Common workflows

Cloud Manager Automation

NetApp
February 28, 2022

This PDF was generated from https://docs.netapp.com/us-en/cloud-manager-automation/cm/wf_common.html on February 28, 2022. Always check docs.netapp.com for the latest.

Table of Contents

Common workflows

Common workflows for all cloud providers

There are several workflows you can use with any of the public cloud providers.

Before using any of the Cloud Manager REST API workflows, review Get started section.

Workflow categories

The common workflows are organized into the following functional categories:

Identity and access

These workflows are typically used to obtain an access token to identify the API caller and manage authorized access to the resources.

SaaS marketplace

These workflows allow you to manage the subscription of a cloud provider account. You can attach a Cloud Manager SaaS subscription to the cloud provider account or retrieve a list of subscriptions.

NetApp Support site

You can perform specific workflows to manage the NSS keys as part of registering a Cloud Manager REST resource for support. Specific NSS keys or all of the available keys can be retrieved. You can also create or delete an NSS key if needed.

Internal task

You can use these workflows to retrieve the information and status of a background task.

SnapMirror Replication

These workflows allow you to manage and monitor SnapMirror relationships to replicate data between source and destination working environments. This supports disaster recovery and backup which optimizes data availability.

Connector setup

You must have a **Connector** for the cloud environment before creating a working environment and performing other activities using the workflows. You can create a Connector using the Cloud Manager web UI. When you create a Connector, Cloud Manager adds the cloud provider account that you deployed the Connector in to your list of available accounts. Your cloud provider account needs to have the right permissions in order to create a Connector.

Review Learn about Connectors to know how to create and deploy a Connector.

Identity and access

Get supported services

You can use this workflow to retrieve information about the Cloud Manager supported services including the *client id* and *account id* values.

This workflow describes how to use the REST API to retrieve the two ID values. You can also use the Cloud Manager web UI to get these values. See Get the client and account identifiers for more information.

1. Get the supported services

This API returns information about the supported services.

HTTP method	Resource path
GET	/occm/api/occm/system/support-services

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/occm/system/support-
services' --header 'Content-Type: application/json' --header 'x-agent-Id:
<AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>'
```

Input parameters

None

Output

The JSON output example includes the supported services information. You can locate the `clientId` value under `auth0Information` and the `accountId` value under `tenancyServiceInformation`.

JSON output example

```
{
    "asupEnabled": true,
    "cognitoEnabled": true,
    "kinesisEnabled": true,
    "intercomEnabled": true,
    "liveChatEnabled": true,
    "volumeViewEnabled": true,
    "portalService": {
        "usePortalAuthentication": true,
        "auth0Information": {
            "domain": "netapp-cloud-account.auth0.com",
            "audience": "https://api.cloud.netapp.com",
            "clientId": "WsefXFuCJJvMKCMppR65jCktHAQBWFs4"
        },
        "portalInformation": {
            "edit_user_url":
"https://services.cloud.netapp.com/?userMenuOpen",
            "portalBackEnd": "https://api.services.cloud.netapp.com",
            "portalFrontEnd": "https://services.cloud.netapp.com"
        },
        "saasFrontEnd": "https://cloudmanager.netapp.com"
    },
    "intercomAppId": "brf2h510",
    "tenancyServiceInformation": {
        "accountId": "account-xxx1234",
        "url": "https://cloudmanager.cloud.netapp.com/tenancy",
        "accountWidgetUrl":
"https://services.cloud.netapp.com/accountWidgetLoader.js",
        "agentsMgmtUrl": "https://cloudmanager.cloud.netapp.com/agents-
mgmt",
        "forwarderUrlOverride": "https://cloudmanager.cloud.netapp.com"
    },
    "saasMpServiceInformation": {
        "productUrlAws":
"http://aws.amazon.com/marketplace/pp/B086BQCW8P",
        "productUrlGcp":
"https://console.cloud.google.com/marketplace/details/netapp-
cloudmanager/cloud-manager",
        "url": "https://cloudmanager.cloud.netapp.com/saas-mp",
        "productUrlAzure": "https://portal.azure.com/#create/netapp.test-
cloud-manager"
    },
    "cvsInformation": "https://services.cloud.netapp.com",
    "backupToS3Information": {
        "enabled": true,
        "disableReason": null,
        "notSubscribedReason": false,
        "cbsEnabled": true,
        "onpremSupported": true
    },
    "complianceEnabled": false,
    "ipaServiceInformation": {
        "url": "https://cloudmanager.cloud.netapp.com/ipa"
    },
    "servicesInformation": {
        "compliance": {
            "enabled": false,
            "govSupported": true,
            "iframeUrl": "https://dev-
components.cloudmanager.netapp.com/cloudmanager_compliance/index.html"
        },
        "cloudSync": {
            "iframeUrl": "https://dev.cloudsync.netapp.com",
            "url": "https://dev.cloudsync.netapp.com",
            "enabled": false,
            "apiUrl": "https://api.dev.cloudsync.netapp.com",
            "govSupported": false
        },
        "monitoring": {
            "enabled": false,
            "govSupported": true,
            "iframeUrl": "https://dev-
components.cloudmanager.netapp.com/monitoring/index.html",
            "apiUrl": "https://cloudmanager.cloud.netapp.com"
        },
        "astra": {
            "enabled": true,
            "govSupported": false,
            "iframeUrl": "https://engint.astra.netapp.io"
        },
        "gfc": {
            "iframeUrl": "https://dev-
components.cloudmanager.netapp.com/cloudmanager_gfc/index.html",
            "dashboardUrl": "https://dev-
components.cloudmanager.netapp.com/cloudmanager_gfc/index.html",
            "enabled": false,
            "apiUrl": "https://cloudmanager.cloud.netapp.com",
            "govSupported": false
        },
        "cloudTiering": {
            "iframeUrl": "https://tiering.cloud.netapp.com",
            "url": "https://tiering.cloud.netapp.com",
            "enabled": false,
            "apiUrl": "https://tiering.cloud.netapp.com",
            "govSupported": false
        },
        "snapCenter": {
            "enabled": false,
            "govSupported": false,
            "iframeUrl": "https://dev-
components.cloudmanager.netapp.com/cloudmanager_snapcenter/index.html",
            "apiUrl": "https://cloudmanager.cloud.netapp.com"
        },
        "k8s": {
            "enabled": false,
            "govSupported": false,
            "iframeUrl": "https://dev-
components.cloudmanager.netapp.com/cloudmanager_k8s/index.html",
            "apiUrl": "https://cloudmanager.cloud.netapp.com"
        },
        "spot": {
            "enabled": true,
            "govSupported": false,
            "iframeUrl": "https://dev-components.cloudmanager.netapp.com/cloudmanager_spot/index.html",
            "apiUrl": "https://cloudmanager.cloud.netapp.com"
        },
        "sfr": {
            "enabled": false,
            "govSupported": true,
            "iframeUrl": "https://dev-components.cloudmanager.netapp.com/cloudmanager_sfr/index.html",
            "dashboardUrl": "https://dev-components.cloudmanager.netapp.com/cloudmanager_sfr/index.html"
        },
        "activeIq": {
            "enabled": true,
            "govSupported": false,
            "iframeUrl": "https://digitaladvisor.aws.techteam.netapp.com"
        }
    },
    "setupInfo": {
        "isSetup": true,
        "isPendingConnectivitySet": false,
        "needCertificate": false,
        "runningInDocker": false
    },
    "useCompliancePrivateIpContainerMode": false
}
```

Get cloud provider accounts

You can retrieve a list of the Cloud Manager accounts available for the supported cloud platforms.

1. Get the cloud provider accounts

HTTP method	Resource path
GET	/occm/api/accounts/cloud-providers

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/accounts/cloud-providers'
--header 'Content-Type: application/json' --header 'x-agent-id:
<AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>'
```

Input parameters

None

Output

The JSON output example shows the cloud provider accounts returned in separate named arrays.

JSON output example

```
{
    "awsAccounts": [
        {
            "publicId": "InstanceProfile",
            "accountName": "Instance Profile",
            "accountType": "INSTANCE_PROFILE",
            "accountId": "733004784675",
            "accessKey": "",
            "assumeRole": null,
            "occmRole": "occm",
            "vsaList": [
                {
                    "publicId": "VsaWorkingEnvironment-N6BPfglr",
                    "name": "ziv04we01ha",
                    "workingEnvironmentType": "AWSHA"
                }
            ],
            "subscriptionId": "subsctionIDshownhere"
        }
    ],
    "azureAccounts": [],
    "gcpStorageAccounts": [],
    "nssAccounts": []
}
```

Get tenants

You can retrieve a list of tenants visible to the user currently signed in through the account workspace.

1. Get the account identifier

Perform the workflow Get supported services to retrieve the account ID.

> You can also get the account identifier through the Cloud Manager web UI. See Get the client and account identifers for more information.

2. Get the account workspace

HTTP method	Resource path
GET	/tenancy/account/{account_id}/workspace

curl example

```
curl --location --request GET
"https://cloudmanager.cloud.netapp.com/tenancy/account/<ACCOUNT_ID>/worksp
ace" --header 'Content-Type: application/json' --header 'Authorization:
Bearer <ACCESS_TOKEN>'
```

Input parameters

Path parameter with the account ID.

Output

The JSON output example shows the list of tenants.

JSON output example

```
[
    {
        "workspacePublicId": "workspace-ced4x9X4",
        "workspaceName": "Workspace-1",
        "associatedAgents": []
    }
]
```

SaaS marketplace

Get SaaS marketplace account

You can use this workflow to retrieve the current marketplace account and subscriptions.

Before you begin

Every cloud account can *optionally* have a subscription associated with it. The subscription identifies how the various cloud services used by the account are charged. When retrieving the following SaaS accounts, notice that the `subscriptionId` values in the `cloudAccounts` array objects match the `id` values in the associated subscription arrays for the three cloud providers.

> ⓘ Each cloud account can have exactly zero or one subscription. You must have a subscription when creating a working environment using PAYGO ("pay as you go").

1. Get the SaaS accounts

HTTP method	Resource path
GET	/occm/api/occm/saas-mp-service/account

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/occm/saas-mp-
service/account' --header 'Content-Type: application/json' --header 'x-
agent-id: <AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>'
```

Input parameters

None

Output

The JSON output example shows the subscriptions and in each case the account it is associated with.

JSON output example

```
{
    "awsSubscriptions": [
        {
            "id": "awsid00000",
            "name": "aws-sub-a2",
            "provider": "aws",
            "active": true
        }
    ],
    "azureSubscriptions": [],
    "gcpSubscriptions": [
        {
            "id": "gcp-xxxx0000",
            "name": "GCP subscription",
            "provider": "gcp",
            "active": true
        }
    ],
    "eligibleForFreeTrialAws": false,
    "eligibleForFreeTrialGcp": false,
    "eligibleForFreeTrialAzure": false,
    "cloudAccounts": [
        {
            "cloudAccountId": "000000",
            "provider": "aws",
            "subscriptionId": "aws-xxxxx000000xxxxxxx0000"
        },
        {
            "cloudAccountId": "occm-dev",
            "provider": "gcp",
            "subscriptionId": "gcp-xxx00000xxx0000"
        },
        {
            "cloudAccountId": "occm-host",
            "provider": "gcp",
            "subscriptionId": "gcp-xxxx000000xxx00000"
        }
    ]
}
```

Attach SaaS subscription

You can use this workflow to attach a Cloud Manager SaaS subscription to a cloud provider account.

1. Select the cloud provider path parameter

Select the cloud provider from the list of allowed values below. You will use this value as the *provider* path parameter in the curl example in step 4.

- aws
- azure
- gcp

2. Determine the account identifier path parameter

Perform the workflow Get cloud provider accounts and choose the `accountId` value for the account.

3. Get the SaaS marketplace account

Perform the workflow Get SaaS marketplace account and choose the `id` value for the `subscriptionId` pamater.

4. Attach the subscription

HTTP method	Resource path
POST	/occm/api/occm/saas-mp-service/attach-subscription/{provider}/{cloudAccountId}

curl example

```
curl --location --request PUT
'https://cloudmanager.cloud.netapp.com/occm/api/occm/saas-mp-
service/attach-subscription/<PROVIDER>/<CLOUD_ACC_ID>' --header 'x-agent-
id: <AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>' --header
'Content-Type: application/json' --d @JSONinput
```

Input parameters

The JSON input example includes the input parameter.

JSON input example

```
{
  "subscriptionId": "gcp-saasMpIntegrationProductId-
saasMpCustomerIdentifier0"
}
```

Output

None

NetApp Support Site

Get an NSS key

You can retrieve the key for a specific NetApp Support Site account.

1. Select the account

Perform the workflow Get supported services and choose the `accountId` value for the path parameter.

2. Select the NSS key to retrieve

Perform the workflow Get NSS keys and choose the `id` value for the `nssKeyId` path parameter.

3. Retrieve the NSS key

HTTP method	Resource path
GET	/ipa/account/{accountId}/nss-keys/{nssKeyId}

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/ipa/account/<ACCOUNT_ID>/nss-
key/<NSS_KEY_ID>' --header 'Content-Type: application/json' --header 'x-
agent-id: <AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>'
```

Input parameters

Path parameters:

- <ACCOUNT_ID> `(accountId)`
- <NSS_KEY_ID> `(nssKeyId)`

Output

The NSS key and related information is returned as shown in the JSON output example.

JSON output example

```
{
    "id": "d69a5214-7d61-486e-8750-8e1f68601c43",
    "username": "misterziv",
    "resources": [],
    "credentialType": "password"
}
```

Get NSS keys

You can retrieve all of the available NetApp Support Site credential keys.

1. Select the account to use

Perform the workflow Get cloud provider accounts and choose the `accountId` value for the path parameter.

2. Retrieve the NSS keys

HTTP method	Resource path
GET	/ipa/account/{accountId}/nss-keys

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/ipa/account/<ACCOUNT_ID>/nss-
keys?includeResources=true' --header 'Content-Type: application/json'
--header 'x-agent-id: <AGENT_ID> --header 'Authorization: Bearer
<ACCESS_TOKEN>'
```

Input parameters

- Path parameter <ACCOUNT_ID> `(accountId)`
- Query parameter `includeResources` is optional

Output

The JSON output example includes a list of the NSS keys.

JSON output example

```
[
    {
        "id": "d69a5214-7d61-486e-8750-8e1f68601c43",
        "username": "zivtest03",
        "resources": [VsaWorkingEnvironment-4UHd0x6l],
        "credentialType": "password"
    },
    {
        "id": "98f2c5db-af76-428f-ae81-08fdfa1e04ae",
        "username": "zivtest23",
        "resources": [],
        "credentialType": "password"
    }
]
```

Delete an NSS key

You can delete an NetApp Support Site (NSS) key.

1. Select the account to use

Perform the workflow Get supported services and choose the `accountId` value for the path parameter.

2. Select the NSS key to delete

Perform the workflow Get NSS keys and choose the `id` value for the `nssKeyID` parameter.

3. Delete the NSS keys

HTTP method	Resource path
DELETE	/ipa/account/{accountId}/nss-keys/{nssKeyId}

curl example

```
curl --location --request DELETE
'https://cloudmanager.cloud.netapp.com/ipa/account/<ACCOUNT_ID>/nss-
key/<NSS_KEY_ID>' --header 'Content-Type: application/json' --header 'x-
agent-id: <AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>'
```

Input parameters

- Path parameter <ACCOUNT_ID> `(accountId)`
- Path parameter <NSS_KEY_ID> `(nssKeyId)`

Output

None

Internal tasks and support

Get active task

All of the requests made to the Cloud Manager REST API are processed asynchronously except those using the HTTP GET method. Each of these asynchronous requests is assigned a unique identifier which is returned to the caller in the response. You can use the request ID to retrieve information about the background task including its status.

1. Get the request identifier

After issuing any REST API call, the associated request identifier is returned in the `X-Response_Id` header. You must extract this value and use it in the path variable in the next step.

2. Get the task

HTTP method	Resource path
GET	/occm/api/audit/activeTask/{request_id}

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/audit/activeTask/<REQUEST_
ID>' --header 'Content-Type: application/json' --header 'x-agent-id:
<AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>'
```

Input parameters

Path parameter that contains the request identifier `<REQUEST_ID>` for the original task.

Output

The JSON output example includes the list of returned values. The `status` contains one of three values indicating the status of the task.

Status	Description
1	The asynchronous task completed successfully.
0	The background task is still running and has not completed.
-1	The asynchronous task completed but failed.

JSON output example

```
{
    "status": 0,
    "closeTime": -2147483648,
    "actionName": "Create Vsa Working Environment",
    "error": null
}
```

SnapMirror replication

Create a SnapMirror replication relationship

You can use this workflow to create a new SnapMirror replication relationship to an ONTAP working environment. You can replicate data between working environments by choosing a one-time data replication for data transfer, or a recurring schedule for disaster recovery or long-term retention.

ⓘ This workflow varies slightly depending on the cloud provider you are using.

1. Select the working environment

Based on the cloud provider, you need to perform a workflow to get the identifier for the working environment as shown in the following table.

Provider	Workflow
AWS	Perform the workflow Get working environments and choose the `publicId` and `svmName` values for the source and destination.

2. Select the LIFs

Perform the workflow Get intercluster LIFs and choose the `address` value for the source and destination.

3. Select the SnapMirror policy

Perform the workflow Get SnapMirror policies and choose the `name` value for the required schedule.

4. Select the SnapMirror schedule

Perform the workflow Get schedules and choose the `name` value for the required policy.

5. Select the volume names and related storage parameters

Based on the cloud provider, you need to perform a workflow to get the volume information.

Provider	Workflow
AWS	Perform the workflow Get volumes and choose the `name` and `svmName` and `aggregateName` values.

6. Create a quote

Based on the cloud provider, you need to perform a workflow to get the volume information.

Provider	Workflow
AWS	Perform the workflow Create quote and choose the `providerVolumeType` and `name` values.

7. Create the relationship

HTTP method	Path
POST	/occm/api/replication/vsa

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/replication/vsa' --header
'Content-Type: application/json' --header 'x-agent-id: <AGENT_ID>'
--header 'Authorization: Bearer <ACCESS_TOKEN>' --d @JSONinput
```

Input

The JSON input example includes the minimum list of input parameters. The `maxTransferRate` is maximum transfer rate limit in KB/s. Specify 0 to indicate no limit or an integer between `1024` and `2,147,482,624`.

JSON input example

```
{
  "replicationRequest": {
    "sourceWorkingEnvironmentId": "vsaworkingenvironment-sfrf3wvj",
    "destinationWorkingEnvironmentId": "vsaworkingenvironment-2qkd75xv",
    "sourceInterclusterLifIps": [
      "10.138.0.147"
    ],
    "destinationInterclusterLifIps": [
      "10.138.0.154"
    ],
    "policyName": "MirrorAllSnapshots",
    "scheduleName": "daily",
    "maxTransferRate": 102400
  },
  "replicationVolume": {
    "sourceSvmName": "svm_zivgcp01we02",
    "sourceVolumeName": "zivagg01vol01",
    "destinationVolumeName": "zivagg01vol03_copy",
    "destinationAggregateName": "aggr1",
    "numOfDisksApprovedToAdd": 0,
    "advancedMode": false,
    "destinationProviderVolumeType": "pd-ssd",
    "destinationSvmName": "svm_zivgcp01we03"
  }
}
```

Output

None

Update a SnapMirror replication relationship

You can use this workflow to update an existing SnapMirror replication relationship.

1. Select the working environment and related values

Perform the workflow Get relationships status and choose the `workingEnvironment` and `svmName` and `volumeName` values for the path parameters. All values are for the destination.

2. Update the relationship

HTTP method	Path
PUT	/occm/api/replication/{workingEnvironmentId}/{destinationSvmName}/{destinationVolumeName}

curl example

```
curl --location --request PUT
'https://cloudmanager.cloud.netapp.com/occm/api/replication/<WORKING_ENV_I
D/<SVM_NAME/<VOLUME_NAME>' --header 'Content-Type: application/json'
--header 'x-agent-id: <AGENT_ID>' --header 'Authorization: Bearer
<ACCESS_TOKEN>' --d @JSONinput
```

Input

Path parameters:

- <WORKING_ENV_ID> `(workingEnvironmentId)`
- <SVM_NAME> `(destinationSvmName)`
- <VOLUME_NAME> `(destinationVolumeName)`

The JSON input example includes some of the parameters you can update.

JSON input example

```
{
  "maxTransferRate": 0
}
```

Output

None

Delete a SnapMirror replication relationship

You can use this workflow to delete an existing SnapMirror replication relationship.

1. Select the working environment and related values

Perform the workflow Get relationships status. Choose the `workingEnvironmentId` and `svmName` and `volumeName` values for the path parameters. All values are for the destination.

2. Delete the relationship

HTTP method	Path
DELETE	/occm/api/replication/{destinationWorkingEnvironmentId}/{destinationSvmName}/{destinationVolumeName}

curl example

```
curl --location --request DELETE
'https://cloudmanager.cloud.netapp.com/occm/api/replication/<WORKING_ENV_I
D>/<SVM_NAME>/<VOLUME_NAME>' --header 'Content-Type: application/json'
--header 'x-agent-id: <AGENT_ID>' --header 'Authorization: Bearer
<ACCESS_TOKEN>'
```

Input

There are three path parameters, all of which apply to the destination:

- <WORKING_ENV_ID> `(destinationWorkingEnvironmentId)`
- <SVM_NAME> `(destinationSvmName)`
- <VOLUME_NAME> `(destinationVolumeName)`

Output

None

Get the SnapMirror relationships

You can retrieve all the SnapMirror relationship pairs.

1. Retrieve the relationships

HTTP method	Path
GET	/occm/api/replication/all-relationships

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/replication/all-
relationships' --header 'Content-Type: application/json' --header 'x-
agent-id: <AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>'
```

Input parameters

None

Output

An array of relationship pairs is returned as shown in the JSON output example.

JSON output example

```
{
    "relationships": [
        {
            "source": {
                "id": "vsaworkingenvironment-sfrf3wvj"
            },
            "target": {
                "id": "vsaworkingenvironment-2qkd75xv"
            }
        }
    ]
}
```

Get the status of the replication relationships

You can use this workflow to retrieve the status of all the SnapMirror replication relationships.

This workflow varies slightly depending on the cloud provider you are using.

1. Optionally select the tenant ID

Based on the cloud provider, you need to perform a workflow to get the tenant ID as shown in the following table.

Provider	Workflow
AWS	Perform the workflow Get working environments and choose the `tenantId` value.

2. Get the status of the relationships

HTTP method	Path
GET	/occm/api/replication/status

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/replication/status'
--header 'Content-Type: application/json' --header 'x-agent-id:
<AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

The query parameter `tenantId` is optional.

Output

The JSON output example includes the list of all the SnapMirror relationships with status.

JSON output example

```
[
    {
        "source": {
            "workingEnvironmentId": "vsaworkingenvironment-sfrf3wvj",
            "workingEnvironmentType": "VSA",
            "workingEnvironmentStatus": "ON",
            "clusterName": "zivgcp01we02",
            "region": "us-west1-b",
            "availabilityZone": null,
            "svmName": "svm_zivgcp01we02",
            "nodeName": null,
            "volumeName": "zivagg01vol01"
        },
        "destination": {
            "workingEnvironmentId": "vsaworkingenvironment-2qkd75xv",
            "workingEnvironmentType": "VSA",
            "workingEnvironmentStatus": "ON",
            "clusterName": "zivgcp01we03",
            "region": "us-west1-b",
            "availabilityZone": null,
            "svmName": "svm_zivgcp01we03",
            "nodeName": "zivgcp01we03-01",
            "volumeName": "zivagg01vol01_copy"
        },
        "mirrorState": "snapmirrored",
        "relationshipType": "extended_data_protection",
        "relationshipStatus": "idle",
        "relationshipProgress": null,
        "policy": "MirrorAllSnapshots",
        "policyType": "async_mirror",
        "schedule": "daily",
        "maxTransferRate": {
            "size": 102400.0,
            "unit": "KB"
        },
        "networkCompressionRatio": "1:1",
        "healthy": true,
        "unhealthyReason": null,
        "lagTime": {
            "length": 14012,
            "unit": "SECONDS"
        },
        "newestSnapshotName": "snapmirror.e7179420-5e45-11eb-8f27-
d7fea0402bd2_2150573386.2021-01-25_123451",
        "newestSnapshotCreated": 1611578092,
        "lastTransferInfo": {
            "transferType": "update",
            "transferSize": {
                "size": 6240.0,
                "unit": "Byte"
            },
            "transferDuration": {
                "length": 4,
                "unit": "SECONDS"
            },
            "transferEnded": 1611578097,
            "transferError": null
        },
        "currentTransferInfo": {
            "transferType": null,
            "transferPriority": null,
            "transferError": null
        },
        "totalTransferTime": {
            "length": 6,
            "unit": "SECONDS"
        },
        "totalTransferSize": {
            "size": 23792.0,
            "unit": "Byte"
        },
        "volumeUsedSize": {
            "size": 1032192.0,
            "unit": "Byte"
        },
        "volumeCapacityTier": {
            "size": 0.0,
            "unit": "Byte"
        }
    }
]
```

Get status of the replication relationships for a working environment

You can use this workflow to retrieve the status of all the SnapMirror replication relationships for a specific working environment.

ⓘ This workflow varies slightly depending on the cloud provider you are using.

1. Select the working environment

Based on the cloud provider, you need to perform a workflow to get the identifier for the working environment as shown in the following table.

Provider	Workflow
AWS	Perform the workflow Get working environments and choose the `publicId` value for the `workingEnvironmentId` query parameter.

2. Get the status of the relationships

HTTP method	Path
GET	/occm/api/replication/status/{workingEnvironmentId}

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/replication/status/<WORKIN
G_ENV_ID>' --header 'Content-Type: application/json' --header 'x-agent-id:
<AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

Path parameters:

- <WORKING_ENV_ID> `(workingEnvironmentId)`

Output

The JSON output example includes the list of all the SnapMirror relationships with status for a specific working environment.

JSON output example

```
[
    {
        "source": {
            "workingEnvironmentId": "vsaworkingenvironment-sfrf3wvj",
            "workingEnvironmentType": "VSA",
            "workingEnvironmentStatus": "ON",
            "clusterName": "zivgcp01we02",
            "region": "us-west1-b",
            "availabilityZone": null,
            "svmName": "svm_zivgcp01we02",
            "nodeName": null,
            "volumeName": "zivagg01vol01"
        },
        "destination": {
            "workingEnvironmentId": "vsaworkingenvironment-2qkd75xv",
            "workingEnvironmentType": "VSA",
            "workingEnvironmentStatus": "ON",
            "clusterName": "zivgcp01we03",
            "region": "us-west1-b",
            "availabilityZone": null,
            "svmName": "svm_zivgcp01we03",
            "nodeName": "zivgcp01we03-01",
            "volumeName": "zivagg01vol01_copy"
        },
        "mirrorState": "snapmirrored",
        "relationshipType": "extended_data_protection",
        "relationshipStatus": "idle",
        "relationshipProgress": null,
        "policy": "MirrorAllSnapshots",
        "policyType": "async_mirror",
        "schedule": "daily",
        "maxTransferRate": {
            "size": 102400.0,
            "unit": "KB"
        },
        "networkCompressionRatio": "1:1",
        "healthy": true,
        "unhealthyReason": null,
        "lagTime": {
            "length": 14886,
            "unit": "SECONDS"
        },
        "newestSnapshotName": "snapmirror.e7179420-5e45-11eb-8f27-
d7fea0402bd2_2150573386.2021-01-25_123451",
        "newestSnapshotCreated": 1611578092,
        "lastTransferInfo": {
            "transferType": "update",
            "transferSize": {
                "size": 6240.0,
                "unit": "Byte"
            },
            "transferDuration": {
                "length": 4,
                "unit": "SECONDS"
            },
            "transferEnded": 1611578097,
            "transferError": null
        },
        "currentTransferInfo": {
            "transferType": null,
            "transferPriority": null,
            "transferError": null
        },
        "totalTransferTime": {
            "length": 6,
            "unit": "SECONDS"
        },
        "totalTransferSize": {
            "size": 23792.0,
            "unit": "Byte"
        },
        "volumeUsedSize": {
            "size": 1032192.0,
            "unit": "Byte"
        },
        "volumeCapacityTier": {
            "size": 0.0,
            "unit": "Byte"
        }
    }
]
```

Get the intercluster LIFs

You can use this workflow to retrieve the intercluster LIFs used in a cluster peering relationship.

> This workflow varies slightly depending on the cloud provider you are using.

1. Select the working environment

Based on the cloud provider, you need to perform a workflow to get the identifier for the working environment as shown in the following table.

Provider	Workflow
AWS	Perform the workflow Get working environments and choose the `publicId` values for the working environment query parameters.

2. Get the intercluster LIFs

HTTP method	Path
GET	/occm/api/replication/intercluster-lifs

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/replication/interclusterli
fs?workingEnvironmentId=<WORKING_ENV_ID>&peerWorkingEnvironmentId=<WORKING
_ENV_ID>' --header 'Content-Type: application/json' --header 'x-agent-id:
<AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

Query parameters:

- <WORKING_ENV_ID> `(workingEnvironmentId)`
- <WORKING_ENV_ID> `(peerWorkingEnvironmentId)`

Output

The JSON output example includes the list of LIFs.

JSON output example

```
{
    "interClusterLifs": [
        {
            "name": "intercluster",
            "address": "10.138.0.154",
            "netmaskLength": 32,
            "port": "e0a",
            "node": "zivgcp01we03-01",
            "status": "up",
            "isPeered": true
        }
    ],
    "peerInterClusterLifs": [
        {
            "name": "intercluster",
            "address": "10.138.0.147",
            "netmaskLength": 32,
            "port": "e0a",
            "node": "zivgcp01we02-01",
            "status": "up",
            "isPeered": true
        }
    ]
}
```

Get the replication schedules

You can use this workflow to retrieve the replication schedules used for a specific working environment.

This workflow varies slightly depending on the cloud provider you are using.

1. Select the working environment

Based on the cloud provider, you need to perform a workflow to get the identifier for the working environment as shown in the following table.

Provider	Workflow
AWS	Perform the workflow Get working environments and choose the `publicId` value for the working environment path parameter.

2. Get the schedules

HTTP method	Path
GET	/occm/api/replication/schedules/{workingEnvironmentId}

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/replication/schedules/<WOR
KING_ENV_ID>' --header 'Content-Type: application/json' --header 'x-agent-
id:<AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

Path parameter <WORKING_ENV_ID> `(workingEnvironment)`

Output

The JSON output example includes the list of replication schedules.

JSON output example

```
[
    {
        "name": "10min",
        "description": "@:00,:10,:20,:30,:40,:50",
        "cronJobSchedule": {
            "months": [],
            "days": [],
            "weekDays": [],
            "hours": [],
            "minutes": [
                0,
                10,
                20,
                30,
                40,
                50
            ]
        }
    },
    {
        "name": "5min",
        "description": "@:00,:05,:10,:15,:20,:25,:30,:35,:40,:45,:50,:55",
        "cronJobSchedule": {
            "months": [],
            "days": [],
            "weekDays": [],
            "hours": [],
            "minutes": [
                0,
                5,
                10,
                15,
                20,
                25,
                30,
                35,
                40,
                45,
                50,
                55
            ]
        }
    }
]
```

Copyright Information

Trademark Information