name> have no LIFs configured with
the iSCSI or FCP protocol for Vserver <server_name>. Use the access-
endpoints API to create a LIF for the LUN.
```

Follow these troubleshooting steps to work around this failure.

1. Create an access endpoint supporting ISCSI/FCP protocol on the SVM on which you tried creating the LUN.

Category	HTTP verb	Path
storage-provider	POST	`/storage-provider/access-endpoints`

Sample cURL

You must enter the details of the access endpoint that you want to create, as the input parameters.

> Ensure that in the input parameter you have added the address to indicate the home node of the LUN and the ha_address to indicate the partner node of the home node. When you run this operation, it creates access endpoints on both the home node and the partner node.

2. Query the job with the Job object key returned in the JSON output to verify that it has run successfully to add the access endpoints on the SVM and that the iSCSI/FCP services have been enabled on the SVM.

Category	HTTP verb	Path
management-server	GET	`/management-server/jobs/{key}`

Sample JSON output

At the end of the output, you can see the key of the access endpoints created. In the following output, the "name": "accessEndpointKey" value indicates the access endpoint created on the home node of the LUN, for which the key is 9c964258-14ef-11ea-95e2-00a098e32c28. The "name": "accessEndpointHAKey" value indicates the access endpoint created on the partner node of the home node, for which the key is 9d347006-14ef-11ea-8760-00a098e3215f.

3. Modify the LUN to update the igroup mapping. For more information about workflow modification, see “Modifying storage workloads”.

Category	HTTP verb	Path
storage-provider	PATCH	`/storage-provider/lun/{key}`

In the input, specify the igroup key with which you want to update the LUN mapping, along with the LUN key.

Sample cURL

The JSON output displays a Job object key that you can use to verify whether the mapping is successful.

4. Verify the LUN mapping by querying with the LUN key.

Category	HTTP verb	Path
storage-provider	GET	`/storage-provider/luns/{key}`

Sample JSON output

In the output you can see the LUN has been successfully mapped with the igroup (key d19ec2fa-fec7-11e8-b23d-00a098e32c28) with which it was initially provisioned.

Modifying storage workloads by using APIs

Modifying storage workloads consists of updating LUNs or file shares with missing parameters, or changing the existing parameters.

This workflow takes the example of updating Performance Service Levels for LUNs and file shares.

The workflow assumes that the LUN or file share has been provisioned with Performance Service Levels.

Modifying file shares

While modifying a file share, you can update the following parameters:

- Capacity or size.
- Online or offline setting.
- Storage Efficiency Policy.
- Performance Service Level.
- Access control list (ACL) settings.
- Export policy settings. You can also delete export policy parameters and revert the default (empty) export policy rules on the file share.

During a single API run, you can update only one parameter.

This procedure describes adding a Performance Service Level to a file share. You can use the same procedure for updating any other file share property.

1. Obtain the CIFS share or NFS file share key of the file share that you want to update. This API queries all the file shares on your data center. Skip this step if you already know the file share key.

Category	HTTP verb	Path
storage-provider	GET	`/storage-provider/file-shares`

2. View the details of the file share by running the following API with the file share key that you obtained.

Category	HTTP verb	Path
storage-provider	GET	`/storage-provider/file-shares/{key}`

View the details of the file share in the output.

```
"assigned_performance_service_level": {
        "key": null,
        "name": "Unassigned",
        "peak_iops": null,
        "expected_iops": null,
        "_links": {}
      },
```

3. Obtain the key for the Performance Service Level that you want to assign on this file share. Currently no policy is assigned to it.

Category	HTTP verb	Path
Performance Service Levels	GET	`/storage-provider/performance-service-levels`

> You can retrieve the details of the system-defined Performance Service Levels by setting the `system_defined` input parameter to `true`. From the output, obtain the key of the Performance Service Level that you want to apply to the file share.

4. Apply the Performance Service Level on the file share.

Category	HTTP verb	Path
Storage Provider	PATCH	`/storage-provider/file-shares/{key}`

In the input, you must specify only the parameter that you want to update, along with the file share key. In this case, it is the key of the Performance Service Level.

Sample cURL

```
curl -X POST "https://<hostname>/api/storage-provider/file-shares"`-H
"accept: application/json" -H "Authorization: Basic
<Base64EncodedCredentials>" -d
"{
\"performance_service_level\": { \"key\": \"1251e51b-069f-11ea-980d-
fa163e82bbf2\" },
}"
```

The JSON output displays a Job object that you can use to verify the whether the access endpoints on the home and partner nodes have been created successfully.

5. Verify whether the Performance Service Level has been added to the file share by using the Job object key displayed in your output.

Category	HTTP verb	Path
Management Server	GET	`/management-server/jobs/{key}`

If you query by the ID of the Job object, you see whether the file share is updated successfully. In case of a failure, troubleshoot the failure and run the API again. On successful creation, query the file share to see the modified object:

Category	HTTP verb	Path
storage-provider	GET	`/storage-provider/file-shares/{key}`

View the details of the file share in the output.

```
"assigned_performance_service_level": {
        "key": "1251e51b-069f-11ea-980d-fa163e82bbf2",
        "name": "Value",
        "peak_iops": 75,
        "expected_iops": 75,
        "_links": {
            "self": {
                "href": "/api/storage-provider/performance-service-
levels/1251e51b-069f-11ea-980d-fa163e82bbf2"
            }
        }
```

Updating LUNs

While updating a LUN, you can modify the following parameters:

- Capacity or size
- Online or offline setting
- Storage Efficiency Policy
- Performance Service Level
- LUN map

During a single API run, you can update only one parameter.

This procedure describes adding a Performance Service Level to a LUN. You can use the same procedure for updating any other LUN property.

1. Obtain the LUN key of the LUN that you want to update. This API returns details of all the LUNS in your data center. Skip this step if you already know the LUN key.

Category	HTTP verb	Path
Storage Provider	GET	`/storage-provider/luns`

2. View the details of the LUN by running the following API with the LUN key that you obtained.

Category	HTTP verb	Path
Storage Provider	GET	`/storage-provider/luns/{key}`

View the details of the LUN in the output. You can see that there is no Performance Service Level assigned to this LUN.

Sample JSON output

```
"assigned_performance_service_level": {
      "key": null,
      "name": "Unassigned",
      "peak_iops": null,
      "expected_iops": null,
      "_links": {}
    },
```

3. Obtain the key for the Performance Service Level that you want to assign to the LUN.

Category	HTTP verb	Path
Performance Service Levels	GET	`/storage-provider/performance-service-levels`

You can retrieve the details of the system-defined Performance Service Levels by setting the `system_defined` input parameter to `true`. From the output, obtain the key of the Performance Service Level that you want to apply on the LUN.

4. Apply the Performance Service Level on the LUN.

Category	HTTP verb	Path
Storage Provider	PATCH	`/storage-provider/lun/{key}`

In the input, you must specify only the parameter that you want to update, along with the LUN key. In this case it is the key of the Performance Service Level.

Sample cURL

```
curl -X PATCH "https://<hostname>/api/storage-provider/luns/7d5a59b3-953a-11e8-8857-00a098dcc959" -H "accept: application/json" -H "Content-Type: application/json" H "Authorization: Basic <Base64EncodedCredentials>" -d "{ \"performance_service_level\": { \"key\": \"1251e51b-069f-11ea-980d-fa163e82bbf2\" }"
```

The JSON output displays a Job object key that you can use to verify the LUN that you updated.

5. View the details of the LUN by running the following API with the LUN key that you obtained.

Category	HTTP verb	Path
Storage Provider	GET	`/storage-provider/luns/{key}`

View the details of the LUN in the output. You can see that the Performance Service Level is assigned to this LUN.

Sample JSON output

```
    "assigned_performance_service_level": {
       "key": "1251e51b-069f-11ea-980d-fa163e82bbf2",
       "name": "Value",
       "peak_iops": 75,
       "expected_iops": 75,
       "_links": {
           "self": {
               "href": "/api/storage-provider/performance-service-
levels/1251e51b-069f-11ea-980d-fa163e82bbf2"
           }
```

Modifying an NFS file share by using APIs to support CIFS

You can modify an NFS file share to support CIFS protocol. During file share creation, it is possible to specify both access control list (ACL) parameters and export policy rules for the same file share. However, if you want to enable CIFS on the same volume where you created an NFS file share, you can update the ACL parameters on that file share to support CIFS.

What you'll need

1. An NFS file share must have been created with only the export policy details. For information, see *Managing file shares* and *Modifying storage workloads*.
2. You must have the file share key to run this operation. For information about viewing file share details and retrieving the file share key by using the Job ID, see *Provisioning CIFS and NFS file shares*.

This is applicable for an NFS file share that you created by adding only export policy rules and not ACL parameters. You modify the NFS file share to include the ACL parameters.

Steps

1. On the NFS file share, perform a `PATCH` operation with the ACL details for allowing CIFS access.

Category	HTTP verb	Path
storage-provider	PATCH	`/storage-provider/file-shares`

Sample cURL

Based on the access privileges you assign to the user group, as displayed in the following sample, an ACL is created and assigned to the file share.

```
{
  "access_control": {
    "acl": [
      {
        "permission": "read",
        "user_or_group": "everyone"
      }
    ],
    "active_directory_mapping": {
      "key": "3b648c1b-d965-03b7-20da-61b791a6263c"
    }
```

Sample JSON output

The operation returns the Job ID of the Job that runs the update.

2. Verify whether the parameters have been added correctly by querying the file share details for the same file share.

Category	HTTP verb	Path
storage-provider	GET	`/storage-provider/file-shares/{key}`

Sample JSON output

```
"access_control": {
        "acl": [
            {
                "user_or_group": "everyone",
                "permission": "read"
            }
        ],
        "export_policy": {
            "id": 1460288880641,
            "key": "7d5a59b3-953a-11e8-8857-
00a098dcc959:type=export_policy,uuid=1460288880641",
            "name": "default",
            "rules": [
                {
                    "anonymous_user": "65534",
                    "clients": [
                        {
                            "match": "0.0.0.0/0"
                        }
                    ],
```

```
                    "index": 1,
                    "protocols": [
                        "nfs3",
                        "nfs4"
                    ],
                    "ro_rule": [
                        "sys"
                    ],
                    "rw_rule": [
                        "sys"
                    ],
                    "superuser": [
                        "none"
                    ]
                },
                {
                    "anonymous_user": "65534",
                    "clients": [
                        {
                            "match": "0.0.0.0/0"
                        }
                    ],
                    "index": 2,
                    "protocols": [
                        "cifs"
                    ],
                    "ro_rule": [
                        "ntlm"
                    ],
                    "rw_rule": [
                        "ntlm"
                    ],
                    "superuser": [
                        "none"
                    ]
                }
            ],
            "_links": {
                "self": {
                    "href": "/api/datacenter/protocols/nfs/export-
policies/7d5a59b3-953a-11e8-8857-
00a098dcc959:type=export_policy,uuid=1460288880641"
                }
            }
        }
    },
```

```
    "_links": {
        "self": {
            "href": "/api/storage-provider/file-shares/7d5a59b3-953a-
11e8-8857-00a098dcc959:type=volume,uuid=e581c23a-1037-11ea-ac5a-
00a098dcc6b6"
        }
    }
```

You can see the ACL assigned along with the export policy to the same file share.