NetApp

Storage workflows

Cloud Manager Automation

NetApp
February 16, 2022

Table of Contents

Storage workflows

AWS workflows

Workflows for Amazon Web Services

There are several workflows you can use with the Amazon Web Services public cloud.

ℹ️   Review the Get started section before using any of the Cloud Manager REST API workflows.

Workflow categories

The AWS workflows are organized into the following functional categories:

- Working environments
- Aggregates
- Volumes
- Metadata
- Miscellaneous

See Understanding the workflow processes for more information on these categories.

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 AWS cloud provider account that you deployed the Connector in to your list of available accounts. Your AWS account needs to have the right permissions in order to create a Connector.

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

Working environments

Create a working environment with PAYGO

You can create a new Cloud Volumes ONTAP working environment using pay-as-you-go (PAYGO) subscription. You can add new volumes when creating the working environment.

> **Note the following when using PAYGO**:
>
> - A marketplace subscription is required.
> - A NetApp Support Site (NSS) key is recommended to register the system for support, but it's not required.
> - You can add more volumes after creating the working environment. You can add volumes using the NFS, CIFS or the iSCSI protocol.

Choose the correct workflow depending on the type of the Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Create working environment for single node

You can use this workflow to create an single node Cloud Volumes ONTAP working environment.

1. Select the region

Perform the workflow Get regions and choose the `code` value for the `region` parameter in step 9.

2. Select the workspace

Perform the workflow Get tenants and choose the `workspacePublicId` value for the `tenantId` parameter in step 9.

3. Select the permutations configuration

Perform the workflow Get permutations and choose the `ontapVersion` and `license: type` and `instanceType` values for the `vsaMetadata` parameter in step 9.

4. Select the VPC

Perform the workflow Get VPCs and do the following:

- Choose the `vpcId` value for the `vpcId` parameter in step 9.
- Choose the `cidrBlock` values for the `ips` value of the volume parameter in step 9.

5. Select the EBS volume configuration

Perform the workflow Get EBS volume types and choose the `size` and `supportedVolumeTypes` values for the `ebsVolumeSize` and `ebsVolumeType` parameters in step 9.

> (i) You need to choose one of the allowed values for the required `ebsVolumeType` parameter. We have used `gp2` value for this parameter in step 9.

6. Attach a marketplace subscription

Perform the workflow Attach SaaS subscription.

7. (Optional) Obtain an NSS key

An NSS key is optional when using PAYGO licensing. If needed, you can create a key or select an existing key, and include the NSS key in the `nssAccount` parameter in step 9.

- To create a new NSS key using the Cloud Manager web user interface, perform the task Generate NSS user ID and choose the NSS ID.

- To select an existing NSS key, perform the workflow Get NSS keys and choose the `id` of the required NSS user.

8. (Optional) Create a new volume

You can optionally add a new volume while creating a working environment.

- Choose the `name` and `size` values for the corresponding `name` and `size:size` input parameters in step 9. The `size:unit` must be one of the following: TB, GB, MB, KB, or Byte.

- Choose the `policyType`, `ips`, and the `nfsVersion` parameters in the `exportPolicyInfo` input parameter in step 9 if you choose to create a volume using the NFS protocol. The `ips` parameter signifies the client IP address (could be multiple addresses) that can access the volume over the network. The `nfsVersion` parameter signifies the version of the NFS protocol that a client will use for data transmission over a network.

> (i) If you choose to create a volume using CIFS protocol, you will need to set the `shareInfo` parameter. If you choose to create a volume using iSCSI protocol, you will need to set the `iscsiInfo` parameter accordingly.

9. Create the working environment

HTTP method	Path
POST	/occm/api/vsa/working-environments

curl example

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

Input

The JSON input example includes the minimum list of parameters.

> (i) This request uses PAYGO subscription as indicated in the `licenseType` parameter.

JSON input example

```json
{
    "name": "ziv01we02",
    "svmPassword": "user_password",
    "vpcId": "vpc-b16c90d4",
    "region": "us-east-1",
    "tenantId": "tenantIDgoeshere",
    "subnetId": "subnet-f4da95ac",
    "dataEncryptionType": "AWS",
    "vsaMetadata": {
        "ontapVersion": "ONTAP-9.9.0.T1",
        "licenseType": "cot-explore-paygo",
        "instanceType": "m5.xlarge"
        },
    "ebsVolumeSize": {
        "size": 100,
        "unit": "GB"
    },
    "ebsVolumeType": "gp2",
    "volume": {
      "name": "ziv02vol01",
      "size": {
        "size": 200,
        "unit": "GB"
      },
      "exportPolicyInfo": {
        "policyType": "custom",
        "ips": [
          "172.31.0.0/16"
        ],
        "nfsVersion": [
          "nfs3",
          "nfs4"
        ]
      },
      "snapshotPolicyName": "default",
      "enableThinProvisioning": true,
      "enableCompression": true,
      "enableDeduplication": true
    },
    "writingSpeedState": "NORMAL"
}
```

Output

The JSON output example includes an example of the `VsaWorkingEnvironmentRresponse`.

JSON output example

```
{
    "publicId": "VsaWorkingEnvironment-0NWsb1aX",
    "name": "ziv01we02",
    "tenantId": "tenantIDgoeshere",
    "svmName": "svm_ziv01we02",
    "creatorUserEmail": "user_email",
    "status": null,
    "awsProperties": null,
    "reservedSize": null,
    "encryptionProperties": null,
    "clusterProperties": null,
    "ontapClusterProperties": null,
    "actionsRequired": null,
    "interClusterLifs": null,
    "cronJobSchedules": null,
    "snapshotPolicies": null,
    "svms": null,
    "activeActions": null,
    "replicationProperties": null,
    "schedules": null,
    "cloudProviderName": "Amazon",
    "isHA": false,
    "workingEnvironmentType": "VSA",
    "supportRegistrationProperties": null,
    "supportRegistrationInformation": null,
    "haProperties": null,
    "capacityFeatures": null,
    "cloudSyncProperties": null,
    "supportedFeatures": null,
    "k8sProperties": null,
    "fpolicyProperties": null,
    "saasProperties": null,
    "cbsProperties": null,
    "complianceProperties": null,
    "monitoringProperties": null
}
```

Create working environment for high availability pair

You can use this workflow to create an HA Cloud Volumes ONTAP working environment.

1. Select the region

Perform the workflow Get regions and choose the `code` value for the `region` parameter in step 11.

2. Select the cloud provider account

Perform the workflow Get cloud provider accounts and choose the `publicId` value of the required account for the `cloudProviderAccount` parameter.

3. Select the workspace

Perform the workflow Get tenants and choose the `workspacePublicId` value for the `tenantId` parameter in step 12.

4. Select the permutations configuration

Perform the workflow Get permutations and choose the `ontapVersion` and `license: type` and `instanceType` values for the `vsaMetadata` parameter in step 12.

5. Select the packages configuration

Perform the Get Packages and choose the `packageName`, `instanceTenancy` and `writingSpeedState` values for the corresponding parameters in step 12.

6. Select the VPC

Perform the workflow Get VPCs and do the following:

- Choose the `vpcId` value for the `vpcId` parameter in step 12.
- Choose three subnets and choose the `subnetId` value for the `haParams:mediatorSubnetId`, `haParams:node1SubnetId, haParams:node2SubnetId` in step 12.
- Select the IPs for the `clusterFloatingIP, dataFloatingIP, dataFloatingIP2` values for the corresponding parameters in step 12.

7. Select route table

Perform the workflow Get route tables and choose the `id` value of the required route table for `haParams:routeTableIds` parameter is step 12.

8. Select the EBS volume configuration

Perform the workflow Get EBS volume types and choose the `size` and `supportedVolumeTypes` values for the `ebsVolumeSize` and `ebsVolumeType` parameters in step 12.

> (i) You need to choose one of the allowed values for the required `ebsVolumeType` parameter. We have used `gp2` value for this parameter in step 12.

9. Attach a marketplace subscription

Perform the workflow Attach SaaS subscription.

10. (Optional) Obtain an NSS key

An NSS key is optional when using PAYGO licensing. If needed, you can create a key or select an existing key, and include the NSS key in the `nssAccount` parameter in step 12.

- To create a new NSS key using the Cloud Manager web user interface, perform the task Generate NSS user ID and choose the NSS ID.

- To select an existing NSS key, perform the workflow Get NSS keys and choose the `id` of the required NSS user.

11. (Optional) Create a new volume

You can optionally add a new volume while creating a working environment.

- Choose the `name` and `size` values for the corresponding `name` and `size:size` input parameters in step 12. The `size:unit` must be one of the following: TB, GB, MB, KB, or Byte.

- Choose the `policyType`, `ips`, and the `nfsVersion` parameters in the `exportPolicyInfo` input parameter in step 12 if you choose to create a volume using the NFS protocol. The `ips` parameter signifies the client IP address (could be multiple addresses) that can access the volume over the network. The `nfsVersion` parameter signifies the version of the NFS protocol that a client will use for data transmission over a network.

> (i) If you choose to create a volume using CIFS protocol, you will need to set the `shareInfo` parameter. If you choose to create a volume using iSCSI protocol, you will need to set the `iscsiInfo` parameter accordingly.

12. Create the working environment

HTTP method	Path
POST	/occm/api/aws/ha/working-environments

curl example

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

Input

The JSON input example includes the minimum list of parameters.

> (i) This request uses PAYGO licensing as indicated in the `licenseType` parameter.

JSON input example

```
{
  "name": "ziv04we02ha",
  "svmPassword": "password",
  "vpcId": "vpc-b16c90d4",
  "region": "us-east-1",
  "tenantId": "tenantIDgoeshere",
```

```json
  "ebsVolumeSize": {
    "size": 100,
    "unit": "GB"
  },
  "ebsVolumeType": "gp2",
  "vsaMetadata": {
    "ontapVersion": "ONTAP-9.9.0X6.T1.ha",
    "licenseType": "ha-cot-explore-paygo",
    "instanceType": "m5.xlarge"
  },
  "dataEncryptionType": "AWS",
  "ontapEncryptionParameters": null,
  "haParams": {
    "node1SubnetId": "subnet-f4da95ac",
    "node2SubnetId": "subnet-b4387a9e",
    "mediatorSubnetId": "subnet-76e6d400",
    "clusterFloatingIP": "4.4.4.4",
    "dataFloatingIP": "5.5.5.5",
    "dataFloatingIP2": "6.6.6.6",
    "mediatorKeyPairName": "Developers_Virginia",
    "routeTableIds": [
      "rtb-02a45467"
    ],
    "failoverMode": "FloatingIP",
    "mediatorAssignPublicIP": true
  },
  "volume": {
    "name": "ziv02vol01",
    "size": {
      "size": 200,
      "unit": "GB"
    },
    "exportPolicyInfo": {
      "policyType": "custom",
      "ips": [
        "172.31.0.0/16"
      ],
      "nfsVersion": [
        "nfs3",
        "nfs4"
      ]
    },
    "snapshotPolicyName": "default",
    "enableThinProvisioning": true,
    "enableCompression": true,
    "enableDeduplication": true
```

```
    },
    "optimizedNetworkUtilization": false,
    "instanceTenancy": "default",
    "packageName": "aws_ha_poc",
    "cloudProviderAccount": "InstanceProfile",
    "backupVolumesToCbs": false,
    "enableMonitoring": "false",
    "writingSpeedState": "NORMAL"
}
```

Output

The JSON output example includes an example of the HA working environment details.

JSON output example

```
{
    "publicId": "VsaWorkingEnvironment-sQ9AELDS",
    "name": "ziv04we02ha",
    "tenantId": "tenantIDshownhere",
    "svmName": "svm_ziv04we02ha",
    "creatorUserEmail": "user_email",
    "status": null,
    "awsProperties": null,
    "reservedSize": null,
    "encryptionProperties": null,
    "clusterProperties": null,
    "ontapClusterProperties": null,
    "actionsRequired": null,
    "interClusterLifs": null,
    "cronJobSchedules": null,
    "snapshotPolicies": null,
    "svms": null,
    "activeActions": null,
    "replicationProperties": null,
    "schedules": null,
    "cloudProviderName": "Amazon",
    "isHA": true,
    "workingEnvironmentType": "VSA",
    "supportRegistrationProperties": null,
    "supportRegistrationInformation": null,
    "haProperties": null,
    "capacityFeatures": null,
    "cloudSyncProperties": null,
    "supportedFeatures": null,
    "k8sProperties": null,
    "fpolicyProperties": null,
    "saasProperties": null,
    "cbsProperties": null,
    "complianceProperties": null,
    "monitoringProperties": null
}
```

Create a working environment with BYOL licensing

You can use this workflow to create a new Cloud Volumes ONTAP working environment using bring-your-own-license (BYOL) licensing.

> **Note the following when using BYOL licensing**:
>
> - A marketplace subscription is not required.
> - A NetApp Support Site (NSS) key is required to register the system for support.
> - You can add a volume after creating the working environment.

To optionally create a new volume with the working environment, you must modify the JSON input provided on the REST API call. See Create a working environment with PAYGO licensing for an example.

Choose the correct workflow depending on the type of the Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Create working environment for single node

You can use this workflow to create a single node Cloud Volumes ONTAP working environment.

1. Select the region

Perform the workflow Get regions and choose the `code` value for the `region` parameter in step 7.

2. Select the workspace

Perform the workflow Get tenants and choose the `workspacePublicId` value for the `tenantId` parameter in step 7.

3. Select the permutations configuration

Perform the workflow Get permutations and choose the `ontapVersion` and `license: type` and `instanceType` values for the `vsaMetadata` parameter in step 7.

You will also need to include the `platformSerialNumber` value in `vsaMetadata` parameter in the REST API call.

4. Select the VPC

Perform the workflow Get VPCs and choose the `vpcId` value for the `vpcId` parameter in step 7.

5. Select the EBS volume configuration

Perform the workflow Get EBS volume types and choose the `size` and `supportedVolumeTypes` values for the `ebsVolumeSize` and `ebsVolumeType` parameters in step 7.

> (i) You need to choose one of the allowed values for the required `ebsVolumeType` parameter. We have used `gp2` value for this parameter in step 7.

6. Obtain the required NSS key

An NSS key is required when using BYOL licensing. You can create a key or select an existing key, and include the NSS key in the `nssAccount` parameter in step 7.

- To create a new NSS key using the Cloud Manager web user interface, perform the task Generate NSS user ID and choose the NSS ID.

- To select an existing NSS key, perform the workflow Get NSS keys and choose the `id` of the required NSS user.

7. Create the working environment

HTTP method	Path
POST	/occm/api/vsa/working-environments

curl example

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

Input

The JSON input example includes the minimum list of parameters. This request uses BYOL licensing as indicated in the `licenseType` parameter. The `platformSerialNumber` is required.

JSON input example

```
{
    "name": "ziv02we03",
    "svmPassword": "password",
    "vpcId": "vpc-b16c90d4",
    "region": "us-east-1",
    "tenantId": "workspace-cxxx123",
    "subnetId": "subnet-f4da95ac",
    "dataEncryptionType": "AWS",
    "nssAccount": "xxx332ce3-xxxx-4000-xx00-000a0601c682",
    "vsaMetadata": {
        "ontapVersion": "ONTAP-9.9.0.T1",
        "licenseType": "cot-premium-byol",
        "instanceType": "m5.xlarge",
        "platformSerialNumber": "00100100000000000020"
        },
        "ebsVolumeSize": {
            "size": 100,
            "unit": "GB"
            },
            "ebsVolumeType": "gp2"
}
```

Output

The JSON output example includes an example of the `VsaWorkingEnvironmentRresponse` response.

JSON output example

```
{
    "publicId": "VsaWorkingEnvironment-wL2MaBJs",
    "name": "username",
    "tenantId": "tenantIDgoeshere",
    "svmName": "svm_ziv02we03",
    "creatorUserEmail": "user_email",
    "status": null,
    "awsProperties": null,
    "reservedSize": null,
    "encryptionProperties": null,
    "clusterProperties": null,
    "ontapClusterProperties": null,
    "actionsRequired": null,
    "interClusterLifs": null,
    "cronJobSchedules": null,
    "snapshotPolicies": null,
    "svms": null,
    "activeActions": null,
    "replicationProperties": null,
    "schedules": null,
    "cloudProviderName": "Amazon",
    "isHA": false,
    "workingEnvironmentType": "VSA",
    "supportRegistrationProperties": null,
    "supportRegistrationInformation": null,
    "haProperties": null,
    "capacityFeatures": null,
    "cloudSyncProperties": null,
    "supportedFeatures": null,
    "k8sProperties": null,
    "fpolicyProperties": null,
    "saasProperties": null,
    "cbsProperties": null,
    "complianceProperties": null,
    "monitoringProperties": null
}
```

Create working environment for high availability pair

You can use this workflow to create an HA Cloud Volumes ONTAP working environment.

1. Select the region

Perform the workflow Get regions and choose the `code` value for the `region` parameter in step 11.

2. Select the cloud provider account

Perform the workflow Get cloud provider accounts and choose the `publicId` value of the required account for the `cloudProviderAccount` parameter.

3. Select the workspace

Perform the workflow Get tenants and choose the `workspacePublicId` value for the `tenantId` parameter in step 11.

4. Select the permutations configuration

Perform the workflow Get permutations and choose the `ontapVersion` and `license: type` and `instanceType` values for the `vsaMetadata` parameter in step 11.

5. Select the packages configuration

Perform the Get Packages and choose the `packageName`, `instanceTenancy` and `writingSpeedState` values for the corresponding parameters in step 11.

6. Select the VPC

Perform the workflow Get VPCs and do the following:

- Choose the `vpcId` value for the `vpcId` parameter in step 11.
- Choose three subnets and choose the `subnetId` value for the `haParams:mediatorSubnetId`, `haParams:node1SubnetId, haParams:node2SubnetId`.
- Attach the licenses serial number to `platformSerialNumberNode1` and `platformSerialNumberNode2` parameters.
- Select the IPs for the `clusterFloatingIP, dataFloatingIP, dataFloatingIP2` values for the corresponding parameters in step 11.

7. Select the route table

Perform the workflow Get route tables and choose the `id` value of the required route table for `haParams:routeTableIds` parameter in step 11.

8. Select the EBS volume configuration

Perform the workflow Get EBS volume types and choose the `size` and `supportedVolumeTypes` values for the `ebsVolumeSize` and `ebsVolumeType` parameters in step 11.

> (i) You need to choose one of the allowed values for the required `ebsVolumeType` parameter. We have used `gp2` value for this parameter in step 11.

9. Get key pairs

Perform the Get key pairs workflow and select the required key for `haParam: mediatorKeyPairName` parameter in step 11.

10. Obtain the required NSS key

An NSS key is required when using BYOL licensing. You can create a key or select an existing key, and include the NSS key in the `nssAccount` parameter.

- To create a new NSS key using the Cloud Manager web user interface, perform the task Generate NSS user ID and choose the NSS ID.

- To select an existing NSS key, perform the workflow Get NSS keys and choose the `id` of the required NSS user.

11. Create the working environment

HTTP method	Path
POST	/occm/api/aws/ha/working-environments

curl example

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

Input

The JSON input example includes the minimum list of parameters. This request uses BYOL licensing as indicated in the `licenseType` parameter. The `platformSerialNumberNode1` and `platformSerialNumberNode2` parameters are required.

JSON input example

```json
{
  "name": "ziv04we02ha",
  "svmPassword": "password",
  "vpcId": "vpc-b16c90d4",
  "region": "us-east-1",
  "tenantId": "tenantIDgoeshere",
  "ebsVolumeSize": {
    "size": 100,
    "unit": "GB"
  },
  "ebsVolumeType": "gp2",
  "vsaMetadata": {
    "ontapVersion": "ONTAP-9.9.0.T1.ha",
    "licenseType": "ha-cot-premium-byol",
    "instanceType": "m5.xlarge"
  },
  "dataEncryptionType": "AWS",
  "ontapEncryptionParameters": null,
  "haParams": {
    "node1SubnetId": "subnet-f4da95ac",
    "node2SubnetId": "subnet-b4387a9e",
    "mediatorSubnetId": "subnet-76e6d400",
    "clusterFloatingIP": "4.4.4.4",
    "dataFloatingIP": "5.5.5.5",
    "dataFloatingIP2": "6.6.6.6",
    "platformSerialNumberNode1": "90120140000000000023",
    "platformSerialNumberNode2": "90120140000000000024",
    "mediatorKeyPairName": "Developers_Virginia",
    "routeTableIds": [
      "rtb-02a45467"
    ],
    "failoverMode": "FloatingIP",
    "mediatorAssignPublicIP": true
  },
  "nssAccount": "54ab5c0d-xx23-xxxd-bcef-6eeda79e747d",
  "optimizedNetworkUtilization": false,
  "instanceTenancy": "default",
  "packageName": "aws_ha_poc",
  "cloudProviderAccount": "InstanceProfile",
  "backupVolumesToCbs": false,
  "enableMonitoring": "false",
  "writingSpeedState": "NORMAL"
}
```

Output

The JSON output example includes an example of the HA working environment details.

JSON output example

```json
{
    "publicId": "VsaWorkingEnvironment-XxCZibxz",
    "name": "ziv04we02ha",
    "tenantId": "tenantIDshownhere",
    "svmName": "svm_ziv04we02ha",
    "creatorUserEmail": "user_email",
    "status": null,
    "awsProperties": null,
    "reservedSize": null,
    "encryptionProperties": null,
    "clusterProperties": null,
    "ontapClusterProperties": null,
    "actionsRequired": null,
    "interClusterLifs": null,
    "cronJobSchedules": null,
    "snapshotPolicies": null,
    "svms": null,
    "activeActions": null,
    "replicationProperties": null,
    "schedules": null,
    "cloudProviderName": "Amazon",
    "isHA": true,
    "workingEnvironmentType": "VSA",
    "supportRegistrationProperties": null,
    "supportRegistrationInformation": null,
    "haProperties": null,
    "capacityFeatures": null,
    "cloudSyncProperties": null,
    "supportedFeatures": null,
    "k8sProperties": null,
    "fpolicyProperties": null,
    "saasProperties": null,
    "cbsProperties": null,
    "complianceProperties": null,
    "monitoringProperties": null
}
```

Get working environments

You can retrieve the public identifier, working environment identifier and the storage virtual machine name for Cloud Volumes ONTAP working environments (visible to

currently logged in user) which would be used in other workflows.

Choose the correct workflow depending on the type of the Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Get working environments for single node

You can use this workflow to retrieve the working environments' details for a single node system.

1. Get the working environments

HTTP method	Path
GET	occm/api/vsa/working-environments

curl example

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

Input

   Optional query parameters:

- fields
- tenantId

Output

   The JSON output example includes an array of VSA working environments and their details.

JSON output example

```
[
    {
        "publicId": "VsaWorkingEnvironment-79VKenHW",
        "name": "ziv01we02",
        "tenantId": "tenantIDshownhere",
        "svmName": "svm_ziv01we02",
        "creatorUserEmail": "user_email",
        "status": null,
        "awsProperties": null,
        "reservedSize": null,
        "encryptionProperties": null,
        "clusterProperties": null,
        "ontapClusterProperties": null,
        "actionsRequired": null,
```

```json
        "interClusterLifs": null,
        "cronJobSchedules": null,
        "snapshotPolicies": null,
        "svms": null,
        "activeActions": null,
        "replicationProperties": null,
        "schedules": null,
        "cloudProviderName": "Amazon",
        "isHA": false,
        "workingEnvironmentType": "VSA",
        "supportRegistrationProperties": null,
        "supportRegistrationInformation": [],
        "haProperties": null,
        "capacityFeatures": null,
        "cloudSyncProperties": null,
        "supportedFeatures": null,
        "k8sProperties": null,
        "fpolicyProperties": null,
        "saasProperties": null,
        "cbsProperties": null,
        "complianceProperties": null,
        "monitoringProperties": null
    },
    {
        "publicId": "VsaWorkingEnvironment-61kN4p5P",
        "name": "ziv01we03",
        "tenantId": "tenantIDshownhere",
        "svmName": "svm_ziv01we03",
        "creatorUserEmail": "user_email",
        "status": null,
        "awsProperties": null,
        "reservedSize": null,
        "encryptionProperties": null,
        "clusterProperties": null,
        "ontapClusterProperties": null,
        "actionsRequired": null,
        "interClusterLifs": null,
        "cronJobSchedules": null,
        "snapshotPolicies": null,
        "svms": null,
        "activeActions": null,
        "replicationProperties": null,
        "schedules": null,
        "cloudProviderName": "Amazon",
        "isHA": false,
        "workingEnvironmentType": "VSA",
```

```json
        "supportRegistrationProperties": null,
        "supportRegistrationInformation": [],
        "haProperties": null,
        "capacityFeatures": null,
        "cloudSyncProperties": null,
        "supportedFeatures": null,
        "k8sProperties": null,
        "fpolicyProperties": null,
        "saasProperties": null,
        "cbsProperties": null,
        "complianceProperties": null,
        "monitoringProperties": null
    },
    {

        "publicId": "VsaWorkingEnvironment-E9WanX81",
        "name": "ziv01we04",
        "tenantId": "tenantIDshownhere",
        "svmName": "svm_ziv01we04",
        "creatorUserEmail": "user_email",
        "status": null,
        "awsProperties": null,
        "reservedSize": null,
        "encryptionProperties": null,
        "clusterProperties": null,
        "ontapClusterProperties": null,
        "actionsRequired": null,
        "interClusterLifs": null,
        "cronJobSchedules": null,
        "snapshotPolicies": null,
        "svms": null,
        "activeActions": null,
        "replicationProperties": null,
        "schedules": null,
        "cloudProviderName": "Amazon",
        "isHA": false,
        "workingEnvironmentType": "VSA",
        "supportRegistrationProperties": null,
        "supportRegistrationInformation": [],
        "haProperties": null,
        "capacityFeatures": null,
        "cloudSyncProperties": null,
        "supportedFeatures": null,
        "k8sProperties": null,
        "fpolicyProperties": null,
        "saasProperties": null,
        "cbsProperties": null,
```

```
        "complianceProperties": null,
        "monitoringProperties": null
    }
]
```

Get working environment for high availability pair

You can use this workflow to get the working environment details for an HA pair.

1. Select the working environment to use

Perform the workflow Get working environments for single node and choose the `publicId` of the required working environment for the `workingEnvironmentId` path parameter.

2. Get the working environment

HTTP method	Path
GET	occm/api/aws/ha/working-environments/<WORKING_ENV_ID>

curl example

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

Input
   Path parameter:

<WORKING_ENV_ID> `workingEnvironmentId`

Optional query parameter:

`fields`

Output
   The JSON output example includes an HA Cloud Volumes ONTAP working environment.

JSON output example

```
[
{
    "publicId": "VsaWorkingEnvironment-N6BPfglr",
    "name": "ziv04we01ha",
    "tenantId": "tenantIDshownhere",
    "svmName": "svm_ziv04we01ha",
    "creatorUserEmail": "user_email",
    "status": null,
    "awsProperties": null,
    "reservedSize": null,
    "encryptionProperties": null,
    "clusterProperties": null,
    "ontapClusterProperties": null,
    "actionsRequired": null,
    "interClusterLifs": null,
    "cronJobSchedules": null,
    "snapshotPolicies": null,
    "svms": null,
    "activeActions": null,
    "replicationProperties": null,
    "schedules": null,
    "cloudProviderName": "Amazon",
    "isHA": true,
    "workingEnvironmentType": "VSA",
    "supportRegistrationProperties": null,
    "supportRegistrationInformation": [],
    "haProperties": null,
    "capacityFeatures": null,
    "cloudSyncProperties": null,
    "supportedFeatures": null,
    "k8sProperties": null,
    "fpolicyProperties": null,
    "saasProperties": null,
    "cbsProperties": null,
    "complianceProperties": null,
    "monitoringProperties": null
}
```

Delete a working environment

You can delete an existing Cloud Volumes ONTAP working environment.

Choose the correct workflow depending on the type of the Cloud Volumes ONTAP deployment:

- Single Node

- [HA pair](#)

Delete working environment for single node

You can use this workflow to delete a working environment for a single node system.

1. Select the working environment to use

Perform the workflow [Get working environments](#) and choose the `publicId` value of the working environment for the `workingEnvironmentId` path parameter.

2. Delete the working environment

HTTP method	Path
DELETE	/occm/api/vsa/working-environments/{workingEnvironmentId}

curl example

```
curl --location --request DELETE
'https://cloudmanager.cloud.netapp.com/occm/api/vsa/working-
environments/<WORKING_ENV_ID>' --header 'Content-Type: application/json'
--header 'x-agent-id: <AGENT_ID>' --header 'Authorization: Bearer
<ACCESS_TOKEN>'
```

Input

   Path parameter <WORKING_ENV_ID> `workingEnvironmentId`

Optional query parameters:

- `localDelete`

   If `true` the Cloud Volumes ONTAP instance in the cloud is not terminated, but Cloud Manager no longer manages it (default is `false`).

- `forceDelete`

   If `true` the working environment is deleted even if it is part of one or more SnapMirror relationships (default is `false`).

   Output
      None

Delete working environment for high availability pair

You can use this workflow to delete a working environment for an HA pair.

1. Select the working environment to use

Perform the workflow [Get working environments](#) and choose the `publicId` value of the working environment for the `workingEnvironmentId` path parameter.

2. Delete the working environment

HTTP method	Path
DELETE	/occm/api/aws/ha/working-environments/{workingEnvironmentId}

curl example

```
curl --location --request DELETE
'https://cloudmanager.cloud.netapp.com/occm/api/aws/ha/working-
environments/<WORKING_ENV_ID>' --header 'Content-Type: application/json'
--header 'x-agent-id: <AGENT_ID>' --header 'Authorization: Bearer
<ACCESS_TOKEN>'
```

Input

Path parameter <WORKING_ENV_ID> `workingEnvironmentId`

Optional query parameters:

- `localDelete`

  If `true` the Cloud Volumes ONTAP instance in the cloud is not terminated, but Cloud Manager no longer manages it (default is `false`).

- `forceDelete`

  If `true` the working environment is deleted even if it is part of one or more SnapMirror relationships (default is `false`).

  Output
    None

Create CIFS server configuration

If you want to create CIFS volumes on your Cloud Volumes ONTAP system, you first need to configure the CIFS server. You can choose to set up the CIFS server in a workgroup or in an Active Directory domain. Review the NetApp docs for more information.

Choose the correct workflow depending on the type of the Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Create CIFS server configuration for single node

You can use this workflow to create a CIFS server configuration for a single node system.

Choose the workflow that is specific to your goal:

- Set up a CIFS server in a workgroup
- Set up a CIFS server in an Active Directory domain

Set up a CIFS server in a workgroup

You can configure a CIFS server in a workgroup when the Microsoft Active Directory domain infrastructure is not available.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value of the working environment for the `workingEnvironmentId` path parameter.

2. Create the CIFS configuration

Create the CIFS server configuration.

HTTP method	Path
POST	/occm/api/vsa/working-environments/{workingEnvironmentId}/cifs-workgroup

curl example

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

Input
- Path parameter <WORKING_ENV_ID> `workingEnvironmentId`

JSON input example

```
{
  "serverName": "SMB_SERVER02",
  "workgroupName": "workgroup02",
  "svmName": "svm_ziv01we01"
}
```

Output
None.

Set up a CIFS server in an Active Directory domain

You can create a CIFS server on the SVM and specify the Active Directory (AD) domain to which it belongs.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value of the working environment for the `workingEnvironmentId` path parameter.

2. Determine the Active Directory configuration

You need the following configuration parameters for an Active Directory server.

Input parameter	Description
dnsDomain	Use the Active Directory domain as the DNS name.
ipAddresses	Define the primary DNS IP address and optionally add a secondary IP address.
netBIOS	Use the CIFS server NetBIOS name.
organizationalUnit	Include the organizational unit as appropriate.
activeDirectoryDomain	Set the Active Directory domain to join.
activeDirectoryUsername	A username with authorization to join the domain.
activeDirectoryPassword	The password for the authorized username.

3. Create the CIFS configuration

Create the CIFS server configuration.

HTTP method	Path
POST	/occm/api/vsa/working-environments/{workingEnvironmentId}/cifs

curl example

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

Input
- Path parameter <WORKING_ENV_ID> `workingEnvironmentId`

JSON input example

```
{
  "dnsDomain": "zivh.netapp.com",
  "ipAddresses": [
    "172.31.5.241"
  ],
  "netBIOS": "zivaws02we03",
  "organizationalUnit": "CN=Computers",
  "activeDirectoryDomain": "zivh.netapp.com",
  "activeDirectoryUsername": "administrator",
  "activeDirectoryPassword": "password"
}
```

Output

None.

Create CIFS server configuration for high availability pair

You can use this workflow to create a CIFS server configuration for an HA working environment.

Choose the workflow that is specific to your goal:

- Set up a CIFS server in a workgroup
- Set up a CIFS server in an Active Directory domain

Set up a CIFS server in a workgroup

You can configure a CIFS server in a workgroup when the Microsoft Active Directory domain infrastructure is not available.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value of the working environment for the `workingEnvironmentId` path parameter.

2. Create the CIFS configuration

Create the CIFS server configuration.

HTTP method	Path
POST	/occm/api/aws/ha/working-environments/{workingEnvironmentId}/cifs-workgroup

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/aws/ha/working-
environments/<WORKING_ENV_ID>/cifs-workgroup' --header 'Content-Type:
application/json' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>' --d @JSONinput
```

Input

- Path parameter <WORKING_ENV_ID> `workingEnvironmentId`

JSON input example

```
{
  "serverName": "SMB_SERVER02",
  "workgroupName": "workgroup02",
  "svmName": "svm_ziv01we01"
}
```

Output

None.

Set up a CIFS server in an Active Directory domain

You can create a CIFS server on the SVM and specify the Active Directory (AD) domain to which it belongs.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value of the working environment for the `workingEnvironmentId` path parameter.

2. Determine the Active Directory configuration

You need the following configuration parameters for an Active Directory server.

Input parameter	Description
dnsDomain	Use the Active Directory domain as the DNS name.
ipAddresses	Define the primary DNS IP address and optionally add a secondary IP address.
netBIOS	Use the CIFS server NetBIOS name.
organizationalUnit	Include the organizational unit as appropriate.
activeDirectoryDomain	Set the Active Directory domain to join.
activeDirectoryUsername	A username with authorization to join the domain.
activeDirectoryPassword	The password for the authorized username.

3. Create the CIFS configuration

Create the CIFS server configuration.

HTTP method	Path
POST	/occm/api/aws/ha/working-environments/{workingEnvironmentId}/cifs

curl example

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

Input

- Path parameter <WORKING_ENV_ID> `workingEnvironmentId`

JSON input example

```json
{
  "dnsDomain": "zivh.netapp.com",
  "ipAddresses": [
    "172.31.5.241"
  ],
  "netBIOS": "zivaws02we03",
  "organizationalUnit": "CN=Computers",
  "activeDirectoryDomain": "zivh.netapp.com",
  "activeDirectoryUsername": "administrator",
  "activeDirectoryPassword": "password"
}
```

Output

None.

Get CIFS server configurations

You can use this workflow to retrieve the CIFS server configurations for an existing Cloud Volumes ONTAP working environment.

Choose the correct workflow depending on the type of the Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Get CIFS server configuration for single node

You can use this workflow to retrieve a CIFS server configuration for a single node system.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value of the working environment for the `workingEnvironmentId` path parameter.

2. Get the CIFS configurations

HTTP method	Path
GET	/occm/api/vsa/working-environments/{workingEnvironmentId}/cifs

curl example

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

Input

- Path parameter <WORKING_ENV_ID> `workingEnvironmentId`

- Optional query parameter `svm`

Output

The JSON output example includes the CIFS configurations for an existing Cloud Volumes ONTAP working environment.

JSON output example

```
[
    {
        "dnsDomain": "zivh.netapp.com",
        "activeDirectoryDomain": "zivh.netapp.com",
        "ipAddresses": [
            "172.31.5.241"
        ],
        "netBIOS": "zivaws02we01",
        "organizationalUnit": "CN=Computers",
        "authenticationType": "domain"
    }
]
```

Get CIFS server configuration for high availability pair

You can use this workflow to retrieve a CIFS server configuration for an HA working environment.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value of the working environment for the `workingEnvironmentId` path parameter.

2. Get the CIFS configurations

HTTP method	Path
GET	/occm/api/aws/ha/working-environments/{workingEnvironmentId}/cifs

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/aws/ha/working-
environments/<WORKING_ENV_ID>/cifs' --header 'Content-Type:
application/json' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

- Path parameter <WORKING_ENV_ID> `workingEnvironmentId`

- Optional query parameter `svm`

Output

The JSON output example includes the CIFS configurations for an existing Cloud Volumes ONTAP working environment.

JSON output example

```
[
    {
        "dnsDomain": "zivh.netapp.com",
        "activeDirectoryDomain": "zivh.netapp.com",
        "ipAddresses": [
            "172.31.5.241"
        ],
        "netBIOS": "zivaws02we01",
        "organizationalUnit": "CN=Computers",
        "authenticationType": "domain"
    }
]
```

Delete CIFS server configuration

You can use this workflow to delete a CIFS server configuration for an existing Cloud Volumes ONTAP working environment.

Choose the correct workflow depending on the type of the Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Delete CIFS server configuration for single node

You can use this workflow to delete a CIFS server configuration for a single node system.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value of the working environment for the `workingEnvironmentId` path parameter.

2. Delete the CIFS configurations

HTTP method	Path
POST	/occm/api/vsa/working-environments/{workingEnvironmentId}/delete-cifs

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/vsa/working-
environments/<WORKING_ENV_ID>/delete-cifs' --header 'Content-Type:
application/json' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

- Path parameter <WORKING_ENV_ID> `workingEnvironmentId`
- Optional JSON body

```
{
  "activeDirectoryUsername": "string",
  "activeDirectoryPassword": "string",
  "svmName": "string"
}
```

Output

None.

Delete CIFS server configuration for high availability pair

You can use this workflow to delete a CIFS server configuration for an HA working environment.

1. Select the working environment

Perform the workflow [Get working environments](#) and choose the `publicId` value of the working environment for the `workingEnvironmentId` path parameter.

2. Delete the CIFS configurations

HTTP method	Path
POST	/occm/api/aws/ha/working-environments/{workingEnvironmentId}/delete-cifs

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/aws/ha/working-
environments/<WORKING_ENV_ID>/delete-cifs' --header 'Content-Type:
application/json' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

- Path parameter <WORKING_ENV_ID> `workingEnvironmentId`
- Optional JSON body

```
{
  "activeDirectoryUsername": "string",
  "activeDirectoryPassword": "string",
  "svmName": "string"
}
```

Output

None.

Aggregates

Get aggregates

You can retrieve a list of available disk aggregates.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- [Single Node](#)
- [HA pair](#)

Get aggregates for single node

You can use this workflow to retrieve the aggregates for a single node working environment.

1. Select the working environment to use

Perform the workflow Get working environments and choose the `publicId` value of the working environment for the `workingEnvironmentId` path parameter.

2. Get the list of aggregates

HTTP method	Path
GET	/occm/api/vsa/aggregates/{workingEnvironmentId}

curl example

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

Input

Path parameter <WORKING_ENV_ID> (`workingEnvironmentId`)

Output

An array of aggregates for the indicated working environment is returned as shown in the JSON output example.

JSON output example

```
[
    {
        "name": "aggr1",
        "availableCapacity": {
            "size": 87.55,
            "unit": "GB"
        },
        "totalCapacity": {
            "size": 88.57,
            "unit": "GB"
        },
        "usedCapacity": {
            "size": 1.02,
            "unit": "GB"
        },
        "volumes": [
            {
                "name": "svm_ziv01we01_root",
```

```
                "totalSize": {
                    "size": 1.0,
                    "unit": "GB"
                },
                "usedSize": {
                    "size": 0.00115203857421875,
                    "unit": "GB"
                },
                "thinProvisioned": false,
                "isClone": false,
                "rootVolume": true
            }
        ],
        "providerVolumes": [
            {
                "id": "vol-066fea889cbc6a65c",
                "name": "vol-066fea889cbc6a65c",
                "size": {
                    "size": 100.0,
                    "unit": "GB"
                },
                "state": "in-use",
                "device": "/dev/xvdg",
                "instanceId": "i-0fa9a2879e67a8829",
                "diskType": "gp2",
                "encrypted": true,
                "iops": null
            }
        ],
        "disks": [
            {
                "name": "NET-1.3",
                "position": "data",
                "ownerNode": "ziv01we01-01",
                "device": "xvdg vol066fea889cbc6a65c",
                "vmDiskProperties": null
            }
        ],
        "state": "online",
        "encryptionType": "cloudEncrypted",
        "encryptionKeyId": null,
        "isRoot": false,
        "homeNode": "ziv01we01-01",
        "ownerNode": "ziv01we01-01",
        "capacityTier": null,
        "capacityTierUsed": null,
```

```
            "sidlEnabled": true,
            "snaplockType": "non_snaplock"
        }
    ]
```

Get aggregates for high availability pair

You can use this workflow to retrieve the aggregates for an HA working environment.

1. Select the working environment to use

Perform the workflow Get working environments and choose the `publicId` value of the working environment for the `workingEnvironmentId` path parameter.

2. Get the list of aggregates

HTTP method	Path
GET	/occm/api/aws/ha/aggregates/{workingEnvironmentId}

curl example

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

Input

Path parameter <WORKING_ENV_ID> (`workingEnvironmentId`)

Output

An array of aggregates for the indicated working environment is returned as shown in the JSON output example.

JSON output example

```
[
    {
        "name": "aggr1",
        "availableCapacity": {
            "size": 83.13,
            "unit": "GB"
        },
        "totalCapacity": {
            "size": 84.14,
            "unit": "GB"
        },
        "usedCapacity": {
```

```
                "size": 1.02,
                "unit": "GB"
        },
        "volumes": [
            {
                "name": "svm_ziv04we01ha_root",
                "totalSize": {
                    "size": 1.0,
                    "unit": "GB"
                },
                "usedSize": {
                    "size": 7.0953369140625E-4,
                    "unit": "GB"
                },
                "thinProvisioned": false,
                "isClone": false,
                "rootVolume": true
            }
        ],
        "providerVolumes": [
            {
                "id": "vol-0d5d1983432218cec",
                "name": "vol-0d5d1983432218cec",
                "size": {
                    "size": 100.0,
                    "unit": "GB"
                },
                "state": "in-use",
                "device": "/dev/xvdh",
                "instanceId": "i-053d9d1bce8121c43",
                "diskType": "gp2",
                "encrypted": true,
                "iops": null
            },
            {
                "id": "vol-0a3fe8eaed2af69de",
                "name": "vol-0a3fe8eaed2af69de",
                "size": {
                    "size": 100.0,
                    "unit": "GB"
                },
                "state": "in-use",
                "device": "/dev/xvdh",
                "instanceId": "i-0392f55ca4bc06322",
                "diskType": "gp2",
                "encrypted": true,
```

```
                "iops": null
            }
        ],
        "disks": [
            {
                "name": "NET-1.4",
                "position": "data",
                "ownerNode": "ziv04we01ha-01",
                "device": "xvdh vol0d5d1983432218cec",
                "vmDiskProperties": null
            },
            {
                "name": "NET-2.4",
                "position": "data",
                "ownerNode": "ziv04we01ha-01",
                "device": "xvdh vol0a3fe8eaed2af69de",
                "vmDiskProperties": null
            }
        ],
        "state": "online",
        "encryptionType": "cloudEncrypted",
        "encryptionKeyId": null,
        "isRoot": false,
        "homeNode": "ziv04we01ha-01",
        "ownerNode": "ziv04we01ha-01",
        "capacityTier": null,
        "capacityTierUsed": null,
        "sidlEnabled": true,
        "snaplockType": "non_snaplock"
    },
    {
        "name": "ziv04we01haagg01",
        "availableCapacity": {
            "size": 84.14,
            "unit": "GB"
        },
        "totalCapacity": {
            "size": 84.14,
            "unit": "GB"
        },
        "usedCapacity": {
            "size": 156.0,
            "unit": "KB"
        },
        "volumes": [],
        "providerVolumes": [
```

```json
        {
            "id": "vol-0149ffa06bb4e92ad",
            "name": "vol-0149ffa06bb4e92ad",
            "size": {
                "size": 100.0,
                "unit": "GB"
            },
            "state": "in-use",
            "device": "/dev/xvdi",
            "instanceId": "i-0392f55ca4bc06322",
            "diskType": "gp2",
            "encrypted": true,
            "iops": null
        },
        {
            "id": "vol-03e6ada9e893b1196",
            "name": "vol-03e6ada9e893b1196",
            "size": {
                "size": 100.0,
                "unit": "GB"
            },
            "state": "in-use",
            "device": "/dev/xvdi",
            "instanceId": "i-053d9d1bce8121c43",
            "diskType": "gp2",
            "encrypted": true,
            "iops": null
        }
    ],
    "disks": [
        {
            "name": "NET-2.5",
            "position": "data",
            "ownerNode": "ziv04we01ha-01",
            "device": "xvdi vol0149ffa06bb4e92ad",
            "vmDiskProperties": null
        },
        {
            "name": "NET-1.5",
            "position": "data",
            "ownerNode": "ziv04we01ha-01",
            "device": "xvdi vol03e6ada9e893b1196",
            "vmDiskProperties": null
        }
    ],
    "state": "online",
```

```
        "encryptionType": "cloudEncrypted",
        "encryptionKeyId": null,
        "isRoot": false,
        "homeNode": "ziv04we01ha-01",
        "ownerNode": "ziv04we01ha-01",
        "capacityTier": null,
        "capacityTierUsed": null,
        "sidlEnabled": true,
        "snaplockType": "non_snaplock"
    }
]
```

Create aggregate

You can create a new aggregate within a working environment using this workflow. Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Create aggregate for single node

You can use this workflow to create an aggregate for a single node working environment.

1. Select the working environment to use

Perform the workflow Get working environments and choose the `publicId` value for the `workingEnvironmentId` parameter.

2. Create the aggregate

HTTP method	Path
POST	occm/api/vsa/aggregates

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/vsa/aggregates' --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.

JSON input example

```
{
  "name": "ziv01agg04",
  "workingEnvironmentId": "VsaWorkingEnvironment-9e6p8LuF",
  "numberOfDisks": 1,
  "diskSize": {
    "size": 100,
    "unit": "GB"
  },
  "providerVolumeType": "gp2"
}
```

Output

None

Create aggregate for high availability pair

You can use this workflow to create an aggregate for an HA working environment.

1. Select the working environment to use

Perform the workflow Get working environments and choose the `publicId` value for the `workingEnvironmentId` parameter.

2. Create the aggregate

HTTP method	Path
POST	occm/api/aws/ha/aggregates

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/aws/ha/aggregates'
--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.

JSON input example

```
{
  "name": "ziv01agg04",
  "workingEnvironmentId": "VsaWorkingEnvironment-9e6p8LuF",
  "numberOfDisks": 1,
  "diskSize": {
    "size": 100,
    "unit": "GB"
  },
  "providerVolumeType": "gp2"
}
```

Output

　None

Add disks to aggregate

You can add disks to an existing aggregate.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Add disks to an aggregate for single node

You can use this workflow to add disks to an aggregate for a single node working environment.

1. Select the working environment to use

Perform the workflow Get working environments and choose the `publicId` value for the `workingEnvironmentId` path parameter.

2. Select the aggregate

Perform the workflow Get aggregates and choose the `name` value for the `aggregateName` path parameter.

3. Add the disks

HTTP method	Path
POST	/occm/api/vsa/aggregates/{workingEnvironmentId}/{aggregateName}/disks

curl example

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

Input

You must include the following path parameters:

- <WORKING_ENV_ID> (workingEnvironmentId)
- <AGGR_NAME> (aggregateName)

Also, the JSON input example includes an input parameter as shown.

JSON input example

```
{
    "numberOfDisks": "1"
}
```

Output

None

Add disks to an aggregate for high availability pair

You can use this workflow to add disks to an aggregate for HA working environment.

1. Select the working environment to use

Perform the workflow Get working environments and choose the `publicId` value for the
`workingEnvironmentId` path parameter.

2. Select the aggregate

Perform the workflow Get aggregates and choose the `name` value for the `aggregateName` path parameter.

3. Add the disks

HTTP method	Path
POST	/occm/api/aws/ha/aggregates/{workingEnvironmentId}/{aggregateName}/disks

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/aws/ha/aggregates/<WORKING
_ENV_ID>/<AGGR_NAME>/disks' --header 'Content-Type: application/json'
--header 'x-agent-id: <AGENT_ID>' --header 'Authorization: Bearer
<ACCESS_TOKEN>' --d @JSONinput
```

Input

You must include the following path parameters:

- <WORKING_ENV_ID> (`workingEnvironmentId`)
- <AGGR_NAME> (`aggregateName`)

Also, the JSON input example includes an input parameter as shown.

JSON input example

```
{
    "numberOfDisks": "1"
}
```

Output

None

Delete aggregate

You can delete an existing disk aggregate. Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Delete aggregate for single node

You can use this workflow to delete an aggregate for a single node working environment.

1. Select the working environment to use

Perform the workflow Get working environments and choose the `publicId` value for the `workingEnvironmentId` path parameter.

2. Select the aggregate to delete

Perform the workflow Get aggregates and choose the `name` value of the required for the `aggregateName` path parameter.

3. Delete the aggregate

HTTP method	Path
DELETE	/occm/api/vsa/aggregates/{workingEnvironmentId}/{aggregateName}

curl example

```
curl --location --request DELETE
'https://cloudmanager.cloud.netapp.com/occm/api/vsa/aggregates/<WORKING_EN
V_ID>/<AGGR_NAME>' --header 'Content-Type: application/json' --header 'x-
agent-id: <AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

Path parameters:

<WORKING_ENV_ID> (`workingEnvironmentId`)
<AGGR_NAME> (`aggregateName`)

Output

None

Delete aggregate for high availability pair

You can use this workflow to delete an aggregate for an HA working environment.

1. Select the working environment to use

Perform the workflow Get working environments and choose the `publicId` value for the
`workingEnvironmentId` path parameter.

2. Select the aggregate to delete

Perform the workflow Get aggregates and choose the `name` value of the required for the `aggregateName`
path parameter.

3. Delete the aggregate

HTTP method	Path
DELETE	/occm/api/aws/ha/aggregates/{workingEnvironmentId}/{aggregateName}

curl example

```
curl --location --request DELETE
'https://cloudmanager.cloud.netapp.com/occm/api/aws/ha/aggregates/<WORKING
_ENV_ID>/<AGGR_NAME>' --header 'Content-Type: application/json' --header
'x-agent-id: <AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

Path parameters:

```
<WORKING_ENV_ID> (workingEnvironmentId)
<AGGR_NAME> (aggregateName)
```

Output

None

Volumes

Create volume using NFS

You can use this workflow to create a volume accessed through NFS protocol.

> (i) If the properties `aggregateName` and `maxNumOfDisksApprovedToAdd` are not provided on the REST API call, the response will fail with a suggested name for the aggregate and the number of disks needed to fulfill the request.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

* Single Node
* HA pair

Create volume using NFS for single node

You can use this workflow to create volume using NFS for a single node system.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` and the `svmName` values for the `workingEnvironmentId` and the `svmName` parameters.

2. Select the aggregate

Perform the workflow Get aggregates and choose the `name` value of the aggregate for the `name` parameter.

> (i) If aggregate name does not exist and the `createAggregateIfNotFound` query parameter is set `true`, the create volume request is allowed if the named aggregate is not found.

3. Select the virtual private cloud

Perform the workflow Get virtual private clouds and choose the `cidrBlock` value of the required VPC for the `ips` parameter or fill in the desired `exportPolicyInfo` value manually.

4. Choose the size for the disk

Choose the size value for the `size:size` parameter. The `size:unit` must be one of the following: `TB`, `GB`, `MB`, `KB`, or `Byte`.

5. Create the quote

Perform the workflow Create quote. This is a recommended step but is not mandatory.

6. Create the volume

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

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/vsa/volumes' --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, including:

- <WORKING_ENV_ID> (`workingEnvironmentId`)

- <SVM_NAME> (`svmName`)

- <AGGR_NAME> (`aggregateName`)

If aggregate name does not exist, you can set the `createAggregateIfNotFound` query parameter to `true` which allows the aggregate not-found condition.

JSON input example

```
{
  "workingEnvironmentId": "VsaWorkingEnvironment-5dZfyKS5",
  "svmName": "svm_ziv01we01",
  "aggregateName": "ziv01agg01",
  "name": "ziv01vol04",
  "size": {
    "size": 100,
    "unit": "GB"
  },
  "exportPolicyInfo": {
    "policyType": "custom",
    "ips": [
      "172.31.0.0/16"
    ],
    "nfsVersion": [
      "nfs3",
      "nfs4"
    ]
  },
  "snapshotPolicyName": "default",
  "enableThinProvisioning": true,
  "enableCompression": true,
  "enableDeduplication": true,
  "maxNumOfDisksApprovedToAdd": 0
}
```

Output

None

Create volume using NFS for high availability pair

You can use this workflow to create volume using NFS for an HA working environment.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` and the `svmName` values for the `workingEnvironmentId` and the `svmName` parameters.

2. Select the aggregate

Perform the workflow Get aggregates and choose the `name` value of the aggregate for the `name` parameter.

> (i) If aggregate name does not exist and the `createAggregateIfNotFound` query parameter is set `true`, the create volume request is allowed if the named aggregate is not found.

3. Select the virtual private cloud

Perform the workflow Get virtual private clouds and choose the `cidrBlock` value of the required VPC for the `ips` parameter or fill in the desired `exportPolicyInfo` value manually.

4. Choose the size for the disk

Choose the size value for the `size:size` parameter. The `size:unit` must be one of the following: `TB`, `GB`, `MB`, `KB`, or `Byte`.

5. Create the quote

Perform the workflow Create quote. This is a recommended step but is not mandatory.

6. Create the volume

HTTP method	Path
POST	/occm/api/aws/ha/volumes

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/aws/ha/volumes' --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, including:

- <WORKING_ENV_ID> (`workingEnvironmentId`)
- <SVM_NAME> (`svmName`)
- <AGGR_NAME> (`aggregateName`)

If aggregate name does not exist, you can set the `createAggregateIfNotFound` query parameter to `true` which allows the aggregate not-found condition.

JSON input example

```
{
  "workingEnvironmentId": "VsaWorkingEnvironment-5dZfyKS5",
  "svmName": "svm_ziv01we01",
  "aggregateName": "ziv01agg01",
  "name": "ziv01vol04",
  "size": {
    "size": 100,
    "unit": "GB"
  },
  "exportPolicyInfo": {
    "policyType": "custom",
    "ips": [
      "172.31.0.0/16"
    ],
    "nfsVersion": [
      "nfs3",
      "nfs4"
    ]
  },
  "snapshotPolicyName": "default",
  "enableThinProvisioning": true,
  "enableCompression": true,
  "enableDeduplication": true,
  "maxNumOfDisksApprovedToAdd": 0
}
```

Output

None

Create volume using CIFS

You can use this workflow to create a volume accessed through CIFS.

> ⓘ If the properties `aggregateName` and `maxNumOfDisksApprovedToAdd` are not provided on the REST API call, the response will fail with a suggested name for the aggregate and the number of disks needed to fulfill the request.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Create volume using CIFS for single node

You can use this workflow to create volume using CIFS protocol for a single node system.

1. Choose the CIFS configuration

A CIFS server configuration must be defined for your working environment. You can do one of the following:

- If a CIFS configuration already exists, perform the workflow Get CIFS server configurations to access the configuration parameters.
- If a CIFS configuration does not exist, perform the workflow Create CIFS server configuration to create one.

2. Select the working environment

Perform the workflow Get working environments and choose the `publicId` and the `svmName` values for the `workingEnvironmentId` (working environment) and the `svmName` (SVM name) parameters.

3. Select the aggregate

Perform the workflow Get aggregates and choose the `name` for the `aggregateName` value.

> ⓘ If aggregate name does not exist and the `createAggregateIfNotFound` query parameter is set `true`, the create volume request is allowed if the named aggregate is not found.

4. Choose the size for the disk

Choose the size value for the `size:size` parameter. The `size:unit` must be one of the following: `TB`, `GB`, `MB`, `KB`, or `Byte`.

5. Create the quote

Perform the workflow Create quote. This is a recommended step but is not mandatory.

6. Create the volume

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

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/vsa/volumes' --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, including:

- <WORKING_ENV_ID> (`workingEnvironmentId`)
- <SVM_NAME> (`svmName`)
- <AGGR_NAME> (`aggregateName`)

If an aggregate name does not exist, you can set the `createAggregateIfNotFound` query parameter to `true` which allows the aggregate not-found condition.

JSON input example

```json
{
  "workingEnvironmentId": "VsaWorkingEnvironment-SfpVUZSc",
  "svmName": "svm_zivaws02we01",
  "aggregateName": "aggr1",
  "name": "zivaws02we02vol02Cifs",
  "size": {
    "size": 100,
    "unit": "GB"
  },
  "shareInfo": {
    "accessControl": {
        "permission": "full_control",
        "users": [
            "Everyone"
        ],
        "users": "Everyone;"
    },
    "shareName": "zivaws02we01vol02Cifs_share"
  },
  "snapshotPolicyName": "default",
  "enableThinProvisioning": true,
  "enableCompression": true,
  "enableDeduplication": true,
  "maxNumOfDisksApprovedToAdd": 0
}
```

Output

None

Create volume using CIFS for high availability pair

You can use this workflow to create volume using CIFS for an HA working environment.

1. Choose the CIFS configuration

A CIFS server configuration must be defined for your working environment. You can do one of the following:

- If a CIFS configuration already exists, perform the workflow Get CIFS server configurations to access the configuration parameters.

- If a CIFS configuration does not exist, perform the workflow Create CIFS server configuration to create one.

2. Select the working environment

Perform the workflow Get working environments and choose the `publicId` and the `svmName` values for the `workingEnvironmentId` (working environment) and the `svmName` (SVM name) parameters.

3. Select the aggregate

Perform the workflow Get aggregates and choose the `name` for the `aggregateName` value.

> ⓘ If aggregate name does not exist and the `createAggregateIfNotFound` query parameter is set `true`, the create volume request is allowed if the named aggregate is not found.

4. Choose the size for the disk

Choose the size value for the `size:size` parameter. The `size:unit` must be one of the following: `TB`, `GB`, `MB`, `KB`, or `Byte`.

5. Create the quote

Perform the workflow Create quote. This is a recommended step but is not mandatory.

6. Create the volume

HTTP method	Path
POST	/occm/api/aws/ha/volumes

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/aws/ha/volumes' --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, including:

- <WORKING_ENV_ID> (`workingEnvironmentId`)
- <SVM_NAME> (`svmName`)
- <AGGR_NAME> (`aggregateName`)

If an aggregate name does not exist, you can set the `createAggregateIfNotFound` query parameter to `true` which allows the aggregate not-found condition.

JSON input example

```
{
  "workingEnvironmentId": "VsaWorkingEnvironment-SfpVUZSc",
  "svmName": "svm_zivaws02we01",
  "aggregateName": "aggr1",
  "name": "zivaws02we02vol02Cifs",
  "size": {
    "size": 100,
    "unit": "GB"
  },
  "shareInfo": {
    "accessControl": {
        "permission": "full_control",
        "users": [
            "Everyone"
        ],
        "users": "Everyone;"
    },
    "shareName": "zivaws02we01vol02Cifs_share"
  },
  "snapshotPolicyName": "default",
  "enableThinProvisioning": true,
  "enableCompression": true,
  "enableDeduplication": true,
  "maxNumOfDisksApprovedToAdd": 0
}
```

Output

None

Create volume using iSCSI

You can use this workflow to create a volume accessed through iSCSI protocol.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Create volume using iSCSI for single node

You can use this workflow to create volume using iSCSI for a single node system.

There are two workflows available depending on whether a new or existing iGroup is used. You need to select the correct workflow:

- Create volume using iSCSI with a new iGroup
- Create volume using iSCSI with an existing iGroup

Create volume using iSCSI with a new iGroup

> ℹ️ If the properties `aggregateName` and `maxNumOfDisksApprovedToAdd` are not provided on the REST API call, the response will fail with a suggested name for the aggregate and the number of disks needed to fulfill the request.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value for the `workingEnvironmentId` parameter and the `svmName` value for the `svmName` parameter.

2. Select the aggregate

Perform the workflow Get aggregates and choose the `name` for the `aggregateName` value.

3. Choose the size for the disk

Choose the size value for the `size:size` parameter. The `size:unit` must be one of the following: `TB`, `GB`, `MB`, `KB`, or `Byte`.

4. Choose the iscasiInfo parameters

You must choose the following values for the REST API call:

- A unique igroup name for `igroupCreationRequest → igroupName` parameter
- The required iqn's to `igroupCreationRequest → initiators` parameter.
- The required operating system for the `osName` parameter from one of the following:
    - windows
    - linux
    - vmware
    - windows_2008
    - windows_gpt

5. Create the quote

Perform the workflow Create quote. This is a recommended step but is not mandatory.

6. Create the volume

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

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/vsa/volumes' --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, including:

- <WORKING_ENV_ID> `(workingEnvironmentId)`

- <SVM_NAME> `(svmName)`

- <AGGR_NAME> `(aggregateName)`

If aggregate name does not exist, you can set the `createAggregateIfNotFound` query parameter to `true` which allows the aggregate not-found condition.

JSON input example

```json
{
  "workingEnvironmentId": "VsaWorkingEnvironment-SfpVUZSc",
  "svmName": "svm_zivaws02we01",
  "aggregateName": "aggr1",
  "name": "zivaws02we01vol01Iscsi",
  "size": {
    "size": 100,
    "unit": "GB"
  },
  "iscsiInfo": {
    "igroupCreationRequest": {
        "igroupName": "zivIgroup",
        "initiators": [
            "iqn.1994-05.com.redhat:96de86825216",
            "iqn.1994-05.com.redhat:96de86823426"
        ]
    },
    "osName": "linux"
  },
  "snapshotPolicyName": "default",
  "enableThinProvisioning": true,
  "enableCompression": true,
  "enableDeduplication": true,
  "maxNumOfDisksApprovedToAdd": 0
}
```

Output

None

Create volume using iSCSI with an existing iGroup

ⓘ If the properties `aggregateName` and `maxNumOfDisksApprovedToAdd` are not provided on the REST API call, the response will fail with a suggested name for the aggregate and the number of disks needed to fulfill the request.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value for the `workingEnvironmentId` parameter and the `svmName` value for the `svmName` parameter.

2. Select the aggregate

Perform the workflow Get aggregates and choose the `name` for the `aggregateName` parameter.

3. Choose the size for the disk

Choose the size value for the `size:size` parameter. The `size:unit` must be one of the following: `TB`, `GB`, `MB`, `KB`, or `Byte`.

4. Choose the iGroup

Perform the workflow Get iGroups and choose the igroups for the `iscasiInfo → igroups` value. Also select the `osType` value for the `iscasiInfo → osName`.

5. Create the quote

Perform the workflow Create quote. This is a recommended step but is not mandatory.

6. Create the volume

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

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/vsa/volumes' --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, including:

- <WORKING_ENV_ID> (`workingEnvironmentId`)
- <SVM_NAME> (`svmName`)

- <AGGR_NAME> (aggregateName)

If an aggregate name does not exist, you can set the `createAggregateIfNotFound` query parameter to `true` which allows the aggregate not-found condition.

JSON input example

```json
{
  "workingEnvironmentId": "VsaWorkingEnvironment-UvFmWXoD",
  "svmName": "svm_zivaws01we01",
  "aggregateName": "aggr1",
  "name": "zivaws01we01vol05Iscsi",
  "size": {
    "size": 100,
    "unit": "GB"
  },
  "iscsiInfo": {
    "igroups": ["zivIgroup1"],
    "osName": "linux"
  },
  "snapshotPolicyName": "default",
  "enableThinProvisioning": true,
  "enableCompression": true,
  "enableDeduplication": true,
  "maxNumOfDisksApprovedToAdd": 0
}
```

Output

None

Create volume using iSCSI for high availability pair

You can use this workflow to create volume using iSCSI for an HA working environment.

There are two workflows available depending on whether a new or existing iGroup is used. You need to select the correct workflow:

- Create volume using iSCSI with a new iGroup
- Create volume using iSCSI with an existing iGroup

Create volume using iSCSI with a new iGroup

ⓘ   If the properties `aggregateName` and `maxNumOfDisksApprovedToAdd` are not provided on the REST API call, the response will fail with a suggested name for the aggregate and the number of disks needed to fulfill the request.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value for the `workingEnvironmentId` parameter and the `svmName` value for the `svmName` parameter.

2. Select the aggregate

Perform the workflow Get aggregates and choose the `name` for the `aggregateName` parameter.

3. Choose the size for the disk

Choose the size value for the `size:size` parameter. The `size:unit` must be one of the following: `TB`, `GB`, `MB`, `KB`, or `Byte`.

4. Choose the iscasiInfo parameters

You must choose the following values for the REST API call:

* A unique igroup name for `igroupCreationRequest → igroupName` parameter
* The required iqn's to `igroupCreationRequest → initiators` parameter.
* The required operating system for the `osName` parameter from one of the following:
  ◦ windows
  ◦ linux
  ◦ vmware
  ◦ windows_2008
  ◦ windows_gpt

5. Create the quote

Perform the workflow Create quote.This is a recommended step but is not mandatory.

6. Create the volume

HTTP method	Path
POST	/occm/api/aws/ha/volumes

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/aws/ha/volumes' --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, including:

* <WORKING_ENV_ID> (`workingEnvironmentId`)

- <SVM_NAME> (`svmName`)
- <AGGR_NAME> (`aggregateName`)

If aggregate name does not exist, you can set the `createAggregateIfNotFound` query parameter to `true` which allows the aggregate not-found condition.

JSON input example

```
{
  "workingEnvironmentId": "VsaWorkingEnvironment-SfpVUZSc",
  "svmName": "svm_zivaws02we01",
  "aggregateName": "aggr1",
  "name": "zivaws02we01vol01Iscsi",
  "size": {
    "size": 100,
    "unit": "GB"
  },
  "iscsiInfo": {
    "igroupCreationRequest": {
        "igroupName": "zivIgroup",
        "initiators": [
            "iqn.1994-05.com.redhat:96de86825216",
            "iqn.1994-05.com.redhat:96de86823426"
        ]
    },
    "osName": "linux"
  },
  "snapshotPolicyName": "default",
  "enableThinProvisioning": true,
  "enableCompression": true,
  "enableDeduplication": true,
  "maxNumOfDisksApprovedToAdd": 0
}
```

Output

None

Create volume using iSCSI with an existing iGroup

(i) If the properties `aggregateName` and `maxNumOfDisksApprovedToAdd` are not provided on the REST API call, the response will fail with a suggested name for the aggregate and the number of disks needed to fulfill the request.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value for the `workingEnvironmentId` parameter and the `svmName` value for the `svmName` parameter.

2. Select the aggregate

Perform the workflow Get aggregates and choose the `name` for the `aggregateName` parameter.

3. Choose the size for the disk

Choose the size value for the `size:size` parameter. The `size:unit` must be one of the following: `TB`, `GB`, `MB`, `KB`, or `Byte`.

4. Choose the iGroup

Perform the workflow Get iGroups and choose the igroups for the `iscasiInfo → igroups` value. Also select the `osType` value for the `iscasiInfo → osName`.

5. Create the quote

Perform the workflow Create quote. This is a recommended step but is not mandatory.

6. Create the volume

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

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/vsa/volumes' --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, including:

- <WORKING_ENV_ID> (`workingEnvironmentId`)
- <SVM_NAME> (`svmName`)
- <AGGR_NAME> (`aggregateName`)

If an aggregate name does not exist, you can set the `createAggregateIfNotFound` query parameter to `true` which allows the aggregate not-found condition.

JSON input example

```
{
  "workingEnvironmentId": "VsaWorkingEnvironment-UvFmWXoD",
  "svmName": "svm_zivaws01we01",
  "aggregateName": "aggr1",
  "name": "zivaws01we01vol05Iscsi",
  "size": {
    "size": 100,
    "unit": "GB"
  },
  "iscsiInfo": {
    "igroups": ["zivIgroup1"],
    "osName": "linux"
  },
  "snapshotPolicyName": "default",
  "enableThinProvisioning": true,
  "enableCompression": true,
  "enableDeduplication": true,
  "maxNumOfDisksApprovedToAdd": 0
}
```

Output

None

Get volumes

You can retrieve the list of volumes.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Get volumes for single node

You can use this workflow to retrieve a list of volumes for a single node working environment.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value of the working environment for the `workingEnvironmentId` query parameter.

2. Get the volumes

HTTP method	Path
GET	/occm/api/vsa/volumes

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/vsa/volumes?workingEnviron
mentId=<WORKING_ENV_ID>' --header 'Content-Type: application/json'
--header 'x-agent-id: <AGENT_ID>' --header 'Authorization: Bearer
<ACCESS_TOKEN>'
```

Input

Query parameter <WORKING_ENV_ID> (workingEnvironmentId)

Output

The JSON output example includes the list of volumes for the working environment.

JSON output example

```
[
    {
        "name": "ziv02vol01",
        "uuid": "cb488216-5bd1-11eb-8a9b-615eb82c79d8",
        "svmName": "svm_ziv01we02",
        "size": {
            "size": 100.0,
            "unit": "GB"
        },
        "usedSize": {
            "size": 5.18798828125E-4,
            "unit": "GB"
        },
        "junctionPath": "/ziv02vol01",
        "volumeTotalInodes": 3112959,
        "volumeUsedInodes": 96,
        "mountPoint": "172.31.1.199:/ziv02vol01",
        "compressionSpaceSaved": {
            "size": 0.0,
            "unit": "GB"
        },
        "deduplicationSpaceSaved": {
            "size": 0.0,
            "unit": "GB"
        },
        "thinProvisioning": true,
        "compression": true,
        "deduplication": true,
        "snapshotPolicy": "default",
        "securityStyle": "unix",
        "exportPolicyInfo": {
```

```json
            "name": "export-svm_ziv01we02-ziv02vol01",
            "policyType": "custom",
            "ips": [
                "172.31.0.0/16"
            ],
            "nfsVersion": [
                "nfs3"
            ]
        },
        "shareNames": [],
        "shareInfo": [],
        "parentVolumeName": "",
        "rootVolume": false,
        "state": "online",
        "volumeType": "rw",
        "aggregateName": "aggr1",
        "parentSnapshot": null,
        "autoSizeMode": "grow",
        "maxGrowSize": {
            "size": 1100.0,
            "unit": "GB"
        },
        "providerVolumeType": "gp2",
        "cloneNames": [],
        "moving": false,
        "primaryNoFailoverMountPoint": null,
        "secondaryNoFailoverMountPoint": null,
        "capacityTier": null,
        "capacityTierUsedSize": null,
        "cifsShareAccessPoint": null,
        "primaryCifsShareAccessPoint": null,
        "secondaryCifsShareAccessPoint": null,
        "tieringPolicy": "none",
        "tierInactiveUserData": {
            "size": 0.0,
            "unit": "GB"
        },
        "tierInactiveUserDataPercent": 0,
        "comment": null,
        "qosPolicyGroupName": null,
        "snaplockType": "non_snaplock",
        "constituentsAggregates": [],
        "snapshotsUsedSize": {
            "size": 1597440.0,
            "unit": "Byte"
        },
```

```
        "cbsBackupsInfo": null,
        "minimumCoolingDays": null,
        "targetName": "iqn.1992-
08.com.netapp:sn.7d147b755bd011ebb076ef46475a0933:vs.2",
        "iscsiEnabled": false,
        "isFlexGroupVolume": false
    }
]
```

Get volumes for high availability pair

You can use this workflow to retrieve a list of volumes for an HA working environment.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value of the working environment
for the `workingEnvironmentId` query parameter.

2. Get the volumes

HTTP method	Path
GET	/occm/api/aws/ha/volumes

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/aws/ha/volumes?workingEnvi
ronmentId=<WORKING_ENV_ID>' --header 'Content-Type: application/json'
--header 'x-agent-id: <AGENT_ID>' --header 'Authorization: Bearer
<ACCESS_TOKEN>'
```

Input

   Query parameter <WORKING_ENV_ID> (`workingEnvironmentId`)

Output

   The JSON output example includes the list of volumes for the working environment.

```
[
{
        "name": "ziv04we01haagg01vol01",
        "uuid": "728ad225-61ca-11eb-81ba-637783e50391",
        "svmName": "svm_ziv04we01ha",
        "size": {
            "size": 100.0,
            "unit": "GB"
        },
```

```json
        "usedSize": {
            "size": 2.93731689453125E-4,
            "unit": "GB"
        },
        "junctionPath": "/ziv04we01haagg01vol01",
        "volumeTotalInodes": 3112959,
        "volumeUsedInodes": 96,
        "mountPoint": "2.2.2.2:/ziv04we01haagg01vol01",
        "compressionSpaceSaved": {
            "size": 0.0,
            "unit": "GB"
        },
        "deduplicationSpaceSaved": {
            "size": 0.0,
            "unit": "GB"
        },
        "thinProvisioning": true,
        "compression": true,
        "deduplication": true,
        "snapshotPolicy": "default",
        "securityStyle": "unix",
        "exportPolicyInfo": {
            "name": "export-svm_ziv04we01ha-ziv04we01haagg01vol01",
            "policyType": "custom",
            "ips": [
                "172.31.0.0/16"
            ],
            "nfsVersion": [
                "nfs3",
                "nfs4"
            ]
        },
        "shareNames": [],
        "shareInfo": [],
        "parentVolumeName": "",
        "rootVolume": false,
        "state": "online",
        "volumeType": "rw",
        "aggregateName": "aggr1",
        "parentSnapshot": null,
        "autoSizeMode": "grow",
        "maxGrowSize": {
            "size": 1100.0,
            "unit": "GB"
        },
        "providerVolumeType": "gp2",
```

```
            "cloneNames": [],
            "moving": false,
            "primaryNoFailoverMountPoint":
 "172.31.1.36:/ziv04we01haagg01vol01",
            "secondaryNoFailoverMountPoint":
 "172.31.2.210:/ziv04we01haagg01vol01",
            "capacityTier": null,
            "capacityTierUsedSize": null,
            "cifsShareAccessPoint": null,
            "primaryCifsShareAccessPoint": null,
            "secondaryCifsShareAccessPoint": null,
            "tieringPolicy": "none",
            "tierInactiveUserData": {
                "size": 0.0,
                "unit": "GB"
            },
            "tierInactiveUserDataPercent": 0,
            "comment": null,
            "qosPolicyGroupName": null,
            "snaplockType": "non_snaplock",
            "constituentsAggregates": [],
            "snapshotsUsedSize": {
                "size": 0.0,
                "unit": "Byte"
            },
            "cbsBackupsInfo": null,
            "minimumCoolingDays": null,
            "targetName": "iqn.1992-
08.com.netapp:sn.c4a88d8c618511eba2c7672081bef253:vs.3",
            "iscsiEnabled": false,
            "isFlexGroupVolume": false
        }
    ]
```

Modify volume

You can modify the configuration of an existing volume.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Modify volume for single node

You can use this workflow to modify the configuration of an existing volume for a single node working environment.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` and `svmName` values for the `workingEnvironmentId` and `svmName` path parameters.

2. Select the volume

Perform the workflow Get volumes and choose the `name` for the `volumeName` path parameter.

3. Modify the volume

HTTP method	Path
PUT	/occm/api/vsa/volumes/{workingEnvironmentId}/{svmName}/{volumeName}

curl example

```
curl --location --request PUT
'https://cloudmanager.cloud.netapp.com/occm/api/vsa/volumes/<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> (svmName)
- <VOLUME_NAME> (volumeName)

The JSON input example includes the minimum list of input parameters.

JSON input example

```
{
  "exportPolicyInfo": {
    "policyType": "custom",
    "ips": [
      "172.31.0.0/16"
    ],
    "nfsVersion": [
      "nfs3"
    ]
  }
}
```

Output

None

Modify volume for high availability pair

You can use this workflow to modify the configuration of an existing volume for an HA working environment.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` and `svmName` values for the `workingEnvironmentId` and `svmName` path parameters.

2. Select the volume

Perform the workflow Get volumes and choose the `name` for the `volumeName` path parameter.

3. Modify the volume

HTTP method	Path
PUT	/occm/api/aws/ha/volumes/{workingEnvironmentId}/{svmName}/{volumeName}

curl example

```
curl --location --request PUT
'https://cloudmanager.cloud.netapp.com/occm/api/aws/ha/volumes/<WORKING_EN
V_ID>/<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> (`svmName`)
- <VOLUME_NAME> (`volumeName`)

The JSON input example includes the minimum list of input parameters.

JSON input example

```
{
  "exportPolicyInfo": {
    "policyType": "custom",
    "ips": [
      "172.31.0.0/16"
    ],
    "nfsVersion": [
      "nfs3"
    ]
  }
}
```

Output

  None

Delete volume

You can delete an existing volume.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Delete volume for single node

You can use this workflow to delete a volume for a single node working environment.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` and `svmName` values of the working environment for the `workingEnvironmentId` and `svmName` path parameters.

2. Select the volume

Perform the workflow Get volumes and choose the `name` of the required volume for the `volumeName` path parameter.

3. Delete the volume

HTTP method	Path
DELETE	/occm/api/vsa/volumes/{workingEnvironmentId}/{svmName}/{volumeName}

curl example

```
curl --location --request DELETE
'https://cloudmanager.cloud.netapp.com/occm/api/vsa/volumes/<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

Path parameters:

- <WORKING_ENV_ID> (`workingEnvironmentId`)

- <SVM_NAME> (`svmName`)

- <VOLUME_NAME> (`volumeName`)

Output

None

Delete volume for high availability pair

You can use this workflow to delete a volume for an HA working environment.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` and `svmName` values of the
working environment for the `workingEnvironmentId` and `svmName` path parameters.

2. Select the volume

Perform the workflow Get volumes and choose the `name` of teh required volume for `volumeName` path
parameter.

3. Delete the volume

HTTP method	Path
DELETE	/occm/api/aws/ha/volumes/{workingEnvironmentId}/{svmName}/{volumeName}

curl example

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

Input

Path parameters:

- <WORKING_ENV_ID> (`workingEnvironmentId`)

- <SVM_NAME> (`svmName`)
- <VOLUME_NAME> (`volumeName`)

Output

None

Create quote

You can create a quote for a new volume which returns a resource quote needed to satisfy the request. The resource quote contains aggregate information where the volume will be created and confirms if the space is available. This is a recommended step but is not mandatory.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Create quote for single node

You can use this workflow to retrieve quote details for a single node working environment.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` and `svmName` values of the working environment for the `workingEnvironmentId` and `svmName` parameters.

2. Select the aggregate

Perform the workflow Get aggregates and choose the `name` value for the `name` parameter.

3. Select the permutations configuration

Perform the workflow Get permutations and choose the `size` and `diskType` values for step 4.

4. Generate the volume quote

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

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/vsa/volumes/quote'
--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 list of input parameters.

JSON input example

```json
{
   "workingEnvironmentId": "VsaWorkingEnvironment-5dZfyKS5",
   "svmName": "svm_ziv01we01",
   "aggregateName": "ziv01agg01",
   "name": "ziv1vol02",
   "size": {
     "size": "100",
     "unit": "GB"
   },
   "enableThinProvisioning": "true",
   "providerVolumeType": "gp2",
   "verifyNameUniqueness": "true"
}
```

Output

The JSON output example includes an example of the quote details.

JSON output example

```json
{
    "numOfDisks": 0,
    "diskSize": {
        "size": 100.0,
        "unit": "GB"
    },
    "aggregateName": "ziv01agg01",
    "newAggregate": false,
    "autoVsaCapacityManagement": true
}
```

Create quote for high availability pair

You can use this workflow to retrieve quote details for an HA working environment.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` and `svmName` values of the working environment for the `workingEnvironmentId` and `svmName` parameters.

2. Select the aggregate

Perform the workflow Get aggregates and choose the `name` value for step 4.

3. Select the package

Perform the Get packages workflow and select the `size` and `diskType` values of the required package for `diskSize` and `providerVolumeType` values.

4. Generate the volume quote

HTTP method	Path
POST	/occm/api/aws/ha/quote

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/aws/ha/volumes/quote'
--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 list of input parameters.

JSON input example

```
{
  "workingEnvironmentId": "VsaWorkingEnvironment-N6BPfglr",
  "svmName": "svm_ziv04we01ha",
  "aggregateName": "aggr1",
  "name": "ziv04we01haagg01vol01",
  "size": {
    "size": "100",
    "unit": "GB"
  },
  "enableThinProvisioning": "true",
  "providerVolumeType": "gp2",
  "verifyNameUniqueness": "true"
}
```

Output

The JSON output example includes an example of the quote details.

JSON output example

```json
{
    "numOfDisks": 0,
    "diskSize": {
        "size": 100.0,
        "unit": "GB"
    },
    "aggregateName": "ziv04we01haagg01",
    "newAggregate": false,
    "autoVsaCapacityManagement": true
}
```

Get iGroups

You can use this workflow to retrieve all the initiator groups (iGroups).

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Get iGroups for single node

You can use this workflow to retrieve the iGroups for a single node system.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` and `svmName` values of the working environment for the `workingEnvironmentId` and `svmName` path parameters.

2. Get the CIFS configurations

HTTP method	Path
GET	/occm/api/vsa/volumes/igroups/{workingEnvironmentId}/{svmName}

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/vsa/volumes/igroups/<WORKI
NG_ENV_ID>/<SVM_NAME>' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>' --header 'Content-Type:
application/json'
```

Input

- Path parameter <WORKING_ENV_ID>
- Path parameter <SVM_NAME>

Output

The JSON output example includes a list of iGroups.

JSON output example

```
[
    {
        "igroupName": "zivIgroup1",
        "osType": "linux",
        "portsetName": "",
        "igroupType": "iscsi",
        "initiators": [
            "iqn.1994-05.com.redhat:1d9ac633937c"
        ]
    },
    {
        "igroupName": "zivIgroup2",
        "osType": "linux",
        "portsetName": "",
        "igroupType": "iscsi",
        "initiators": [
            "iqn.1994-05.com.redhat:96de86825216"
        ]
    }
]
```

Get iGroups for high availability pair

You can use this workflow to retrieve the iGroups for an HA working environment.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` and `svmName` values of the working environment for the `workingEnvironmentId` and `svmName` path parameters.

2. Get the CIFS configurations

HTTP method	Path
GET	/occm/api/aws/ha/volumes/igroups/{workingEnvironmentId}/{svmName}

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/aws/ha/volumes/igroups/<WO
RKING_ENV_ID>/<SVM_NAME>' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>' --header 'Content-Type:
application/json'
```

Input

- Path parameter <WORKING_ENV_ID>
- Path parameter <SVM_NAME>

Output

The JSON output example includes a list of iGroups.

JSON output example

```
[
    {
        "igroupName": "zivIgroup1",
        "osType": "linux",
        "portsetName": "",
        "igroupType": "iscsi",
        "initiators": [
            "iqn.1994-05.com.redhat:1d9ac633937c"
        ]
    },
    {
        "igroupName": "zivIgroup2",
        "osType": "linux",
        "portsetName": "",
        "igroupType": "iscsi",
        "initiators": [
            "iqn.1994-05.com.redhat:96de86825216"
        ]
    }
]
```

Metadata

Get regions

This workflow retrieves the AWS regions in which an Cloud Volumes ONTAP working environment may be created.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Get regions for single node

You can use this workflow to retrieve the regions for a single node system.

1. Get the list of regions

HTTP method	Path
GET	/occm/api/vsa/metadata/regions

curl example

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

Input

None

Output

The JSON output provides an example of a list of AWS regions.

JSON output example

```
[
    {
        "name": "US East",
        "code": "us-east-1",
        "location": "N. Virginia",
        "s3Region": "US"
    },
    {
        "name": "US West",
        "code": "us-west-1",
        "location": "N. California",
        "s3Region": "us-west-1"
    },
    {
        "name": "EU",
        "code": "eu-central-1",
        "location": "Frankfurt",
        "s3Region": "eu-central-1"
    },
    {
        "name": "Asia Pacific",
        "code": "ap-south-1",
        "location": "Mumbai",
        "s3Region": "ap-south-1"
    },
]
```

Get regions for high availability pair

You can use this workflow to retrieve the regions for an HA pair.

1. Get the list of regions

HTTP method	Path
GET	/occm/api/aws/ha/metadata/regions

curl example

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

Input

None

Output

The JSON output provides an example of a list of AWS regions.

JSON output example

```
[
    {
        "name": "US East",
        "code": "us-east-1",
        "location": "N. Virginia",
        "s3Region": "US"
    },
    {
        "name": "US East",
        "code": "us-east-2",
        "location": "Ohio",
        "s3Region": "us-east-2"
    },
    {
        "name": "US West",
        "code": "us-west-1",
        "location": "N. California",
        "s3Region": "us-west-1"
    }
]
```

Get permutations

You can use the permutations endpoint to retrieve the Cloud Volumes ONTAP configuration information.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Get permutations for single node

You can use this workflow to retrieve the configuration information for a single node working environment.

1. Get the permutations

HTTP method	Path
GET	/occm/api/vsa/metadata/permutations

curl example

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

Input

There are several optional query parameters you can use:

- region
- version
- license
- instance_type
- default_instance_type
- feature
- latest_only
- ami

Output

The JSON output example includes the list of Cloud Volumes ONTAP configurations.

JSON output example

```
[
    {
        "ontapVersion": "ONTAP-9.9.0X4.T1",
        "license": {
            "type": "cot-explore-paygo",
            "name": "Cloud Volumes ONTAP Explore",
            "description": "Suitable for smaller capacity applications.
Supports up to 2 TB of underlying AWS storage.",
            "subName": "",
            "subDescription":"Support of tiering to object storage is not
included.",
            "capacity_limit": "2TB",
            "platformLicenseRequired": false,
            "default":false,
            "capacityLimit": {"size":2.0, "unit": "TB"}
        },
        "instanceType": "m5.xlarge",
        "region": {
            "name": "EU",
            "code": "eu-central-1",
            "location": "Frankfurt",
            "s3Region": "eu-central-1"
        },
        "defaultInstance":true,
        "features": ["ena","kvm","network-utilization"],
        "upgradeableFrom": ["9.8","9.9.0"]
    }
]
```

Get permutations for high availability pair

You can use this workflow to retrieve the configuration information for an HA working environment.

1. Get the permutations

HTTP method	Path
GET	/occm/api/aws/ha/metadata/permutations

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/aws/ha/metadata/permutatio
ns?latest_only=true' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>' --header 'Content-Type:
application/json'
```

Input

There are several optional query parameters you can use:

- region

- version

- license

- instance_type

- default_instance_type

- feature

- latest_only

- ami

Output

The JSON output example includes the list of Cloud Volumes ONTAP configurations.

JSON output example

```
[
    {
        "ontapVersion": "ONTAP-9.9.0X6.T1.ha",
        "license": {
            "type": "ha-cot-explore-paygo",
            "name": "Cloud Volumes ONTAP Explore",
            "description": "Suitable for smaller capacity applications.
Supports up to 2 TB of underlying AWS storage.",
            "subName": "",
            "subDescription": "Support of tiering to object storage is not
included.",
            "capacity_limit": "2TB",
            "platformLicenseRequired": false,
            "default": false,
            "capacityLimit": {
                "size": 2.0,
                "unit": "TB"
            }
        },
        "instanceType": "m5.xlarge",
        "region": {
            "name": "EU",
            "code": "eu-central-1",
            "location": "Frankfurt",
            "s3Region": "eu-central-1"
        },
        "defaultInstance": true,
        "features": [
            "ena",
```

```
            "kvm",
            "network-utilization"
        ],
        "upgradeableFrom": [
            "9.8",
            "9.9.0"
        ]
    },
    {
        "ontapVersion": "ONTAP-9.9.0X6.T1.ha",
        "license": {
            "type": "ha-cot-explore-paygo",
            "name": "Cloud Volumes ONTAP Explore",
            "description": "Suitable for smaller capacity applications.
Supports up to 2 TB of underlying AWS storage.",
            "subName": "",
            "subDescription": "Support of tiering to object storage is not
included.",
            "capacity_limit": "2TB",
            "platformLicenseRequired": false,
            "default": false,
            "capacityLimit": {
                "size": 2.0,
                "unit": "TB"
            }
        },
        "instanceType": "m5.xlarge",
        "region": {
            "name": "EU",
            "code": "eu-west-1",
            "location": "Ireland",
            "s3Region": "eu-west-1"
        },
        "defaultInstance": true,
        "features": [
            "ena",
            "kvm",
            "network-utilization"
        ],
        "upgradeableFrom": [
            "9.8",
            "9.9.0"
        ]
    }
]
```

Get list of virtual private clouds

You can use this workflow to retrieve a list of the available virtual private clouds (VPCs).

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Get VPCs for single node

You can use this workflow to retrieve the VPCs for a single node working environment.

1. Select the region to use

Perform the workflow Get regions and choose the `code` value for the `region` query parameter.

2. Get the VPCs

HTTP method	Path
GET	/occm/api/vsa/metadata/vpcs?region=<REGION>

Curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/vsa/metadata/vpcs?region=<
REGION>' --header 'Content-Type: application/json' --header 'x-agent-id:
<AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

Query parameters:

- <REGION> `region`

Output

The JSON output example includes the list of virtual private clouds.

JSON output example

```
[
    {
        "vpcId": "vpc-b16c90d4",
        "state": "available",
        "cidrBlock": "172.31.0.0/16",
        "tags": [
            {
                "key": "Name",
                "value": "VPC for VSA"
            },
```

```
                    {
                        "key": "last",
                        "value": "ioio"
                    }
                ],
                "default": true,
                "subnets": [
                    {
                        "subnetId": "subnet-c1d99699",
                        "cidr": "172.31.5.0/24",
                        "subnetName": "subnet5",
                        "availabilityZone": "us-east-1a",
                        "availableIps": 247,
                        "minimumRequiredIps": 8,
                        "outpostArn": null
                    },
                    {
                        "subnetId": "subnet-deebdbe3",
                        "cidr": "172.31.6.0/24",
                        "subnetName": "Proxy Subnet",
                        "availabilityZone": "us-east-1e",
                        "availableIps": 248,
                        "minimumRequiredIps": 8,
                        "outpostArn": null
                    }
                ],
                "securityGroups": [
                    {
                        "securityGroupId": "xx-xxxa1ne9xxx67xcvf",
                        "description": "NetApp OCCM Instance External Security
Group",
                        "name": "hguyiuukOCCM1590415972561-OCCMSecurityGroup-
yryrytt"
                    },
                    {
                        "securityGroupId": "xx-xxxa1ne9xxx67xcvf",
                        "description": "Enable HTTP and NFS for Cloud Restore
Instance",
                        "name": "Cloud-Restore-Instance-account-xxxxxx-
CloudRestoreInstanceSecurityGroup-sdweretttrrt"
                    }
                ],
                "tenancy": "default"
            },
            {
                "vpcId": "vpc-fe5c1f98",
```

```
        "state": "available",
        "cidrBlock": "140.30.0.0/16",
        "tags": [
            {
                "key": "Name",
                "value": "VPCWIthNOS3"
            }
        ],
        "default": false,
        "subnets": [
            {
                "subnetId": "subnet-e84722d4",
                "cidr": "140.30.5.0/24",
                "subnetName": "subnet1",
                "availabilityZone": "us-east-1e",
                "availableIps": 250,
                "minimumRequiredIps": 8,
                "outpostArn": null
            }
        ],
        "securityGroups": [
            {
                "securityGroupId": "xx-xxxa1ne9xxx67xcvf",
                "description": "ONTAP Cloud firewall rules for management
and data interface",
                "name": "xx-Version10-duahpJbS-
NetAppExternalSecurityGroup-rtyunht"
            },
            {
                "securityGroupId": "xx-xxxa1ne9xxx67xcvf",
                "description": "default VPC security group",
                "name": "default"
            }
        ],
        "tenancy": "default"
    }
]
```

Get VPCs for high availability pair

You can use this workflow to retrieve the VPCs for an HA working environment.

1. Select the region to use

Perform the workflow Get regions and choose the `code` value for the `region` query parameter.

2. Get the VPCs

HTTP method	Path
GET	/occm/api/aws/ha/metadata/vpcs?region=<REGION>

Curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/aws/ha/metadata/vpcs?regio
n=<REGION>' --header 'Content-Type: application/json' --header 'x-agent-
id: <AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

Query parameters:

- <REGION> region

Output

The JSON output example includes the list of virtual private clouds.

JSON output example

```
[
    {
        "vpcId": "vpc-b16c90d4",
        "state": "available",
        "cidrBlock": "172.31.0.0/16",
        "tags": [
            {
                "key": "Name",
                "value": "VPC for VSA"
            },
            {
                "key": "last",
                "value": "ioio"
            }
        ],
        "default": true,
        "subnets": [
            {
                "subnetId": "subnet-c1d99699",
                "cidr": "172.31.5.0/24",
                "subnetName": "subnet5",
                "availabilityZone": "us-east-1a",
                "availableIps": 247,
                "minimumRequiredIps": 8,
                "outpostArn": null
```

```
            },
            {
                "subnetId": "subnet-deebdbe3",
                "cidr": "172.31.6.0/24",
                "subnetName": "Proxy Subnet",
                "availabilityZone": "us-east-1e",
                "availableIps": 248,
                "minimumRequiredIps": 8,
                "outpostArn": null
            }
        ],
        "securityGroups": [
            {
                "securityGroupId": "xx-xxxa1ne9xxx67xcvf",
                "description": "NetApp OCCM Instance External Security
Group",
                "name": "lilush2000OCCM1590415972561-OCCMSecurityGroup-
JDB72N6W90UG"
            },
            {
                "securityGroupId": "xx-xxxa1ne9xxx67xcvf",
                "description": "Enable HTTP and NFS for Cloud Restore
Instance",
                "name": "Cloud-Restore-Instance-account-xxxxxx-
CloudRestoreInstanceSecurityGroup-sdweretttrrt"
            }
        ],
        "tenancy": "default"
    },
    {
        "vpcId": "vpc-fe5c1f98",
        "state": "available",
        "cidrBlock": "140.30.0.0/16",
        "tags": [
            {
                "key": "Name",
                "value": "VPCWIthNOS3"
            }
        ],
        "default": false,
        "subnets": [
            {
                "subnetId": "subnet-e84722d4",
                "cidr": "140.30.5.0/24",
                "subnetName": "subnet1",
                "availabilityZone": "us-east-1e",
```

```
                "availableIps": 250,
                "minimumRequiredIps": 8,
                "outpostArn": null
            }
        ],
        "securityGroups": [
            {
                "securityGroupId": "xx-xxxa1ne9xxx67xcvf",
                "description": "ONTAP Cloud firewall rules for management
 and data interface",
                "name": "XX-Version10-yuiloJbS-
 NetAppExternalSecurityGroup-TYUOPR"
            },
            {
                "securityGroupId": "xx-xxxa1ne9xxx67xcvf",
                "description": "default VPC security group",
                "name": "default"
            }
        ],
        "tenancy": "default"
    }
]
```

Get EBS volume types

You can use this workflow to retrieve the supported EBS volume types.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Get EBS volume types for single node

You can use this workflow to get the EBS volume types for a single node working environment.

1. Get the EBS volume types

HTTP method	Path
GET	/occm/api/vsa/metadata/ebs-volume-types

curl example

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

Input

  None

Output

  The JSON output example includes the list of EBS volume types.

JSON output example

```
[
    {
        "description": "",
        "size": {
            "size": 100.0,
            "unit": "GB"
        },
        "supportedVolumeTypes": [
            "standard",
            "io1",
            "io2",
            "gp2"
        ],
        "supportedOccmLicenses": [
            "Explore (hourly)",
            "Standard (hourly)",
            "Standard (BYOL)",
            "Cloud Volumes ONTAP Explore",
            "Cloud Volumes ONTAP Standard",
            "Cloud Volumes ONTAP Premium",
            "Cloud Volumes ONTAP BYOL",
            "Cloud Volumes ONTAP Explore",
            "Cloud Volumes ONTAP Standard",
            "Cloud Volumes ONTAP Premium",
            "Cloud Volumes ONTAP BYOL"
        ],
        "isDefault": false
    },
    {
        "description": "",
        "size": {
            "size": 500.0,
            "unit": "GB"
```

```
        },
        "supportedVolumeTypes": [
            "standard",
            "io1",
            "io2",
            "gp2",
            "st1"
        ],
        "supportedOccmLicenses": [
            "Explore (hourly)",
            "Standard (hourly)",
            "Standard (BYOL)",
            "Cloud Volumes ONTAP Explore",
            "Cloud Volumes ONTAP Standard",
            "Cloud Volumes ONTAP Premium",
            "Cloud Volumes ONTAP BYOL",
            "Cloud Volumes ONTAP Explore",
            "Cloud Volumes ONTAP Standard",
            "Cloud Volumes ONTAP Premium",
            "Cloud Volumes ONTAP BYOL"
        ],
        "isDefault": false
    }
]
```

Get EBS volume types for high availability pair

You can use this workflow to get the EBS volume types for an HA working environment.

1. Get the EBS volume types

HTTP method	Path
GET	/occm/api/aws/ha/metadata/ebs-volume-types

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/aws/ha/metadata/ebs-
volume-types' --header 'Content-Type: application/json' --header 'x-agent-
id: <AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

None

Output

The JSON output example includes the list of EBS volume types.

JSON output example

```json
[
    {
        "description": "",
        "size": {
            "size": 100.0,
            "unit": "GB"
        },
        "supportedVolumeTypes": [
            "standard",
            "io1",
            "io2",
            "gp2",
            "gp3"
        ],
        "supportedOccmLicenses": [
            "Explore (hourly)",
            "Standard (hourly)",
            "Standard (BYOL)",
            "Cloud Volumes ONTAP Explore",
            "Cloud Volumes ONTAP Standard",
            "Cloud Volumes ONTAP Premium",
            "Cloud Volumes ONTAP BYOL",
            "Cloud Volumes ONTAP Explore",
            "Cloud Volumes ONTAP Standard",
            "Cloud Volumes ONTAP Premium",
            "Cloud Volumes ONTAP BYOL",
            "Cloud Volumes ONTAP Explore",
            "Cloud Volumes ONTAP Standard",
            "Cloud Volumes ONTAP Premium",
            "Cloud Volumes ONTAP BYOL",
            "Cloud Volumes ONTAP Standard",
            "Cloud Volumes ONTAP Premium",
            "Cloud Volumes ONTAP BYOL",
            "Cloud Volumes ONTAP Explore",
            "Cloud Volumes ONTAP Standard",
            "Cloud Volumes ONTAP Premium",
            "Cloud Volumes ONTAP BYOL",
            "Cloud Volumes ONTAP Explore",
            "Cloud Volumes ONTAP Standard",
            "Cloud Volumes ONTAP Premium",
            "Cloud Volumes ONTAP BYOL"
        ],
        "isDefault": false
    },
```

```json
    {
        "description": "",
        "size": {
            "size": 500.0,
            "unit": "GB"
        },
        "supportedVolumeTypes": [
            "standard",
            "io1",
            "io2",
            "gp2",
            "gp3",
            "st1"
        ],
        "supportedOccmLicenses": [
            "Explore (hourly)",
            "Standard (hourly)",
            "Standard (BYOL)",
            "Cloud Volumes ONTAP Explore",
            "Cloud Volumes ONTAP Standard",
            "Cloud Volumes ONTAP Premium",
            "Cloud Volumes ONTAP BYOL",
            "Cloud Volumes ONTAP Explore",
            "Cloud Volumes ONTAP Standard",
            "Cloud Volumes ONTAP Premium",
            "Cloud Volumes ONTAP BYOL",
            "Cloud Volumes ONTAP Explore",
            "Cloud Volumes ONTAP Standard",
            "Cloud Volumes ONTAP Premium",
            "Cloud Volumes ONTAP BYOL",
            "Cloud Volumes ONTAP Standard",
            "Cloud Volumes ONTAP Premium",
            "Cloud Volumes ONTAP BYOL",
            "Cloud Volumes ONTAP Explore",
            "Cloud Volumes ONTAP Standard",
            "Cloud Volumes ONTAP Premium",
            "Cloud Volumes ONTAP BYOL",
            "Cloud Volumes ONTAP Explore",
            "Cloud Volumes ONTAP Standard",
            "Cloud Volumes ONTAP Premium",
            "Cloud Volumes ONTAP BYOL"
        ],
        "isDefault": false
    }
]
```

Get packages

You can use the packages endpoint to retrieve the Cloud Volumes ONTAP packages configuration information.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Get packages for single node

You can use this workflow to retrieve the packages information for a single node system.

1. Get the packages

HTTP method	Path
GET	/occm/api/vsa/metadata/packages

curl example

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

Input

None

Output

The JSON output example includes the list of Cloud Volumes ONTAP packages.

JSON output example

```json
[
    {
        "name": "aws_ha_poc",
        "displayName": "POC and small workloads",
        "description": "No description yet",
        "licenseType": "ha-cot-explore-paygo",
        "instanceTypeMapping": [
            {
                "region": "default",
                "instanceType": "m5.xlarge"
            }
        ],
        "diskType": "gp2",
        "diskSize": {
            "size": 100.0,
            "unit": "GB"
        },
        "capacityTier": null,
        "instanceTenancy": "default",
        "writingSpeedState": "NORMAL"
    },
    {
        "name": "aws_ha_standard",
        "displayName": "Database and application data production
workloads",
        "description": "No description yet",
        "licenseType": "ha-cot-standard-paygo",
        "instanceTypeMapping": [
            {
                "region": "default",
                "instanceType": "m5.2xlarge"
            }
        ],
        "diskType": "gp2",
        "diskSize": {
            "size": 100.0,
            "unit": "GB"
        },
        "capacityTier": null,
        "instanceTenancy": "default",
        "writingSpeedState": "NORMAL"
    }
]
```

Get packages for high availability pair

You can use this workflow to retrieve the packages information for an HA working environment.

1. Get the packages

HTTP method	Path
GET	/occm/api/aws/ha/metadata/packages

curl example

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

Input

None

Output

The JSON output example includes the list of Cloud Volumes ONTAP packages.

JSON output example

```
[
    {
        "name": "aws_ha_poc",
        "displayName": "POC and small workloads",
        "description": "No description yet",
        "licenseType": "ha-cot-explore-paygo",
        "instanceTypeMapping": [
            {
                "region": "default",
                "instanceType": "m5.xlarge"
            }
        ],
        "diskType": "gp2",
        "diskSize": {
            "size": 100.0,
            "unit": "GB"
        },
        "capacityTier": null,
        "instanceTenancy": "default",
        "writingSpeedState": "NORMAL"
    },
    {
        "name": "aws_ha_standard",
        "displayName": "Database and application data production
workloads",
        "description": "No description yet",
        "licenseType": "ha-cot-standard-paygo",
        "instanceTypeMapping": [
            {
                "region": "default",
                "instanceType": "m5.2xlarge"
            }
        ],
        "diskType": "gp2",
        "diskSize": {
            "size": 100.0,
            "unit": "GB"
        },
        "capacityTier": null,
        "instanceTenancy": "default",
        "writingSpeedState": "NORMAL"
    }
]
```

Get route tables

You can use the route tables endpoint to retrieve the route tables labels for each VPC and their subnet association information.

> ⓘ We use the Get route tables workflow only for the high availability pair.

1. Get the region

Perform the workflow Get regions and choose the `code` value of the required region for the `region` query parameter.

2. Get the VPC

Perform the workflow Get VPCs and choose the `vpcId` value of the required VPC for the `vpcId` query parameter.

3. Get the route tables

HTTP method	Path
GET	/occm/api/aws/ha/metadata/route-tables

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/aws/ha/metadata/route-
tables?region=<REGION>&vpcId=<VPC_ID>' --header 'x-agent-id: <AGENT_ID>'
--header 'Authorization: Bearer <ACCESS_TOKEN>' --header 'Content-Type:
application/json'
```

Input

  Query parameters:

   • <REGION> `region`

   • <VPCID> `vpcId`

Optional parameters:

 • `roleArn`: String

 • `cloudProviderAccountId`: String

   Output

     The output includes an array of route tables.

   JSON output example

```
[
    {
```

```
        "id": "rtb-3338784b",
        "main": false,
        "subnets": [],
        "tags": [
            {
                "key": "Name",
                "value": "route3"
            }
        ]
    },
    {
        "id": "rtb-e13a7a99",
        "main": false,
        "subnets": [],
        "tags": [
            {
                "key": "Name",
                "value": "route2"
            }
        ]
    },
    {
        "id": "rtb-7a3b7b02",
        "main": false,
        "subnets": [],
        "tags": [
            {
                "key": "Name",
                "value": "route4"
            }
        ]
    },
    {
        "id": "rtb-0a41155f5c924872e",
        "main": false,
        "subnets": [],
        "tags": [
            {
                "key": "Name",
                "value": "long"
            }
        ]
    },
    {
        "id": "rtb-d0e847b6",
        "main": false,
```

```
        "subnets": [],
        "tags": [
            {
                "key": "Name",
                "value": "IC Route"
            }
        ]
    },
    {
        "id": "rtb-3333734b",
        "main": false,
        "subnets": [],
        "tags": [
            {
                "key": "Name",
                "value": "route1"
            }
        ]
    },
    {
        "id": "rtb-02a45467",
        "main": true,
        "subnets": [],
        "tags": [
            {
                "key": "Name",
                "value": "main"
            }
        ]
    }
]
```

Get key pairs

You can use the key pairs endpoint to retrieve the AWS key pairs for specific regions.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Get key pairs for single node

You can use this workflow to retrieve the key pairs for a single node working environment.

1. Get the region

Perform the workflow Get regions and choose the `code` value of the required region for the `region` query parameter.

2. Get the key pairs

HTTP method	Path
GET	/occm/api/vsa/metadata/key-pairs

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/vsa/metadata/key-
pairs?region=<REGION>' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>' --header 'Content-Type:
application/json'
```

Input

Query parameter:

- <REGION> `region`

Optional parameters for filter:

- `roleArn`: String
- `cloudProviderAccountId`: String

Output

The output shows the AWS user key pairs for a specific region.

JSON output example

```
[
    "Developers_Virginia",
    "gfcqa",
    "kubernetes-net6vbp8sd",
    "kubernetes-netpn77b32",
    "occm_qa"
]
```

Get key pairs for high availability pair

You can use this workflow to retrieve the key pairs for an HA working environment.

1. Get the region

Perform the workflow Get regions and choose the `code` value of the required region for the `region` query

parameter.

> ℹ️ Ensure that you choose the path value for the HA pair while performing the Get Regions workflow.

2. Get the key pairs

HTTP method	Path
GET	/occm/api/aws/ha/metadata/key-pairs

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/aws/ha/metadata/key-
pairs?region=<REGION>' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>' --header 'Content-Type:
application/json'
```

Input

Query parameter:

- <REGION> region

Optional parameters for filter:

- roleArn: String

- cloudProviderAccountId: String

Output

The output shows the AWS user key pairs for a specific region.

JSON output example

```
[
    "Developers_Virginia",
    "gfcqa",
    "kubernetes-net6vbp8sd",
    "kubernetes-netpn77b32",
    "occm_qa"
]
```

Miscellaneous

Create AWS cloud provider account

You can use this workflow to create an AWS cloud provider account.

Before you begin

You must have AWS credentials.

1. Get the SaaS marketplace account

Perform the workflow Get SaaS marketplace account and choose the `id` value for the required subscription for `subscriptionId` parameter.

2. Create the account

HTTP method	Path
POST	/occm/api/accounts/aws

curl example

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

Input

The JSON input example includes the list of parameters.

JSON input example

```
{
  "accountName": "zivAccountTest2",
  "providerKeys": {
    "awsAccessKeys": {
      "accessKey": "accesskeystring",
      "secretKey": "secretkeystring"
    }
  },
  "subscriptionId": "subscriptionIDgoeshere"
}
```

Output

The JSON output provides an example of the cloud provider details.

JSON output example

```
{
    "publicId": "CloudProviderAccount-LCwgVOy7",
    "accountName": "zivAccountTest2",
    "accountType": "AWS_KEYS",
    "accountId": "accountIDshownhere",
    "accessKey": " accesskeyshownhere",
    "assumeRole": null,
    "occmRole": null,
    "vsaList": [],
    "subscriptionId": "subscriptionIDshownhere"
}
```

Azure workflows

Before you begin

There are several workflows you can use with the Azure public cloud.

ⓘ Review the Get started section before using any of the Cloud Manager REST API workflows.

Workflow categories

The Azure workflows are organized into the following categories:

- Working environments
- Aggregates
- Volumes
- Metadata
- Miscellaneous

See Understanding the workflow processes for more information on these categories.

Azure credentials

Before beginning to use the Azure workflows, you will need to create an Azure account through the Microsoft Azure portal and obtain the Azure credentials and the subscription ID.

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 Azure cloud provider account that you deployed the Connector in to your list of available accounts. Your Azure account needs to have the right permissions in order to create a Connector.

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

Working environments

Create a working environment with PAYGO

You can use this workflow to create a new Azure Cloud Volumes ONTAP working environment using pay-as-you-go (PAYGO) subscription.

> **Note the following when using PAYGO**:
>
> - A marketplace subscription is required.
> - A NetApp Support Site (NSS) key is recommended to register the system for support, but it's not required.
> - You can add more volumes after creating the working environment. You can choose to create a volume using either NFS, CIFS, or iSCSI protocol.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Create working environment for single node

You can use this workflow to create single node working environment with PAYGO.

1. Select the region

Perform the workflow Get regions and choose the `name` value of the required region for the `region` parameter in step 11.

2. Select the workspace

Perform the workflow Get tenants and choose the `workspacePublicId` value for the `tenantId` parameter in step 11.

3. Select the VNets

Perform the workflow Get VNets and do the following:

- Choose the `id` value of the desired vnet from the `virtualNetworks` for the `vnetId` parameter in step 11.
- Choose the `cidr` value of the desired cidr from the `virtualNetworks` for the `cidr` parameter in step 11.
- Choose the `id` of the desired subnet from the `virtualNetworks→ cidrs→ subnets` for the `subnetId` parameter in step 11.
- Choose the `id` of the desired security group from the `securityGroups` field for the `securityGroupId` parameter in step 11.

4. Select the Azure availability zone

Perform the workflow Get Azure Availability Zones and choose the number from `zones` for the

`availabilityZone` parameter in step 11.

5. Get the Azure packages configuration

Perform the Get Azure Packages workflow and choose the `name` of the desired package item for the `packageName` parameter.

6. Attach a marketplace subscription

Perform the workflow Attach SaaS subscription.

7. (Optional) Obtain an NSS key

An NSS key is **optional** when using PAYGO licensing. If needed, you can create a key or select an existing key, and include the NSS key in the `nssAccount` parameter in step 11.

- To create a new NSS key using the Cloud Manager web user interface, perform the task Generate NSS user ID and choose the `id`.
- To select an existing NSS key, perform the workflow Get NSS keys and choose the `id` of the required NSS user.

8. Select the Azure cloud provider accounts

Perform the Create Azure cloud provider accounts workflow and choose the `publicId` of the desired cloud provider account item from the `azureAccounts` for the `cloudProviderAccount` parameter.

9. Select the Azure permutations

Perform the Get Azure Permutations workflow and pick the desired permutation:

- Choose the `ontapVersion` of the desired permutation for the `ontapVersion` field inside the `vsaMetadata` parameter in step 11.
- Choose the `type` of the desired license item from the `license` parameter for the `licenseType` field inside the `vsaMetadata` parameter in step 11.
- Choose the `instanceType` of the desired permutation for the `instanceType` field inside the `vsaMetadata` parameter in step 11.

10. Select the Azure storage account types

Perform the Get Azure Storage Account Types workflow and pick the desired storage type.

- Choose the `diskType` of the desired storage type item for the `storageType` parameter in step 11.
- Choose the `size` of the desired disk size from the `sizes → size` field for the `size` field inside the `diskSize` parameter in step 11.
- Choose the `unit` of the desired disk size from the `sizes → size` field for the `unit` field inside the `diskSize` parameter in step 11.

11. Create the working environment

HTTP method	Path
POST	/occm/api/azure/vsa/working-environments

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/azure/vsa/working-
environments' --header 'x-agent-id: <AGENT_ID>' //<1> --header
'Authorization: Bearer <ACCESS_TOKEN>' //<2> ---header 'Content-Type:
application/json' --d @JSONinput
```

(1) Replace <AGENT_ID> with your agent ID.
(2) Replace <ACCESS_TOKEN> with your obtained access bearer token.

Input

The JSON input example includes the minimum list of parameters.

ⓘ   This request uses PAYGO licensing as indicated in the `licenseType` parameter.

JSON input example

```json
{
    "name":"Azure123",
    "tenantId":"tenantID",
    "region":"westeurope",
    "packageName":"azure_poc",
    "dataEncryptionType":"AZURE",
    "vsaMetadata":{
        "ontapVersion":"ONTAP-9.10.0.T1.azure",
        "licenseType":"azure-cot-explore-paygo",
        "instanceType":"Standard_DS3_v2"
    },
    "writingSpeedState":"NORMAL",
    "subnetId":"/subscriptions/x000xx00-0x00-0000-
000x/resourceGroups/occm_group_westeurope/providers/Microsoft.Network/virt
ualNetworks/Vnet1/subnets/Subnet2",
    "svmPassword":"Netapp123",
    "vnetId":"/subscriptions/x000xx00-0x00-0000-
000x/resourceGroups/occm_group_westeurope/providers/Microsoft.Network/virt
ualNetworks/Vnet1",
    "cidr":"10.0.0.0/16",
    "ontapEncryptionParameters":null,
    "securityGroupId":"/subscriptions/x000xx00-0x00-0000-
000x/resourceGroups/Cloud-Compliance-1nThiJkG05ZgcyucIJvCSbtBdpVnK-
2020894989899/providers/Microsoft.Network/networkSecurityGroups/Cloud-
Compliance-1nTxxxtkG05ZgcxxxxxxXXXXX-2000000000000000",
    "skipSnapshots":false,
    "diskSize":{
        "size":500,
        "unit":"GB",
        "_identifier":"500 GB"
    },
    "storageType":"Premium_LRS",
    "azureTags":[],
    "subscriptionId":"x000xx00-0x00-0000-000x",
    "cloudProviderAccount":"ManagedServiceIdentity",
    "backupVolumesToCbs":false,
    "enableCompliance":false,
    "enableMonitoring":false,
    "availabilityZone":1,
    "allowDeployInExistingRg":true,
    "resourceGroup":"occm_group_westeurope"
}
```

Output

The JSON output example includes an example of the `VsaWorkingEnvironmentResponse`.

JSON output example

```json
{
    "publicId": "VsaWorkingEnvironment-uFPaNkrv",
    "name": "Azure123",
    "tenantId": "tenantID",
    "svmName": "svm_Azure123",
    "creatorUserEmail": "user_mail",
    "status": null,
    "providerProperties": null,
    "reservedSize": null,
    "clusterProperties": null,
    "ontapClusterProperties": null,
    "cloudProviderName": "Azure",
    "snapshotPolicies": null,
    "actionsRequired": null,
    "activeActions": null,
    "replicationProperties": null,
    "schedules": null,
    "svms": null,
    "workingEnvironmentType": "VSA",
    "supportRegistrationProperties": null,
    "supportRegistrationInformation": null,
    "capacityFeatures": null,
    "encryptionProperties": null,
    "supportedFeatures": null,
    "isHA": false,
    "haProperties": null,
    "k8sProperties": null,
    "fpolicyProperties": null,
    "saasProperties": null,
    "cbsProperties": null,
    "complianceProperties": null,
    "monitoringProperties": null
}
```

Create working environment for high availability pair

You can use this workflow to create an HA working environment with PAYGO.

1. Select the region

Perform the workflow Get regions and choose the `name` value of the required region for the `region` parameter in step 11.

2. Select the workspace

Perform the workflow Get tenants and choose the `workspacePublicId` value for the `tenantId` parameter in step 11.

3. Select the VNets

Perform the workflow Get VNets and do the following:

- Choose the `id` value of the desired vnet from the `virtualNetworks` for the `vnetId` parameter in step 11.

- Choose the `cidr` value of the desired cidr from the `virtualNetworks` for the `cidr` parameter in step 11.

- Choose the `id` of the desired subnet from the `virtualNetworks`→ `cidrs`→ `subnets` for the `subnetId` parameter in step 11.

- Choose the `id` of the desired security group from the `securityGroups` field for the `securityGroupId` parameter in step 11.

4. Select the Azure availability zone

Perform the workflow Get Azure Availability Zones and choose the number from `zones` for the `availabilityZone` parameter in step 11.

5. Get Azure packages configuration

Perform the Get Azure Packages workflow and choose the `name` of the desired package item for the `packageName` parameter.

6. Attach a marketplace subscription

Perform the workflow Attach SaaS subscription.

7. (Optional) Obtain an NSS key

An NSS key is **optional** when using PAYGO licensing. If needed, you can create a key or select an existing key, and include the NSS key in the `nssAccount` parameter in step 11.

- To create a new NSS key using the Cloud Manager web user interface, perform the task Generate NSS user ID and choose the `id`.

- To select an existing NSS key, perform the workflow Get NSS keys and choose the `id` of the required NSS user.

8. Select Azure cloud provider accounts

Perform the Create Azure Cloud Provider Accounts workflow and choose the `publicId` of the desired cloud provider account item from the `azureAccounts` for the `cloudProviderAccount` parameter.

9. Select Azure permutations

Perform the Get Azure Permutations workflow and pick the desired permutation:

- Choose the `ontapVersion` of the desired permutation for the `ontapVersion` field inside the

`vsaMetadata` parameter in step 11.

- Choose the `type` of the desired license item from the `license` parameter for the `licenseType` field inside the `vsaMetadata` parameter in step 11.

- Choose the `instanceType` of the desired permutation for the `instanceType` field inside the `vsaMetadata` parameter in step 11.

10. Select Azure storage account types

Perform the Get Azure Storage Account Types workflow and pick the desired storage type.

- Choose the `diskType` of the desired storage type item for the `storageType` parameter in step 11.

- Choose the `size` of the desired disk size from the `sizes → size` field for the `size` field inside the `diskSize` parameter in step 11.

- Choose the `unit` of the desired disk size from the `sizes → size` field for the `unit` field inside the `diskSize` parameter in step 11.

11. Create the working environment

HTTP method	Path
POST	/occm/api/azure/ha/working-environments

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/azure/ha/working-
environments' --header 'x-agent-id: <AGENT_ID>' //<1> --header
'Authorization: Bearer <ACCESS_TOKEN>' //<2> ---header 'Content-Type:
application/json' --d @JSONinput
```

(1) Replace <AGENT_ID> with your agent ID.
(2) Replace <ACCESS_TOKEN> with your obtained access bearer token.

Input

The JSON input example includes the minimum list of parameters.

ⓘ   This request uses PAYGO licensing as indicated in the `licenseType` parameter.

JSON input example

```
{
   "name":"ranukazure12",
   "volume":{
      "exportPolicyInfo":{
         "policyType":"custom",
         "ips":[
```

```
                    "10.0.0.0/16"
            ],
            "nfsVersion":[
                "nfs3",
                "nfs4"
            ]
        },
        "snapshotPolicyName":"default",
        "name":"ranukvol12",
        "enableThinProvisioning":true,
        "enableDeduplication":true,
        "enableCompression":true,
        "size":{
            "size":100,
            "unit":"GB"
        },
        "tieringPolicy":"auto"
    },
    "tenantId":"tenantIDgoeshere",
    "region":"westeurope",
    "packageName":"azure_ha_standard",
    "dataEncryptionType":"AZURE",
    "capacityTier":"Blob",
    "vsaMetadata":{
        "ontapVersion":"ONTAP-9.10.1X7.T1.azureha",
        "licenseType":"azure-ha-cot-standard-paygo",
        "instanceType":"Standard_DS4_v2"
    },
    "writingSpeedState":"NORMAL",
    "subnetId":"/subscriptions/x000xx00-0x00-0000-000x
/resourceGroups/occm_group_westeurope/providers/Microsoft.Network/virtualN
etworks/Vnet1/subnets/Subnet2",
    "svmPassword":"password",
    "vnetId":"/subscriptions/x000xx00-0x00-0000-000x/resourceGroups
/occm_group_westeurope/providers/Microsoft.Network/virtualNetworks/Vnet1",
    "cidr":"10.0.0.0/16",
    "ontapEncryptionParameters":null,
    "skipSnapshots":false,
    "diskSize":{
        "size":1,
        "unit":"TB",
        "_identifier":"1 TB"
    },
    "storageType":"Premium_LRS",
    "azureTags":[],
    "subscriptionId":"x000xx00-0x00-0000-000x",
```

```
    "cloudProviderAccount":"ManagedServiceIdentity",
    "backupVolumesToCbs":true,
    "enableCompliance":true,
    "enableMonitoring":true,
    "availabilityZone":null,
    "resourceGroup":"ranukazure12-rg"
}
```

Output

The JSON output example includes an example of the `VsaWorkingEnvironmentResponse`.

JSON output example

```
{
    "publicId":"VsaWorkingEnvironment-1m76JaRt",
    "name":"ranukazure12",
    "tenantId":"tenantID",
    "svmName":"svm_ranukazure12",
    "creatorUserEmail":"user_email",
    "status":null,
    "providerProperties":null,
    "reservedSize":null,
    "clusterProperties":null,
    "ontapClusterProperties":null,
    "cloudProviderName":"Azure",
    "snapshotPolicies":null,
    "actionsRequired":null,
    "activeActions":null,
    "replicationProperties":null,
    "schedules":null,
    "svms":null,
    "workingEnvironmentType":"VSA",
    "supportRegistrationProperties":null,
    "supportRegistrationInformation":null,
    "capacityFeatures":null,
    "encryptionProperties":null,
    "supportedFeatures":null,
    "isHA":true,
    "haProperties":null,
    "fpolicyProperties":null,
    "saasProperties":null,
    "cbsProperties":null,
    "complianceProperties":null,
    "monitoringProperties":null,
    "licensesInformation":null
}
```

Create a working environment with BYOL licensing

You can use this workflow to create a new Cloud Volumes ONTAP working environment using bring-your-own-license (BYOL) licensing.

> **Note the following when using BYOL licensing**:
>
> - A marketplace subscription is not required.
> - A NetApp Support Site (NSS) key is required to register the system for support.
> - You can add more volumes after creating the working environment. You can choose to create a volume using either NFS, CIFS, or iSCSI protocol.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Create working environment for single node

You can use this workflow to create single node working environment with BYOL licensing.

1. Select the region

Perform the workflow Get regions and choose the `name` value of the required region for the `region` parameter in step 10.

2. Select the workspace

Perform the workflow Get tenants and choose the `workspacePublicId` value for the `tenantId` parameter in step 10.

3. Select the VNets

Perform the workflow Get VNets and do the following:

- Choose the `id` value of the desired vnet from the `virtualNetworks` for the `vnetId` parameter in step 10.
- Choose the `cidr` value of the desired cidr from the `virtualNetworks` for the `cidr` parameter in step 10.
- Choose the `id` of the desired subnet from the `virtualNetworks→ cidrs→ subnets` for the `subnetId` parameter in step 10.
- Choose the `id` of the desired security group from the `securityGroups` field for the `securityGroupId` parameter in step 10.

4. Select the Azure availability zone

Perform the workflow Get Availability Zones and choose the number from `zones` for the `availabilityZone` parameter in step 10.

5. Select Azure packages configuration

Perform the Get Azure Packages workflow and choose the `name` of the desired package item for the `packageName` parameter.

6. Obtain the required NSS key

An NSS key is **required** when using BYOL licensing. You can create a key or select an existing key, and include the NSS key in the `nssAccount` parameter in step 10.

- To create a new NSS key using the Cloud Manager web user interface, perform the task Generate NSS user ID and choose the `id`.

- To select an existing NSS key, perform the workflow Get NSS keys and choose the `id` of the required NSS user.

7. Select the Azure cloud provider accounts

Perform the Get Azure Cloud Provider Accounts workflow and choose the `publicId` of the desired cloud provider account item from the `azureAccounts` for the `cloudProviderAccount` parameter.

8. Select the Azure permutations

Perform the Get Azure Permutations workflow and pick the desired permutation:

- Choose the `ontapVersion` of the desired permutation for the `ontapVersion` field inside the `vsaMetadata` parameter in step 10.

- Choose the `type` of the desired license item from the `license` parameter for the `licenseType` field inside the `vsaMetadata` parameter in step 10.

- Choose the `instanceType` of the desired permutation for the `instanceType` field inside the `vsaMetadata` parameter in step 10.

You will also need to include the `serialNumber` value in the REST API call.
=== 9. Select the Azure storage account types

Perform the Get Azure Storage Account Types workflow and pick the desired storage type.

- Choose the `diskType` of the desired storage type item for the `storageType` parameter in step 10.

- Choose the `size` of the desired disk size from the `sizes → size` field for the `size` field inside the `diskSize` parameter in step 10.

- Choose the `unit` of the desired disk size from the `sizes → size` field for the `unit` field inside the `diskSize` parameter in step 10.

10. Create the working environment

HTTP method	Path
POST	/occm/api/azure/vsa/working-environments

curl example

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

Input

The JSON input example includes the minimum list of parameters. This request uses BYOL licensing as indicated in the `licenseType` parameter. The `serialNumber` is required.

JSON input example

```
{
  "name": "Azure123",
  "tenantId": "tenantID",
  "region": "eastus2",
  "packageName": "azure_custom",
  "dataEncryptionType": "AZURE",
  "vsaMetadata": {
    "ontapVersion": "ONTAP-9.9.0X5.T1.azure",
    "licenseType": "azure-cot-premium-byol",
    "instanceType": "Standard_DS3_v2"
  },
  "nssAccount": "x0x0x000-0000-000x-00xx-x0000cx0000xx",
  "subnetId": "/subscriptions/x000xx00-0x00-0000-
000x/resourceGroups/occm_group_eastus2/providers/Microsoft.Network/virtual
Networks/Vnet1/subnets/ProxySubnet",
  "svmPassword": "password",
  "vnetId": "/subscriptions/x000xx00-0x00-0000-
000x/resourceGroups/occm_group_eastus2/providers/Microsoft.Network/virtual
Networks/Vnet1",
  "cidr": "10.0.0.0/16",
  "diskSize": {
    "size": 1,
    "unit": "TB"
  },
  "storageType": "Premium_LRS",
  "resourceGroup": "Azure000-xx",
  "serialNumber": "000001100000000000001",
  "subscriptionId": "x000xx00-0x00-0000-000x",
  "cloudProviderAccount": "ManagedServiceIdentity",
  "availabilityZone": 2
}
```

Output

The JSON output example includes an example of the `VsaWorkingEnvironmentRresponse` response.

JSON output example

```json
{
    "publicId": "VsaWorkingEnvironment-uFPaNkrv",
    "name": "Azure123",
    "tenantId": "tenantID",
    "svmName": "svm_Azure123",
    "creatorUserEmail": "user_mail",
    "status": null,
    "providerProperties": null,
    "reservedSize": null,
    "clusterProperties": null,
    "ontapClusterProperties": null,
    "cloudProviderName": "Azure",
    "snapshotPolicies": null,
    "actionsRequired": null,
    "activeActions": null,
    "replicationProperties": null,
    "schedules": null,
    "svms": null,
    "workingEnvironmentType": "VSA",
    "supportRegistrationProperties": null,
    "supportRegistrationInformation": null,
    "capacityFeatures": null,
    "encryptionProperties": null,
    "supportedFeatures": null,
    "isHA": false,
    "haProperties": null,
    "k8sProperties": null,
    "fpolicyProperties": null,
    "saasProperties": null,
    "cbsProperties": null,
    "complianceProperties": null,
    "monitoringProperties": null
}
```

Create working environment for high availability pair

You can use this workflow to create an HA working environment with BYOL licensing.

1. Select the region

Perform the workflow Get regions and choose the `name` value of the required region for the `region` parameter in step 10.

2. Select the workspace

Perform the workflow Get tenants and choose the `workspacePublicId` value for the `tenantId` parameter in step 10.

3. Select the VNets

Perform the workflow Get VNets and do the following:

- Choose the `id` value of the desired vnet from the `virtualNetworks` for the `vnetId` parameter in step 10.
- Choose the `cidr` value of the desired cidr from the `virtualNetworks` for the `cidr` parameter in step 10.
- Choose the `id` of the desired subnet from the `virtualNetworks→ cidrs→ subnets` for the `subnetId` parameter in step 10.
- Choose the `id` of the desired security group from the `securityGroups` field for the `securityGroupId` parameter in step 10.

4. Select the Azure availability zone

Perform the workflow Get Availability Zones and choose the number from `zones` for the `availabilityZone` parameter in step 10.

5. Select Azure packages configuration

Perform the Get Azure Packages workflow and choose the `name` of the desired package item for the `packageName` parameter.

6. Obtain the required NSS key

An NSS key is **required** when using BYOL licensing. You can create a key or select an existing key, and include the NSS key in the `nssAccount` parameter in step 10.

- To create a new NSS key using the Cloud Manager web user interface, perform the task Generate NSS user ID and choose the `id`.
- To select an existing NSS key, perform the workflow Get NSS keys and choose the `id` of the required NSS user.

7. Select Azure cloud provider accounts

Perform the Get Azure Cloud Provider Accounts workflow and choose the `publicId` of the desired cloud provider account item from the `azureAccounts` for the `cloudProviderAccount` parameter.

8. Select the Azure permutations

Perform the Get Azure Permutations workflow and pick the desired permutation:

- Choose the `ontapVersion` of the desired permutation for the `ontapVersion` field inside the `vsaMetadata` parameter in step 10.
- Choose the `type` of the desired license item from the `license` parameter for the `licenseType` field inside the `vsaMetadata` parameter in step 10.

- Choose the `instanceType` of the desired permutation for the `instanceType` field inside the `vsaMetadata` parameter in step 10.

You will also need to include the `serialNumber` value in the REST API call.
=== 9. Select the Azure storage account types

Perform the Get Azure Storage Account Types workflow and pick the desired storage type.

- Choose the `diskType` of the desired storage type item for the `storageType` parameter in step 10.

- Choose the `size` of the desired disk size from the `sizes → size` field for the `size` field inside the `diskSize` parameter in step 10.

- Choose the `unit` of the desired disk size from the `sizes → size` field for the `unit` field inside the `diskSize` parameter in step 10.

10. Create the working environment

HTTP method	Path
POST	/occm/api/azure/ha/working-environments

curl example

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

Input

The JSON input example includes the minimum list of parameters. This request uses BYOL licensing as indicated in the `licenseType` parameter. The `platformSerialNumberNode1` and `platformSerialNumberNode2` parameters are required.

JSON input example

```
{
  "name": "ShirleyHa2701",
  "tenantId": "tenantID",
  "region": "eastus2",
  "packageName": "azure_ha_standard",
  "dataEncryptionType": "AZURE",
  "capacityTier": "Blob",
  "vsaMetadata": {
    "ontapVersion": "ONTAP-9.9.0X5.T1.azureha",
    "licenseType": "azure-ha-cot-premium-byol",
    "instanceType": "Standard_DS4_v2"
  },
  "nssAccount": "x0x0x000-0000-000x-00xx-x0000cx0000xx",
  "writingSpeedState": "NORMAL",
  "subnetId": "/subscriptions/x000xx00-0x00-0000-
000x/resourceGroups/occm_group_eastus2/providers/Microsoft.Network/virtual
Networks/Vnet1/subnets/Subnet1",
  "svmPassword": "password",
  "vnetId": "/subscriptions/x000xx00-0x00-0000-
000x/resourceGroups/occm_group_eastus2/providers/Microsoft.Network/virtual
Networks/Vnet1",
  "cidr": "10.0.0.0/16",
  "ontapEncryptionParameters": null,
  "skipSnapshots": false,
  "diskSize": {
    "size": 500,
    "unit": "GB",
    "_identifier": "500 GB"
  },
  "storageType": "Premium_LRS",
  "azureTags": [],
  "resourceGroup": "ShirleyHa2701-rg",
  "subscriptionId": "x000xx00-0x00-0000-000x,
  "cloudProviderAccount": "ManagedServiceIdentity",
  "backupVolumesToCbs": false,
  "enableCompliance": false,
  "enableMonitoring": false,
  "availabilityZone": null,
  "haParams": {
    "platformSerialNumberNode1": "00000110000000000001",
    "platformSerialNumberNode2": "00000110000000000002"
  }
}
```

Output

The JSON output example includes an example of the `VsaWorkingEnvironmentRresponse` response.

JSON output example

```json
{
    "publicId": "VsaWorkingEnvironment-Kms14Nkv",
    "name": "ShirleyHa2701",
    "tenantId": "tenantID",
    "svmName": "svm_ShirleyHa2701",
    "creatorUserEmail": "user_email",
    "status": null,
    "providerProperties": null,
    "reservedSize": null,
    "clusterProperties": null,
    "ontapClusterProperties": null,
    "cloudProviderName": "Azure",
    "snapshotPolicies": null,
    "actionsRequired": null,
    "activeActions": null,
    "replicationProperties": null,
    "schedules": null,
    "svms": null,
    "workingEnvironmentType": "VSA",
    "supportRegistrationProperties": null,
    "supportRegistrationInformation": null,
    "capacityFeatures": null,
    "encryptionProperties": null,
    "supportedFeatures": null,
    "isHA": true,
    "haProperties": null,
    "k8sProperties": null,
    "fpolicyProperties": null,
    "saasProperties": null,
    "cbsProperties": null,
    "complianceProperties": null,
    "monitoringProperties": null
}
```

Get working environment

You can retrieve the public identifier, working environment ID, the storage virtual machine name for Cloud Volumes ONTAP working environments and other Cloud Volumes ONTAP related details (visible to currently logged in user) which would be used in other workflows.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node

- HA pair

Get working environment for single node

You can use this workflow to retrieve the working environments' details of a single node system.

1. Create the working environment

Perform the Create Azure single node working environment workflow and select the `publicId` from the output for the `workingEnvironmentId` path parameter.

2. Get the working environment

HTTP method	Path
GET	/occm/api/azure/vsa/working-environments/{workingEnvironmentId}

curl

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/azure/vsa/working-
environments/<WORKING_ENV_ID>?fields=status,ontapClusterProperties.fields(
upgradeVersions,nodes),reservedSize,saasProperties,complianceProperties,mo
nitoringProperties,providerProperties' --header 'Content-Type:
application/json' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

   Path parameters:

- <WORKING_ENV_ID> `workingEnvironmentId` string

- (Optional) `fields` string

Output

   The JSON output example includes details of a single node Azure working environment.

JSON output example

```
{
    "publicId": "VsaWorkingEnvironment-zGQWVOyo",
    "name": "pradipm",
    "tenantId": "tenantID",
    "svmName": "svm_pradipm",
    "creatorUserEmail": "user_email",
    "status": {
        "status": "OFF",
```

```json
        "message": "",
        "failureCauses": {
            "invalidOntapCredentials": false,
            "noCloudProviderConnection": false,
            "invalidCloudProviderCredentials": false
        },
        "extendedFailureReason": null
    },
    "providerProperties": {
        "regionName": "westeurope",
        "resourceGroup": {
            "name": "occm_group_westeurope",
            "location": "westeurope",
            "tags": {
                "KeepMe": "true"
            }
        },
        "vnetCidr": "10.0.0.0/16",
        "tags": {
            "KeepMe": "true"
        },
        "subscriptionId": "x000xx00-0x00-0000-000x",
        "deploymentId": "",
        "creationTime": 1631783479373,
        "instanceType": "Standard_DS3_v2",
        "numOfNics": 3,
        "singleNetworkInterface": true,
        "subscriptionName": "OCCM Dev",
        "cloudProviderAccountId": null,
        "availabilityZone": null,
        "dataDisks": [
            {
                "name": "pradipm-disk-root",
                "diskSizeGB": 0,
                "lun": 0,
                "id": "/subscriptions/x000xx00-0x00-0000-
000x/resourceGroups/OCCM_GROUP_WESTEUROPE/providers/Microsoft.Compute/disk
s/pradipm-disk-root",
                "caching": "ReadOnly",
                "accountType": "NA",
                "managed": true,
                "encryptionSet": null
            },
            {
                "name": "pradipm-disk-nvram",
                "diskSizeGB": 0,
```

```
                "lun": 1,
                "id": "/subscriptions/dx000xx00-0x00-0000-
000x/resourceGroups/occm_group_westeurope/providers/Microsoft.Compute/disk
s/pradipm-disk-nvram",
                "caching": "None",
                "accountType": "NA",
                "managed": true,
                "encryptionSet": null
            },
            {
                "name": "pradipm-disk-core",
                "diskSizeGB": 0,
                "lun": 2,
                "id": "/subscriptions/x000xx00-0x00-0000-
000x/resourceGroups/occm_group_westeurope/providers/Microsoft.Compute/disk
s/pradipm-disk-core",
                "caching": "ReadOnly",
                "accountType": "NA",
                "managed": true,
                "encryptionSet": null
            },
            {
                "name": "pradipmdatadisk3",
                "diskSizeGB": 0,
                "lun": 3,
                "id": "/subscriptions/x000xx00-0x00-0000-
000xfbbce1b18/resourceGroups/OCCM_GROUP_WESTEUROPE/providers/Microsoft.Com
pute/disks/pradipmdatadisk3",
                "caching": "None",
                "accountType": "NA",
                "managed": true,
                "encryptionSet": null
            }
        ]
    },
    "reservedSize": {
        "size": 0.0,
        "unit": "GB"
    },
    "clusterProperties": null,
    "ontapClusterProperties": {
        "nodes": [],
        "clusterName": "",
        "clusterUuid": "xxxxx0000000000x00x0x00x0x0x0",
        "ontapVersion": "",
        "systemManagerUrl": "https://10.0.1.5/sysmgr/SysMgr.html",
```

```
        "creationTime": 1631783479373,
        "licenseType": {
            "name": "Cloud Volumes ONTAP Capacity Based Charging",
            "capacityLimit": {
                "size": 500.0,
                "unit": "GB"
            }
        },
        "licensePackageName": null,
        "lastModifiedOffbox": 1632392140549,
        "offboxTarget": false,
        "upgradeVersions": null,
        "writingSpeedState": null,
        "broadcastDomainInfos": [],
        "evaluation": false,
        "capacityTierInfo": null,
        "canConfigureCapacityTier": false,
        "usedCapacity": {
            "size": 0.0,
            "unit": "GB"
        },
        "userName": "admin",
        "wormEnabled": false,
        "isSpaceReportingLogical": false
    },
    "cloudProviderName": "Azure",
    "snapshotPolicies": null,
    "actionsRequired": null,
    "activeActions": null,
    "replicationProperties": null,
    "schedules": null,
    "svms": null,
    "workingEnvironmentType": "VSA",
    "supportRegistrationProperties": null,
    "supportRegistrationInformation": [],
    "capacityFeatures": null,
    "encryptionProperties": null,
    "supportedFeatures": null,
    "isHA": false,
    "haProperties": null,
    "fpolicyProperties": null,
    "saasProperties": {
        "subscription": null,
        "freeTrialExpiry": null,
        "saasEnabled": null,
        "capacityLicensePackage": null
```

```
        },
        "cbsProperties": null,
        "complianceProperties": {
            "scanStatus": "SCAN_DISABLED",
            "complianceStatus": null,
            "lastDeploymentError": null,
            "complianceBackupStatus": null
        },
        "monitoringProperties": {
            "monitoringStatus": "MONITORING_DISABLED",
            "monitoringInfo": null,
            "tenantUrl": null
        },
        "licensesInformation": null
    }
```

Get working environment for high availability pair

You can use this workflow to retrieve the working environments' details of an HA working environment.

1. Create an Azure HA working environment

Perform the Create Azure dual node working environment workflow and select the `publicId` from the output for the `workingEnvironmentId` path parameter.

2. Get the working environment

HTTP method	Path
GET	/occm/api/azure/ha/working-environments/{workingEnvironmentId}

curl

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/azure/ha/working-
environments/<WORKING_ENV_ID>?fields=status,ontapClusterProperties.fields(
upgradeVersions,nodes),reservedSize,saasProperties,complianceProperties,mo
nitoringProperties,providerProperties' --header 'Content-Type:
application/json' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

Path parameters:

- <WORKING_ENV_ID> `workingEnvironmentId` string

- (Optional) `fields` string

Output

The JSON output example includes details of a single node Azure working environment.

JSON output example

```json
{
    "publicId": "VsaWorkingEnvironment-Kms14Nkv",
    "name": "ShirleyHa2701",
    "tenantId": "Tenant-c6wmZaze",
    "svmName": "svm_ShirleyHa2701",
    "creatorUserEmail": "useremail",
    "status": {
        "status": "ON",
        "message": "",
        "failureCauses": {
            "invalidOntapCredentials": false,
            "noCloudProviderConnection": false,
            "invalidCloudProviderCredentials": false
        },
        "extendedFailureReason": null
    },
    "providerProperties": {
        "regionName": "eastus2",
        "resourceGroup": {
            "name": "ShirleyHa2701-rg",
            "location": "eastus2",
            "tags": {}
        },
        "vnetCidr": "10.0.0.0/16",
        "tags": {},
        "subscriptionId": "x000xx00-0x00-0000-000x",
        "deploymentId": "",
        "creationTime": 1611698774849,
        "instanceType": "Standard_DS4_v2",
        "numOfNics": 8,
        "singleNetworkInterface": false,
        "subscriptionName": "OCCM Dev",
        "cloudProviderAccountId": null,
        "availabilityZone": null,
        "dataDisks": [
            {
                "name": "ShirleyHa2701-vm1-disk-nvram",
                "diskSizeGB": 128,
                "lun": 1,
                "id": "/subscriptions/x000xx00-0x00-0000-
000x/resourceGroups/ShirleyHa2701-
rg/providers/Microsoft.Compute/disks/ShirleyHa2701-vm1-disk-nvram",
```

```json
                "caching": "None",
                "accountType": "Premium_LRS",
                "managed": true,
                "encryptionSet": null
            },
            {
                "name": "ShirleyHa2701-vm1-disk-core",
                "diskSizeGB": 1024,
                "lun": 2,
                "id": "/subscriptions/x000xx00-0x00-0000-
000x/resourceGroups/ShirleyHa2701-
rg/providers/Microsoft.Compute/disks/ShirleyHa2701-vm1-disk-core",
                "caching": "ReadOnly",
                "accountType": "Standard_LRS",
                "managed": true,
                "encryptionSet": null
            },
            {
                "name": "ShirleyHa2701-vm2-disk-nvram",
                "diskSizeGB": 128,
                "lun": 1,
                "id": "/subscriptions/x000xx00-0x00-0000-
000x/resourceGroups/ShirleyHa2701-
rg/providers/Microsoft.Compute/disks/ShirleyHa2701-vm2-disk-nvram",
                "caching": "None",
                "accountType": "Premium_LRS",
                "managed": true,
                "encryptionSet": null
            },
            {
                "name": "ShirleyHa2701-vm2-disk-core",
                "diskSizeGB": 1024,
                "lun": 2,
                "id": "/subscriptions/x000xx00-0x00-0000-
000x/resourceGroups/ShirleyHa2701-
rg/providers/Microsoft.Compute/disks/ShirleyHa2701-vm2-disk-core",
                "caching": "ReadOnly",
                "accountType": "Standard_LRS",
                "managed": true,
                "encryptionSet": null
            }
        ]
    },
    "reservedSize": {
        "size": 1.0,
        "unit": "GB"
```

```
        },
        "clusterProperties": {
            "lifs": [],
            "serialNumber": "",
            "systemId": "",
            "clusterName": "",
            "ontapVersion": "",
            "accountId": "",
            "productCode": "",
            "amiId": "",
            "systemManagerUrl": "",
            "creationTime": 0,
            "instanceId": "",
            "platformLicense": "",
            "licenseExpiryDate": 0,
            "instanceType": "",
            "publicIp": null,
            "publicDnsName": null,
            "licenseType": {
                "name": "",
                "capacityLimit": {
                    "size": 0.0,
                    "unit": "GB"
                }
            },
            "lastModifiedOffbox": null,
            "offboxTarget": false,
            "upgradeVersions": null,
            "writingSpeedState": null
        },
        "ontapClusterProperties": {
            "nodes": [
                {
                    "name": "ShirleyHa2701-01",
                    "lifs": [
                        {
                            "ip": "10.0.0.29",
                            "netmask": "255.255.255.0",
                            "lifType": "Cluster",
                            "dataProtocols": [],
                            "nodeName": "ShirleyHa2701-01",
                            "privateIp": true
                        },
                        {
                            "ip": "10.0.0.26",
                            "netmask": "255.255.255.0",
```

```json
                "lifType": "Node Management",
                "dataProtocols": [],
                "nodeName": "ShirleyHa2701-01",
                "privateIp": true
            },
            {
                "ip": "10.0.0.13",
                "netmask": "255.255.255.0",
                "lifType": "Cluster Management",
                "dataProtocols": [],
                "nodeName": "ShirleyHa2701-01",
                "privateIp": true
            },
            {
                "ip": "10.0.0.27",
                "netmask": "255.255.255.0",
                "lifType": "Intercluster",
                "dataProtocols": [],
                "nodeName": "ShirleyHa2701-01",
                "privateIp": true
            },
            {
                "ip": "10.0.0.14",
                "netmask": "255.255.255.0",
                "lifType": "Data",
                "dataProtocols": [
                    "nfs",
                    "cifs"
                ],
                "nodeName": "ShirleyHa2701-01",
                "privateIp": true
            },
            {
                "ip": "10.0.0.28",
                "netmask": "255.255.255.0",
                "lifType": "Data",
                "dataProtocols": [
                    "iscsi"
                ],
                "nodeName": "ShirleyHa2701-01",
                "privateIp": true
            },
            {
                "ip": "10.0.0.16",
                "netmask": "255.255.255.0",
                "lifType": "SVM Management",
```

```
                "dataProtocols": [],
                "nodeName": "ShirleyHa2701-01",
                "privateIp": true
            }
        ],
        "serialNumber": "0000000000000000",
        "systemId": "2315255834",
        "platformLicense": null,
        "platformSerialNumber": null,
        "cloudProviderId": "",
        "healthy": true,
        "inTakeover": false
    },
    {
        "name": "ShirleyHa2701-02",
        "lifs": [
            {
                "ip": "10.0.0.24",
                "netmask": "255.255.255.0",
                "lifType": "Cluster",
                "dataProtocols": [],
                "nodeName": "ShirleyHa2701-02",
                "privateIp": true
            },
            {
                "ip": "10.0.0.18",
                "netmask": "255.255.255.0",
                "lifType": "Node Management",
                "dataProtocols": [],
                "nodeName": "ShirleyHa2701-02",
                "privateIp": true
            },
            {
                "ip": "10.0.0.19",
                "netmask": "255.255.255.0",
                "lifType": "Intercluster",
                "dataProtocols": [],
                "nodeName": "ShirleyHa2701-02",
                "privateIp": true
            },
            {
                "ip": "10.0.0.15",
                "netmask": "255.255.255.0",
                "lifType": "Data",
                "dataProtocols": [
                    "nfs",
```

```
                            "cifs"
                        ],
                        "nodeName": "ShirleyHa2701-02",
                        "privateIp": true
                    },
                    {
                        "ip": "10.0.0.20",
                        "netmask": "255.255.255.0",
                        "lifType": "Data",
                        "dataProtocols": [
                            "iscsi"
                        ],
                        "nodeName": "ShirleyHa2701-02",
                        "privateIp": true
                    }
                ],
                "serialNumber": "0000000000000000",
                "systemId": "2315255826",
                "platformLicense": null,
                "platformSerialNumber": null,
                "cloudProviderId": "",
                "healthy": true,
                "inTakeover": false
            }
        ],
        "clusterName": "ShirleyHa2701",
        "clusterUuid": "xxx000000e-xxx00-xxx00-xxx00-xxx00xxx00",
        "ontapVersion": "9.9.0X5",
        "systemManagerUrl": "https://10.0.0.13/sysmgr/SysMgr.html",
        "creationTime": 1611698774849,
        "licenseType": {
            "name": "Cloud Volumes ONTAP BYOL",
            "capacityLimit": {
                "size": 368.0,
                "unit": "TB"
            }
        },
        "lastModifiedOffbox": null,
        "offboxTarget": true,
        "upgradeVersions": null,
        "writingSpeedState": null,
        "broadcastDomainInfos": [
            {
                "broadcastDomain": "Cluster",
                "ipSpace": "Cluster",
                "mtu": 1500
```

```
            },
            {
                "broadcastDomain": "Default",
                "ipSpace": "Default",
                "mtu": 1500
            }
        ],
        "evaluation": false,
        "capacityTierInfo": {
            "capacityTierUsedSize": {
                "size": 0.0,
                "unit": "GB"
            },
            "s3BucketName": "qxtj1b4zpsieeen.blob.core.windows.net",
            "tierLevel": "normal"
        },
        "canConfigureCapacityTier": false,
        "usedCapacity": {
            "size": 9.863281247817213E-4,
            "unit": "TB"
        },
        "userName": "admin",
        "wormEnabled": false
    },
    "cloudProviderName": "Azure",
    "snapshotPolicies": [
        {
            "name": "default",
            "schedules": [
                {
                    "frequency": "hourly",
                    "retention": 6
                },
                {
                    "frequency": "daily",
                    "retention": 2
                },
                {
                    "frequency": "weekly",
                    "retention": 2
                }
            ],
            "description": "Default policy with hourly, daily & weekly
schedules."
        },
        {
```

```json
            "name": "default-1weekly",
            "schedules": [
                {
                    "frequency": "hourly",
                    "retention": 6
                },
                {
                    "frequency": "daily",
                    "retention": 2
                },
                {
                    "frequency": "weekly",
                    "retention": 1
                }
            ],
            "description": "Default policy with 6 hourly, 2 daily & 1
weekly schedule."
        },
        {
            "name": "none",
            "schedules": [],
            "description": "Policy for no automatic snapshots."
        }
    ],
    "actionsRequired": [
        {
            "actionType": "licenseGracePeriod",
            "parameters": {
                "aggregateName": "",
                "numOfDisks": 0,
                "diskSize": null,
                "volumeNames": null,
                "maxCapacity": null,
                "licenseExpiryDate": 1611698969000,
                "serialNumber": "000000000000000000",
                "volumeMoveParameters": null,
                "workingEnvironmentId": "",
                "licenseParameters": null,
                "resourcesToDelete": null,
                "instances": [],
                "moreInfo": null,
                "providerVolumeType": null,
                "volumeInfo": null,
                "currentInstanceType": null
            },
            "severity": "error"
```

```
        },
        {
            "actionType": "licenseGracePeriod",
            "parameters": {
                "aggregateName": "",
                "numOfDisks": 0,
                "diskSize": null,
                "volumeNames": null,
                "maxCapacity": null,
                "licenseExpiryDate": 1611699072000,
                "serialNumber": "000000000000000000000",
                "volumeMoveParameters": null,
                "workingEnvironmentId": "",
                "licenseParameters": null,
                "resourcesToDelete": null,
                "instances": [],
                "moreInfo": null,
                "providerVolumeType": null,
                "volumeInfo": null,
                "currentInstanceType": null
            },
            "severity": "error"
        }
    ],
    "activeActions": [],
    "replicationProperties": {
        "peers": [],
        "replicationTargets": []
    },
    "schedules": [
        {
            "name": "day",
            "schedule": [
                {
                    "stop": {
                        "day": 0,
                        "hour": 20,
                        "minute": 0
                    },
                    "start": {
                        "day": 1,
                        "hour": 8,
                        "minute": 0
                    }
                },
                {
```

```
                "stop": {
                    "day": 1,
                    "hour": 20,
                    "minute": 0
                },
                "start": {
                    "day": 2,
                    "hour": 8,
                    "minute": 0
                }
            },
            {
                "stop": {
                    "day": 2,
                    "hour": 20,
                    "minute": 0
                },
                "start": {
                    "day": 3,
                    "hour": 8,
                    "minute": 0
                }
            },
            {
                "stop": {
                    "day": 3,
                    "hour": 20,
                    "minute": 0
                },
                "start": {
                    "day": 4,
                    "hour": 8,
                    "minute": 0
                }
            },
            {
                "stop": {
                    "day": 4,
                    "hour": 20,
                    "minute": 0
                },
                "start": {
                    "day": 5,
                    "hour": 8,
                    "minute": 0
                }
```

```
                },
                {
                    "stop": {
                        "day": 5,
                        "hour": 20,
                        "minute": 0
                    },
                    "start": {
                        "day": 6,
                        "hour": 8,
                        "minute": 0
                    }
                },
                {
                    "stop": {
                        "day": 6,
                        "hour": 20,
                        "minute": 0
                    },
                    "start": {
                        "day": 0,
                        "hour": 8,
                        "minute": 0
                    }
                }
            ],
            "enabled": false
        },
        {
            "name": "Weekend",
            "schedule": [
                {
                    "stop": {
                        "day": 6,
                        "hour": 8,
                        "minute": 0
                    },
                    "start": {
                        "day": 1,
                        "hour": 8,
                        "minute": 0
                    }
                }
            ],
            "enabled": false
        },
```

```
        {
            "name": "Weekdays",
            "schedule": [
                {
                    "stop": {
                        "day": 1,
                        "hour": 20,
                        "minute": 0
                    },
                    "start": {
                        "day": 2,
                        "hour": 8,
                        "minute": 0
                    }
                },
                {
                    "stop": {
                        "day": 2,
                        "hour": 20,
                        "minute": 0
                    },
                    "start": {
                        "day": 3,
                        "hour": 8,
                        "minute": 0
                    }
                },
                {
                    "stop": {
                        "day": 3,
                        "hour": 20,
                        "minute": 0
                    },
                    "start": {
                        "day": 4,
                        "hour": 8,
                        "minute": 0
                    }
                },
                {
                    "stop": {
                        "day": 4,
                        "hour": 20,
                        "minute": 0
                    },
                    "start": {
```

```json
                    "day": 5,
                    "hour": 8,
                    "minute": 0
                }
            },
            {
                "stop": {
                    "day": 5,
                    "hour": 20,
                    "minute": 0
                },
                "start": {
                    "day": 6,
                    "hour": 8,
                    "minute": 0
                }
            }
        ],
        "enabled": false
    }
],
"svms": [
    {
        "name": "svm_ShirleyHa2701",
        "state": "running",
        "language": "c.utf_8",
        "allowedAggregates": [
            "aggr1"
        ],
        "ver3Enabled": true,
        "ver4Enabled": true
    }
],
"workingEnvironmentType": "VSA",
"supportRegistrationProperties": {
    "supportRegistrationStatus": "registered",
    "licenseExpiryDate": 1604102400000
},
"supportRegistrationInformation": [
    {
        "supportRegistrationStatus": "registered",
        "serialNumber": "00000000000000",
        "licenseExpiryDate": 1611698969000,
        "cloudLicenseExists": true,
        "nssAccountId": "x0x0x000-0000-0000x-00de-x000xxxx00000"
    },
```

```
        {
            "supportRegistrationStatus": "registered",
            "serialNumber": "0000000001111",
            "licenseExpiryDate": 1611699072000,
            "cloudLicenseExists": true,
            "nssAccountId": "x0x0x000-0000-0000x-00de-x000xxxx00000"
        }
    ],
    "capacityFeatures": {
        "providerVolumesType": [
            {
                "size": {
                    "size": 500.0,
                    "unit": "GB"
                },
                "supportedVolumeTypes": [
                    "Premium_LRS"
                ],
                "maxDisksAllow": {
                    "numOfDisks": 12,
                    "reason": null
                }
            },
            {
                "size": {
                    "size": 1.0,
                    "unit": "TB"
                },
                "supportedVolumeTypes": [
                    "Premium_LRS"
                ],
                "maxDisksAllow": {
                    "numOfDisks": 12,
                    "reason": null
                }
            },
            {
                "size": {
                    "size": 2.0,
                    "unit": "TB"
                },
                "supportedVolumeTypes": [
                    "Premium_LRS"
                ],
                "maxDisksAllow": {
                    "numOfDisks": 12,
```

```
                    "reason": null
                }
            },
            {
                "size": {
                    "size": 4.0,
                    "unit": "TB"
                },
                "supportedVolumeTypes": [
                    "Premium_LRS"
                ],
                "maxDisksAllow": {
                    "numOfDisks": 12,
                    "reason": null
                }
            },
            {
                "size": {
                    "size": 8.0,
                    "unit": "TB"
                },
                "supportedVolumeTypes": [
                    "Premium_LRS"
                ],
                "maxDisksAllow": {
                    "numOfDisks": 12,
                    "reason": null
                }
            }
        ],
        "defaultProviderVolumeType": {
            "size": {
                "size": 500.0,
                "unit": "GB"
            },
            "diskType": "Premium_LRS",
            "capacityTier": "Blob",
            "iops": null
        },
        "supportedCapacityTiers": {
            "supportedCapacityTiersPerVolumeType": [
                {
                    "volumeType": "Premium_LRS",
                    "supportedCapacityTiers": [
                        "Blob"
                    ],
```

```
                    "availableTieringPolicies": [
                        "none",
                        "snapshot_only",
                        "auto",
                        "all"
                    ]
                }
            ],
            "capacityTiersDisableReasons": [],
            "compositeSupported": true,
            "forceCompositeVersion": true
        },
        "maxDisksPerAggregate": 12,
        "existingIops": []
    },
    "encryptionProperties": {
        "ontapEncryption": false,
        "awsVolumeEncryption": false,
        "azureVolumeEncryption": true,
        "gcpVolumeEncryption": false,
        "keyManagers": [],
        "encryptionCertificates": [],
        "awsEncryptionKey": null
    },
    "supportedFeatures": {
        "supportsMixedAggregates": false,
        "supportsTieringWithServiceAccount": false
    },
    "isHA": true,
    "haProperties": {
        "loadBalancerName": "ShirleyHa2701-rg-lb",
        "node1Info": {
            "instanceName": "ShirleyHa2701-vm1",
            "instanceId": null,
            "primaryIp": "10.0.0.26",
            "state": "running",
            "serialNumber": "000000000000000",
            "availabilitySet": {
                "faultDomain": 0,
                "updateDomain": 0
            }
        },
        "node2Info": {
            "instanceName": "ShirleyHa2701-vm2",
            "instanceId": null,
            "primaryIp": "10.0.0.18",
```

```json
            "state": "running",
            "serialNumber": "00000000000000000000000",
            "availabilitySet": {
                "faultDomain": 1,
                "updateDomain": 1
            }
        }
    },
    "k8sProperties": {
        "isConnected": false,
        "mainClusterIds": [],
        "connectedClusterIds": []
    },
    "fpolicyProperties": {
        "status": 0,
        "fPolicyProtocolStatus": {
            "nfsv3": 0,
            "nfsv4": 0,
            "cifs": -1
        },
        "fileExtensions": [
            "micro",
            "encrypted",
            "locked",
            "crypto",
            "crypt",
            "crinf",
            "r5a",
            "XRNT",
            "XTBL",
            "R16M01D05",
            "pzdc",
            "good",
            "LOL!",
            "OMG!",
            "RDM",
            "RRK",
            "encryptedRS",
            "crjoker",
            "EnCiPhErEd",
            "LeChiffre"
        ]
    },
    "saasProperties": {
        "subscription": null,
        "freeTrialExpiry": null,
```

```
            "saasEnabled": false
        },
        "cbsProperties": {
            "cbsBackupStatus": "off",
            "cbsRules": [],
            "numberOfBackedUpVolumes": 0,
            "objectStoreName": null,
            "providerSpecific": null,
            "cbsPolicyName": null,
            "usedCapacity": null,
            "ipSpace": null,
            "region": null,
            "providerAccountName": null,
            "exclusionReason": null
        },
        "complianceProperties": {
            "scanStatus": "SCAN_DISABLED",
            "complianceStatus": null,
            "lastDeploymentError": null,
            "complianceBackupStatus": null
        },
        "monitoringProperties": {
            "monitoringStatus": "MONITORING_DISABLED",
            "monitoringInfo": null,
            "tenantUrl": null
        }
    }
```

Delete a working environment

You can delete an existing Azure Cloud Volumes ONTAP working environment.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Delete working environment for single node

You can use this workflow to delete a single node working environment.

1. Create the working environment to use

Perform the workflow Create Azure single node working environment and choose the `publicId` value of the working environment for the `workingEnvironmentId` path parameter.

2. Delete the working environment

HTTP method	Path
DELETE	/occm/api/azure/vsa/working-environments/{workingEnvironmentId}

curl example

```
curl --location --request DELETE
'https://cloudmanager.cloud.netapp.com/occm/api/azure/vsa/working-
environments/<WORKING_ENV_ID>' --header 'Content-Type: application/json'
--header 'x-agent-id: <AGENT_ID>' --header 'Authorization: Bearer
<ACCESS_TOKEN>'
```

Input

Path parameter <WORKING_ENV_ID> (`workingEnvironmentId`) string

(Optional) Query parameters:

- `localDelete` boolean

  If `true` the Cloud Volumes ONTAP instance in the cloud is not terminated, but Cloud Manager no longer manages it (default is `false`).

- `forceDelete` boolean

  If `true` the working environment is deleted even if it is part of one or more SnapMirror relationships (default is `false`).

  Output

  None

Delete working environment for high availability pair

You can use this workflow to delete an HA working environment.

1. Create the working environment to use

Perform the workflow Create Azure HA working environment and choose the `publicId` value of the working environment for the `workingEnvironmentId` path parameter.

2. Delete the working environment

HTTP method	Path
DELETE	/occm/api/azure/ha/working-environments/{workingEnvironmentId}

curl example

```
curl --location --request DELETE
'https://cloudmanager.cloud.netapp.com/occm/api/azure/ha/working-
environments/<WORKING_ENV_ID>' --header 'Content-Type: application/json'
--header 'x-agent-id: <AGENT_ID>' --header 'Authorization: Bearer
<ACCESS_TOKEN>'
```

Input

Path parameter <WORKING_ENV_ID> `(workingEnvironmentId)` string

(Optional) Query parameters:

- `localDelete` boolean

  If `true` the Cloud Volumes ONTAP instance in the cloud is not terminated, but Cloud Manager no longer manages it (default is `false`).

- `forceDelete` boolean

  If `true` the working environment is deleted even if it is part of one or more SnapMirror relationships (default is `false`).

  Output
  None

Create CIFS server configuration

If you want to create CIFS volumes on your Cloud Volumes ONTAP system, you first need to configure the CIFS server. You can choose to set up the CIFS server in a workgroup or in an Active Directory domain. Review the ONTAP documentation for more information.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Create CIFS server configuration for single node

You can use this workflow to create CIFS server configuration for single node system.

Choose the workflow that is specific to your goal:

- Set up a CIFS server in a workgroup
- Set up a CIFS server in an Active Directory domain

Set up a CIFS server in a workgroup

You can configure a CIFS server in a workgroup when the Microsoft Active Directory domain infrastructure is not available.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value for the working environment used in the `workingEnvironmentId` path parameter.

2. Create the CIFS configuration

Create the CIFS server configuration.

HTTP method	Path
POST	/occm/api/azure/vsa/working-environments/{workingEnvironmentId}/cifs-workgroup

curl example

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

Input
- Path parameter <WORKING_ENV_ID> `workingEnvironmentId` string

JSON input example

```
{
  "serverName": "SMB_SERVER02",
  "workgroupName": "workgroup02",
  "svmName": "svm_ziv01we01"
}
```

Output

None.

Set up a CIFS server in an Active Directory domain

You can create a CIFS server on the SVM and specify the Active Directory (AD) domain to which it belongs.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value for the working environment used in the `workingEnvironmentId` path parameter.

2. Determine the Active Directory configuration

You need the following configuration parameters for an Active Directory server.

Input parameter	Description
dnsDomain	Use the Active Directory domain as the DNS name.
ipAddresses	Define the primary DNS IP address and optionally add a secondary IP address.
netBIOS	Use the CIFS server NetBIOS name.
organizationalUnit	Include the organizational unit as appropriate.
activeDirectoryDomain	Set the Active Directory domain to join.
activeDirectoryUsername	A username with authorization to join the domain.
activeDirectoryPassword	The password for the authorized username.

3. Create the CIFS configuration

Create the CIFS server configuration.

HTTP method	Path
POST	/occm/api/azure/vsa/working-environments/{workingEnvironmentId}/cifs

curl example

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

Input

- Path parameter <WORKING_ENV_ID> `workingEnvironmentId` string

JSON input example

```
{
  "dnsDomain": "zivh.netapp.com",
  "ipAddresses": [
    "172.31.5.241"
  ],
  "netBIOS": "zivaws02we03",
  "organizationalUnit": "CN=Computers",
  "activeDirectoryDomain": "zivh.netapp.com",
  "activeDirectoryUsername": "administrator",
  "activeDirectoryPassword": "password"
}
```

Output

None.

Create CIFS server configuration for high availability pair

You can use this workflow to create to create CIFS server configuration for an HA working environment.

Choose the workflow that is specific to your goal:

- Set up a CIFS server in a workgroup
- Set up a CIFS server in an Active Directory domain

Set up a CIFS server in a workgroup

You can configure a CIFS server in a workgroup when the Microsoft Active Directory domain infrastructure is not available.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value for the working environment used in the `workingEnvironmentId` path parameter.

2. Create the CIFS configuration

Create the CIFS server configuration.

HTTP method	Path
POST	/occm/api/azure/ha/working-environments/{workingEnvironmentId}/cifs-workgroup

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/azure/ha/working-
environments/<WORKING_ENV_ID>/cifs-workgroup' --header 'Content-Type:
application/json' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>' --d @JSONinput
```

Input

- Path parameter <WORKING_ENV_ID> `workingEnvironmentId` string

JSON input example

```
{
  "serverName": "SMB_SERVER02",
  "workgroupName": "workgroup02",
  "svmName": "svm_ziv01we01"
}
```

Output

None.

Set up a CIFS server in an Active Directory domain

You can create a CIFS server on the SVM and specify the Active Directory (AD) domain to which it belongs.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value for the working environment used in the `workingEnvironmentId` path parameter.

2. Determine the Active Directory configuration

You need the following configuration parameters for an Active Directory server.

Input parameter	Description
dnsDomain	Use the Active Directory domain as the DNS name.
ipAddresses	Define the primary DNS IP address and optionally add a secondary IP address.
netBIOS	Use the CIFS server NetBIOS name.
organizationalUnit	Include the organizational unit as appropriate.
activeDirectoryDomain	Set the Active Directory domain to join.
activeDirectoryUsername	A username with authorization to join the domain.
activeDirectoryPassword	The password for the authorized username.

3. Create the CIFS configuration

Create the CIFS server configuration.

HTTP method	Path
POST	/occm/api/azure/ha/working-environments/{workingEnvironmentId}/cifs

curl example

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

Input

- Path parameter <WORKING_ENV_ID> `workingEnvironmentId` string

JSON input example

```json
{
    "dnsDomain": "mydomain.com",
    "activeDirectoryDomain": "mydomain.com",
    "ipAddresses": ["10.10.10.20", "172.xx.yy.xx"],
    "netBIOS": "azureHAPayGo",
    "organizationalUnit": "CN=Computers",
    "activeDirectoryUsername": "administrator",
    "activeDirectoryPassword": "password",
    "svmName": "svm_azureHAPayGo"
}
```

Output

None.

Get CIFS server configurations

You can use this workflow to retrieve the CIFS server configurations for an existing Cloud Volumes ONTAP working environment.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Get CIFS server configuration for single node

You can use this workflow to retrieve CIFS server configuration for a single node system.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value for the working environment used in the `workingEnvironmentId` path parameter.

2. Get the CIFS configurations

HTTP method	Path
GET	/occm/api/azure/vsa/working-environments/{workingEnvironmentId}/cifs

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/azure/vsa/working-
environments/<WORKING_ENV_ID>/cifs' --header 'Content-Type:
application/json' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

- Path parameter <WORKING_ENV_ID> `workingEnvironmentId` string

- (Optional) Query parameter `svm` string

Output

The JSON output example includes the CIFS configurations for an existing Cloud Volumes ONTAP ONTAP working environment.

JSON output example

```
[
    {
        "dnsDomain": "zivh.netapp.com",
        "activeDirectoryDomain": "zivh.netapp.com",
        "ipAddresses": [
            "172.31.5.241"
        ],
        "netBIOS": "zivaws02we01",
        "organizationalUnit": "CN=Computers",
        "authenticationType": "domain"
    }
]
```

Get CIFS server configuration for high availability pair

You can use this workflow to retrieve CIFS server configuration for an HA working environment.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value for the working environment used in the `workingEnvironmentId` path parameter.

2. Get the CIFS configurations

HTTP method	Path
GET	/occm/api/azure/ha/working-environments/{workingEnvironmentId}/cifs

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/azure/ha/working-
environments/<WORKING_ENV_ID>/cifs' --header 'Content-Type:
application/json' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

- Path parameter <WORKING_ENV_ID> `workingEnvironmentId` string
- (Optional) Query parameter `svm` string

Output

The JSON output example includes the CIFS configurations for an existing Cloud Volumes ONTAP working environment.

JSON output example

```
[
  {
   "dnsDomain": "mydomain.com",
   "activeDirectoryDomain": "mydomain.com",
   "ipAddresses": ["10.10.10.20", "172.xx.yy.xx"],
   "netBIOS": "azureHAPayGo",
   "organizationalUnit": "CN=Computers",
   "activeDirectoryUsername": "administrator",
   "activeDirectoryPassword": "password",
   "svmName": "svm_azureHAPayGo"
  }
]
```

Delete CIFS server configuration

You can use this workflow to delete a CIFS server configuration for an existing Cloud Volumes ONTAP working environment.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Delete CIFS server configuration for single node

You can use this workflow to delete CIFS server configuration for a single node system.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value for the working environment used in the `workingEnvironmentId` path parameter.

2. Delete the CIFS configurations

HTTP method	Path
POST	/occm/api/azure/vsa/working-environments/{workingEnvironmentId}/delete-cifs

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/azure/vsa/working-
environments/<WORKING_ENV_ID>/delete-cifs' --header 'Content-Type:
application/json' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

- Path parameter <WORKING_ENV_ID> `workingEnvironmentId` string

- Optional JSON body

```
{
    "activeDirectoryUsername": "string",
    "activeDirectoryPassword": "string",
    "svmName": "string"
}
```

Output

None.

Delete CIFS server configuration for high availability pair

You can use this workflow to delete CIFS server configuration for an HA working environment.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value for the working environment used in the `workingEnvironmentId` path parameter.

2. Delete the CIFS configurations

HTTP method	Path
POST	/occm/api/azure/ha/working-environments/{workingEnvironmentId}/delete-cifs

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/azure/ha/working-
environments/<WORKING_ENV_ID>/delete-cifs' --header 'Content-Type:
application/json' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

- Path parameter <WORKING_ENV_ID> `workingEnvironmentId` string

- Optional JSON body

```
{
  "activeDirectoryUsername": "string",
  "activeDirectoryPassword": "string",
  "svmName": "string"
}
```

Output

None.

Aggregates

Get aggregates

You can retrieve a list of available disk aggregates of an Azure working environment.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Get aggregates for single node

You can use this workflow to retrieve the aggregates for a single node working environment.

1. Create the working environment

Perform the workflow Create Azure single node working environment and choose the `publicId` value of the working environment for the `workingEnvironmentId` path parameter.

2. Get the list of aggregates

HTTP method	Path
GET	/occm/api/azure/vsa/aggregates/{workingEnvironmentId}

curl example

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

Input

Path parameter:

- <WORKING_ENV_ID> (`workingEnvironmentId`) string

Output

An array of aggregates for the indicated working environment is returned as shown in the JSON output example.

> ⓘ The capacity (sizes) in the output are in MB/GB/TB (1000th order) because these are ONTAP aggregates, whereas in Cloud Manager the capacity is specified as MiB, GiB (1024 order).

JSON output example

```
[
    {
        "name": "aggr1",
        "availableCapacity": {
            "size": 905.27,
            "unit": "GB"
        },
        "totalCapacity": {
            "size": 906.29,
            "unit": "GB"
        },
        "usedCapacity": {
            "size": 1.02,
            "unit": "GB"
        },
        "volumes": [
            {
                "name": "svm_ShirleyAzureVsa2601_root",
                "totalSize": {
                    "size": 1.0,
                    "unit": "GB"
                },
                "usedSize": {
                    "size": 3.62396240234375E-4,
                    "unit": "GB"
                },
                "thinProvisioned": false,
                "isClone": false,
                "rootVolume": true
            }
        ],
        "providerVolumes": [
            {
                "id": "/subscriptions/x000xx00-0x00-0000-
000x/resourceGroups/ShirleyAzureVsa2601-
rg/providers/Microsoft.Compute/disks/ShirleyAzureVsa2601datadisk3",
                "name": "ShirleyAzureVsa2601datadisk3",
                "size": {
```

```
                    "size": 1.0,
                    "unit": "TB"
                },
                "state": "available",
                "device": "3",
                "instanceId": "/subscriptions/x000xx00-0x00-0000-
000x/resourceGroups/ShirleyAzureVsa2601-
rg/providers/Microsoft.Compute/virtualMachines/ShirleyAzureVsa2601",
                "diskType": "Premium_LRS",
                "encrypted": false,
                "iops": null
            }
        ],
        "disks": [
            {
                "name": "NET-1.2",
                "position": "data",
                "ownerNode": "ShirleyAzureVsa2601-01",
                "device": "LUN 5.3",
                "vmDiskProperties": null
            }
        ],
        "state": "online",
        "encryptionType": "notEncrypted",
        "encryptionKeyId": null,
        "isRoot": false,
        "homeNode": "ShirleyAzureVsa2601-01",
        "ownerNode": "ShirleyAzureVsa2601-01",
        "capacityTier": null,
        "capacityTierUsed": null,
        "sidlEnabled": false,
        "snaplockType": "non_snaplock"
    },
    {
        "name": "aggr2",
        "availableCapacity": {
            "size": 906.29,
            "unit": "GB"
        },
        "totalCapacity": {
            "size": 906.29,
            "unit": "GB"
        },
        "usedCapacity": {
            "size": 3.48,
            "unit": "MB"
```

```json
        },
        "volumes": [],
        "providerVolumes": [
            {
                "id": "/subscriptions/x000xx00-0x00-0000-
000x/resourceGroups/ShirleyAzureVsa2601-
rg/providers/Microsoft.Compute/disks/ShirleyAzureVsa2601datadisk4",
                "name": "ShirleyAzureVsa2601datadisk4",
                "size": {
                    "size": 1.0,
                    "unit": "TB"
                },
                "state": "available",
                "device": "4",
                "instanceId": "/subscriptions/x000xx00-0x00-0000-
000x/resourceGroups/ShirleyAzureVsa2601-
rg/providers/Microsoft.Compute/virtualMachines/ShirleyAzureVsa2601",
                "diskType": "Premium_LRS",
                "encrypted": false,
                "iops": null
            }
        ],
        "disks": [
            {
                "name": "NET-1.3",
                "position": "data",
                "ownerNode": "ShirleyAzureVsa2601-01",
                "device": "LUN 5.4",
                "vmDiskProperties": null
            }
        ],
        "state": "online",
        "encryptionType": "notEncrypted",
        "encryptionKeyId": null,
        "isRoot": false,
        "homeNode": "ShirleyAzureVsa2601-01",
        "ownerNode": "ShirleyAzureVsa2601-01",
        "capacityTier": null,
        "capacityTierUsed": null,
        "sidlEnabled": false,
        "snaplockType": "non_snaplock"
    },
    {
        "name": "aggr3",
        "availableCapacity": {
            "size": 1.77,
```

```json
            "unit": "TB"
        },
        "totalCapacity": {
            "size": 1.77,
            "unit": "TB"
        },
        "usedCapacity": {
            "size": 5.78,
            "unit": "MB"
        },
        "volumes": [],
        "providerVolumes": [
            {
                "id": "/subscriptions/x000xx00-0x00-0000-
000x/resourceGroups/ShirleyAzureVsa2601-
rg/providers/Microsoft.Compute/disks/ShirleyAzureVsa2601datadisk5",
                "name": "ShirleyAzureVsa2601datadisk5",
                "size": {
                    "size": 1.0,
                    "unit": "TB"
                },
                "state": "available",
                "device": "5",
                "instanceId": "/subscriptions/x000xx00-0x00-0000-
000x/resourceGroups/ShirleyAzureVsa2601-
rg/providers/Microsoft.Compute/virtualMachines/ShirleyAzureVsa2601",
                "diskType": "Premium_LRS",
                "encrypted": false,
                "iops": null
            },
            {
                "id": "/subscriptions/x000xx00-0x00-0000-
000x/resourceGroups/ShirleyAzureVsa2601-
rg/providers/Microsoft.Compute/disks/ShirleyAzureVsa2601datadisk6",
                "name": "ShirleyAzureVsa2601datadisk6",
                "size": {
                    "size": 1.0,
                    "unit": "TB"
                },
                "state": "available",
                "device": "6",
                "instanceId": "/subscriptions/x000xx00-0x00-0000-
000x/resourceGroups/ShirleyAzureVsa2601-
rg/providers/Microsoft.Compute/virtualMachines/ShirleyAzureVsa2601",
                "diskType": "Premium_LRS",
                "encrypted": false,
```

```
                "iops": null
            }
        ],
        "disks": [
            {
                "name": "NET-1.4",
                "position": "data",
                "ownerNode": "ShirleyAzureVsa2601-01",
                "device": "LUN 5.5",
                "vmDiskProperties": null
            },
            {
                "name": "NET-1.5",
                "position": "data",
                "ownerNode": "ShirleyAzureVsa2601-01",
                "device": "LUN 5.6",
                "vmDiskProperties": null
            }
        ],
        "state": "online",
        "encryptionType": "notEncrypted",
        "encryptionKeyId": null,
        "isRoot": false,
        "homeNode": "ShirleyAzureVsa2601-01",
        "ownerNode": "ShirleyAzureVsa2601-01",
        "capacityTier": null,
        "capacityTierUsed": null,
        "sidlEnabled": false,
        "snaplockType": "non_snaplock"
    }
]
```

Get aggregates for high availability pair

You can use this workflow to retrieve the aggregates for an HA working environment.

1. Create the working environment

Perform the workflow Create Azure HA working environment and choose the `publicId` value of the working environment for the `workingEnvironmentId` path parameter.

2. Get the list of aggregates

HTTP method	Path
GET	/occm/api/azure/ha/aggregates/{workingEnvironmentId}

curl example

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

Input

Path parameter:

- <WORKING_ENV_ID> (workingEnvironmentId) string

Output

An array of aggregates for the indicated working environment is returned as shown in the JSON output example.

ⓘ   The capacity (sizes) in the output are in MB/GB/TB (1000th order) because these are ONTAP aggregates, whereas in Cloud Manager the capacity is specified as MiB, GiB (1024 order).

JSON output example

```
[
    {
        "name": "aggr1",
        "availableCapacity": {
            "size": 906.16,
            "unit": "GB"
        },
        "totalCapacity": {
            "size": 907.18,
            "unit": "GB"
        },
        "usedCapacity": {
            "size": 1.01,
            "unit": "GB"
        },
        "volumes": [
            {
                "name": "svm_ShirleyHa2801_root",
                "totalSize": {
                    "size": 1.0,
                    "unit": "GB"
                },
                "usedSize": {
                    "size": 3.24249267578125E-4,
                    "unit": "GB"
                },
```

```json
            "thinProvisioned": false,
            "isClone": false,
            "rootVolume": true
        }
    ],
    "providerVolumes": [
        {
            "id": "ki4cw3n3oyha",
            "name": "ki4cw3n3oyha",
            "size": {
                "size": 1.0,
                "unit": "TB"
            },
            "state": "available",
            "device": "",
            "instanceId": "",
            "diskType": "Premium_LRS",
            "encrypted": true,
            "iops": null
        }
    ],
    "disks": [
        {
            "name": "NET-1.3",
            "position": "data",
            "ownerNode": "ShirleyHa2801-01",
            "device": "",
            "vmDiskProperties": {
                "objectName": "ki4cw3n3oyha",
                "storageAccountName": "rootsacnqfypfg",
                "containerName": "blobcontainer"
            }
        }
    ],
    "state": "online",
    "encryptionType": "cloudEncrypted",
    "encryptionKeyId": null,
    "isRoot": false,
    "homeNode": "ShirleyHa2801-01",
    "ownerNode": "ShirleyHa2801-01",
    "capacityTier": "Blob",
    "capacityTierUsed": {
        "size": 0.0,
        "unit": "GB"
    },
    "sidlEnabled": true,
```

```json
            "snaplockType": "non_snaplock"
        },
        {
            "name": "aggr2",
            "availableCapacity": {
                "size": 907.18,
                "unit": "GB"
            },
            "totalCapacity": {
                "size": 907.18,
                "unit": "GB"
            },
            "usedCapacity": {
                "size": 500.0,
                "unit": "KB"
            },
            "volumes": [],
            "providerVolumes": [
                {
                    "id": "1102qyj51rwt",
                    "name": "1102qyj51rwt",
                    "size": {
                        "size": 1.0,
                        "unit": "TB"
                    },
                    "state": "available",
                    "device": "",
                    "instanceId": "",
                    "diskType": "Premium_LRS",
                    "encrypted": true,
                    "iops": null
                }
            ],
            "disks": [
                {
                    "name": "NET-1.4",
                    "position": "data",
                    "ownerNode": "ShirleyHa2801-01",
                    "device": "",
                    "vmDiskProperties": {
                        "objectName": "1102qyj51rwt",
                        "storageAccountName": "rootsacnqfypfg",
                        "containerName": "blobcontainer"
                    }
                }
            ],
```

```
        "state": "online",
        "encryptionType": "cloudEncrypted",
        "encryptionKeyId": null,
        "isRoot": false,
        "homeNode": "ShirleyHa2801-01",
        "ownerNode": "ShirleyHa2801-01",
        "capacityTier": "Blob",
        "capacityTierUsed": {
            "size": 0.0,
            "unit": "GB"
        },
        "sidlEnabled": true,
        "snaplockType": "non_snaplock"
    }
]
```

Create aggregate

You can create a new aggregate within an Azure working environment using this workflow.

1. Create the working environment

Perform the workflow Create Azure single node working environment and choose the `publicId` value for the `workingEnvironmentId` parameter in the JSON input.

2. Select the Azure Storage Account Types

Perform the Get Azure Storage Account Types workflow and choose the desired storage type:

- Choose the `diskType` of the desired storage type item for the `providerVolumeType` parameter in the JSON input.

- Choose the `size` value of the desired disk from the `sizes → size` field for the `size` field inside `diskSize` parameter in the JSON input.

- Choose the `unit` value of the desired disk from the `sizes → size` field for the `unit` field inside `diskSize` parameter in the JSON input.

3. Create the aggregate

HTTP method	Path
POST	occm/api/azure/vsa/aggregates

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/azure/vsa/aggregates'
--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.

JSON input example

```json
{
  "name": "aggr3",
  "workingEnvironmentId": "VsaWorkingEnvironment-IsDYFJf8",
  "numberOfDisks": "2",
  "diskSize": {
    "size": "1",
    "unit": "TB"
  },
  "providerVolumeType": "Premium_LRS"
}
```

Output

None

Add disks to aggregate

You can add disks to an existing aggregate.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Add disks to an aggregate for single node

You can use this workflow to add disks to an aggregate for a single node working environment.

1. Create the working environment

Perform the workflow Create Azure single node working environment and choose the `publicId` value for the `workingEnvironmentId` path parameter.

2. Create the aggregate

Perform the workflow Create aggregate to create an aggregate with the name `aggr2` and choose `aggr2` for the `aggregateName` path parameter.

3. Add the disks to the aggregate

HTTP method	Path
POST	/occm/api/azure/vsa/aggregates/{workingEnvironmentId}/{aggregateName}/disks

curl example

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

Input

You must include the following path parameters:

- <WORKING_ENV_ID> (workingEnvironmentId) string
- <AGGR_NAME> (aggregateName) string

Also, the JSON input example includes an input parameter as shown.

JSON input example

```
{
    "numberOfDisks": "1"
}
```

Output

None

Add disks to an aggregate for high availability pair

You can use this workflow to add disks to an aggregate for HA working environment.

1. Create the working environment

Perform the workflow Create Azure HA working environment and choose the `publicId` value for the `workingEnvironmentId` path parameter.

2. Create the aggregate

Perform the workflow Create aggregate to create an aggregate with the name `aggr2` and choose `aggr2` for the `aggregateName` path parameter.

3. Add the disks to the aggregate

HTTP method	Path
POST	/occm/api/azure/ha/aggregates/{workingEnvironmentId}/{aggregateName}/disks

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/azure/ha/aggregates/<WORKI
NG_ENV_ID>/<AGGR_NAME>/disks' --header 'Content-Type: application/json'
--header 'x-agent-id: <AGENT_ID>' --header 'Authorization: Bearer
<ACCESS_TOKEN>' --d @JSONinput
```

Input

You must include the following path parameters:

- <WORKING_ENV_ID> (`workingEnvironmentId`) string
- <AGGR_NAME> (`aggregateName`) string

Also, the JSON input example includes an input parameter as shown.

JSON input example

```
{
    "numberOfDisks": "2"
}
```

Output

None

Delete aggregate

You can delete an existing disk aggregate in an Azure working environment.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Delete aggregate for single node

You can use this workflow to delete an aggregate for a single node working environment.

1. Create the working environment

Perform the workflow Create Azure single node working environment and choose the `publicId` value of the working environment for the `workingEnvironmentId` path parameter.

2. Create an aggregate to delete

Perform the workflow Create aggregate to create an aggregate with the name `aggr2`. Use `aggr2` value for the `aggregateName` path parameter.

3. Delete the aggregate

HTTP method	Path
DELETE	/occm/api/azure/vsa/aggregates/{workingEnvironmentId}/{aggregateName}

curl example

```
curl --location --request DELETE
'https://cloudmanager.cloud.netapp.com/occm/api/azure/vsa/aggregates/<WORK
ING_ENV_ID>/<AGGR_NAME>' --header 'Content-Type: application/json'
--header 'x-agent-id: <AGENT_ID>' --header 'Authorization: Bearer
<ACCESS_TOKEN>'
```

Input

Path parameters:

- <WORKING_ENV_ID> (`workingEnvironmentId`) string
- <AGGR_NAME> (`aggregateName`) string

Output

None

Delete aggregate for high availability pair

You can use this workflow to delete an aggregate for an HA working environment.

1. Create the working environment

Perform the workflow Create Azure HA working environment and choose the `publicId` value of the working environment for the `workingEnvironmentId` path parameter.

2. Create an aggregate to delete

Perform the workflow Create aggregate to create an aggregate with the name `aggr2`. Use `aggr2` value for the `aggregateName` path parameter.

3. Delete the aggregate

HTTP method	Path
DELETE	/occm/api/azure/ha/aggregates/{workingEnvironmentId}/{aggregateName}

curl example

```
curl --location --request DELETE
'https://cloudmanager.cloud.netapp.com/occm/api/azure/ha/aggregates/<WORKI
NG_ENV_ID>/<AGGR_NAME>' --header 'Content-Type: application/json' --header
'x-agent-id: <AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

Path parameters:

- <WORKING_ENV_ID> (`workingEnvironmentId`) string
- <AGGR_NAME> (`aggregateName`) string

Output

None

Delete aggregate for high availability pair

You can use this workflow to delete an aggregate for an HA working environment.

1. Create the working environment

Perform the workflow Create Azure HA working environment and choose the `publicId` value of the working environment for the `workingEnvironmentId` path parameter.

2. Create an aggregate to delete

Perform the workflow Create aggregate to create an aggregate with the name `aggr2`. Use `aggr2` value for the `aggregateName` path parameter.

3. Delete the aggregate

HTTP method	Path
DELETE	/occm/api/azure/ha/aggregates/{workingEnvironmentId}/{aggregateName}

curl example

```
curl --location --request DELETE
'https://cloudmanager.cloud.netapp.com/occm/api/azure/ha/aggregates/<WORKI
NG_ENV_ID>/aggr2' --header 'Content-Type: application/json' --header 'x-
agent-id: <AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

Path parameters:

- <WORKING_ENV_ID> (`workingEnvironmentId`) string
- <AGGR_NAME> (`aggregateName`) string

Output

None

Volumes

Create a volume using NFS

You can use this workflow to create a volume accessed through NFS protocol.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Create volume using NFS for single node

You can use this workflow to create a volume using NFS protocol for a single node working environment.

1. Select the working environment

Perform the workflow Create working environments and choose the `publicId` value for the `workingEnvironmentId` parameter in the JSON input.

2. Create the volume

HTTP method	Path
POST	/occm/api/azure/vsa/volumes

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/azure/vsa/volumes'
--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.

JSON input example

```json
{
  "providerVolumeType": "Premium_LRS",
  "verifyNameUniqueness": true,
  "name": "vol2",
  "size": {
    "size": 10,
    "unit": "GB"
  },
  "enableCompression": true,
  "enableDeduplication": true,
  "createAggregateIfNotFound": "true",
  "enableThinProvisioning": true,
  "aggregateName": "aggr1",
  "maxNumOfDisksApprovedToAdd": 0,
  "svmName": "svm_ShirleyAzureVsa2601",
  "iops": null,
  "snapshotPolicyName": "default",
  "autoVsaCapacityManagementEnabled": true,
  "exportPolicyInfo": {
    "_ips": "10.0.0.0/16",
    "ips": [
      "10.0.0.0/16"
    ],
    "nfsVersion": [
      "nfs3",
      "nfs4"
    ],
    "policyType": "custom"
  },
  "workingEnvironmentId": "VsaWorkingEnvironment-IsDYFJf8"
}
```

Output

None

Create volume using NFS for high availability pair

You can use this workflow to create a volume using NFS protocol for an HA working environment.

1. Select the working environment

Perform the workflow Create working environments and choose the `publicId` value for the `workingEnvironmentId` parameter in the JSON input.

2. Create the volume

HTTP method	Path
POST	/occm/api/azure/ha/volumes

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/azure/ha/volumes' --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.

JSON input example

```
{
  "providerVolumeType": "Premium_LRS",
  "verifyNameUniqueness": true,
  "name": "vol3",
  "size": {
    "size": 100,
    "unit": "GB"
  },
  "enableCompression": true,
  "enableDeduplication": true,
  "createAggregateIfNotFound": "true",
  "enableThinProvisioning": true,
  "aggregateName": "aggr1",
  "maxNumOfDisksApprovedToAdd": 0,
  "svmName": "svm_ShirleyHa2901",
  "iops": null,
  "snapshotPolicyName": "default",
  "autoVsaCapacityManagementEnabled": true,
  "exportPolicyInfo": {
    "_ips": "10.0.0.0/16",
    "ips": [
      "10.0.0.0/16"
    ],
    "nfsVersion": [
      "nfs3",
      "nfs4"
    ],
    "policyType": "custom"
  },
  "workingEnvironmentId": "VsaWorkingEnvironment-LUeyohBV"
}
```

Output

None

Create a volume using CIFS

You can use this workflow to create a volume accessed through CIFS protocol.

ⓘ   If the properties `aggregateName` and `maxNumOfDisksApprovedToAdd` are not provided on the REST API call, the response will fail with a suggested name for the aggregate and the number of disks needed to fulfill the request.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node

- [HA pair](#)

Create volume using CIFS for single node

You can use this workflow to create a volume using CIFS protocol for a single node working environment.

1. Choose the CIFS configuration

A CIFS server configuration must be defined for your working environment. You can do one of the following:

- If a CIFS configuration already exists, perform the workflow [Get CIFS server configurations](#) to access the configuration parameters.
- If a CIFS configuration does not exist, perform the workflow [Create CIFS server configuration](#) to create one.

2. Select the working environment

Perform the workflow [Get working environments](#) and choose the `publicId` value for the `workingEnvironmentId` (working environment) and the `svmName` (SVM name).

3. Select the aggregate

Perform the workflow [Get aggregates](#) and choose the `name` for the `aggregateName` value.

> ⓘ If aggregate name does not exist and the `createAggregateIfNotFound` query parameter is set `true`, the create volume request is allowed if the named aggregate is not found.

4. Choose the size for the disk

Choose the size value for the `size:size` parameter. The `size:unit` must be one of the following: `TB`, `GB`, `MB`, `KB`, or `Byte`.

5. Create the quote

Perform the workflow [Create quote](#). This is a recommended step but is not mandatory.

6. Create the volume

HTTP method	Path
POST	/occm/api/azure/vsa/volumes

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/azure/vsa/volumes'
--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, including:

- <WORKING_ENV_ID> (`workingEnvironmentId`) string

- <SVM_NAME> (`svmName`) string

- <AGGR_NAME> (`aggregateName`) string

If an aggregate name does not exist, you can set the `createAggregateIfNotFound` query parameter to `true` which allows the aggregate not-found condition.

JSON input example

```
{
  "workingEnvironmentId": "VsaWorkingEnvironment-SfpVUZSc",
  "svmName": "svm_zivaws02we01",
  "aggregateName": "aggr1",
  "name": "zivaws02we02vol02Cifs",
  "size": {
    "size": 100,
    "unit": "GB"
  },
  "shareInfo": {
    "accessControl": {
        "permission": "full_control",
        "users": [
            "Everyone"
        ],
        "users": "Everyone;"
    },
    "shareName": "zivaws02we01vol02Cifs_share"
  },
  "snapshotPolicyName": "default",
  "enableThinProvisioning": true,
  "enableCompression": true,
  "enableDeduplication": true,
  "maxNumOfDisksApprovedToAdd": 0
}
```

Output

None

Create volume using CIFS for high availability pair

You can use this workflow to create a volume using CIFS protocol for an HA working environment.

1. Choose the CIFS configuration

A CIFS server configuration must be defined for your working environment. You can do one of the following:

- If a CIFS configuration already exists, perform the workflow Get CIFS server configurations to access the configuration parameters.
- If a CIFS configuration does not exist, perform the workflow Create CIFS server configuration to create one.

2. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value for the `workingEnvironmentId` (working environment) and the `svmName` (SVM name).

3. Select the aggregate

Perform the workflow Get aggregates and choose the `name` for the `aggregateName` value.

> ⓘ If aggregate name does not exist and the `createAggregateIfNotFound` query parameter is set `true`, the create volume request is allowed if the named aggregate is not found.

4. Choose the size for the disk

Choose the size value for the `size:size` parameter. The `size:unit` must be one of the following: `TB`, `GB`, `MB`, `KB`, or `byte`.

5. Create the quote

Perform the workflow Create quote. This is a recommended step but is not mandatory.

6. Create the volume

HTTP method	Path
POST	/occm/api/azure/ha/volumes

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/azure/ha/volumes' --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, including:

- <WORKING_ENV_ID> (`workingEnvironmentId`) string
- <SVM_NAME> (`svmName`) string
- <AGGR_NAME> (`aggregateName`) string

If an aggregate name does not exist, you can set the `createAggregateIfNotFound` query parameter to `true` which allows the aggregate not-found condition.

JSON input example

```
{
  "workingEnvironmentId": "VsaWorkingEnvironment-SfpVUZSc",
  "svmName": "svm_zivaws02we01",
  "aggregateName": "aggr1",
  "name": "zivaws02we02vol02Cifs",
  "size": {
    "size": 100,
    "unit": "GB"
  },
  "shareInfo": {
    "accessControl": {
        "permission": "full_control",
        "users": [
            "Everyone"
        ],
        "users": "Everyone;"
    },
    "shareName": "zivaws02we01vol02Cifs_share"
  },
  "snapshotPolicyName": "default",
  "enableThinProvisioning": true,
  "enableCompression": true,
  "enableDeduplication": true,
  "maxNumOfDisksApprovedToAdd": 0
}
```

Output

None

Create a volume using iSCSI

You can use this workflow to create a volume accessed through iSCSI.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Create volume using iSCSI for single node

You can use this workflow to create a volume using iSCSI protocol for a single node working environment.

There are two workflows available depending on whether a new or existing iGroup is used. You need to select

the correct workflow:

- [Create volume using iSCSI with a new iGroup](#)
- [Create volume using iSCSI with an existing iGroup](#)

Create volume using iSCSI with a new iGroup

ⓘ   If the properties `aggregateName` and `maxNumOfDisksApprovedToAdd` are not provided on the REST API call, the response will fail with a suggested name for the aggregate and the number of disks needed to fulfill the request.

1. Select the working environment

Perform the workflow [Get working environments](#) and choose the `publicId` value for the `workingEnvironmentId` parameter and the `svmName` value for the `svmName` parameter.

2. Select the aggregate

Perform the workflow [Get aggregates](#) and choose the `name` for the `aggregateName` value.

3. Choose the size for the disk

Choose the size value for the `size:size` parameter. The `size:unit` must be one of the following: `TB`, `GB`, `MB`, `KB`, or `Byte`.

4. Choose the iscsiInfo parameters

You must choose the following values for the REST API call:

- A unique igroup name for `igroupCreationRequest → igroupName` parameter
- The required iqn's to `igroupCreationRequest → initiators` parameter.
- The required operating system for the `osName` parameter from one of the following:
  - windows
  - linux
  - vmware
  - windows_2008
  - windows_gpt

5. Create the quote

Perform the workflow [Create quote](#).This is a recommended step but is not mandatory.

6. Create the volume

HTTP method	Path
POST	/occm/api/azure/vsa/volumes

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/azure/vsa/volumes'
--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, including:

- <WORKING_ENV_ID> (workingEnvironmentId) string
- <SVM_NAME> (svmName) string
- <AGGR_NAME> (aggregateName) string

If aggregate name does not exist, you can set the `createAggregateIfNotFound` query parameter to `true` which allows the aggregate not-found condition.

JSON input example

```json
{
  "workingEnvironmentId": "VsaWorkingEnvironment-SfpVUZSc",
  "svmName": "svm_zivaws02we01",
  "aggregateName": "aggr1",
  "name": "zivaws02we01vol01Iscsi",
  "size": {
    "size": 100,
    "unit": "GB"
  },
  "iscsiInfo": {
    "igroupCreationRequest": {
        "igroupName": "zivIgroup",
        "initiators": [
            "iqn.1994-05.com.redhat:00xx00000000",
            "iqn.1994-05.com.redhat:00xx00000000"
        ]
    },
    "osName": "linux"
  },
  "snapshotPolicyName": "default",
  "enableThinProvisioning": true,
  "enableCompression": true,
  "enableDeduplication": true,
  "maxNumOfDisksApprovedToAdd": 0
}
```

Output

None

Create volume using iSCSI with an existing iGroup

> ⓘ If the properties `aggregateName` and `maxNumOfDisksApprovedToAdd` are not provided on the REST API call, the response will fail with a suggested name for the aggregate and the number of disks needed to fulfill the request.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value for the `workingEnvironmentId` parameter and the `svmName` value for the `svmName` parameter.

2. Select the aggregate

Perform the workflow Get aggregates and choose the `name` for the `aggregateName` value.

3. Choose the size for the disk

Choose the size value for the `size:size` parameter. The `size:unit` must be one of the following: `TB`, `GB`, `MB`, `KB`, or `byte`.

4. Choose the iGroup

Perform the workflow Get iGroups and choose the igroups for the `iscasiInfo → igroups` value. Also select the `osType` value for the `iscasiInfo → osName`.

5. Create the quote

Perform the workflow Create quote. This is a recommended step but is not mandatory.

6. Create the volume

HTTP method	Path
POST	/occm/api/azure/vsa/volumes

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/azure/vsa/volumes'
--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, including:

- <WORKING_ENV_ID> (`workingEnvironmentId`) string

- <SVM_NAME> (`svmName`) string

- <AGGR_NAME> (aggregateName) string

If an aggregate name does not exist, you can set the `createAggregateIfNotFound` query parameter to `true` which allows the aggregate not-found condition.

JSON input exmaple

```json
{
  "workingEnvironmentId": "VsaWorkingEnvironment-UvFmWXoD",
  "svmName": "svm_zivaws01we01",
  "aggregateName": "aggr1",
  "name": "zivaws01we01vol05Iscsi",
  "size": {
    "size": 100,
    "unit": "GB"
  },
  "iscsiInfo": {
    "igroups": ["zivIgroup1"],
    "osName": "linux"
  },
  "snapshotPolicyName": "default",
  "enableThinProvisioning": true,
  "enableCompression": true,
  "enableDeduplication": true,
  "maxNumOfDisksApprovedToAdd": 0
}
```

Output

None

Create volume using iSCSI for high availability

You can use this workflow to create a volume using iSCSI protocol for an HA working environment.

There are two workflows available depending on whether a new or existing iGroup is used. You need to select the correct workflow:

- Create volume using iSCSI with a new iGroup
- Create volume using iSCSI with an existing iGroup

Create volume using iSCSI with a new iGroup

ⓘ   If the properties `aggregateName` and `maxNumOfDisksApprovedToAdd` are not provided on the REST API call, the response will fail with a suggested name for the aggregate and the number of disks needed to fulfill the request.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value for the `workingEnvironmentId` parameter and the `svmName` value for the `svmName` parameter.

2. Select the aggregate

Perform the workflow Get aggregates and choose the `name` for the `aggregateName` value.

3. Choose the size for the disk

Choose the size value for the `size:size` parameter. The `size:unit` must be one of the following: `TB`, `GB`, `MB`, `KB`, or `byte`.

4. Choose the iscsiInfo parameters

You must choose the following values for the REST API call:

* A unique igroup name for `igroupCreationRequest → igroupName` parameter
* The required iqn's to `igroupCreationRequest → initiators` parameter.
* The required operating system for the `osName` parameter from one of the following:
  - windows
  - linux
  - vmware
  - windows_2008
  - windows_gpt

5. Create the quote

Perform the workflow Create quote.This is a recommended step but is not mandatory.

6. Create the volume

HTTP method	Path
POST	/occm/api/azure/ha/volumes

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/azure/ha/volumes' --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, including:

* <WORKING_ENV_ID> (`workingEnvironmentId`) string

- <SVM_NAME> (`svmName`) string

- <AGGR_NAME> (`aggregateName`) string

If aggregate name does not exist, you can set the `createAggregateIfNotFound` query parameter to `true` which allows the aggregate not-found condition.

JSON input example

```json
{
    "workingEnvironmentId":"VsaWorkingEnvironment-1m76JaRt",
    "svmName":"svm_ranukazure12",
    "snapshotPolicyName":"default",
    "name":"ranukvoliscsi",
    "iops":null,
    "throughput":null,
    "providerVolumeType":"Premium_LRS",
    "capacityTier":"Blob",
    "tieringPolicy":"auto",
    "verifyNameUniqueness":true,
    "iscsiInfo":{
        "igroupCreationRequest":{
            "igroupName":"ig1",
            "initiators":[
                "iqn.1991-05.com.microsoft:pradipm02-pc"
            ]
        },
        "osName":"windows"
    },
    "size":{
        "size":200,
        "unit":"GB"
    },
    "enableThinProvisioning":true,
    "enableDeduplication":true,
    "enableCompression":true,
    "maxNumOfDisksApprovedToAdd":0,
    "aggregateName":"aggr1"
}
```

Output

None

Create volume using iSCSI with an existing iGroup

	If the properties `aggregateName` and `maxNumOfDisksApprovedToAdd` are not provided on the REST API call, the response will fail with a suggested name for the aggregate and the number of disks needed to fulfill the request.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value for the `workingEnvironmentId` parameter and the `svmName` value for the `svmName` parameter.

2. Select the aggregate

Perform the workflow Get aggregates and choose the `name` for the `aggregateName` value.

3. Choose the size for the disk

Choose the size value for the `size:size` parameter. The `size:unit` must be one of the following: `TB`, `GB`, `MB`, `KB`, or `byte`.

4. Choose the iGroup

Perform the workflow Get iGroups and choose the igroups for the `iscasiInfo → igroups` value. Also select the `osType` value for the `iscasiInfo → osName`.

5. Create the quote

Perform the workflow Create quote. This is a recommended step but is not mandatory.

6. Create the volume

HTTP method	Path
POST	/occm/api/azure/ha/volumes

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/azure/ha/volumes' --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, including:

- <WORKING_ENV_ID> (`workingEnvironmentId`) string

- <SVM_NAME> (`svmName`) string

- <AGGR_NAME> (`aggregateName`) string

If an aggregate name does not exist, you can set the `createAggregateIfNotFound` query parameter to `true` which allows the aggregate not-found condition.

JSON input exmaple

```json
{
  "workingEnvironmentId": "VsaWorkingEnvironment-UvFmWXoD",
  "svmName": "svm_zivaws01we01",
  "aggregateName": "aggr1",
  "name": "zivaws01we01vol05Iscsi",
  "size": {
    "size": 100,
    "unit": "GB"
  },
  "iscsiInfo": {
    "igroups": ["zivIgroup1"],
    "osName": "linux"
  },
  "snapshotPolicyName": "default",
  "enableThinProvisioning": true,
  "enableCompression": true,
  "enableDeduplication": true,
  "maxNumOfDisksApprovedToAdd": 0
}
```

Output

None

Get volumes

You can retrieve the list of volumes of an Azure working environment.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Get volume for single node

You can use this workflow to retrieve volumes for a single node working environment.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Get volume for single node

You can use this workflow to retrieve volume for a single node working environment.

1. Create the working environment

Perform the workflow Create Azure single node working environment and choose the `publicId` value of the working environment for the `workingEnvironmentId` query parameter.

2. Get the volumes

HTTP method	Path
GET	/occm/api/azure/vsa/volumes

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/azure/vsa/volumes?workingE
nvironmentId=<WORKING_ENV_ID>' --header 'Content-Type: application/json'
--header 'x-agent-id: <AGENT_ID>' --header 'Authorization: Bearer
<ACCESS_TOKEN>'
```

Input

Query parameter <WORKING_ENV_ID> (`workingEnvironmentId`) string

Output

The JSON output example includes the list of volumes for the working environment.

JSON output example

```
[
    {
        "name": "vol1",
        "uuid": "x0000x00-0000-xx00-x000-000x0x0xxxxx",
        "svmName": "svm_ShirleyAzureVsa2601",
        "size": {
            "size": 10.0,
            "unit": "GB"
        },
        "usedSize": {
            "size": 2.74658203125E-4,
            "unit": "GB"
        },
        "junctionPath": "/vol1",
        "volumeTotalInodes": 311287,
        "volumeUsedInodes": 96,
        "mountPoint": "10.0.0.10:/vol1",
        "compressionSpaceSaved": {
            "size": 0.0,
            "unit": "GB"
        },
```

```json
        "deduplicationSpaceSaved": {
            "size": 0.0,
            "unit": "GB"
        },
        "thinProvisioning": true,
        "compression": true,
        "deduplication": true,
        "snapshotPolicy": "default",
        "securityStyle": "unix",
        "exportPolicyInfo": {
            "name": "export-svm_ShirleyAzureVsa2601-vol1",
            "policyType": "custom",
            "ips": [
                "10.0.0.0/16"
            ],
            "nfsVersion": [
                "nfs3",
                "nfs4"
            ]
        },
        "shareNames": [],
        "shareInfo": [],
        "parentVolumeName": "",
        "rootVolume": false,
        "state": "online",
        "volumeType": "rw",
        "aggregateName": "aggr1",
        "parentSnapshot": null,
        "autoSizeMode": "grow",
        "maxGrowSize": {
            "size": 110.0,
            "unit": "GB"
        },
        "providerVolumeType": "Premium_LRS",
        "cloneNames": [],
        "moving": false,
        "primaryNoFailoverMountPoint": null,
        "secondaryNoFailoverMountPoint": null,
        "capacityTier": null,
        "capacityTierUsedSize": null,
        "cifsShareAccessPoint": null,
        "primaryCifsShareAccessPoint": null,
        "secondaryCifsShareAccessPoint": null,
        "tieringPolicy": "none",
        "tierInactiveUserData": {
            "size": 0.0,
```

```
            "unit": "GB"
        },
        "tierInactiveUserDataPercent": 0,
        "comment": null,
        "qosPolicyGroupName": null,
        "snaplockType": "non_snaplock",
        "constituentsAggregates": [],
        "snapshotsUsedSize": {
            "size": 0.0,
            "unit": "Byte"
        },
        "cbsBackupsInfo": null,
        "minimumCoolingDays": null,
        "targetName": "iqn.1992-
08.com.netapp:sn.65c7e1cc600d11eb8ca3000d3a7e6f7c:vs.2",
        "iscsiEnabled": false,
        "isFlexGroupVolume": false
    }
]
```

Get volume for high availability pair

You can use this workflow to retrieve volume for an HA working environment.

1. Create the working environment

Perform the workflow Create Azure HA working environment and choose the `publicId` value of the working environment for the `workingEnvironmentId` query parameter.

2. Get the volumes

HTTP method	Path
GET	/occm/api/azure/ha/volumes

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/azure/ha/volumes?workingEn
vironmentId=<WORKING_ENV_ID>' --header 'Content-Type: application/json'
--header 'x-agent-id: <AGENT_ID>' --header 'Authorization: Bearer
<ACCESS_TOKEN>'
```

Input

Query parameter <WORKING_ENV_ID> (`workingEnvironmentId`) string

Output

The JSON output example includes the list of volumes for the working environment.

JSON output example

```json
[
    {
        "name": "vol1",
        "uuid": "x0000x00-0000-xx00-x000-000x0x0xxxxx",
        "svmName": "svm_ShirleyHa2901",
        "size": {
            "size": 100.0,
            "unit": "GB"
        },
        "usedSize": {
            "size": 2.93731689453125E-4,
            "unit": "GB"
        },
        "junctionPath": "/vol1",
        "volumeTotalInodes": 3112959,
        "volumeUsedInodes": 96,
        "mountPoint": "10.0.0.9:/vol1",
        "compressionSpaceSaved": {
            "size": 0.0,
            "unit": "GB"
        },
        "deduplicationSpaceSaved": {
            "size": 0.0,
            "unit": "GB"
        },
        "thinProvisioning": true,
        "compression": true,
        "deduplication": true,
        "snapshotPolicy": "default",
        "securityStyle": "unix",
        "exportPolicyInfo": {
            "name": "export-svm_ShirleyHa2901-vol1",
            "policyType": "custom",
            "ips": [
                "10.0.0.0/16"
            ],
            "nfsVersion": [
                "nfs3",
                "nfs4"
            ]
        },
        "shareNames": [],
```

```json
        "shareInfo": [],
        "parentVolumeName": "",
        "rootVolume": false,
        "state": "online",
        "volumeType": "rw",
        "aggregateName": "aggr1",
        "parentSnapshot": null,
        "autoSizeMode": "grow",
        "maxGrowSize": {
            "size": 1100.0,
            "unit": "GB"
        },
        "providerVolumeType": "Premium_LRS",
        "cloneNames": [],
        "moving": false,
        "primaryNoFailoverMountPoint": null,
        "secondaryNoFailoverMountPoint": null,
        "capacityTier": null,
        "capacityTierUsedSize": null,
        "cifsShareAccessPoint": null,
        "primaryCifsShareAccessPoint": null,
        "secondaryCifsShareAccessPoint": null,
        "tieringPolicy": "none",
        "tierInactiveUserData": {
            "size": 0.0,
            "unit": "GB"
        },
        "tierInactiveUserDataPercent": 0,
        "comment": null,
        "qosPolicyGroupName": null,
        "snaplockType": "non_snaplock",
        "constituentsAggregates": [],
        "snapshotsUsedSize": {
            "size": 0.0,
            "unit": "Byte"
        },
        "cbsBackupsInfo": null,
        "minimumCoolingDays": null,
        "targetName": "iqn.1992-
08.com.netapp:sn.fc000x0000000xx0x000000xae000005:vs.3",
        "iscsiEnabled": false,
        "isFlexGroupVolume": false
    },
    {
        "name": "vol2",
        "uuid": "x0000x00-0000-xx00-x000-000x0x0xxxxx",
```

```json
        "svmName": "svm_ShirleyHa2901",
        "size": {
            "size": 30.0,
            "unit": "GB"
        },
        "usedSize": {
            "size": 2.6702880859375E-4,
            "unit": "GB"
        },
        "junctionPath": "/vol2",
        "volumeTotalInodes": 933887,
        "volumeUsedInodes": 96,
        "mountPoint": "10.0.0.9:/vol2",
        "compressionSpaceSaved": {
            "size": 0.0,
            "unit": "GB"
        },
        "deduplicationSpaceSaved": {
            "size": 0.0,
            "unit": "GB"
        },
        "thinProvisioning": false,
        "compression": false,
        "deduplication": false,
        "snapshotPolicy": "default",
        "securityStyle": "unix",
        "exportPolicyInfo": {
            "name": "export-svm_ShirleyHa2901-vol2",
            "policyType": "custom",
            "ips": [
                "10.0.0.0/16"
            ],
            "nfsVersion": [
                "nfs3",
                "nfs4"
            ]
        },
        "shareNames": [],
        "shareInfo": [],
        "parentVolumeName": "",
        "rootVolume": false,
        "state": "online",
        "volumeType": "rw",
        "aggregateName": "aggr1",
        "parentSnapshot": null,
        "autoSizeMode": "grow",
```

```
        "maxGrowSize": {
            "size": 330.0,
            "unit": "GB"
        },
        "providerVolumeType": "Premium_LRS",
        "cloneNames": [],
        "moving": false,
        "primaryNoFailoverMountPoint": null,
        "secondaryNoFailoverMountPoint": null,
        "capacityTier": null,
        "capacityTierUsedSize": null,
        "cifsShareAccessPoint": null,
        "primaryCifsShareAccessPoint": null,
        "secondaryCifsShareAccessPoint": null,
        "tieringPolicy": "none",
        "tierInactiveUserData": {
            "size": 0.0,
            "unit": "GB"
        },
        "tierInactiveUserDataPercent": 0,
        "comment": null,
        "qosPolicyGroupName": null,
        "snaplockType": "non_snaplock",
        "constituentsAggregates": [],
        "snapshotsUsedSize": {
            "size": 0.0,
            "unit": "Byte"
        },
        "cbsBackupsInfo": null,
        "minimumCoolingDays": null,
        "targetName": "iqn.1992-
08.com.netapp:sn.fc000x0000000xx0x000000xae000005:vs.3",
        "iscsiEnabled": false,
        "isFlexGroupVolume": false
    }
]
```

Modify volume

You can modify the configuration of an existing volume.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Modify volume for single node

You can use this workflow to modify the volume configuration for a single node working environment.

1. Create the working environment

Perform the workflow Create Azure single node working environment and choose the `publicId` value of the working environment to be used as the `workingEnvironmentId` path parameter.

2. Select the volume

Perform the workflow Get volumes and choose the `name` and the `svmName` values of the volume to modify and use them for the `volumeName` and `svmName` path parameters.

3. Modify the volume

HTTP method	Path
PUT	/occm/api/azure/vsa/volumes/{workingEnvironmentId}/{svmName}/{volumeName}

curl example

```
curl --location --request PUT
'https://cloudmanager.cloud.netapp.com/occm/api/azure/vsa/volumes/<WORKING
_ENV_ID>/<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`) string

- <SVM_NAME> (`svmName`) string

- <VOLUME_NAME> (`volumeName`) string

The JSON input example includes the minimum list of input parameters.

JSON input example

```
{
  "volumeName": "vol5",
  "svmName": "svm_ShirleyAzureVsa2601",
  "originalVolumeInfo": {
    "exportPolicyInfo": {
      "ips": [
        "10.0.0.0/16"
      ],
      "name": "export-svm_ShirleyAzureVsa2601-vol5",
      "nfsVersion": [
        "nfs3",
        "nfs4"
      ],
      "policyType": "custom"
    },
    "shareInfo": [],
    "snapshotPolicyName": "default"
  },
  "workingEnvironmentName": "ShirleyAzureVsa2601",
  "snapshotPolicyName": "default",
  "exportPolicyInfo": {
    "ips": [],
    "policyType": "none"
  },
  "workingEnvironmentId": "VsaWorkingEnvironment-IsDYFJf8"
}
```

Output

None

Modify volume for high availability pair

You can use this workflow to modify the volume configuration for an HA working environment.

1. Create the working environment

Perform the workflow Create Azure HA working environment and choose the `publicId` value of the working environment to be used as the `workingEnvironmentId` path parameter.

2. Select the volume

Perform the workflow Get volumes and choose the `name` and the `svmName` values of the volume to modify and use them for the `volumeName` and `svmName` path parameters.

3. Modify the volume

HTTP method	Path
PUT	/occm/api/azure/ha/volumes/{workingEnvironmentId}/{svmName}/{volumeName}

curl example

```
curl --location --request PUT
'https://cloudmanager.cloud.netapp.com/occm/api/azure/ha/volumes/<WORKING_
ENV_ID>/<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) string

- <SVM_NAME> (svmName) string

- <VOLUME_NAME> (volumeName) string

The JSON input example includes the minimum list of input parameters.

JSON input example

```
{
  "volumeName": "vol2",
  "svmName": "svm_ShirleyHa2901",
  "originalVolumeInfo": {
    "exportPolicyInfo": {
      "ips": [
        "10.0.0.0/16"
      ],
      "name": "export-svm_ShirleyHa2901-vol2",
      "nfsVersion": [
        "nfs3",
        "nfs4"
      ],
      "policyType": "custom"
    },
    "shareInfo": [],
    "snapshotPolicyName": "default"
  },
  "workingEnvironmentName": "ShirleyHa2901",
  "snapshotPolicyName": "default",
  "exportPolicyInfo": {
    "ips": [
      "20.0.0.0/16"
    ],
    "nfsVersion": [
      "nfs4"
    ],
    "policyType": "custom"
  },
  "workingEnvironmentId": "VsaWorkingEnvironment-LUeyohBV"
}
```

Output

  None

Delete volume

You can delete an existing volume.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Delete volume for single node

You can use this workflow to delete volume for a single node working environment.

1. Create the working environment

Perform the Create Azure single node working environment workflow and choose the `publicId` value for the working environment field in the input parameter.

2. Select the volume

Perform the workflow Get volumes and choose the `name` and the `svmName` values of the volume to delete and use them for the `volumeName` and `svmName` path parameters.

3. Delete the volume

HTTP method	Path
DELETE	/occm/api/azure/vsa/volumes/{workingEnvironmentId}/{svmName}/{volumeName}

curl example

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

Input

Path parameters:

- <WORKING_ENV_ID> (`workingEnvironmentId`) string

- <SVM_NAME> (`svmName`) string

- <VOLUME_NAME> (`volumeName`) string

Output

None

Delete volume for high availability pair

You can use this workflow to delete volume for an HA working environment.

1. Create the working environment

Perform the Create Azure HA working environment workflow and choose the `publicId` value for the working environment field in the input parameter.

2. Select the volume

Perform the workflow Get volumes and choose the `name` and the `svmName` values of the volume to delete and use them for the `volumeName` and `svmName` path parameters.

3. Delete the volume

HTTP method	Path
DELETE	/occm/api/azure/ha/volumes/{workingEnvironmentId}/{svmName}/{volumeName}

curl example

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

Input

Path parameters:

- <WORKING_ENV_ID> (workingEnvironmentId) string

- <SVM_NAME> (svmName) string

- <VOLUME_NAME> (volumeName) string

Output

None

Create quote

You can create a quote for a new volume which returns a resource quote needed to satisfy the request. The resource quote contains aggregate information where the volume will be created and confirms if the space is available. This is a recommended step but is not mandatory.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Create quote for single node

You can perform this workflow to create a volume quote for a single node working environment.

1. Select the working environment

Perform the workflow Create Azure single node working environment and choose the `publicId` value for the working environment field in the input parameter.

2. Generate the volume quote

HTTP method	Path
POST	/occm/api/azure/vsa/quote

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/azure/vsa/volumes/quote'
--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 list of input parameters.

JSON input example

```
{
  "providerVolumeType": "Premium_LRS",
  "verifyNameUniqueness": true,
  "name": "vol3",
  "size": {
    "size": 10,
    "unit": "GB"
  },
  "enableCompression": true,
  "enableDeduplication": true,
  "enableThinProvisioning": true,
  "svmName": "svm_ShirleyAzureVsa2601",
  "iops": null,
  "snapshotPolicyName": "default",
  "autoVsaCapacityManagementEnabled": true,
  "exportPolicyInfo": {
    "_ips": "10.0.0.0/16",
    "ips": [
      "10.0.0.0/16"
    ],
    "policyType": "custom"
  },
  "workingEnvironmentId": "VsaWorkingEnvironment-IsDYFJf8"
}
```

Output

The JSON output example includes an example of the quote details.

JSON output example

```
{
    "numOfDisks": 0,
    "diskSize": {
        "size": 1.0,
        "unit": "TB"
    },
    "aggregateName": "aggr1",
    "newAggregate": false,
    "autoVsaCapacityManagement": true
}
```

Create quote for high availability pair

You can use this workflow to create a volume quote for an HA working environment.

1. Create the working environment

Perform the workflow Create Azure HA working environment and choose the `publicId` value for the working environment field in the input parameter.

2. Generate the volume quote

HTTP method	Path
POST	/occm/api/azure/ha/quote

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/azure/ha/volumes/quote'
--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 list of input parameters.

JSON input example

```
{
    "providerVolumeType": "Premium_LRS",
    "verifyNameUniqueness": true,
    "name": "vol3",
    "size": {
        "size": 10,
        "unit": "GB"
    },
    "enableCompression": true,
    "enableDeduplication": true,
    "enableThinProvisioning": true,
    "svmName": "svm_ShirleyAzureVsa2601",
    "iops": null,
    "snapshotPolicyName": "default",
    "autoVsaCapacityManagementEnabled": true,
    "exportPolicyInfo": {
        "_ips": "10.0.0.0/16",
        "ips": [
            "10.0.0.0/16"
        ],
        "policyType": "custom"
    },
    "workingEnvironmentId": "VsaWorkingEnvironment-IsDYFJf8"
}
```

Output

The JSON output example includes an example of the quote details.

JSON output example

```
{
    "numOfDisks": 0,
    "diskSize": {
        "size": 1.0,
        "unit": "TB"
    },
    "aggregateName": "aggr1",
    "newAggregate": false,
    "autoVsaCapacityManagement": true
}
```

Get iGroups

You can use this workflow to retrieve all the initiator groups (iGroups).

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node

- HA pair

Get iGroups for single node

You can use this workflow to retrieve iGroups for a single node working environment.

1. Select the working environment

Perform the workflow Create Azure single node working environment and choose the `publicId` and `svmName` values for the working environment `workingEnvironmentId` and `svmName` path parameters.

2. Get the CIFS configurations

HTTP method	Path
GET	/occm/api/azure/vsa/volumes/igroups/{workingEnvironmentId}/{svmName}

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/azure/vsa/volumes/igroups/
<WORKING_ENV_ID>/<SVM_NAME>' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>' --header 'Content-Type:
application/json'
```

Input

- Path parameter <WORKING_ENV_ID> `workingEnvironmentId` string

- Path parameter <SVM_NAME> `svmName` string

Output
The JSON output example includes a list of iGroups.

JSON output example

```
[
    {
        "igroupName": "zivIgroup1",
        "osType": "linux",
        "portsetName": "",
        "igroupType": "iscsi",
        "initiators": [
            "iqn.1994-05.com.redhat:0x0xx000000x"
        ]
    },
    {
        "igroupName": "zivIgroup2",
        "osType": "linux",
        "portsetName": "",
        "igroupType": "iscsi",
        "initiators": [
            "iqn.1994-05.com.redhat:0x0xx000000x"
        ]
    }
]
```

Get iGroups for high availability pair

You can use this workflow to retrieve iGroups for an HA working environment.

1. Select the working environment

Perform the workflow Create Azure single node working environment and choose the `publicId` and `svmName` values for the working environment `workingEnvironmentId` and `svmName` path parameters.

2. Get the CIFS configurations

HTTP method	Path
GET	/occm/api/azure/ha/volumes/igroups/{workingEnvironmentId}/{svmName}

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/azure/ha/volumes/igroups/<
WORKING_ENV_ID>/<SVM_NAME>' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>' --header 'Content-Type:
application/json'
```

Input
- Path parameter <WORKING_ENV_ID> `workingEnvironmentId` string

- Path parameter <SVM_NAME> `svmName` string

Output

The JSON output example includes a list of iGroups.

JSON output example

```json
[
    {
        "igroupName": "zivIgroup1",
        "osType": "linux",
        "portsetName": "",
        "igroupType": "iscsi",
        "initiators": [
            "iqn.1994-05.com.redhat:0x0xx000000x"
        ]
    },
    {
        "igroupName": "zivIgroup2",
        "osType": "linux",
        "portsetName": "",
        "igroupType": "iscsi",
        "initiators": [
            "iqn.1994-05.com.redhat:0x0xx000000x"
        ]
    }
]
```

Metadata

Get Azure regions

This workflow retrieves the Azure regions in which an Cloud Volumes ONTAP working environment may be created.

ⓘ   Ensure that you've the Azure subscription ID. You can easily obtain the ID from the Azure portal.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Get Azure regions for single node

You can perform this workflow to retrieve the Azure regions for a single node working environment.

1. Get the list of regions

HTTP method	Path
GET	/occm/api/azure/vsa/metadata/regions

Curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/azure/vsa/metadata/regions
?subscriptionId=<SUBSCRIPTION_ID>' --header 'Content-Type:
application/json' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

Query parameters (Optional):

- `fields`, string

- `subscriptionId`, string

- `cloudProviderAccountId`, string

Output

The JSON output provides an example of a list of Azure regions.

JSON output example

```
[
    {
        "displayName": "Central US",
        "name": "centralus",
        "vnets": null
    },
    {
        "displayName": "East US",
        "name": "eastus",
        "vnets": null
    },
    {
        "displayName": "East US 2",
        "name": "eastus2",
        "vnets": null
    },
    {
        "displayName": "West US 2",
        "name": "westus2",
        "vnets": null
    }
]
```

Get Azure regions for high availability pair

You can perform this workflow to retrieve the Azure regions for an HA working environment.

1. Get the list of regions

HTTP method	Path
GET	/occm/api/azure/ha/metadata/regions

Curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/azure/ha/metadata/regions?
subscriptionId=<SUBSCRIPTION_ID>' --header 'Content-Type:
application/json' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

(Optional) Query parameters:

- `fields`, string

- `subscriptionId`, string

- `cloudProviderAccountId`, string

Output

The JSON output provides an example of a list of Azure regions.

JSON output example

```
[
    {
        "displayName": "Central US",
        "name": "centralus",
        "vnets": null
    },
    {
        "displayName": "East US",
        "name": "eastus",
        "vnets": null
    },
    {
        "displayName": "East US 2",
        "name": "eastus2",
        "vnets": null
    },
    {
        "displayName": "West US 2",
        "name": "westus2",
        "vnets": null
    }
]
```

Get Azure permutations

You can use the permutations endpoint to retrieve the Cloud Volumes ONTAP configuration information.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Get permutations for single node

You can use this workflow to retrieve the Cloud Volumes ONTAP configurations information for a single node working environment.

1. Get the permutations

HTTP method	Path
GET	/occm/api/azure/vsa/metadata/permutations

curl example

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

Input

There are several optional query parameters you can use:

- `region` string

- `version` string

- `license` string

- `instance_type` string

- `default_instance_type` string

- `feature` string

- `latest_only` string

- `marketplace_version` string

- `marketplace_sku` string

Output

The JSON output example includes the list of Cloud Volumes ONTAP configurations.

JSON output example

```
[
    {
        "ontapVersion": "ONTAP-9.9.0X5.T1.azure",
        "license": {
            "type": "azure-cot-explore-paygo",
            "name": "Cloud Volumes ONTAP Explore",
            "description": "Suitable for smaller capacity applications.
Supports up to 2 TB of underlying Azure storage.",
            "subName": "",
            "subDescription": "Support of tiering to object storage is not
included.",
            "capacity_limit": "2TB",
            "platformLicenseRequired": false,
            "default": false,
            "capacityLimit": {
```

```
                "size": 2.0,
                "unit": "TB"
            }
        },
        "instanceType": "Standard_DS3_v2",
        "region": {
            "name": "East US 2",
            "code": "eastus2",
            "location": "Virginia",
            "s3Region": null
        },
        "defaultInstance": true,
        "features": [
            "four-nics",
            "no-sidl"
        ],
        "upgradeableFrom": [
            "9.8",
            "9.9"
        ]
    },
    {
        "ontapVersion": "ONTAP-9.9.0X5.T1.azure",
        "license": {
            "type": "azure-cot-explore-paygo",
            "name": "Cloud Volumes ONTAP Explore",
            "description": "Suitable for smaller capacity applications.
Supports up to 2 TB of underlying Azure storage.",
            "subName": "",
            "subDescription": "Support of tiering to object storage is not
included.",
            "capacity_limit": "2TB",
            "platformLicenseRequired": false,
            "default": false,
            "capacityLimit": {
                "size": 2.0,
                "unit": "TB"
            }
        },
        "instanceType": "Standard_DS3_v2",
        "region": {
            "name": "East US 2",
            "code": "eastus2",
            "location": "Virginia",
            "s3Region": null
        },
```

```
        "defaultInstance": true,
        "features": [
            "four-nics",
            "no-sidl"
        ],
        "upgradeableFrom": [
            "9.8",
            "9.9"
        ]
    }
]
```

Get permutations for high availability pair

You can use this workflow to retrieve the Cloud Volumes ONTAP configurations information for an HA working environment.

1. Get the permutations

HTTP method	Path
GET	/occm/api/azure/ha/metadata/permutations

curl example

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

Input

There are several optional query parameters you can use:

- `region` string

- `version` string

- `license` string

- `instance_type` string

- `default_instance_type` string

- `feature` string

- `latest_only` string

- `marketplace_version` string

- `marketplace_sku` string

Output

The JSON output example includes the list of Cloud Volumes ONTAP configurations.

JSON output example

```
[
    {
        "ontapVersion": "ONTAP-9.9.0X5.T1.azureha",
        "license": {
            "type": "azure-ha-cot-standard-paygo",
            "name": "Cloud Volumes ONTAP Standard",
            "description": "Flexible performance and larger capacity for a
wider range of applications. Supports up to 10 TB of underlying Azure
storage.",
            "subName": "",
            "subDescription": "Supports tiering to object storage of
replicated volumes and snapshots.",
            "capacity_limit": "10TB",
            "platformLicenseRequired": false,
            "default": true,
            "capacityLimit": {
                "size": 10.0,
                "unit": "TB"
            }
        },
        "instanceType": "Standard_DS4_v2",
        "region": {
            "name": "Southeast Asia",
            "code": "southeastasia",
            "location": "Singapore",
            "s3Region": null
        },
        "defaultInstance": true,
        "features": [],
        "upgradeableFrom": [
            "9.8",
            "9.9"
        ]
    },
    {
        "ontapVersion": "ONTAP-9.9.0X5.T1.azureha",
        "license": {
            "type": "azure-ha-cot-standard-paygo",
            "name": "Cloud Volumes ONTAP Standard",
            "description": "Flexible performance and larger capacity for a
wider range of applications. Supports up to 10 TB of underlying Azure
storage.",
```

```
            "subName": "",
            "subDescription": "Supports tiering to object storage of
 replicated volumes and snapshots.",
            "capacity_limit": "10TB",
            "platformLicenseRequired": false,
            "default": true,
            "capacityLimit": {
                "size": 10.0,
                "unit": "TB"
            }
        },
        "instanceType": "Standard_DS4_v2",
        "region": {
            "name": "Southeast Asia",
            "code": "southeastasia",
            "location": "Singapore",
            "s3Region": null
        },
        "defaultInstance": true,
        "features": [],
        "upgradeableFrom": [
            "9.8",
            "9.9"
        ]
    }
]
```

Get Azure storage account types

You can retrieve the supported Azure storage account types.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Get Azure storage account types for single node

You can perform this workflow to retrieve the Azure storage account types for a single node working environment.

1. Get the storage account types

HTTP method	Path
GET	/occm/api/azure/vsa/metadata/storage-account-types

curl example

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

Input

None

Output

The JSON output example includes the list of Azure storage account types.

JSON output example

```
[{
        "diskType": "Premium",
        "availabilityTypes": ["Premium_LRS"],
        "sizes": [{
                "size": {
                        "size": 500.0,
                        "unit": "GB"
                },
                "description": "",
                "supportedOccmLicenses": ["Explore (hourly)", "Standard
(hourly)", "Standard (BYOL)", "Cloud Volumes ONTAP Explore", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP
Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL",
"Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes ONTAP
Capacity Based Charging", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": false
        }, {
                "size": {
                        "size": 1.0,
                        "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Explore (hourly)", "Standard
(hourly)", "Standard (BYOL)", "Cloud Volumes ONTAP Explore", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
```

ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP
Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL",
"Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes ONTAP
Capacity Based Charging", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": true
            }, {
                "size": {
                    "size": 2.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Explore (hourly)", "Standard
(hourly)", "Standard (BYOL)", "Cloud Volumes ONTAP Explore", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP
Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL",
"Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes ONTAP
Capacity Based Charging", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": false
            }, {
                "size": {
                    "size": 4.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Standard (hourly)", "Standard
(BYOL)", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes
ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity
Based Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP
Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes

ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL"],
                "isDefault": **false**
            }, {
                "size": {
                    "size": **8.0**,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Standard (hourly)", "Standard
(BYOL)", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes
ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity
Based Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP
Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL"],
                "isDefault": **false**
            }, {
                "size": {
                    "size": **16.0**,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Standard (BYOL)", "Cloud
Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP
Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity Based
Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes
ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": **false**
            }, {
                "size": {
                    "size": **32.0**,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Standard (BYOL)", "Cloud
Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP
Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity Based
Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes
ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium",

```
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": false
            }
        ]
    }, {
        "diskType": "Standard",
        "availabilityTypes": ["Standard_LRS"],
        "sizes": [{
                "size": {
                    "size": 100.0,
                    "unit": "GB"
                },
                "description": "",
                "supportedOccmLicenses": ["Explore (hourly)", "Standard
(hourly)", "Standard (BYOL)", "Cloud Volumes ONTAP Explore", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP
Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL",
"Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes ONTAP
Capacity Based Charging", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": false
            }, {
                "size": {
                    "size": 500.0,
                    "unit": "GB"
                },
                "description": "",
                "supportedOccmLicenses": ["Explore (hourly)", "Standard
(hourly)", "Standard (BYOL)", "Cloud Volumes ONTAP Explore", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP
Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL",
"Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes ONTAP
Capacity Based Charging", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP Standard",
```

```
                "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                    "isDefault": false
            }, {
                "size": {
                    "size": 1.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Explore (hourly)", "Standard
(hourly)", "Standard (BYOL)", "Cloud Volumes ONTAP Explore", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP
Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL",
"Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes ONTAP
Capacity Based Charging", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                    "isDefault": true
            }, {
                "size": {
                    "size": 2.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Explore (hourly)", "Standard
(hourly)", "Standard (BYOL)", "Cloud Volumes ONTAP Explore", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP
Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL",
"Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes ONTAP
Capacity Based Charging", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                    "isDefault": false
            }, {
                "size": {
                    "size": 4.0,
                    "unit": "TB"
```

```
                },
                "description": "",
                "supportedOccmLicenses": ["Standard (hourly)", "Standard
(BYOL)", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes
ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity
Based Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP
Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL"],
                "isDefault": false
            }, {
                "size": {
                    "size": 8.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Standard (hourly)", "Standard
(BYOL)", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes
ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity
Based Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP
Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL"],
                "isDefault": false
            }, {
                "size": {
                    "size": 16.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Standard (BYOL)", "Cloud
Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP
Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity Based
Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes
ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": false
```

```
        }, {
            "size": {
                "size": 32.0,
                "unit": "TB"
            },
            "description": "",
            "supportedOccmLicenses": ["Standard (BYOL)", "Cloud
Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP
Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity Based
Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes
ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
            "isDefault": false
        }
    ]
}, {
    "diskType": "StandardSSD",
    "availabilityTypes": ["StandardSSD_LRS"],
    "sizes": [{
            "size": {
                "size": 100.0,
                "unit": "GB"
            },
            "description": "",
            "supportedOccmLicenses": ["Explore (hourly)", "Standard
(hourly)", "Standard (BYOL)", "Cloud Volumes ONTAP Explore", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP
Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL",
"Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes ONTAP
Capacity Based Charging", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
            "isDefault": false
        }, {
            "size": {
                "size": 500.0,
                "unit": "GB"
            },
            "description": "",
            "supportedOccmLicenses": ["Explore (hourly)", "Standard
```

(hourly)", "Standard (BYOL)", "Cloud Volumes ONTAP Explore", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP
Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL",
"Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes ONTAP
Capacity Based Charging", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": false
            }, {
                "size": {
                    "size": 1.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Explore (hourly)", "Standard
(hourly)", "Standard (BYOL)", "Cloud Volumes ONTAP Explore", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP
Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL",
"Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes ONTAP
Capacity Based Charging", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": true
            }, {
                "size": {
                    "size": 2.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Explore (hourly)", "Standard
(hourly)", "Standard (BYOL)", "Cloud Volumes ONTAP Explore", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP
Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL",
"Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes ONTAP
Capacity Based Charging", "Cloud Volumes ONTAP Explore", "Cloud Volumes

ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": false
            }, {
                "size": {
                    "size": 4.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Standard (hourly)", "Standard
(BYOL)", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes
ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity
Based Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP
Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL"],
                "isDefault": false
            }, {
                "size": {
                    "size": 8.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Standard (hourly)", "Standard
(BYOL)", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes
ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity
Based Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP
Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL"],
                "isDefault": false
            }, {
                "size": {
                    "size": 16.0,

223

```
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Standard (BYOL)", "Cloud
Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP
Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity Based
Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes
ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": false
            }, {
                "size": {
                    "size": 32.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Standard (BYOL)", "Cloud
Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP
Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity Based
Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes
ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": false
            }
        ]
    }
]
```

Get Azure storage account types for high availability pair

You can perform this workflow to retrieve the Azure storage account types for an HA working environment.

1. Get the storage account types

HTTP method	Path
GET	/occm/api/azure/ha/metadata/storage-account-types

curl example

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

Input

   None

Output

   The JSON output example includes the list of Azure storage account types.

JSON output example

```json
[{
        "diskType": "Premium",
        "availabilityTypes": ["Premium_LRS"],
        "sizes": [{
                "size": {
                        "size": 500.0,
                        "unit": "GB"
                },
                "description": "",
                "supportedOccmLicenses": ["Explore (hourly)", "Standard
(hourly)", "Standard (BYOL)", "Cloud Volumes ONTAP Explore", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP
Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL",
"Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes ONTAP
Capacity Based Charging", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": false
            }, {
                "size": {
                        "size": 1.0,
                        "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Explore (hourly)", "Standard
(hourly)", "Standard (BYOL)", "Cloud Volumes ONTAP Explore", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP
Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL",
"Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes ONTAP
Capacity Based Charging", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes
```

```
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": true
            }, {
                "size": {
                    "size": 2.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Explore (hourly)", "Standard
(hourly)", "Standard (BYOL)", "Cloud Volumes ONTAP Explore", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP
Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL",
"Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes ONTAP
Capacity Based Charging", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": false
            }, {
                "size": {
                    "size": 4.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Standard (hourly)", "Standard
(BYOL)", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes
ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity
Based Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP
Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL"],
                "isDefault": false
            }, {
                "size": {
                    "size": 8.0,
                    "unit": "TB"
```

226

```
                },
                "description": "",
                "supportedOccmLicenses": ["Standard (hourly)", "Standard
(BYOL)", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes
ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity
Based Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP
Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL"],
                "isDefault": false
            }, {
                "size": {
                    "size": 16.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Standard (BYOL)", "Cloud
Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP
Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity Based
Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes
ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": false
            }, {
                "size": {
                    "size": 32.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Standard (BYOL)", "Cloud
Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP
Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity Based
Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes
ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": false
            }
        ]
    }, {
        "diskType": "Standard",
```

```
        "availabilityTypes": ["Standard_LRS"],
        "sizes": [{
                "size": {
                        "size": 100.0,
                        "unit": "GB"
                },
                "description": "",
                "supportedOccmLicenses": ["Explore (hourly)", "Standard
(hourly)", "Standard (BYOL)", "Cloud Volumes ONTAP Explore", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP
Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL",
"Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes ONTAP
Capacity Based Charging", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": false
        }, {
                "size": {
                        "size": 500.0,
                        "unit": "GB"
                },
                "description": "",
                "supportedOccmLicenses": ["Explore (hourly)", "Standard
(hourly)", "Standard (BYOL)", "Cloud Volumes ONTAP Explore", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP
Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL",
"Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes ONTAP
Capacity Based Charging", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": false
        }, {
                "size": {
                        "size": 1.0,
                        "unit": "TB"
                },
```

```
                "description": "",
                "supportedOccmLicenses": ["Explore (hourly)", "Standard
(hourly)", "Standard (BYOL)", "Cloud Volumes ONTAP Explore", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP
Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL",
"Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes ONTAP
Capacity Based Charging", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": true
            }, {
                "size": {
                    "size": 2.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Explore (hourly)", "Standard
(hourly)", "Standard (BYOL)", "Cloud Volumes ONTAP Explore", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP
Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL",
"Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes ONTAP
Capacity Based Charging", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": false
            }, {
                "size": {
                    "size": 4.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Standard (hourly)", "Standard
(BYOL)", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes
ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity
Based Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud
```

Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP
Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL"],
                    "isDefault": false
            }, {
                "size": {
                    "size": 8.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Standard (hourly)", "Standard
(BYOL)", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes
ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity
Based Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP
Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL"],
                    "isDefault": false
            }, {
                "size": {
                    "size": 16.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Standard (BYOL)", "Cloud
Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP
Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity Based
Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes
ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                    "isDefault": false
            }, {
                "size": {
                    "size": 32.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Standard (BYOL)", "Cloud

```
Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP
Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity Based
Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes
ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": false
            }
        ]
    }, {
        "diskType": "StandardSSD",
        "availabilityTypes": ["StandardSSD_LRS"],
        "sizes": [{
                "size": {
                    "size": 100.0,
                    "unit": "GB"
                },
                "description": "",
                "supportedOccmLicenses": ["Explore (hourly)", "Standard
(hourly)", "Standard (BYOL)", "Cloud Volumes ONTAP Explore", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP
Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL",
"Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes ONTAP
Capacity Based Charging", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": false
            }, {
                "size": {
                    "size": 500.0,
                    "unit": "GB"
                },
                "description": "",
                "supportedOccmLicenses": ["Explore (hourly)", "Standard
(hourly)", "Standard (BYOL)", "Cloud Volumes ONTAP Explore", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP
Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL",
"Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes ONTAP
Capacity Based Charging", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
```

BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": false
            }, {
                "size": {
                    "size": 1.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Explore (hourly)", "Standard
(hourly)", "Standard (BYOL)", "Cloud Volumes ONTAP Explore", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP
Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL",
"Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes ONTAP
Capacity Based Charging", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": true
            }, {
                "size": {
                    "size": 2.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Explore (hourly)", "Standard
(hourly)", "Standard (BYOL)", "Cloud Volumes ONTAP Explore", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP
Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL",
"Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes ONTAP
Capacity Based Charging", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL", "Cloud Volumes ONTAP Explore", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": false

```
            }, {
                "size": {
                    "size": 4.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Standard (hourly)", "Standard
(BYOL)", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes
ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity
Based Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP
Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL"],
                "isDefault": false
            }, {
                "size": {
                    "size": 8.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Standard (hourly)", "Standard
(BYOL)", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium",
"Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes
ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity
Based Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud
Volumes ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes
ONTAP BYOL", "Cloud Volumes ONTAP Standard", "Cloud Volumes ONTAP
Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Standard",
"Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes
ONTAP Standard", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP
BYOL"],
                "isDefault": false
            }, {
                "size": {
                    "size": 16.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Standard (BYOL)", "Cloud
Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP
Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity Based
Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes
```

```
ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium",
 "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes
 ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": false
            }, {
                "size": {
                    "size": 32.0,
                    "unit": "TB"
                },
                "description": "",
                "supportedOccmLicenses": ["Standard (BYOL)", "Cloud
 Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP
 Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Capacity Based
 Charging", "Cloud Volumes ONTAP Capacity Based Charging", "Cloud Volumes
 ONTAP Premium", "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium",
 "Cloud Volumes ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes
 ONTAP BYOL", "Cloud Volumes ONTAP Premium", "Cloud Volumes ONTAP BYOL"],
                "isDefault": false
            }
        ]
    }
]
```

Get Azure VNets

You can perform this workflow to retrieve the network extended information.

> ℹ️   Ensure that you've the Azure subscription ID. You can easily obtain the ID from the Azure portal.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Get Azure VNets for single node

You can perform this workflow to retrieve the Azure VNets for a single node working environment.

1. Get the VNets

HTTP method	Path
GET	/occm/api/azure/vsa/metadata/vnets

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/azure/vsa/metadata/vnets?r
egion=<REGION>&subscriptionId=<SUBSCRIPTION_ID>' --header 'x-agent-id:
<AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>' --header
'Content-Type: application/json'
```

Input

Query parameters:

- <REGION> `region` string

- <SUBSCRIPTION_ID> `subscriptionId` string

- (Optional) `cloudProviderAccountId` string

Output

The JSON output example includes the list of Azure storage account types.

JSON output example

```
{
    "virtualNetworks": [
        {
            "name": "Vnet1",
            "id":
"/subscriptions/xxxxxxxx00000000000000/resourceGroups/occm_group_eastus2/p
roviders/Microsoft.Network/virtualNetworks/Vnet1",
            "cidrs": [
                {
                    "cidr": "10.0.0.0/16",
                    "subnets": [
                        {
                            "id":
"/subscriptions/xxxxxxxx00000000000000/resourceGroups/occm_group_eastus2/p
roviders/Microsoft.Network/virtualNetworks/Vnet1/subnets/Subnet1",
                            "cidr": "10.0.0.0/24",
                            "name": "Subnet1",
                            "availableIps": 250,
                            "minimumRequiredIps": 6
                        },
                        {
                            "id":
"/subscriptions/xxxxxxxx00000000000000/resourceGroups/occm_group_eastus2/p
roviders/Microsoft.Network/virtualNetworks/Vnet1/subnets/Subnet2",
                            "cidr": "10.0.1.0/24",
                            "name": "Subnet2",
                            "availableIps": 251,
```

```
                        "minimumRequiredIps": 6
                    },
                    {
                        "id":
"/subscriptions/xxxxxxx00000000000000/resourceGroups/occm_group_eastus2/p
roviders/Microsoft.Network/virtualNetworks/Vnet1/subnets/ProxySubnet",
                        "cidr": "10.0.2.0/24",
                        "name": "ProxySubnet",
                        "availableIps": 245,
                        "minimumRequiredIps": 6
                    }
                ]
            }
        ],
        "resourceGroup": "occm_group_eastus2",
        "tags": []
    },
    {
        "name": "Vnet2",
        "id":
"/subscriptions/xxxxxxx00000000000000/resourceGroups/occm_group_eastus2/p
roviders/Microsoft.Network/virtualNetworks/Vnet2",
        "cidrs": [
            {
                "cidr": "10.1.0.0/16",
                "subnets": [
                    {
                        "id":
"/subscriptions/xxxxxxx00000000000000/resourceGroups/occm_group_eastus2/p
roviders/Microsoft.Network/virtualNetworks/Vnet2/subnets/Subnet1",
                        "cidr": "10.1.0.0/24",
                        "name": "Subnet1",
                        "availableIps": 251,
                        "minimumRequiredIps": 6
                    },
                    {
                        "id":
"/subscriptions/xxxxxxx00000000000000/resourceGroups/occm_group_eastus2/p
roviders/Microsoft.Network/virtualNetworks/Vnet2/subnets/Subnet2",
                        "cidr": "10.1.1.0/24",
                        "name": "Subnet2",
                        "availableIps": 251,
                        "minimumRequiredIps": 6
                    },
                    {
                        "id":
```

```
    "/subscriptions/xxxxxxx00000000000000/resourceGroups/occm_group_eastus2/p
    roviders/Microsoft.Network/virtualNetworks/Vnet2/subnets/ProxySubnet",
                            "cidr": "10.1.2.0/24",
                            "name": "ProxySubnet",
                            "availableIps": 251,
                            "minimumRequiredIps": 6
                        }
                    ]
                }
            ],
            "resourceGroup": "occm_group_eastus2",
            "tags": []
        }
    ],
    "securityGroups": [
        {
            "id":
"/subscriptions/xxxxxxx00000000000000/resourceGroups/occm_group_eastus2/p
roviders/Microsoft.Network/networkSecurityGroups/OCCM-SG-467",
            "name": "OCCM-SG-467",
            "resourceGroup": "occm_group_eastus2"
        },
        {
            "id":
"/subscriptions/xxxxxxx00000000000000/resourceGroups/occm_group_eastus2/p
roviders/Microsoft.Network/networkSecurityGroups/OCCM-SG-491",
            "name": "OCCM-SG-491",
            "resourceGroup": "occm_group_eastus2"
        },
        {
            "id":
"/subscriptions/xxxxxxx00000000000000/resourceGroups/occm_group_eastus2/p
roviders/Microsoft.Network/networkSecurityGroups/OCCM-SG-589",
            "name": "OCCM-SG-589",
            "resourceGroup": "occm_group_eastus2"
        }
    ]
}
```

Get Azure VNets for high availability pair

You can perform this workflow to retrieve the Azure network extended information for an HA working environment.

1. Get the VNets

HTTP method	Path
GET	/occm/api/azure/ha/metadata/vnets

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/azure/ha/metadata/vnets?re
gion=<REGION>&subscriptionId=<SUBSCRIPTION_ID>' --header 'x-agent-id:
<AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>' --header
'Content-Type: application/json'
```

Input

Query parameters:

- <REGION> `region` string

- <SUBSCRIPTION_ID> `subscriptionId` string

- (Optional) `cloudProviderAccountId` string

Output

The JSON output example includes the list of Azure storage account types.

JSON output example

```
{
    "virtualNetworks": [
        {
            "name": "Vnet1",
            "id":
"/subscriptions/xxxxxxxx00000000000000/resourceGroups/occm_group_eastus2/p
roviders/Microsoft.Network/virtualNetworks/Vnet1",
            "cidrs": [
                {
                    "cidr": "10.0.0.0/16",
                    "subnets": [
                        {
                            "id":
"/subscriptions/xxxxxxxx00000000000000/resourceGroups/occm_group_eastus2/p
roviders/Microsoft.Network/virtualNetworks/Vnet1/subnets/Subnet1",
                            "cidr": "10.0.0.0/24",
                            "name": "Subnet1",
                            "availableIps": 250,
                            "minimumRequiredIps": 6
                        },
                        {
                            "id":
"/subscriptions/xxxxxxxx00000000000000/resourceGroups/occm_group_eastus2/p
```

```
roviders/Microsoft.Network/virtualNetworks/Vnet1/subnets/Subnet2",
                        "cidr": "10.0.1.0/24",
                        "name": "Subnet2",
                        "availableIps": 251,
                        "minimumRequiredIps": 6
                    },
                    {
                        "id":
"/subscriptions/xxxxxxxx00000000000000/resourceGroups/occm_group_eastus2/p
roviders/Microsoft.Network/virtualNetworks/Vnet1/subnets/ProxySubnet",
                        "cidr": "10.0.2.0/24",
                        "name": "ProxySubnet",
                        "availableIps": 245,
                        "minimumRequiredIps": 6
                    }
                ]
            }
        ],
        "resourceGroup": "occm_group_eastus2",
        "tags": []
    },
    {
        "name": "Vnet2",
        "id":
"/subscriptions/xxxxxxxx00000000000000/resourceGroups/occm_group_eastus2/p
roviders/Microsoft.Network/virtualNetworks/Vnet2",
        "cidrs": [
            {
                "cidr": "10.1.0.0/16",
                "subnets": [
                    {
                        "id":
"/subscriptions/dxxxxxxxx000000000000008/resourceGroups/occm_group_eastus2
/providers/Microsoft.Network/virtualNetworks/Vnet2/subnets/Subnet1",
                        "cidr": "10.1.0.0/24",
                        "name": "Subnet1",
                        "availableIps": 251,
                        "minimumRequiredIps": 6
                    },
                    {
                        "id":
"/subscriptions/xxxxxxxx00000000000000/resourceGroups/occm_group_eastus2/p
roviders/Microsoft.Network/virtualNetworks/Vnet2/subnets/Subnet2",
                        "cidr": "10.1.1.0/24",
                        "name": "Subnet2",
                        "availableIps": 251,
```

```
                           "minimumRequiredIps": 6
                        },
                        {
                           "id":
"/subscriptions/xxxxxxxx00000000000000/resourceGroups/occm_group_eastus2/p
roviders/Microsoft.Network/virtualNetworks/Vnet2/subnets/ProxySubnet",
                           "cidr": "10.1.2.0/24",
                           "name": "ProxySubnet",
                           "availableIps": 251,
                           "minimumRequiredIps": 6
                        }
                     ]
                  }
            ],
            "resourceGroup": "occm_group_eastus2",
            "tags": []
         }
      ],
      "securityGroups": [
         {
            "id":
"/subscriptions/dxxxxxxxx00000000000000/resourceGroups/occm_group_eastus2/
providers/Microsoft.Network/networkSecurityGroups/OCCM-SG-467",
            "name": "OCCM-SG-467",
            "resourceGroup": "occm_group_eastus2"
         },
         {
            "id":
"/subscriptions/xxxxxxxx00000000000000/resourceGroups/occm_group_eastus2/p
roviders/Microsoft.Network/networkSecurityGroups/OCCM-SG-491",
            "name": "OCCM-SG-491",
            "resourceGroup": "occm_group_eastus2"
         },
         {
            "id":
"/subscriptions/xxxxxxxx00000000000000/resourceGroups/occm_group_eastus2/p
roviders/Microsoft.Network/networkSecurityGroups/OCCM-SG-589",
            "name": "OCCM-SG-589",
            "resourceGroup": "occm_group_eastus2"
         }
      ]
}
```

Get Azure availability zones

You can perform this workflow to retrieve the Azure availability zones by region.

ⓘ Ensure that you've the Azure subscription ID. You can easily obtain the ID from the Azure portal.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node
- HA pair

Get availability zones for single node

You can use this workflow to retrieve the availability zones by region for a single node working environment.

1. Get the availability zones

HTTP method	Path
GET	/occm/api/azure/vsa/metadata/availability-zones

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/azure/vsa/metadata/availab
ility-zones?region=<REGION>&subscriptionId=<SUBSCRIPTION_ID>' --header 'x-
agent-id: <AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>'
--header 'Content-Type: application/json'
```

Input

  Query parameters:

- <REGION> `region` string
- (Optional) <SUBSCRIPTION_ID> `subscriptionId` string
- (Optional) `cloudProviderAccountId` string

Output

  The JSON output example includes the list of Azure availability zones.

JSON output example

```
{
    "region": "eastus2",
    "zones": [
        1,
        3,
        2
    ]
}
```

Get availability zones for high availability pair

You can use this workflow to retrieve the availability zones by region for a high availability pair working environment.

1. Get the availability zones

HTTP method	Path
GET	/occm/api/azure/ha/metadata/availability-zones

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/azure/ha/metadata/availabi
lity-zones?region=<REGION>&subscriptionId=<SUBSCRIPTION_ID>' --header 'x-
agent-id: <AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>'
--header 'Content-Type: application/json'
```

Input

Query parameters:

- <REGION> `region` string

- (Optional) <SUBSCRIPTION_ID> `subscriptionId` string

- (Optional) `cloudProviderAccountId` string

Output

The JSON output example includes the list of Azure availability zones.

JSON output example

```
{
    "region": "eastus2",
    "zones": [
        1,
        3,
        2
    ]
}
```

Get availability zones for high availability pair

You can use this workflow to retrieve the availability zones by region for a high availability pair working environment.

1. Get the availability zones

HTTP method	Path
GET	/occm/api/azure/ha/metadata/availability-zones

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/azure/ha/metadata/availabi
lity-zones?region=<REGION>&subscriptionId=<SUBSCRIPTION_ID>' --header 'x-
agent-id: <AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>'
--header 'Content-Type: application/json'
```

Input

Query parameters:

- <REGION> `region` string

- (Optional) <SUBSCRIPTION_ID> `subscriptionId` string

- (Optional) `cloudProviderAccountId` string

Output

The JSON output example includes the list of Azure storage account types.

JSON output example

```
{
    "region": "eastus2",
    "zones": [
        1,
        3,
        2
    ]
}
```

Get Azure packages

You can perform this workflow to retrieve the pre-defined packages configuration.

Choose the correct workflow depending on the type of Cloud Volumes ONTAP deployment:

- Single Node

- HA pair

Get packages for single node

You can use this workflow to retrieve the packages information for a single node working environment.

1. Get the packages

HTTP method	Path
GET	/occm/api/azure/vsa/metadata/packages

curl example

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

Input

None

Output

The JSON output example includes the packages information.

JSON output example

```
Output example  [
    {
        "name": "azure_poc",
        "displayName": "POC and small workloads",
        "description": "No description yet",
        "licenseType": "azure-cot-explore-paygo",
        "instanceTypeMapping": [
            {
                "region": "default",
                "instanceType": "Standard_DS3_v2"
            }
        ],
        "diskType": "Premium_LRS",
        "diskSize": {
            "size": 100.0,
            "unit": "GB"
        },
        "capacityTier": null,
        "instanceTenancy": null,
        "writingSpeedState": "NORMAL"
    },
    {
        "name": "azure_standard",
        "displayName": "Database and application data production
workloads",
        "description": "No description yet",
        "licenseType": "azure-cot-standard-paygo",
```

```json
    "instanceTypeMapping": [
        {
            "region": "default",
            "instanceType": "Standard_DS4_v2"
        }
    ],
    "diskType": "Premium_LRS",
    "diskSize": {
        "size": 100.0,
        "unit": "GB"
    },
    "capacityTier": null,
    "instanceTenancy": null,
    "writingSpeedState": "NORMAL"
},
{

    "name": "azure_dr",
    "displayName": "Cost effective DR",
    "description": "No description yet",
    "licenseType": "azure-cot-standard-paygo",
    "instanceTypeMapping": [
        {
            "region": "default",
            "instanceType": "Standard_DS4_v2"
        }
    ],
    "diskType": "Standard_LRS",
    "diskSize": {
        "size": 100.0,
        "unit": "GB"
    },
    "capacityTier": null,
    "instanceTenancy": null,
    "writingSpeedState": "NORMAL"
},
{

    "name": "azure_fastest",
    "displayName": "Highest performance production workloads",
    "description": "No description yet",
    "licenseType": "azure-cot-premium-paygo",
    "instanceTypeMapping": [
        {
            "region": "default",
            "instanceType": "Standard_DS5_v2"
        }
    ],
```

```
        "diskType": "Premium_LRS",
        "diskSize": {
            "size": 100.0,
            "unit": "GB"
        },
        "capacityTier": null,
        "instanceTenancy": null,
        "writingSpeedState": "NORMAL"
    }
]
```

Get packages for high availability pair

You can use this workflow to retrieve the packages information for an HA working environment.

1. Get the packages

HTTP method	Path
GET	/occm/api/azure/ha/metadata/packages

curl example

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

Input

None

Output

The JSON output example includes the packages information.

JSON output example

```
Output example  [
    {
        "name": "azure_poc",
        "displayName": "POC and small workloads",
        "description": "No description yet",
        "licenseType": "azure-cot-explore-paygo",
        "instanceTypeMapping": [
            {
                "region": "default",
                "instanceType": "Standard_DS3_v2"
            }
        ],
```

```
        "diskType": "Premium_LRS",
        "diskSize": {
            "size": 100.0,
            "unit": "GB"
        },
        "capacityTier": null,
        "instanceTenancy": null,
        "writingSpeedState": "NORMAL"
    },
    {
        "name": "azure_standard",
        "displayName": "Database and application data production
workloads",
        "description": "No description yet",
        "licenseType": "azure-cot-standard-paygo",
        "instanceTypeMapping": [
            {
                "region": "default",
                "instanceType": "Standard_DS4_v2"
            }
        ],
        "diskType": "Premium_LRS",
        "diskSize": {
            "size": 100.0,
            "unit": "GB"
        },
        "capacityTier": null,
        "instanceTenancy": null,
        "writingSpeedState": "NORMAL"
    },
    {
        "name": "azure_dr",
        "displayName": "Cost effective DR",
        "description": "No description yet",
        "licenseType": "azure-cot-standard-paygo",
        "instanceTypeMapping": [
            {
                "region": "default",
                "instanceType": "Standard_DS4_v2"
            }
        ],
        "diskType": "Standard_LRS",
        "diskSize": {
            "size": 100.0,
            "unit": "GB"
        },
```

```
        "capacityTier": null,
        "instanceTenancy": null,
        "writingSpeedState": "NORMAL"
    },
    {

        "name": "azure_fastest",
        "displayName": "Highest performance production workloads",
        "description": "No description yet",
        "licenseType": "azure-cot-premium-paygo",
        "instanceTypeMapping": [
            {
                "region": "default",
                "instanceType": "Standard_DS5_v2"
            }
        ],
        "diskType": "Premium_LRS",
        "diskSize": {
            "size": 100.0,
            "unit": "GB"
        },
        "capacityTier": null,
        "instanceTenancy": null,
        "writingSpeedState": "NORMAL"
    }
]
```

Get packages for high availability pair

You can use this workflow to retrieve the packages for an HA working environment.

1. Get the packages

HTTP method	Path
GET	/occm/api/azure/ha/metadata/packages

curl example

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

Input

None

Output

The JSON output example includes the list of Azure storage account types.

JSON output example

```
Output example [
    {
        "name": "azure_poc",
        "displayName": "POC and small workloads",
        "description": "No description yet",
        "licenseType": "azure-cot-explore-paygo",
        "instanceTypeMapping": [
            {
                "region": "default",
                "instanceType": "Standard_DS3_v2"
            }
        ],
        "diskType": "Premium_LRS",
        "diskSize": {
            "size": 100.0,
            "unit": "GB"
        },
        "capacityTier": null,
        "instanceTenancy": null,
        "writingSpeedState": "NORMAL"
    },
    {
        "name": "azure_standard",
        "displayName": "Database and application data production
workloads",
        "description": "No description yet",
        "licenseType": "azure-cot-standard-paygo",
        "instanceTypeMapping": [
            {
                "region": "default",
                "instanceType": "Standard_DS4_v2"
            }
        ],
        "diskType": "Premium_LRS",
        "diskSize": {
            "size": 100.0,
            "unit": "GB"
        },
        "capacityTier": null,
        "instanceTenancy": null,
        "writingSpeedState": "NORMAL"
    },
```

```
    {
        "name": "azure_dr",
        "displayName": "Cost effective DR",
        "description": "No description yet",
        "licenseType": "azure-cot-standard-paygo",
        "instanceTypeMapping": [
            {
                "region": "default",
                "instanceType": "Standard_DS4_v2"
            }
        ],
        "diskType": "Standard_LRS",
        "diskSize": {
            "size": 100.0,
            "unit": "GB"
        },
        "capacityTier": null,
        "instanceTenancy": null,
        "writingSpeedState": "NORMAL"
    },
    {
        "name": "azure_fastest",
        "displayName": "Highest performance production workloads",
        "description": "No description yet",
        "licenseType": "azure-cot-premium-paygo",
        "instanceTypeMapping": [
            {
                "region": "default",
                "instanceType": "Standard_DS5_v2"
            }
        ],
        "diskType": "Premium_LRS",
        "diskSize": {
            "size": 100.0,
            "unit": "GB"
        },
        "capacityTier": null,
        "instanceTenancy": null,
        "writingSpeedState": "NORMAL"
    }
]
```

Miscellaneous

Create Azure cloud provider account

You can use this workflow to create an Azure cloud provider account. An Azure cloud provider account holds Azure access keys required to perform actions in Azure.

Before you begin

You must have Azure credentials.

1. Get the account

HTTP method	Path
GET	/occm/api/accounts?providerType=AZURE

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/accounts?providerType=AZUR
E' --header 'x-agent-id: <AGENT_ID>' --header 'Authorization: Bearer
<ACCESS_TOKEN>' --header 'Content-Type: application/json'
```

Input

Query parameter (Optional):

`providerType` string

Output

The JSON output provides an example of the cloud provider details.

JSON output example

```
{
    "awsAccounts": [],
    "azureAccounts": [
        {
            "publicId": "ManagedServiceIdentity",
            "accountName": "Managed Service Identity",
            "accountType": "SERVICE_IDENTITY",
            "tenantId": "",
            "applicationId": "",
            "occmRole": "Azure",
            "vsaList": [
                {
                    "publicId": "VsaWorkingEnvironment-uFPaNkrv",
                    "name": "Azure123",
                    "workingEnvironmentType": "AZURE"
                }
            ]
        }
    ],
    "gcpStorageAccounts": [],
    "nssAccounts": []
}
```

GCP workflows

Before you begin

There are several workflows that you can use to deploy and manage Cloud Volumes ONTAP in Google Cloud.

ⓘ | Review the Get started section before using any of the Cloud Manager REST API workflows.

Workflow categories

The GCP workflows are organized into the following categories:

- Working environments
- Aggregates
- Volumes
- Metadata

See Understanding the workflow processes for more information on these categories.

ⓘ Cloud Volumes ONTAP requires a Google Cloud service account to run several background automation tasks such as data tiering and backup service. Learn more about GCP service accounts.

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 GCP cloud provider account that you deployed the Connector in to your list of available accounts. Your GCP account needs to have the right permissions in order to create a Connector.

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

Working environments

Create a working environment with PAYGO

You can use this workflow to create a new GCP Cloud Volumes ONTAP working environment using pay-as-you-go (PAYGO) subscription.

> **Note the following when using PAYGO**:
>
> - A marketplace subscription is required.
> - A NetApp Support Site (NSS) key is recommended to register the system for support, but it's not required.
> - You can add more volumes after creating the working environment. You can choose to create a volume using either NFS, CIFS, or iSCSI protocol.

1. Select the region

Perform the workflow Get regions and do the following:

- Choose the `name` value of the required region for the `region` parameter in step 8.

- Choose one of the VPCs. Choose `name` for `vpcId` parameter and `subnets: path` for `subnetId` and `subnetPath` parameters in step 8.

2. Select the workspace

Perform the workflow Get tenants and choose the `workspacePublicId` value for the `tenantId` parameter in step 8.

3. Select the projects

Perform the workflow Get projects and choose the `projectId` value of the required project for `project` parameter in step 8.

4. Select the permutations

Perform the workflow Get permutations and choose the `ontapVersion`, `license: type`, and `instanceType` values of the required `vsaMetadata` parameter in step 8.

5. Select the packages configuration

Perform the workflow Get packages and search the `licenseType` from permutations:

- Choose the `name` for `packageName` parameter.

- Choose the `diskSize` for `gcpVolumeSize`.

- Choose the `diskType` and `writingSpeedState` for the `gcpVolumeType` and `writingSpeedState` parameters.

6. Select the service account

Perform the workflow Get service accounts workflow and choose the `email` value of the required service accounts for the `gcpServiceAccount` parameter in step 8.

7. (Optional) Obtain an NSS key

An NSS key is **optional** when using PAYGO subscription. If needed, you can create a key or select an existing key, and include the NSS key in the `nssAccount` parameter in step 8.

- To create a new NSS key using the Cloud Manager web user interface, perform the task Generate NSS user ID and choose the `id`.

- To select an existing NSS key, perform the Get NSS keys workflow and choose the `id` value of the required NSS user.

8. Create the working environment

HTTP method	Path
POST	/occm/api/gcp/vsa/working-environments

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/gcp/vsa/working-
environments' --header 'x-agent-id: <AGENT_ID>' //<1> --header
'Authorization: Bearer <ACCESS_TOKEN>' //<2> ---header 'Content-Type:
application/json' --d @JSONinput
```

(1) Replace <AGENT_ID> with your agent ID.
(2) Replace <ACCESS_TOKEN> with your obtained bearer access token.

Input

The JSON input example includes the minimum list of parameters.

JSON input example

```json
{
  "name": "zivgcp01we03",
  "svmPassword": "password",
  "vpcId": "default",
  "region": "us-west1-b",
  "tenantId": "tenantID",
  "subnetPath": "projects/occm-dev/regions/us-west1/subnetworks/default",
  "subnetId": "projects/occm-dev/regions/us-west1/subnetworks/default",
  "dataEncryptionType": "GCP",
  "vsaMetadata": {
    "ontapVersion": "ONTAP-9.9.0X4.T1.gcp",
    "licenseType": "gcp-cot-explore-paygo",
    "instanceType": "custom-4-16384"
  },
  "gcpVolumeSize": {
    "size": 100,
    "unit": "GB"
  },
  "gcpVolumeType": "pd-ssd",
  "gcpLabels": [],
  "writingSpeedState": "NORMAL",
  "packageName": "gcp_poc",
  "gcpServiceAccount": "xxxxx@occm-dev.iam.xxx.com",
  "project": "occm-dev",
  "backupVolumesToCbs": false
}
```

JSON output example

```
{
    "publicId": "vsaworkingenvironment-2qkd75xv",
    "name": "zivgcp01we03",
    "tenantId": "tenantIDshownhere",
    "svmName": "svm_zivgcp01we03",
    "creatorUserEmail": "user_email",
    "status": null,
    "providerProperties": null,
    "reservedSize": null,
    "clusterProperties": null,
    "ontapClusterProperties": null,
    "cloudProviderName": "GCP",
    "snapshotPolicies": null,
    "actionsRequired": null,
    "activeActions": null,
    "replicationProperties": null,
    "schedules": null,
    "svms": null,
    "workingEnvironmentType": "VSA",
    "supportRegistrationProperties": null,
    "supportRegistrationInformation": null,
    "capacityFeatures": null,
    "encryptionProperties": null,
    "supportedFeatures": null,
    "isHA": false,
    "haProperties": null,
    "k8sProperties": null,
    "fpolicyProperties": null,
    "saasProperties": null,
    "cbsProperties": null,
    "complianceProperties": null,
    "monitoringProperties": null
}
```

Create a working environment with BYOL

You can use this workflow to create a new Cloud Volumes ONTAP working environment using bring your own license (BYOL) licensing.

> **Note the following when using BYOL licensing**:
>
> - A marketplace subscription is not required.
> - A NetApp Support Site (NSS) key is required to register the system for support.
> - You can add more volumes after creating the working environment. You can choose to create a volume using either NFS, CIFS, or iSCSI protocol.

1. Select the region

Perform the workflow Get regions and do the following:

- Choose the `name` value of the required region for the `region` parameter in step 8.
- Choose one of the VPCs. Choose `name` for `vpcId` parameter and `subnets: path` for `subnetId` and `subnetPath` parameters in step 8.

2. Select the workspace

Perform the workflow Get tenants and choose the `workspacePublicId` value for the `tenantId` parameter in step 8.

3. Select the project

Perform the workflow Get projects and choose the `projectId` value of the required project for `project` parameter in step 8.

4. Select the permutations

Perform the workflow Get permutations and choose the `ontapVersion`, `license: type`, and `instanceType` values of the required `vsaMetadata` parameter in step 8.

5. Select the packages configuration

Perform the workflow Get packages and search the `licenseType` from permutations:

- Choose the `name` for `packageName` parameter.
- Choose the `diskSize` for `gcpVolumeSize`.
- Choose the `diskType` and `writingSpeedState` for the `gcpVolumeType` and `writingSpeedState` parameters.

6. Select the service account

Perform the workflow Get service accounts workflow and choose the `email` value of the required service accounts for the `gcpServiceAccount` parameter in step 8.

7. Obtain an NSS key

An NSS key is **required** when using BYOL subscription. If needed, you can create a key or select an existing key, and include the NSS key in the `nssAccount` parameter in step 8.

- To create a new NSS key using the Cloud Manager web user interface, perform the task Generate NSS user ID and choose the `id`.

- To select an existing NSS key, perform the Get NSS keys workflow and choose the `id` value of the required NSS user.

8. Create the working environment

HTTP method	Path
POST	/occm/api/gcp/vsa/working-environments

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/gcp/vsa/working-
environments' --header 'x-agent-id: <AGENT_ID>' //<1> --header
'Authorization: Bearer <ACCESS_TOKEN>' //<2> ---header 'Content-Type:
application/json' --d @JSONinput
```

(1) Replace <AGENT_ID> with your agent ID.
(2) Replace <ACCESS_TOKEN> with your obtained bearer access token.

Input

The JSON input example includes the minimum list of parameters. This request uses BYOL licensing as indicated in the `licenseType` parameter. The `serialNumber` is required.

JSON input example

```
{
name: "gcpwe123"
backupVolumesToCbs: true
capacityTier: "cloudStorage"
dataEncryptionType: "GCP"
enableCompliance: true
gcpLabels: []
gcpServiceAccount: "fabric-pool@occm-dev.iam.gserviceaccount.com"
gcpVolumeSize: {size: 500, unit: "GB", _identifier: "500 GB"}
gcpVolumeType: "pd-ssd"
name: "gcpwe123"
nssAccount: "0xxx-000-4c70-9cee-304f36b74db6"
packageName: "gcp_poc"
project: "occm-dev"
region: "europe-west3-c"
serialNumber: "00000108000000000000"
subnetId: "projects/occm-dev/regions/europe-west3/subnetworks/vpc4qa-2-
europe-west3"
subnetPath: "projects/occm-dev/regions/europe-west3/subnetworks/vpc4qa-2-
europe-west3"
svmPassword: "Netapp123"
tenantId: "workspaceNqaJyVMz"
tierLevel: "standard"
volume: {
  exportPolicyInfo: {
    policyType: "custom",
    ips: ["172.22.13.0/24"],
    nfsVersion: ["nfs3", "nfs4"]
  }
}
vpcId: "vpc4qa-2"
vsaMetadata: {
  ontapVersion: "ONTAP-9.10.1RC1.T1.gcp",
  licenseType: "gcp-cot-premium-byol"
}
instanceType: "n2-standard-4"
licenseType: "gcp-cot-premium-byol"
ontapVersion: "ONTAP-9.10.1RC1.T1.gcp"
writingSpeedState: "NORMAL"
}
```

Output

The JSON output example includes an example of the `VsaWorkingEnvironmentRresponse` response.

JSON output example

```
{
  "publicId": "vsaworkingenvironment-9nhkrtu0",
  "name": "yuvalbyol3101",
  "tenantId": "tenantIDshownhere",
  "svmName": "svm_yuvalbyol3101",
  "creatorUserEmail": "user_email",
  "status": null,
  "providerProperties": null,
  "reservedSize": null,
  "clusterProperties": null,
  "ontapClusterProperties": null,
  "cloudProviderName": "GCP",
  "snapshotPolicies": null,
  "actionsRequired": null,
  "activeActions": null,
  "replicationProperties": null,
  "schedules": null,
  "svms": null,
  "workingEnvironmentType": "VSA",
  "supportRegistrationProperties": null, "supportRegistrationInformation":
 null,
  "capacityFeatures": null,
  "encryptionProperties": null,
  "supportedFeatures": null,
  "isHA": false,
  "haProperties": null,
  "fpolicyProperties": null,
  "saasProperties": null,
  "cbsProperties": null,
  "complianceProperties": null,
  "monitoringProperties": null,
  "licensesInformation": null
}
```

Get working environment

You can retrieve the public identifier, working environment ID, the storage virtual machine name for Cloud Volumes ONTAP working environments and other Cloud Volumes ONTAP related details (visible to currently logged in user) which would be used in other workflows.

1. Get the working environments

HTTP method	Path
GET	/occm/api/gcp/vsa/working-environments/{workingEnvironmentId}

curl

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

Input

(Optional) Query parameters:

- `fields` string

- `tenantId` string

Output

The JSON output example includes details of a single node Google Cloud working environment.

JSON output example

```
[
    {
        "publicId": "vsaworkingenvironment-sfrf3wvj",
        "name": "zivgcp01we02",
        "tenantId": "tenantIDshownhere",
        "svmName": "svm_zivgcp01we02",
        "creatorUserEmail": "user_email",
        "status": null,
        "providerProperties": null,
        "reservedSize": null,
        "clusterProperties": null,
        "ontapClusterProperties": null,
        "cloudProviderName": "GCP",
        "snapshotPolicies": null,
        "actionsRequired": null,
        "activeActions": null,
        "replicationProperties": null,
        "schedules": null,
        "svms": null,
        "workingEnvironmentType": "VSA",
        "supportRegistrationProperties": null,
        "supportRegistrationInformation": [],
        "capacityFeatures": null,
        "encryptionProperties": null,
        "supportedFeatures": null,
        "isHA": false,
        "haProperties": null,
        "k8sProperties": null,
        "fpolicyProperties": null,
```

```
        "saasProperties": null,
        "cbsProperties": null,
        "complianceProperties": null,
        "monitoringProperties": null
    },
    {

        "publicId": "vsaworkingenvironment-2qkd75xv",
        "name": "zivgcp01we03",
        "tenantId": "tenantIdshownhere",
        "svmName": "svm_zivgcp01we03",
        "creatorUserEmail": "user_email",
        "status": null,
        "providerProperties": null,
        "reservedSize": null,
        "clusterProperties": null,
        "ontapClusterProperties": null,
        "cloudProviderName": "GCP",
        "snapshotPolicies": null,
        "actionsRequired": null,
        "activeActions": null,
        "replicationProperties": null,
        "schedules": null,
        "svms": null,
        "workingEnvironmentType": "VSA",
        "supportRegistrationProperties": null,
        "supportRegistrationInformation": [],
        "capacityFeatures": null,
        "encryptionProperties": null,
        "supportedFeatures": null,
        "isHA": false,
        "haProperties": null,
        "k8sProperties": null,
        "fpolicyProperties": null,
        "saasProperties": null,
        "cbsProperties": null,
        "complianceProperties": null,
        "monitoringProperties": null
    }
]
```

Delete a working environment

You can delete an existing GCP Cloud Volumes ONTAP working environment.

1. Select the working environment

Perform the workflow Get working environment and choose the `publicId` value of the working environment for the `workingEnvironmentId` path parameter.

2. Delete the working environment

HTTP method	Path
DELETE	/occm/api/gcp/vsa/working-environments/{workingEnvironmentId}

Curl example

```
curl --location --request DELETE
'https://cloudmanager.cloud.netapp.com/occm/api/gcp/vsa/working-
environments/<WORKING_ENV_ID>' --header 'Content-Type: application/json'
--header 'x-agent-id: <AGENT_ID>' --header 'Authorization: Bearer
<ACCESS_TOKEN>'
```

Input

  Path parameter <WORKING_ENV_ID> (`workingEnvironmentId`) string

(Optional) Query parameters:

* `localDelete` boolean

  If `true` the Cloud Volumes ONTAP instance in the cloud is not terminated, but Cloud Manager no longer manages it (default is `false`). If `false` the Cloud Volumes ONTAP instance is deleted including all the cloud resources created for this working environment.

* `forceDelete` boolean

  If `true` the working environment is deleted even if it is part of one or more SnapMirror relationships (default is `false`).

  Output
    None

Create CIFS server configuration

If you want to create CIFS volumes on your Cloud Volumes ONTAP system, you first need to configure the CIFS server. You can choose to set up the CIFS server in a workgroup or in an Active Directory domain. Review the ONTAP documentation for more information.

Choose the workflow that is specific to your goal:

* Set up a CIFS server in a workgroup
* Set up a CIFS server in an Active Directory domain

Set up a CIFS server in a workgroup

You can configure a CIFS server in a workgroup when the Microsoft Active Directory domain infrastructure is not available.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value for the working environment used in the `workingEnvironmentId` path parameter.

2. Create the CIFS configuration

Create the CIFS server configuration.

HTTP method	Path
POST	/occm/api/gcp/vsa/working-environments/{workingEnvironmentId}/cifs-workgroup

curl example

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

Input
- Path parameter <WORKING_ENV_ID> `workingEnvironmentId` string

JSON input example

```
{
  "serverName": "SMB_SERVER02",
  "workgroupName": "workgroup02",
  "svmName": "svm_ziv01we01"
}
```

Output
  None.

Set up a CIFS server in an Active Directory domain

You can create a CIFS server on the SVM and specify the Active Directory (AD) domain to which it belongs.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value for the working environment used in the `workingEnvironmentId` path parameter.

2. Determine the Active Directory configuration

You need the following configuration parameters for an Active Directory server.

Input parameter	Description
dnsDomain	Use the Active Directory domain as the DNS name.
ipAddresses	Define the primary DNS IP address and optionally add a secondary IP address.
netBIOS	Use the CIFS server NetBIOS name.
organizationalUnit	Include the organizational unit as appropriate.
activeDirectoryDomain	Set the Active Directory domain to join.
activeDirectoryUsername	A username with authorization to join the domain.
activeDirectoryPassword	The password for the authorized username.

3. Create the CIFS configuration

Create the CIFS server configuration.

HTTP method	Path
POST	/occm/api/gcp/vsa/working-environments/{workingEnvironmentId}/cifs

curl example

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

Input

- Path parameter <WORKING_ENV_ID> `workingEnvironmentId` string

JSON input example

```
{
  "dnsDomain": "zivh.netapp.com",
  "ipAddresses": [
    "172.31.5.241"
  ],
  "netBIOS": "zivaws02we03",
  "organizationalUnit": "CN=Computers",
  "activeDirectoryDomain": "zivh.netapp.com",
  "activeDirectoryUsername": "administrator",
  "activeDirectoryPassword": "password"
}
```

Output

None.

Get CIFS server configurations

You can use this workflow to retrieve the CIFS server configurations for an existing Cloud Volumes ONTAP working environment.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value for the working environment used in the `workingEnvironmentId` path parameter.

2. Get the CIFS configurations

HTTP method	Path
GET	/occm/api/gcp/vsa/working-environments/{workingEnvironmentId}/cifs

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/gcp/vsa/working-
environments/<WORKING_ENV_ID>/cifs' --header 'Content-Type:
application/json' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

- Path parameter <WORKING_ENV_ID> `workingEnvironmentId` string

- (Optional) Query parameter `svm` string

Output

The JSON output example includes the CIFS configurations for an existing Cloud Volumes ONTAP ONTAP working environment.

JSON output example

```
[
    {
        "dnsDomain": "zivh.netapp.com",
        "activeDirectoryDomain": "zivh.netapp.com",
        "ipAddresses": [
            "172.31.5.241"
        ],
        "netBIOS": "zivaws02we01",
        "organizationalUnit": "CN=Computers",
        "authenticationType": "domain"
    }
]
```

Delete CIFS server configuration

You can use this workflow to delete a CIFS server configuration for an existing Cloud Volumes ONTAP working environment.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value for the working environment used in the `workingEnvironmentId` path parameter.

2. Delete the CIFS configurations

HTTP method	Path
POST	/occm/api/gcp/vsa/working-environments/{workingEnvironmentId}/delete-cifs

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/gcp/vsa/working-
environments/<WORKING_ENV_ID>/delete-cifs' --header 'Content-Type:
application/json' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

- Path parameter <WORKING_ENV_ID> `workingEnvironmentId` string
- Optional JSON body

```
{
    "activeDirectoryUsername": "string",
    "activeDirectoryPassword": "string",
    "svmName": "string"
}
```

Output

None.

Aggregates

Get aggregates

You can retrieve a list of available disk aggregates for Cloud Volumes ONTAP in Google Cloud.

1. Select the working environment

Perform the workflow Get working environment and choose the `publicId` value of the working environment for the `workingEnvironmentId` path parameter.

2. Get the list of aggregates

HTTP method	Path
GET	/occm/api/gcp/vsa/aggregates/{workingEnvironmentId}

curl example

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

Input

Path parameter:

- <WORKING_ENV_ID> (`workingEnvironmentId`) string

Output

An array of aggregates for the indicated working environment is returned as shown in the JSON output example.

> ⓘ The capacity (sizes) in the output are in MB/GB/TB (1000th order) because these are ONTAP aggregates, whereas in Cloud Manager the capacity is specified as MiB, GiB (1024 order).

JSON output example

```
[
    {
        "name": "aggr1",
        "availableCapacity": {
            "size": 87.55,
            "unit": "GB"
        },
        "totalCapacity": {
            "size": 88.57,
            "unit": "GB"
        },
        "usedCapacity": {
            "size": 1.02,
            "unit": "GB"
        },
        "volumes": [
            {
                "name": "svm_zivgcp01we02_root",
                "totalSize": {
                    "size": 1.0,
                    "unit": "GB"
                },
                "usedSize": {
                    "size": 7.59124755859375E-4,
                    "unit": "GB"
                },
                "thinProvisioned": false,
                "isClone": false,
                "rootVolume": true
            }
        ],
        "providerVolumes": [
            {
                "id": "00000000000000000",
                "name": "zivgcp01we02datadisk1",
                "size": {
                    "size": 100.0,
                    "unit": "GB"
                },
                "state": "READY",
                "device": "zivgcp01we02datadisk1",
                "instanceId": "zivgcp01we02",
                "diskType": "pd-ssd",
                "encrypted": true,
```

```
                "iops": null
            }
        ],
        "disks": [
            {
                "name": "NET-1.2",
                "position": "data",
                "ownerNode": "zivgcp01we02-01",
                "device": "zivgcp01we02datadisk1",
                "vmDiskProperties": null
            }
        ],
        "state": "online",
        "encryptionType": "cloudEncrypted",
        "encryptionKeyId": null,
        "isRoot": false,
        "homeNode": "zivgcp01we02-01",
        "ownerNode": "zivgcp01we02-01",
        "capacityTier": null,
        "capacityTierUsed": null,
        "sidlEnabled": true,
        "snaplockType": "non_snaplock"
    }
]
```

Create aggregate

You can create new aggregate within a Cloud Volumes ONTAP working environment using this workflow.

1. Select the working environment

Perform the workflow Get working environment and choose the `publicId` value for the `workingEnvironmentId` parameter in the JSON input.

2. Select the GCP disk types

Perform the Get GCP disk types workflow and choose the `size` and `supportedDiskType` values of the required `diskSize` and `providerVolumeType` parameters in the JSON input.

3. Create the aggregate

HTTP method	Path
POST	occm/api/gcp/vsa/aggregates

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/gcp/vsa/aggregates'
--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.

JSON input example

```
{
  "name": "ziv01agg01",
  "workingEnvironmentId": "vsaworkingenvironment-sfrf3wvj",
  "numberOfDisks": 1,
  "diskSize": {
    "size": 100,
    "unit": "GB"
  },
  "providerVolumeType": "pd-ssd"
}
```

Output

None

Add disks to aggregate

You can add disks to an existing aggregate.

1. Select the working environment

Perform the workflow Get GCP single node working environment and choose the `publicId` value for the `workingEnvironmentId` path parameter.

2. Select the aggregate

Perform the workflow Get aggregates and choose `name` of the required aggregate for the `aggregateName` path parameter.

3. Add the disks to the aggregate

HTTP method	Path
POST	/occm/api/gcp/vsa/aggregates/{workingEnvironmentId}/{aggregateName}/disks

curl example

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

Input

You must include the following path parameters:

- <WORKING_ENV_ID> `(workingEnvironmentId)` string
- <AGGR_NAME> `(aggregateName)` string

Also, the JSON input example includes an input parameter as shown.

JSON input example

```
{
    "numberOfDisks": "1"
}
```

Output

None

Delete aggregate

You can delete an existing disk aggregate in a Cloud Volumes ONTAP working environment.

1. Select the working environment

Perform the workflow Get working environment and choose the `publicId` value of the working environment for the `workingEnvironmentId` path parameter.

2. Select the aggregate

Perform the workflow Get aggregates and choose the `name` value of the required aggregate for the `aggregateName` path parameter.

3. Delete the aggregate

HTTP method	Path
DELETE	/occm/api/gcp/vsa/aggregates/{workingEnvironmentId}/{aggregateName}

curl example

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

Input

Path parameters:

- <WORKING_ENV_ID> (`workingEnvironmentId`) string
- <AGGR_NAME> (`aggregateName`) string

Output

None

Volumes

Create a volume using NFS

You can use this workflow to create a volume accessed through NFS protocol.

> ⓘ   If the properties `aggregateName` and `maxNumOfDisksApprovedToAdd` are not provided on
> the REST API call, the response will fail with a suggested name for the aggregate and the
> number of disks needed to fulfill the request.

1. Select the working environment

Perform the workflow [Get working environement](#) and choose the `publicId` and the `svmName` values for the
`workingEnvironmentId` and the `svmName` parameters.

2. Select the aggregate

Perform the workflow [Get aggregates](#) and choose the `name` value of the aggregate for the `name` parameter.

> ⓘ   If aggregate name does not exist and the `createAggregateIfNotFound` query parameter is
> set `true`, the create volume request is allowed if the named aggregate is not found.

3. Choose the size for the disk

Choose the size value for the `size:size` parameter. The `size:unit` must be one of the following: `TB`, `GB`,
`MB`, `KB`, or `Byte`.

4. Select the region

Perform the workflow [Get regions](#) workflow and pick `ipCidrRange` value of the required `region→ subnets`
for `exportPolicy→ips` value.

5. Create the quote

Perform the workflow Create quote. This is a recommended step but is not mandatory.

6. Create the volume

HTTP method	Path
POST	/occm/api/gcp/vsa/volumes

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/gcp/vsa/volumes' --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, including:

- <WORKING_ENV_ID> (workingEnvironmentId)
- <SVM_NAME> (svmName)
- <AGGR_NAME> (aggregateName)

If aggregate name does not exist, you can set the `createAggregateIfNotFound` query parameter to `true` which allows the aggregate not-found condition.

JSON input example

```
{
  "workingEnvironmentId": "vsaworkingenvironment-sfrf3wvj",
  "svmName": "svm_zivgcp01we02",
  "aggregateName": "ziv01agg01",
  "name": "zivagg01vol01",
  "size": {
    "size": 100,
    "unit": "GB"
  },
  "snapshotPolicyName": "default",
  "enableThinProvisioning": true,
  "enableCompression": true,
  "enableDeduplication": true,
  "maxNumOfDisksApprovedToAdd": 0,
  "exportPolicyInfo": {
      "ips": [
          "10.138.0.0/20"
      ],
      "nfsVersion": [
          "nfs3", "nfs4"
      ],
      "policyType": "custom"
  }
}
```

Output

  None

Create a volume using CIFS

You can use this workflow to create a volume accessed through CIFS.

> ⓘ If the properties `aggregateName` and `maxNumOfDisksApprovedToAdd` are not provided on
> the REST API call, the response will fail with a suggested name for the aggregate and the
> number of disks needed to fulfill the request.

1. Choose the CIFS configuration

A CIFS server configuration must be defined for your working environment. You can do one of the following:

- If a CIFS configuration already exists, perform the workflow Get CIFS server configurations to access the
  configuration parameters.
- If a CIFS configuration does not exist, perform the workflow Create CIFS server configuration to create
  one.

2. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value for the `workingEnvironmentId` (working environment) and the `svmName` (SVM name).

3. Select the aggregate

Perform the workflow Get aggregates and choose the `name` for the `aggregateName` value.

> (i) If aggregate name does not exist and the `createAggregateIfNotFound` query parameter is set `true`, the create volume request is allowed if the named aggregate is not found.

4. Choose the size for the disk

Choose the size value for the `size:size` parameter. The `size:unit` must be one of the following: `TB`, `GB`, `MB`, `KB`, or `Byte`.

5. Create the quote

Perform the workflow Create quote. This is a recommended step but is not mandatory.

6. Create the volume

HTTP method	Path
POST	/occm/api/gcp/vsa/volumes

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/gcp/vsa/volumes' --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, including:

- <WORKING_ENV_ID> (`workingEnvironmentId`) string
- <SVM_NAME> (`svmName`) string
- <AGGR_NAME> (`aggregateName`) string

If an aggregate name does not exist, you can set the `createAggregateIfNotFound` query parameter to `true` which allows the aggregate not-found condition.

JSON input example

```
{
   "workingEnvironmentId": "VsaWorkingEnvironment-SfpVUZSc",
   "svmName": "svm_zivaws02we01",
   "aggregateName": "aggr1",
   "name": "zivaws02we02vol02Cifs",
   "size": {
     "size": 100,
     "unit": "GB"
   },
   "shareInfo": {
     "accessControl": {
         "permission": "full_control",
         "users": [
             "Everyone"
         ],
         "users": "Everyone;"
     },
     "shareName": "zivaws02we01vol02Cifs_share"
   },
   "snapshotPolicyName": "default",
   "enableThinProvisioning": true,
   "enableCompression": true,
   "enableDeduplication": true,
   "maxNumOfDisksApprovedToAdd": 0
}
```

Output

   None

Create a volume using iSCSI

You can use this workflow to create a volume accessed through iSCSI. There are two workflows available depending on whether a new or existing iGroup is used. You need to select the correct workflow:

- Create volume using iSCSI with a new iGroup
- Create volume using iSCSI with an existing iGroup

Create volume using iSCSI with a new iGroup

> (i) If the properties `aggregateName` and `maxNumOfDisksApprovedToAdd` are not provided on the REST API call, the response will fail with a suggested name for the aggregate and the number of disks needed to fulfill the request.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value for the `workingEnvironmentId` parameter and the `svmName` value for the `svmName` parameter.

2. Select the aggregate

Perform the workflow Get aggregates and choose the `name` for the `aggregateName` value.

3. Choose the size for the disk

Choose the size value for the `size:size` parameter. The `size:unit` must be one of the following: TB, GB, MB, KB, or Byte.

4. Choose the iscasiInfo parameters

You must choose the following values for the REST API call:

* A unique igroup name for `igroupCreationRequest → igroupName` parameter
* The required iqn's to `igroupCreationRequest → initiators` parameter.
* The required operating system for the `osName` parameter from one of the following:
  ◦ windows
  ◦ linux
  ◦ vmware
  ◦ windows_2008
  ◦ windows_gpt

5. Create the quote

Perform the workflow Create quote.This is a recommended step but is not mandatory.

6. Create the volume

HTTP method	Path
POST	/occm/api/gcp/vsa/volumes

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/gcp/vsa/volumes' --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, including:

* <WORKING_ENV_ID> (`workingEnvironmentId`) string

- <SVM_NAME> (`svmName`) string

- <AGGR_NAME> (`aggregateName`) string

If aggregate name does not exist, you can set the `createAggregateIfNotFound` query parameter to `true` which allows the aggregate not-found condition.

JSON input example

```json
{
  "workingEnvironmentId": "VsaWorkingEnvironment-SfpVUZSc",
  "svmName": "svm_zivaws02we01",
  "aggregateName": "aggr1",
  "name": "zivaws02we01vol01Iscsi",
  "size": {
    "size": 100,
    "unit": "GB"
  },
  "iscsiInfo": {
    "igroupCreationRequest": {
        "igroupName": "zivIgroup",
        "initiators": [
            "iqn.1994-05.com.redhat:96de86825216",
            "iqn.1994-05.com.redhat:96de86823426"
        ]
    },
    "osName": "linux"
  },
  "snapshotPolicyName": "default",
  "enableThinProvisioning": true,
  "enableCompression": true,
  "enableDeduplication": true,
  "maxNumOfDisksApprovedToAdd": 0
}
```

Output

None

Create volume using iSCSI with an existing iGroup

> (i) If the properties `aggregateName` and `maxNumOfDisksApprovedToAdd` are not provided on the REST API call, the response will fail with a suggested name for the aggregate and the number of disks needed to fulfill the request.

1. Select the working environment

Perform the workflow Get working environments and choose the `publicId` value for the `workingEnvironmentId` parameter and the `svmName` value for the `svmName` parameter.

2. Select the aggregate

Perform the workflow Get aggregates and choose the `name` for the `aggregateName` value.

3. Choose the size for the disk

Choose the size value for the `size:size` parameter. The `size:unit` must be one of the following: `TB`, `GB`, `MB`, `KB`, or `Byte`.

4. Choose the iGroup

Perform the workflow Get iGroups and choose the igroups for the `iscasiInfo → igroups` value. Also select the `osType` value for the `iscasiInfo → osName`.

5. Create the quote

Perform the workflow Create quote. This is a recommended step but is not mandatory.

6. Create the volume

HTTP method	Path
POST	/occm/api/gcp/vsa/volumes

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/gcp/vsa/volumes' --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, including:

- <WORKING_ENV_ID> (`workingEnvironmentId`) string
- <SVM_NAME> (`svmName`) string
- <AGGR_NAME> (`aggregateName`) string

If an aggregate name does not exist, you can set the `createAggregateIfNotFound` query parameter to `true` which allows the aggregate not-found condition.

JSON input exmaple

```
{
  "workingEnvironmentId": "VsaWorkingEnvironment-UvFmWXoD",
  "svmName": "svm_zivaws01we01",
  "aggregateName": "aggr1",
  "name": "zivaws01we01vol05Iscsi",
  "size": {
    "size": 100,
    "unit": "GB"
  },
  "iscsiInfo": {
    "igroups": ["zivIgroup1"],
    "osName": "linux"
  },
  "snapshotPolicyName": "default",
  "enableThinProvisioning": true,
  "enableCompression": true,
  "enableDeduplication": true,
  "maxNumOfDisksApprovedToAdd": 0
}
```

Output

None

Get volumes

You can retrieve the list of volumes of a single node Azure working environment.

1. Select the working environment

Perform the workflow Get working environment and choose the `publicId` value of the working environment for the `workingEnvironmentId` query parameter.

2. Get the volumes

HTTP method	Path
GET	/occm/api/gcp/vsa/volumes

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/gcp/vsa/volumes?workingEnv
ironmentId=<WORKING_ENV_ID>' --header 'Content-Type: application/json'
--header 'x-agent-id: <AGENT_ID>' --header 'Authorization: Bearer
<ACCESS_TOKEN>'
```

Input

Query parameter <WORKING_ENV_ID> (`workingEnvironmentId`) string

Output

The JSON output example includes the list of volumes for the working environment.

JSON output example

```
[
    {
        "name": "zivagg01vol01",
        "uuid": "0x00000-0xx0-00xx-xx00-00xxxx000",
        "svmName": "svm_zivgcp01we02",
        "size": {
            "size": 100.0,
            "unit": "GB"
        },
        "usedSize": {
            "size": 2.93731689453125E-4,
            "unit": "GB"
        },
        "junctionPath": "/zivagg01vol01",
        "volumeTotalInodes": 3112959,
        "volumeUsedInodes": 96,
        "mountPoint": "10.138.0.150:/zivagg01vol01",
        "compressionSpaceSaved": {
            "size": 0.0,
            "unit": "GB"
        },
        "deduplicationSpaceSaved": {
            "size": 0.0,
            "unit": "GB"
        },
        "thinProvisioning": true,
        "compression": true,
        "deduplication": true,
        "snapshotPolicy": "default",
        "securityStyle": "unix",
        "exportPolicyInfo": {
            "name": "export-svm_zivgcp01we02-zivagg01vol01",
            "policyType": "custom",
            "ips": [
                "10.138.0.0/20"
            ],
            "nfsVersion": [
                "nfs3",
                "nfs4"
```

```
        ]
    },
    "shareNames": [],
    "shareInfo": [],
    "parentVolumeName": "",
    "rootVolume": false,
    "state": "online",
    "volumeType": "rw",
    "aggregateName": "ziv01agg01",
    "parentSnapshot": null,
    "autoSizeMode": "grow",
    "maxGrowSize": {
        "size": 1100.0,
        "unit": "GB"
    },
    "providerVolumeType": "pd-ssd",
    "cloneNames": [],
    "moving": false,
    "primaryNoFailoverMountPoint": null,
    "secondaryNoFailoverMountPoint": null,
    "capacityTier": null,
    "capacityTierUsedSize": null,
    "cifsShareAccessPoint": null,
    "primaryCifsShareAccessPoint": null,
    "secondaryCifsShareAccessPoint": null,
    "tieringPolicy": "none",
    "tierInactiveUserData": null,
    "tierInactiveUserDataPercent": null,
    "comment": null,
    "qosPolicyGroupName": null,
    "snaplockType": "non_snaplock",
    "constituentsAggregates": [],
    "snapshotsUsedSize": {
        "size": 0.0,
        "unit": "Byte"
    },
    "cbsBackupsInfo": null,
    "minimumCoolingDays": null,
    "targetName": "iqn.1992-
08.com.netapp:sn.986656ab5e3e11eb9cb735a0758d479a:vs.2",
    "iscsiEnabled": false,
    "isFlexGroupVolume": false
    }
]
```

Modify volume

You can modify the configuration of an existing volume.

1. Select the working environment

Perform the workflow Get working environment and choose the `publicId` and `svmName` values of the working environment for used as the `workingEnvironmentId` and `svmName` path parameters.

2. Select the volume

Perform the workflow Get volumes and choose the `name` for the `volumeName` path parameter.

3. Modify the volume

HTTP method	Path
PUT	/occm/api/gcp/vsa/volumes/{workingEnvironmentId}/{svmName}/{volumeName}

curl example

```
curl --location --request PUT
'https://cloudmanager.cloud.netapp.com/occm/api/gcp/vsa/volumes/<WORKING_E
NV_ID>/<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`) string

- <SVM_NAME> (`svmName`) string

- <VOLUME_NAME> (`volumeName`) string

The JSON input example includes the minimum list of input parameters.

JSON input example

```
{
  "exportPolicyInfo": {
    "policyType": "custom",
    "ips": [
      "10.000.0.0/20"
    ],
    "nfsVersion": [
      "nfs3"
    ]
  }
}
```

Output

  None

Delete volume

You can delete an existing volume.

1. Select the working environment

Perform the Get working environment workflow and choose the `publicId` and `svmName` values of the required working environment for `workingEnvironmentId` and `svmName` path parameters.

2. Select the volume

Perform the workflow Get volumes and choose the `name` for the `volumeName` path parameter.

3. Delete the volume

HTTP method	Path
DELETE	/occm/api/gcp/vsa/volumes/{workingEnvironmentId}/{svmName}/{volumeName}

curl example

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

Input

  Path parameters:

- <WORKING_ENV_ID> (workingEnvironmentId) string

- <SVM_NAME> (svmName) string

- <VOLUME_NAME> (`volumeName`) string

Output

None

Create quote

You can create a quote for a new volume which returns a resource quote needed to satisfy the request. The resource quote contains aggregate information where the volume will be created and confirms if the space is available. This is a recommended step but is not mandatory.

1. Select the working environment

Perform the workflow Get working environment and choose the `publicId` and 'svmName' values of the required working environment for `workingEnvironmentId` and `svmName` parameters in the JSON input.

2. Select the aggregate

Perform the Get aggregates workflow and choose the `name` value of the required aggregate for the `aggregateName` parameter in the JSON input.

3. Select the gcp disk type

Perform the Get gcp disk types workflow and choose the `size` and `supportedDiskType` values of the required disk type for the `diskSize` and `providerVolumeType` parameters in the JSON input.

4. Generate the volume quote

HTTP method	Path
POST	/occm/api/gcp/vsa/quote

curl example

```
curl --location --request POST
'https://cloudmanager.cloud.netapp.com/occm/api/gcp/vsa/volumes/quote'
--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 list of input parameters.

JSON input example

```
{
    "workingEnvironmentId": "vsaworkingenvironment-sfrf3wvj",
    "svmName": "svm_zivgcp01we02",
    "aggregateName": "ziv01agg01",
    "name": "zivagg01vol01",
    "size": {
      "size": "100",
      "unit": "GB"
    },
    "enableThinProvisioning": "true",
    "providerVolumeType": "pd-ssd",
    "verifyNameUniqueness": "true"
}
```

Output

The JSON output example includes an example of the quote details.

JSON output example

```
{
    "numOfDisks": 0,
    "diskSize": {
        "size": 100.0,
        "unit": "GB"
    },
    "aggregateName": "ziv01agg01",
    "newAggregate": false,
    "autoVsaCapacityManagement": true
}
```

Get iGroups

You can use this workflow to retrieve all the initiator groups (iGroups).

1. Create the working environment

Perform the workflow Create GCP single node working environment and choose the `publicId` and `svmName` values for the working environment `workingEnvironmentId` and `svmName` path parameters.

2. Get the CIFS configurations

HTTP method	Path
GET	/occm/api/gcp/vsa/volumes/igroups/{workingEnvironmentId}/{svmName}

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/gcp/vsa/volumes/igroups/<W
ORKING_ENV_ID>/<SVM_NAME>' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>' --header 'Content-Type:
application/json'
```

Input

- Path parameter <WORKING_ENV_ID> `workingEnvironmentId` string

- Path parameter <SVM_NAME> `svmName` string

Output

The JSON output example includes a list of iGroups.

JSON output example

```
[
    {
        "igroupName": "zivIgroup1",
        "osType": "linux",
        "portsetName": "",
        "igroupType": "iscsi",
        "initiators": [
            "iqn.1994-05.com.redhat:1d9ac633937c"
        ]
    },
    {
        "igroupName": "zivIgroup2",
        "osType": "linux",
        "portsetName": "",
        "igroupType": "iscsi",
        "initiators": [
            "iqn.1994-05.com.redhat:96de86825216"
        ]
    }
]
```

Metadata

Get GCP regions

This workflow retrieves the GCP regions in which a Cloud Volumes ONTAP working
environment might be created.

1. Get the list of regions

HTTP method	Path
GET	/occm/api/gcp/vsa/metadata/regions

curl example

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

Input

None

Output

The JSON output provides an example of a list of GCP regions.

JSON output example

```
[
    {
        "displayName": "asia-east1",
        "name": "asia-east1",
        "zones": [
            {
                "name": "asia-east1-a"
            },
            {
                "name": "asia-east1-b"
            },
            {
                "name": "asia-east1-c"
            }
        ],
        "vpcs": [
            {
                "name": "default",
                "subnets": [
                    {
                        "ipCidrRange": "10.140.0.0/20",
                        "name": "default",
                        "path": "projects/occm-dev/regions/asia-
east1/subnetworks/default",
                        "availableIps": 4090,
                        "minimumRequiredIps": 6
                    }
```

```json
                ],
                "firewalls": [
                    {
                        "name": "allow-all",
                        "vpc": "default"
                    },
                    {
                        "name": "allow-ssh-netapp",
                        "vpc": "default"
                    }
                ]
            },
            {
                "name": "eli-vpc",
                "subnets": [
                    {
                        "ipCidrRange": "10.0.0.0/00",
                        "name": "eli-subnet",
                        "path": "projects/occm-dev/regions/asia-east1/subnetworks/eli-subnet",
                        "availableIps": 250,
                        "minimumRequiredIps": 6
                    }
                ],
                "firewalls": [
                    {
                        "name": "eli-vpc-allow-http",
                        "vpc": "eli-vpc"
                    },
                    {
                        "name": "eli-vpc-allow-https",
                        "vpc": "eli-vpc"
                    }
                ]
            },
        ],
    },
    {
        "displayName": "asia-northeast1",
        "name": "asia-northeast1",
        "zones": [
            {
                "name": "asia-northeast1-a"
            },
            {
                "name": "asia-northeast1-b"
```

```
            },
            {
                "name": "asia-northeast1-c"
            }
        ],
        "vpcs": [
            {
                "name": "default",
                "subnets": [
                    {
                        "ipCidrRange": "00.000.0.0/20",
                        "name": "default",
                        "path": "projects/occm-dev/regions/asia-
northeast1/subnetworks/default",
                        "availableIps": 4090,
                        "minimumRequiredIps": 6
                    }
                ],
                "firewalls": [
                    {
                        "name": "allow-all",
                        "vpc": "default"
                    },
                    {
                        "name": "allow-ssh-netapp",
                        "vpc": "default"
                    }
                ]
            }
        ]
    }
}
```

Get GCP permutations

You can use the permutations endpoint to retrieve the Cloud Volumes ONTAP configuration information such as `ontapVersion`, `license`, `instanceType`, `region` and more. You can check the possible permutations that can potentially be provided for the GCP workflows while provisioning a Cloud Volumes ONTAP instance.

1. Get the permutations

HTTP method	Path
GET	/occm/api/gcp/vsa/metadata/permutations

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/gcp/vsa/metadata/permutati
ons?latest_only=true' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>' --header 'Content-Type:
application/json'
```

Input

There are several **optional** query parameters you can use:

- `region` string

- `version` string

- `license` string

- `machine_type` string

- `latest_only` string

Output

The JSON output example includes the list of Cloud Volumes ONTAP configurations.

JSON output example

```json
[
    {
        "ontapVersion": "ONTAP-9.9.0X4.T1.gcp",
        "license": {
            "type": "gcp-cot-explore-paygo",
            "name": "Cloud Volumes ONTAP Explore",
            "description": "Suitable for smaller capacity applications.
Supports up to 2 TB of underlying GCP storage.",
            "subName": "",
            "subDescription": "Support of tiering to object storage is not
included.",
            "capacity_limit": "2TB",
            "platformLicenseRequired": false,
            "default": false,
            "capacityLimit": {
                "size": 2.0,
                "unit": "TB"
            }
        },
        "instanceType": "custom-4-16384",
        "region": {
            "name": "asia east 1",
            "code": "asia-east1",
```

```
            "location": "Changhua County, Taiwan",
            "s3Region": null
        },
        "defaultInstance": false,
        "features": [
            "cpu:Intel Skylake"
        ],
        "upgradeableFrom": [
            "9.8",
            "9.9.0"
        ]
    },
    {
        "ontapVersion": "ONTAP-9.9.0X4.T1.gcp",
        "license": {
            "type": "gcp-cot-explore-paygo",
            "name": "Cloud Volumes ONTAP Explore",
            "description": "Suitable for smaller capacity applications.
Supports up to 2 TB of underlying GCP storage.",
            "subName": "",
            "subDescription": "Support of tiering to object storage is not
included.",
            "capacity_limit": "2TB",
            "platformLicenseRequired": false,
            "default": false,
            "capacityLimit": {
                "size": 2.0,
                "unit": "TB"
            }
        },
        "instanceType": "custom-4-16384",
        "region": {
            "name": "asia east 2",
            "code": "asia-east2",
            "location": "Hong Kong",
            "s3Region": null
        },
        "defaultInstance": false,
        "features": [
            "cpu:Intel Skylake"
        ],
        "upgradeableFrom": [
            "9.8",
            "9.9.0"
        ]
    }
```

```
    ]
```

Get tag keys

You can perform this workflow to retrieve all the labels in the specified project and zone.

1. Select the project

Perform the workflow Get projects and choose `projectId` value of the required project for `projectId` query parameter.

2. Select the region

Perform the Get regions workflow and choose the `zone: name` value of the required region for the `zone` query parameter.

3. Get the tag keys

HTTP method	Path
GET	/occm/api/gcp/vsa/metadata/tag-keys

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/gcp/vsa/metadata/tag-
keys?projectId=<PROJECT_ID>&zone=<ZONE>' --header 'x-agent-id: <AGENT_ID>'
--header 'Authorization: Bearer <ACCESS_TOKEN>' --header 'Content-Type:
application/json'
```

Input
Query parameters:

- <PROJECT_ID> `projectId` string
- <ZONE> `zone` string

Output
The JSON output example includes the list of Azure storage account types.

JSON output example

```
[
    {
        "key": "working-environment-id",
        "values": [
            "vsaworkingenvironment-sfrf3wvj",
            "vsaworkingenvironment-2qkd75xv"
        ]
    },
```

```json
    {
        "key": "count-down",
        "values": [
            "3",
            "0",
            "2"
        ]
    },
    {
        "key": "username",
        "values": [
            "administrator"
        ]
    },
    {
        "key": "keepme",
        "values": [
            "10"
        ]
    },
    {
        "key": "cloud-ontap-version",
        "values": [
            "9_9_0x4"
        ]
    },
    {
        "key": "cloud-ontap-dm",
        "values": [
            "zivgcp01we02-deployment",
            "zivgcp01we03-deployment"
        ]
    },
    {
        "key": "platform-serial-number",
        "values": [
            "00000030000000000009",
            "00000000000000096011"
        ]
    }
]
    },
    {
        "key": "netapp:cloud-compliance:cloudManager:ClientId",
        "values": [
            "sNwn2FzHxFrucwz8j1huxNIYI7aRNqTC"
```

```
              ]
          }
      ]
```

Create Buckets

You can perform this workflow to create a new bucket. This is a GCP storage bucket (data container in GCP) required for data tiering and backup.

1. Create a bucket

HTTP method	Path
POST	/occm/api/gcp/vsa/metadata/create-bucket

curl example

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

JSON input example

```
{
    "projectId": "occm-dev",
    "bucketName": "zivgcpbucket02",
    "location": "us-west1",
    "storageClass": "standard"
}
```

Output
   None

Get buckets

You can perform this workflow to retrieve the S3 buckets. The S3 buckets are the GCP storage buckets (data containers in GCP) required for data tiering and backups.

1. Get the buckets

HTTP method	Path
GET	/occm/api/vsa/metadata/buckets

curl example

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

Input

None

Output

The JSON output example includes the list of S3 buckets.

JSON output example

```
[
    {
        "bucketName": "3.9.0.bins.08112020",
        "region": "us-east-1",
        "tags": {}
    },
    {
        "bucketName": "0000000000-awsmacietrail-dataevent",
        "region": "us-east-1",
        "tags": {}
    }
]
```

Get GCP packages

You can perform this workflow to retrieve the pre-defined packages configuration.

1. Get the packages

HTTP method	Path
GET	/occm/api/gcp/vsa/metadata/packages

curl example

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

Input

None

Output

The JSON output example includes the list of GCP packages.

JSON output example

```json
[
    {
        "name": "gcp_poc",
        "displayName": "POC and small workloads",
        "description": "No description yet",
        "licenseType": "gcp-cot-explore-paygo",
        "instanceTypeMapping": [
            {
                "region": "default",
                "instanceType": "custom-4-16384"
            }
        ],
        "diskType": "pd-ssd",
        "diskSize": {
            "size": 100.0,
            "unit": "GB"
        },
        "capacityTier": null,
        "instanceTenancy": null,
        "writingSpeedState": "NORMAL"
    },
    {
        "name": "gcp_standard",
        "displayName": "Database and application data production
 workloads",
        "description": "No description yet",
        "licenseType": "gcp-cot-standard-paygo",
        "instanceTypeMapping": [
            {
                "region": "default",
                "instanceType": "n1-standard-8"
            }
        ],
        "diskType": "pd-ssd",
        "diskSize": {
            "size": 100.0,
            "unit": "GB"
        },
        "capacityTier": "GCP",
        "instanceTenancy": null,
        "writingSpeedState": "NORMAL"
    },
```

```
    {
        "name": "gcp_dr",
        "displayName": "Cost effective DR",
        "description": "No description yet",
        "licenseType": "gcp-cot-standard-paygo",
        "instanceTypeMapping": [
            {
                "region": "default",
                "instanceType": "n1-standard-8"
            }
        ],
        "diskType": "pd-standard",
        "diskSize": {
            "size": 100.0,
            "unit": "GB"
        },
        "capacityTier": "GCP",
        "instanceTenancy": null,
        "writingSpeedState": "NORMAL"
    },
    {
        "name": "gcp_fastest",
        "displayName": "Highest performance production workloads",
        "description": "No description yet",
        "licenseType": "gcp-cot-premium-paygo",
        "instanceTypeMapping": [
            {
                "region": "default",
                "instanceType": "n1-standard-32"
            }
        ],
        "diskType": "pd-ssd",
        "diskSize": {
            "size": 100.0,
            "unit": "GB"
        },
        "capacityTier": "GCP",
        "instanceTenancy": null,
        "writingSpeedState": "NORMAL"
    }
]
```

Get Snapshot policies

You can perform this workflow to retrieve the default snapshot policies available on the cluster.

1. Get the snapshot policies

HTTP method	Path
GET	/occm/api/gcp/vsa/metadata/default-snapshot-policies

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/gcp/vsa/metadata/default-
snapshot-policies' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>' --header 'Content-Type:
application/json'
```

Input

None

Output

The JSON output example includes the list of default snapshot policies available on the cluster.

JSON output example

```
[
    {
        "name": "default",
        "schedules": [
            {
                "frequency": "hourly",
                "retention": 6
            },
            {
                "frequency": "daily",
                "retention": 2
            },
            {
                "frequency": "weekly",
                "retention": 2
            }
        ],
        "description": "Default policy with hourly, daily & weekly
 schedules."
    },
    {
        "name": "none",
        "schedules": [],
        "description": "Policy for no automatic snapshots."
    }
]
```

Get supported features

You can perform this workflow to retrieve and check the supported features while provisioning a Cloud Volumes ONTAP instance.

1. Select the permutations

Perform the workflow Get permutations and choose the `ontapVersion`, `license: type`, `instanceType`, and `region: code` values of the required permutations for the query parameters of `supported features` request.

2. Get the supported features

HTTP method	Path
GET	/occm/api/gcp/vsa/metadata/supported-features

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/gcp/vsa/metadata/supported
-features?region=<REGION>&ontapVersion=<ONTAP_VERSION>&dataEncryptionType=
<ENCRP_TYPE>&licenseType=<LICENSE_TYPE>&instanceType=<INST_TYPE>' --header
'x-agent-id: <AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>'
--header 'Content-Type: application/json'
```

Input

Query parameters:

- <REGION> `region` string

- <ONTAP_VERSION> `ontapVersion` string

- <LICENSE_TYPE> `licenseType` string

- <INST_TYPE> `instanceType` string

- <ENCRP_TYPE> `dataEncryptionType` string

JSON output example

```
{
    "wormSupportedVersion": true,
    "cbsSupportedVersion": true,
    "httpsStorageAccountSupportedVersion": false,
    "tieringWithServiceAccount": true
}
```

Get supported capacity tiers

You can retrieve the supported capacity tiers for Google Cloud disk types.

1. Select the permutations

Perform the workflow Get permutations and choose the `ontapVersion`, `license: type`, `instanceType`, and `region: code` values of the required permutations for the query parameters of `supported capacity tiers` request.

2. Get the supported features

HTTP method	Path
GET	/occm/api/gcp/vsa/metadata/supported-capacity-tiers

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/gcp/vsa/metadata/supported
-capacity
-tiers?region=<REGION>&ontapVersion=<ONTAP_VERSION>&dataEncryptionType=<EN
CRP_TYPE>&licenseType=<LICENSE_TYPE>&instanceType=<INST_TYPE>' --header
'x-agent-id: <AGENT_ID>' --header 'Authorization: Bearer <ACCESS_TOKEN>'
--header 'Content-Type: application/json'
```

Input

Query parameters:

- <REGION> `region` string

- <ONTAP_VERSION> `ontapVersion` string

- <LICENSE_TYPE> `licenseType` string

- <INST_TYPE> `instanceType` string

- <ENCRP_TYPE> `dataEncryptionType` string

Output

The output shows a list of supported capacity tiers for the GCP disk types.

JSON output example

```
{
    "supportedCapacityTiersPerVolumeType": [
        {
            "volumeType": "pd-standard",
            "supportedCapacityTiers": [
                "cloudStorage"
            ],
            "availableTieringPolicies": [
                "none",
                "snapshot_only"
            ]
        },
        {
            "volumeType": "pd-ssd",
            "supportedCapacityTiers": [
                "cloudStorage"
            ],
            "availableTieringPolicies": [
                "none",
                "snapshot_only"
            ]
        }
    ],
    "capacityTiersDisableReasons": [
        "Cannot create capacity tiered volume on Cloud Volumes ONTAP
 Explore license"
    ],
    "compositeSupported": true,
    "forceCompositeVersion": false
}
```

Get service accounts

This workflow retrieves a list of service accounts from the specified project.

1. Get the list of service accounts

HTTP method	Path
GET	/occm/api/gcp/vsa/metadata/service-accounts

curl example

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

Input

None

Output

The JSON output retrieves a list of service accounts from the project.

JSON output example

```
{
"accounts": [
    {
        "name": "projects/occm-dev/serviceAccounts/00000-
compute@serviceaccount.com",
        "projectId": "occm-dev",
        "email": "00000-compute@serviceaccount.com",
        "displayName": "Compute Engine default service account",
        "isEnabled": false
    },
    {
        "name": "projects/occm-dev/serviceAccounts/xxx-
000@occmaccount.com",
        "projectId": "occm-dev",
        "email": "xxxx-000@occmaccount.com",
        "displayName": "admin",
        "isEnabled": false
    }
]
}
```

Get projects

This workflow retrieves the list of projects that the caller has permission on.

1. Get the list of regions

HTTP method	Path
GET	/occm/api/gcp/vsa/metadata/projects

curl example

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

Input

None

Output

The JSON output provides an example of a list of authorized projects.

JSON output example

```
{
    "projects": [
        {
            "isDefault": false,
            "projectNumber": "958377592668",
            "projectId": "occm-host",
            "lifecycleState": "ACTIVE",
            "name": "OCCM-host",
            "createTime": "2019-07-24T14:36:32.472Z",
            "parent": {
                "type": "folder",
                "id": "339830134733"
            },
            "subscriptionId": null
        },
        {
            "isDefault": false,
            "projectNumber": "844924364732",
            "projectId": "occm-slave",
            "lifecycleState": "ACTIVE",
            "name": "OCCM-slave",
            "createTime": "2019-07-24T14:36:32.405Z",
            "parent": {
                "type": "folder",
                "id": "339830134733"
            },
            "subscriptionId": null
        },
        {
            "isDefault": true,
            "projectNumber": "92083494653",
            "projectId": "occm-dev",
            "lifecycleState": "ACTIVE",
            "name": "OCCM-Dev",
            "createTime": "2018-05-24T17:23:50.505Z",
            "parent": {
                "type": "folder",
                "id": "339830134733"
            },
            "subscriptionId": "gcp-saasMpIntegrationProductId-
saasMpCustomerIdentifier6"
        }
    ]
}
```

Get GCP encryption keys

This workflow retrieves the GCP encryption keys for a specific region.

1. Select the region

Perform the workflow Get regions and choose the `name` value of the required region for the `region` path parameter.

2. Select the project

Perform the get projects workflow and choose the `projectId` value of the required project for the `project` path parameter.

3. Get the encryption keys

HTTP method	Path
GET	/occm/api/gcp/vsa/metadata/gcp-encryption-keys

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/gcp/vsa/metadata/gcp-
encryption-keys?region=<REGION>&project=<PROJECT>' --header 'Content-Type:
application/json' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>'
```

Input
Required query parameters:

- <REGION> `region`: string
- <PROJECT> `project`: string

Output
The JSON output provides an example of a list of GCP encryption keys for a specific region.

JSON output example

```
[
    {
        "name": "key1",
        "id": "projects/occm-
dev/locations/global/keyRings/test/cryptoKeys/key1",
        "keyRing": "test",
        "location": "global"
    },
    {
        "name": "key2",
        "id": "projects/occm-
dev/locations/global/keyRings/test/cryptoKeys/key2",
        "keyRing": "test",
        "location": "global"
    }
]
```

Get GCP disk types

This workflow retrieves the GCP supported disk types.

1. Get the disk types

HTTP method	Path
GET	/occm/api/gcp/vsa/metadata/gcp-disk-types

curl example

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

Input

None

Output

The JSON output provides an example of a list of GCP supported disk types.

JSON output example

```
[
    {
        "size": {
            "size": 100.0,
            "unit": "GB"
        },
        "supportedDiskTypes": [
            "pd-standard",
            "pd-ssd"
        ],
        "supportedOccmLicenses": [
            "Standard (BYOL)",
            "Cloud Volumes ONTAP Standard",
            "Cloud Volumes ONTAP Explore",
            "Standard (hourly)",
            "Cloud Volumes ONTAP BYOL",
            "Explore (hourly)",
            "Cloud Volumes ONTAP Premium"
        ]
    },
    {
        "size": {
            "size": 500.0,
            "unit": "GB"
        },
        "supportedDiskTypes": [
            "pd-standard",
            "pd-ssd"
        ],
        "supportedOccmLicenses": [
            "Standard (BYOL)",
            "Cloud Volumes ONTAP Standard",
            "Cloud Volumes ONTAP Explore",
            "Standard (hourly)",
            "Cloud Volumes ONTAP BYOL",
            "Explore (hourly)",
            "Cloud Volumes ONTAP Premium"
        ]
    }
]
```

Get instance types not supporting acceleration and capacity tiering

This workflow retrieves the GCP instance types which do not support the acceleration and capacity tiering. This means that these kind instance types do not use any sort of

hardware accelerator. You can check the instance types while provisioning the Cloud Volumes ONTAP.

1. Get the instance types

HTTP method	Path
GET	occm/api/gcp/vsa/metadata/instance-types-not-supporting-acceleration-and-capacity-tiering

curl example

```
curl --location --request GET
'https://cloudmanager.cloud.netapp.com/occm/api/gcp/vsa/metadata/instance-
types-not-supporting-acceleration-and-capacity-tiering' --header 'Content-
Type: application/json' --header 'x-agent-id: <AGENT_ID>' --header
'Authorization: Bearer <ACCESS_TOKEN>'
```

Input

None

Output

The JSON output provides an example of a list of instance types.

JSON output example

```
{
    "instanceTypes": []
}
```

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

```json
{
    "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",
```

```json
        "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.

> (i) 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

```json
{
  "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}/{destination VolumeName}

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

```json
[
    {
        "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.

ℹ️ 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",
```

```json
            "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.

> ℹ️ 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

```json
[
    {
        "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
            ]
        }
    }
]
```