Oracle Private Cloud Appliance

User Guide

F74804-15
May 2025

ORACLE

Oracle Private Cloud Appliance User Guide,

F74804-15

Contents

2 Identity and Access Management

5 Load Balancing

6 Compute Images

7 Compute Instance Deployment

11 File System Storage

12 Object Storage

Preface

This publication is part of the customer documentation set for Oracle Private Cloud Appliance Release 3.0. Note that the documentation follows the release numbering scheme of the appliance software, not the hardware on which it is installed. All Oracle Private Cloud Appliance product documentation is available at https://docs.oracle.com/en/engineered-systems/private-cloud-appliance/index.html.

Oracle Private Cloud Appliance Release 3.x is a flexible general purpose Infrastructure as a Service solution, engineered for optimal performance and compatibility with Oracle Cloud Infrastructure. It allows customers to consume the core cloud services from the safety of their own network, behind their own firewall.

Audience

This documentation is intended for owners, administrators and operators of Oracle Private Cloud Appliance. It provides architectural and technical background information about the engineered system components and services, as well as instructions for installation, administration, monitoring and usage.

Oracle Private Cloud Appliance has two strictly separated operating areas, known as enclaves. The Compute Enclave offers a practically identical experience to Oracle Cloud Infrastructure: It allows users to build, configure and manage cloud workloads using compute instances and their associated cloud resources. The Service Enclave is where privileged administrators configure and manage the appliance infrastructure that provides the foundation for the cloud environment. The target audiences of these enclaves are distinct groups of users and administrators. Each enclave also provides its own separate interfaces.

It is assumed that readers have experience with system administration, network and storage configuration, and are familiar with virtualization technologies. Depending on the types of workloads deployed on the system, it is advisable to have a general understanding of container orchestration, and UNIX and Microsoft Windows operating systems.

Feedback

Provide feedback about this documentation at https://www.oracle.com/goto/docfeedback.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.

Convention	Meaning
`monospace`	Monospace type indicates commands within a paragraph, code in examples, text that appears on the screen, or text that you enter.
`$` prompt	The dollar sign (`$`) prompt indicates a command run as a non-root user.
`#` prompt	The pound sign (`#`) prompt indicates a command run as the `root` user.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at https://www.oracle.com/corporate/accessibility/.

Access to Oracle Support for Accessibility

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit https://www.oracle.com/corporate/accessibility/learning-support.html#support-tab.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

1

Working in the Compute Enclave

The Compute Enclave is the part of the Private Cloud Appliance where you work with and manage cloud resources.

This section describes the general usage principles of the Compute Enclave graphical user interface, command line interface, and REST API.

This section also describes how you can view your resource limits.

Using the Compute Web UI

The Compute Web UI is the graphical interface to the Compute Enclave. You can use the Compute Web UI on its own or with the OCI CLI to complete tasks. The Compute Web UI provides the same core functionality as the OCI CLI; however, the OCI CLI has some additional functionality.

This section provides instructions for logging into the Compute Web UI, navigating the dashboard, and working with resources using resource type and resource detail pages. Within the rest of the *Oracle Private Cloud Appliance User Guide* you learn how to use the Compute Web UI to complete tasks within the context of the step-by-step procedures.

> **Note:**
>
> You access the Compute Web UI using a web browser. For support information, please refer to the Oracle software web browser support policy.

Logging In

Before you log into the Compute Web UI, make sure you have the Private Cloud Appliance system and domain names, the tenancy name, and your user name and password. If you do not have these details, ask your administrator. If you have access to the Service Web UI, you can locate the tenancy name and the system and domain names for your Private Cloud Appliance.

To log into the Compute Web UI, complete the following steps.

1. From a browser, enter the URL for your Private Cloud Appliance.

   For example, `https://console.pcasys1.example.com` where ***`pcasys1`*** is the name of your Private Cloud Appliance and ***`example.com`*** is your domain.

   The Compute Enclave Select Tenancy page is displayed.

2. Enter your tenancy name and click Continue.

   The Sign In page is displayed.

3. Enter your Username and Password, and then click Sign In.

   The Private Cloud Appliance dashboard displays with quick action tiles.

> **Note:**
>
> If you are prompted to change a temporary password, see Setting Your Password.

Navigating the Dashboard

When you log into the Compute Enclave, the dashboard is displayed with quick action tiles for common tasks, such as viewing compute instances, block and file storage, and VCNs. There is also a quick action tile to create a virtual machine instance.

> **Note:**
>
> The dashboard is static and not configurable.

You can click on or tab to the dashboard tiles and the navigation menu. The navigation menu (the three lines to the left of "Oracle Private Cloud Appliance") is a list of services. When you click on a service, the sub-menu expands and displays the resource types for that service. When you click on a resource type, a page is displayed that contains a tabular list of resources related to that resource type. The following table provides the Private Cloud Appliance services and their respective resource types as they are displayed in the navigation menu.

Service	Resource Types in Sub-Menu
Compute	• Instances • Instance Exports • Instance Imports • • Instance Configurations • Instance Pools • Autoscaling Configurations • Custom Images For more information, see Compute Instance Deployment.
Block Storage	• Block Volumes • Block Volume Backups • Boot Volumes • Boot Volume Backups • Volume Groups • Volume Group Backups • Backup Policies For more information, see Block Volume Storage.
File Storage	• File Systems • Mount Targets For more information, see File System Storage.
Object Storage	• Object Storage For more information, see Object Storage.

Service	Resource Types in Sub-Menu
IP Management	• Reserved Public IPs For more information, see Reserving a Public IP Address.
Networking	• Virtual Cloud Networks • Load Balancers • Network Load Balancers • Dynamic Routing Gateways For more information, see Networking.
Containers	• Kubernetes Clusters (OKE) For more information, see the Oracle Private Cloud Appliance Kubernetes Engine user guide.
DNS	• Zones • Steering Policies • TSIG Keys For more information, see Networking.
Identity	• Users • Groups • Dynamic Groups • Policies • Compartments • Federation For more information, see Identity and Access Management.
Governance	• Tag Namespaces For more information, see Creating and Managing Tag Namespaces.

Using Resource Type and Resource Detail Pages

Resource type and resource detail pages are what you use to work with resources in a tenancy or other compartment. A resource type page displays a list of all resources of that type and also contains the service's sub-menu. When you click on a resource in the list, its own detail page is displayed. Every resource detail page has some general information about the resource, such as its OCID, when it was created, the compartment it is in, and any tags associated with it.

About Resource Type Pages

A resource type page contains a list of resources in table format, one resource per row. The rows of the table are in alphabetical order by the name of the resource.

Columns in a resource type table include the name of the resource type, State, Created, and Actions, as well as columns that are specific to that resource type. The Actions column contains the Actions menu (three dots) for the resource, which contains options such as View Details, Edit, Delete, and Copy OCID, as well as options that are specific to that resource type.

To the left of the table is a menu that shows all resource types available for this service. Click a different type to show the page for that resource type.

Above the table, you can select Auto Reload, Refresh, and Filter by Tags. The Compute service instance resource type page also has a Filter by Status option.

The number of resources in the table is displayed above and below the table, along with page navigation buttons if the list of resources is longer than one page.

At the top of the page is the compartment menu, which enables you to view resources of this type in a different compartment. Click the name of the compartment to view a hierarchical list of all compartments in the tenancy.

The top of the page also has a button to create a new resource of this type.

Each resource listed on a resource type page has its own page with more detail about the resource. See About Resource Detail Pages. To view the details page for the resource, click the name of the resource, or select View Details from the Actions menu.

The remainder of this section describes information in resource lists that is specific to each resource type.

Compute

The following compute resource types have additional information in their resource list.

Resource Type	Resource-specific Elements
Instances	Status or Filter by Status - Allows you to see the status of an instance or filter by the instance state: • Creating Image • Provisioning • Running • Starting • Stopped • Stopping • Terminated • Terminating Shape - The shape of the instance, which determines the number of CPUs and the amount of memory allocated to the instance. Fault Domain - The name of the fault domain (a grouping of hardware and infrastructure) the instance is running in. Fault domains let you distribute your instances so that they are not on the same physical hardware. For more information, see Working with Instances.

Resource Type	Resource-specific Elements
Instance Pools	Lifecycle State - The current state of the instance pool: • Provisioning • Scaling • Starting • Running • Stopping • Stopped • Terminating • Terminated Target Instance Count - Number of instances in a pool. Instance Configuration - The name of the instance configuration associated with the instance pool. For more information, see Working with Instance Pools.
Custom Images	Status - The current state of a custom image: • Provisioning • Importing • Available • Exporting • Stopping • Disabled • Deleted For more information, see Managing Custom Images.

Block Storage

The following block storage resource types have additional information in their resource list.

Resource Type	Resource-specific Elements
Block Volumes	Status - The current state of a volume: • Provisioning • Restoring • Available • Terminating • Terminated • Faulty Size - The size of the volume in GBs. Backup Policy - The name of the backup policy. For more information, see Managing Block Volumes.

Resource Type	Resource-specific Elements
Block Volume Backups	Status - The current state of a volume backup: • Creating • Available • Terminating • Terminated • Faulty • Request Received Total Size - The size used by the backup, in GBs, which is typically smaller than size of the block volume depending on the space consumed on the boot volume and whether the backup is full or incremental. For more information, see Backing Up Block Volumes.
Boot Volumes	State - The current state of a boot volume: • Provisioning • Restoring • Available • Terminating • Terminated • Faulty Attached to Instance - Displays Yes if the boot volume is attached to an instance and No if it is not. Size in GB - The size of the boot volume in GBs. For more information, see Managing Boot Volumes.
Boot Volume Backups	Status - The current state of a boot volume backup: • Creating • Available • Terminating • Terminated • Faulty • Request Received Total Size - The size used by the backup, in GBs, which is typically smaller than size of the boot volume depending on the space consumed on the boot volume. For more information, see Backing Up Block Volumes.

Resource Type	Resource-specific Elements
Volume Groups	Status - The current state of a volume group: • Provisioning • Available • Terminating • Terminated • Faulty Total Size - The aggregate size of the volume group in GBs. Source Volume Group - Specifies the source for a volume group which can be a volume group backup ID, a volume group ID, or a volume ID. For more information, see Managing Volume Groups.
Volume Group Backups	Status - The current state of a volume group backup: • Creating • Committed • Available • Terminating • Terminated • Faulty • Request Received Backup Size (in GB) - The aggregate size of the volume group backup, in GBs, which is typically smaller than the size of a volume group depending on the space consumed on the volume group. For more information, see Backing Up Block Volumes.

File Storage

The following file storage resource types have additional information in their resource list.

Resource Type	Resource-specific Elements
File Systems	State - The current state of the file system: • Creating • Active • Deleting • Deleted Utilization - The number of bytes consumed by the file system, including any snapshots. This number reflects the metered size of the file system and is updated asynchronously with respect to updates to the file system. For more information, see Managing File Systems.

Resource Type	Resource-specific Elements
Mount Targets	State - The current state of the mount target: • Creating • Active • Deleting • Deleted • Failed For more information, see Managing Mount Targets and Exports.

Object Storage

The following object storage resource types have additional information in their resource list.

Resource Type	Resource-specific Elements
Object Storage	Default Storage Tier - The storage tier type for every bucket is Standard, which means objects uploaded or copied to the bucket will be in the standard storage tier. Visibility - Whether this bucket is read only. By default, a bucket is not read-only. A bucket is set to read-only when it is configured as a destination in a replication policy. For more information, see Managing Object Storage Buckets.

Networking

The following networking resource types have additional information in their resource list.

Resource Type	Resource-specific Elements
Virtual Cloud Networks	Status - The current state of the virtual cloud network: • Provisioning • Available • Terminating • Terminated • Updating CIDR Block - The list of IPv4 CIDR blocks the VCN uses. DNS Domain Name - Name of the associated DNS domain. For more information, see Managing VCNs and Subnets.

Resource Type	Resource-specific Elements
Dynamic Routing Gateways	Status - The current state of the dynamic routing gateway: • Provisioning • Available • Terminating • Terminated For more information, see Connecting to the On-Premises Network through a Dynamic Routing Gateway.

DNS

The following DNS resource types have additional information in their resource list.

Resource Type	Resource-specific Elements
Zones	Status - The current state of the zone resource: • Creating • Active • Deleting • Deleted • Failed • Updating Zone Type - The type of the zone which must be either Primary or Secondary. For more information, see Managing Public DNS Zones.
Steering Policies	Status - The current state of the steering policy: • Creating • Active • Deleting • Deleted Policy Type - The type of steering policy which is either Load Balancer or IP Prefix Steering. Load Balancer policies allow distribution of traffic across multiple endpoints. Endpoints can be assigned equal weights to distribute traffic evenly across the endpoints or custom weights may be assigned for ratio load balancing. IP Prefix steering policies enable customers to steer DNS traffic based on the IP Prefix of the originating query. For more information, see Managing Traffic with Steering Policies.

Resource Type	Resource-specific Elements
TSIG Keys	Status - The current state of the tag signature key: • Creating • Active • Deleting • Deleted • Failed • Updating Algorithm - The type of algorithm used for the tag signature key: • hmac-md5 • hmac-sha1 • hmac-sha224 • hmac-sha256 • hmac-h384 • hmac-sha512 For more information, see Working with Transaction Signature Keys.

Identity

The following identity resource types have additional information in their resource list.

Resource Type	Resource-specific Elements
Users	Status - The current state of the user: • Creating • Active • Inactive • Deleting • Deleted Email - The email address assigned to the user which does not have to be unique across all users in the tenancy; multiple user accounts can have the same email address. For more information, see Creating and Managing User Accounts.

Resource Type	Resource-specific Elements
Federation	Status - The current state of an identity provider: • Creating • Active • Inactive • Deleting • Deleted Type - The type identity provider service or product which is either Security Assertion Markup Language (SAML) 2.0 protocol or Microsoft Active Directory Federation Services (ADFS). Redirect URL - The identity provider-provided URL that enables a service provider to get required information to federate with that identity provider. For more information, see Federating with Microsoft Active Directory.
Groups	Status - The current state of a group: • Creating • Active • Inactive • Deleting • Deleted For more information, see Creating and Managing User Groups.
Policies	Status - The current state of a policy: • Creating • Active • Inactive • Deleting • Deleted Statements - The number of statements attached to a policy. For more information, see Managing Policies.
Compartments	Status - The current state of a compartment: • Creating • Active • Inactive • Deleting • Deleted For more information, see Creating and Managing Compartments.

Governance

The following governance resource types have additional information in their resource list.

Resource Type	Resource-specific Elements
Tag Namespaces	Status - The current state of a tag namespace: • Active • Inactive • Deleting • Deleted For more information, see Creating and Managing Tag Namespaces.

About Resource Detail Pages

A resource details page has a section of general information about a particular resource, such as its OCID, when it was created, and the compartment it is in. Tags associated with the resource are shown on a separate Tags tab in the general information section. Some resources have other tabs as well. For example, instance details pages have separate tabs for Configuration and Networking information.

Above the general information section is the name of the resource and often one or more buttons that enable you to perform operations similar to the operations on the Actions menu on the resource type page.

To the left of the general information section is a box that shows the status of the resource.

Below the resource status box is a Resources box that lists resources that are associated with the resource that is named at the top of this resource details page. For example, on a VCN details page, the Resources box includes subnets, route tables, and security lists.

Click on a resource type in the Resources box to list all the resources of that type that are associated with the resource named at the top of this page. These resources are listed in a table that is very similar to the tables on resource type pages described in About Resource Type Pages. The resource tables in a Resources section have similar rows and columns, including an Actions menu for each resource, and have a compartment selector and usually a create button above the table.

One difference between resource tables on a resource type page and in the Resources section of a resource details page is that the column headings of resource tables in a Resources section have arrow buttons that enable you to sort the rows of the table by the content in that column. You are not limited to sorting the table only by the name of the resource. For example, you can click the arrows to sort an IP address table by the private or public IP address or by the create date of the resource.

Click the name of a resource in a Resources section table to display the details page for that resource.

The following table shows you some of the tasks you can only do from a resource's detail page.

Task	Resource Detail Page
Create a volume group clone	Volume group
Create a schedule for a backup policy	Backup policy
Create file system export	Mount target
Create file system snapshot or create file system export	File system

Task	Resource Detail Page
Upload an API key	User
View or configure policy statements	Policy
View or create a tag key definition	Tag namespace
Attach a DRG to a VCN	Dynamic routing gateway
View or add the following for VCNs: • Subnets • Route Tables View or add route rules from a route table's detail page • Internet Gateways • Local Peering Gateway • DHCP Options • Security Lists View or create ingress or egress rules from a security list's detail page • NAT Gateways • Network Security Groups • Service Gateways • Dynamic Routing Gateway	Virtual cloud network
View or create DNS zone records	Zone
View or add attached domains page	Steering policy

Locating Tenancy and Profile Information

For many tasks in Private Cloud Appliance OCI CLI, you need the tenancy OCID, which you can find on a tenancy's detail page in the Compute Web UI. You can find this page by clicking your user name in the top menu bar and selecting Tenancy or using the OCI CLI after you have installed and configured it.

A tenancy detail page provides you with some general information (which includes the OCID), object storage settings, and any associated tags. Within the Compute Web UI you cannot make any changes to a tenancy; rather, this is done by an administrator of the Service Web UI. For more information about tenancies, see Understanding the Tenancy.

Every user in the system has an associated profile. The information in a user's profile can be found in the user detail pages. Users with administrative privileges (or through group membership or policies) have access to all user profiles.

You can find your profile page by logging into a tenancy for which you have access, clicking your user name in the top menu bar and selecting Profile. From your profile page, you can view general information and your OCID, view any tags associated with your profile, and view, add or delete API keys. You can also see which groups you belong to; however, you cannot change any of your group assignments unless you have administrative privileges.

For more information about user profiles, see Creating and Managing User Accounts.

Using the OCI CLI

This section provides instructions for installing and configuring the OCI CLI as well as some general information to help you use it. The rest of the *Oracle Private Cloud Appliance User*

Guide shows you how to use the OCI CLI to complete tasks within the context of the step-by-step procedures.

The OCI CLI is the command line interface to the Compute Enclave. You can use the OCI CLI on its own or with the Compute Web UI to complete tasks. The OCI CLI provides the same core functionality as the Compute Web UI, plus additional commands, such as the ability to run scripts that extend the functionality. The OCI CLI's functionality is based on REST APIs which you can access from a browser with this URL:

`https://console.pcasysname.example.com/api-reference`

where ***pcasysname*** is the name of your Private Cloud Appliance and ***example.com*** is your domain. You can find the system and domain names on the dashboard of the Service Web UI or you can ask an administrator for this information.

Before You Begin

To install and use the OCI CLI, you must have:

- A user account for the Compute Web UI.
- An API signing key pair, required to sign API requests. If you do not already have an RSA public/private key pair in PEM format, you can create one in the OCI CLI configuration steps. To add the public key to your user account, see Adding an API Public Key to a User Profile.
- A Private Cloud Appliance self-signed certificate.

  This requirement is satisfied during the configuration steps.

You can install the OCI CLI on macOS, Microsoft Windows, or any supported Linux/UNIX operating system:

- Oracle Linux 7 and Oracle Linux 8
- CentOS 7.0 and CentOS 8.x
- Ubuntu 16.04, Ubuntu 18.04, and Ubuntu 20.04

Installing the OCI CLI

You can install the OCI CLI on Oracle Linux or macOS operating systems using a package manager. To install on Microsoft Windows or some other operating system, use the install script.

> **Important:**
>
> If you already have the CLI installed and configured, you can skip to Configuring the OCI CLI to learn how to further configure the CLI for Private Cloud Appliance.

To install the CLI, its dependencies, and Python, follow the steps for your operating system. During installation, respond to the prompts for information as described in "Responding to the Install Script Prompts."

Oracle Linux 8

Run the following commands to install the CLI:

```
$ sudo dnf -y install oraclelinux-developer-release-el8
$ sudo dnf install python36-oci-cli
```

To uninstall the CLI, run:

```
$ sudo dnf remove python36-oci-cli
```

Oracle Linux 7

Run the following command to install the CLI:

```
$ sudo yum install python36-oci-cli
```

To uninstall the CLI, run:

```
$ sudo yum remove python36-oci-cli
```

macOS

You can use Homebrew to install, upgrade, and uninstall the CLI on macOS.

> **Note:**
>
> Optionally, you can install the CLI using the install script. See "Using the Install Script for Other Operating Systems" in this section for details.

- To install the CLI, run:

  ```
  $ brew update && brew install oci-cli
  ```

- To upgrade the CLI, run:

  ```
  $ brew update && brew upgrade oci-cli
  ```

- To uninstall the CLI, run:

  ```
  $ brew uninstall oci-cli
  ```

Microsoft Windows

You can use Microsoft Windows PowerShell to install the CLI.

1. Open the PowerShell console using the Run as Administrator option.
2. Set the `http_proxy` and `https_proxy` environment variables.

   > **Important:**
   >
   > The value of `https_proxy` is the hostname or IP address of your HTTP proxy server.

   ```
   $Env:http_proxy="http://www-proxy.example.com:80"
   $Env:https_proxy="http://www-proxy.example.com:80"
   ```

   If your proxy server requires a user name and password, or uses a port number other than 80, include that information, as shown in the following example:

   ```
   $Env:https_proxy=http://username:password@proxy.example.com:port
   ```

Check that your proxy variables are set correctly. Make sure you can connect to internet locations.

```
$Env:http_proxy
$Env:https_proxy
ping https://raw.githubusercontent.com
```

3. The installer enables auto-complete by installing and running a script. To allow this script to run, you must enable the RemoteSigned execution policy.

   To configure the remote execution policy for PowerShell, run the following command:

   ```
   $ Set-ExecutionPolicy RemoteSigned
   ```

4. Force PowerShell to use TLS 1.2 for Microsoft Windows 2012 and Microsoft Windows 2016:

   ```
   $ [Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
   ```

5. Download the install script:

   ```
   $ Invoke-WebRequest ^
   https://raw.githubusercontent.com/oracle/oci-cli/master/scripts/install/install.ps1 ^
   -OutFile install.ps1
   ```

6. Run the install script with or without prompts.

   Run the `install.ps1` script that you downloaded in the previous step.

   To avoid prompts and accept the default values, run the script with the following option:

   ```
   $ install.ps1 -AcceptAllDefaults
   ```

7. Unset the proxy environment variables.

   ```
   $Env:http_proxy=""
   $Env:https_proxy=""
   ```

Using the Install Script for Other Operating Systems

For any other operating system, run the following install script to install the CLI, its dependencies, and Python.

```
$ bash -c "$(curl -L https://raw.githubusercontent.com/oracle/oci-cli/master/scripts/
install/install.sh)"
```

To avoid prompts and accept the default values, add the `--accept-all-defaults` option.

Responding to the Install Script Prompts

- If you do not have a compatible version of Python installed in Linux or Microsoft Windows, you are prompted to provide a location for installing the binaries and executables. The script installs Python for you.
- If you do not have a compatible version of Python installed in macOS, you are notified that your version of Python is incompatible. You must upgrade before you can proceed with the installation. The script does not install Python for you.
- When prompted to upgrade the CLI to the newest version, respond with **Y** to overwrite an existing installation.
- When prompted to update your PATH, respond with **Y** to be able to invoke the `oci` command without providing the full path to the executable.

Configuring the OCI CLI

Before using the OCI CLI, you must configure it for working with Private Cloud Appliance and obtain the system's certificate authority (CA) chain. You can complete the configuration manually or use the config setup tool to help you.

> **Important:**
>
> If you are already using the OCI CLI and have it configured for other purposes, read this section entirely before proceeding with any of the configuration steps.

Obtain the Required Information

Whether you manually configure the OCI CLI or use the setup config tool, there is required information you must provide for the configuration file. Before you begin the configuration process, ensure you have the following:

- User OCID

  The user's OCID is in `ocid1.user.`***`unique_ID`*** format. You can copy the user's OCID from the user details page in the Compute Web UI. To navigate to your user details page, click your user name in the Compute Web UI dashboard and then click My Profile.

- Tenancy OCID

  The tenancy OCID is in `ocid1.tenancy..`***`unique_ID`*** format. You can copy the tenancy OCID from the tenancy details page in the Compute Web UI. To navigate to the tenancy details page, click your user name in the Compute Web UI dashboard and then click Tenancy.

- Region name

  The region name is in ***`pcasys1.example.com`*** format, where ***`pcasys1`*** is the name of your Private Cloud Appliance and ***`example.com`*** is your domain.

  If you have access to the Service Web UI, you can find the system and domain names on the dashboard. Otherwise, ask a Service Web UI administrator for the information.

If you do not already have an existing API public and private key pair, we recommend that you create them as part of the manual or automated OCI CLI configuration. For more information, see the Manual Configuration or Automated Configuration section.

If you have an existing API public and private key pair that you want to use, make sure:

- They are in PEM format.
- Your public key is added to your user profile.
- You know the full path and file name of the private key. For example, `~/.oci/oci_api_key.pem` .
- You have your public key fingerprint, which is in ***`xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx`*** format. You can find this fingerprint on your profile page in the Compute Web UI or through a terminal using a command. For example:

  ```
  openssl rsa -pubout -outform DER -in ~/.oci/oci_api_key.pem | openssl md5 -c
  ```

Manual Configuration

Complete the following steps to manually configure the OCI CLI for Private Cloud Appliance. Ensure you have gathered all the required information.

The steps below assume you are on a Linux system and that you have already created a user through the Compute Web UI. However, the basic procedure is the same for other system types.

1. From a terminal, log into the system where you installed the CLI and create an API key pair. For example:

   ```
   $ oci setup keys
   Enter a passphrase for your private key (empty for no passphrase):
   Public key written to: /home/username/.oci/oci_api_key_public.pem
   Private key written to: /home/username/.oci/oci_api_key.pem
   Public key fingerprint: xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx
   ```

2. From a browser, log into the Web UI.
3. Navigate to your user details page. Click your user name in the top right of the page, and then click My Profile. Your user details page is displayed.
4. In the Resources section of your user details page, click API Keys, and then click the Add API Key button.
5. Navigate to the location of your public key or paste the public key contents and then click Upload Key.
6. In your `/home/username/.oci` directory, create a file named `config`. Add a profile section with the required information:

   ```
   [PCA1]
   user=ocid1.user...unique_id
   key_file=/home/username/.oci/oci_api_key.pem
   tenancy=ocid1.tenancy.unique_ID
   region=pcasys1.example.com
   fingerprint=xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx
   ```

   In this example, ***pcasys1*** is the name of your Private Cloud Appliance and ***example.com*** is your domain.

   If you have access to the Service Web UI, you can find the system and domain names on the dashboard. Otherwise, ask a Service Web UI administrator for the information.

Automated Configuration

If this is the first time you are using the OCI CLI, the setup config tool helps you walk you through setup process. When you enter the `oci setup config` command it prompts you for the information required for the config file and the API public/private keys and then generates an API key pair and creates the config file.

To configure the OCI CLI using the setup config tool:

1. From a command window, enter `oci setup config` and follow the prompts, for example:

   ```
   $ oci setup config
   This command provides a walkthrough of creating a valid CLI config file.
   Enter a location for your config [/home/myuserdir/.oci/config]:
   Enter a user OCID: ocid1.user.unique_ID
   Enter a tenancy OCID: ocid1.tenancy.unique_ID
   ```

> **Important:**
>
> For the step *Enter a region by index or name*, you cannot enter the region in the required `system.domain` format. Rather, enter any value from the list as the value is meaningless to Private Cloud Appliance. In Step 2 you will modify the config file to provide the information needed by Private Cloud Appliance.

```
Enter a region by index or name(e.g.
1: ap-chiyoda-1, 2: ap-chuncheon-1, 3: ap-hyderabad-1, 4: ap-melbourne-1, 5: ap-
mumbai-1,
6: ap-osaka-1, 7: ap-seoul-1, 8: ap-sydney-1, 9: ap-tokyo-1, 10: ca-montreal-1,
11: ca-toronto-1, 12: eu-amsterdam-1, 13: eu-frankfurt-1, 14: eu-zurich-1, 15: me-
dubai-1,
16: me-jeddah-1, 17: sa-santiago-1, 18: sa-saopaulo-1, 19: sa-vinhedo-1, 20: uk-
cardiff-1,
21: uk-gov-cardiff-1, 22: uk-gov-london-1, 23: uk-london-1, 24: us-ashburn-1,
25: us-gov-ashburn-1, 26: us-gov-chicago-1, 27: us-gov-phoenix-1, 28: us-langley-1,
29: us-luke-1, 30: us-phoenix-1, 31: us-sanjose-1): 24

Do you want to generate a new API Signing RSA key pair?
(If you decline you will be asked to supply the path to an existing key.) [Y/n]: Y
Enter a directory for your keys to be created [/home/myuserdir/.oci]:
Enter a name for your key [oci_api_key]:
Public key written to: /home/myuserdir/.oci/oci_api_key_public.pem
Enter a passphrase for your private key (empty for no passphrase):
Private key written to: /home/myuserdir/.oci/oci_api_key.pem
Fingerprint: xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx
Config written to /home/myuserdir/.oci/config
```

2. Navigate to the `~/`***`myuserdir`***`/.oci` directory and modify the `config` file to use the correct region, for example:

   ```
   [DEFAULT]
   user=ocid1.user.unique_ID
   fingerprint=xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx
   key_file=/home/myuserdir/.oci/oci_api_key.pem
   tenancy=ocid1.tenancy.unique_ID
   region=pcasys1.example.com
   ```

   where ***`pcasys1`*** is the name of your Private Cloud Appliance and ***`example.com`*** is your domain.

   If you have access to the Service Web UI, you can find the system and domain names on the dashboard. Otherwise, ask a Service Web UI administrator for the information.

3. If you haven't already, upload your API Signing public key through the Compute Web UI. For more information, see Adding an API Public Key to a User Profile.

Obtaining the Certificate Authority Bundle

Whether you configured the CLI manually or used the automated tool, you must obtain the Private Cloud Appliance external silo CA chain before you can run commands.

The external silo CA chain must be copied to the system where you are installing the CLI and referenced in the `oci_cli_rc` file.

1. Navigate to your `~/.oci` directory.
2. Copy the external silo CA chain from the following location:

```
https://iaas.system-name.domain-name/cachain
```

Save the CA chain in a file. In this example, the file is named `ca.crt` and is saved in the `~/.oci` directory.

3. In the `~/.oci` directory, create a file named `oci_cli_rc`. Add the profile name and the path to your copy of the external silo CA chain. For example:

```
[PCA1]
cert-bundle=/home/username/.oci/ca.crt
```

4. Set the `OCI_CLI_CERT_BUNDLE` environment variable to the same path as in the previous step.

Testing the OCI CLI Configuration

> **Important:**
>
> If you followed the manual configuration process and attempt to test the configuration by running the commands in this section, you might encounter a warning message stating the permissions on the config file are too open. If this happens, follow the instructions in the warning message to resolve the issue.

After you have installed and configured the OCI CLI, enter a list command to verify that the OCI CLI is working correctly. For example:

```
$ oci iam user list
```

Using Multiple Profiles

The OCI CLI configuration files `config` and `oci_cli_rc` can define more than one profile. Each profile section in the `config` file references a tenancy within a Private Cloud Appliance. The tenancies can be on different appliances.

In the following example `~/.oci/config` file, the PCA1 profile is for a tenancy on the ***pcasys1*** appliance, the PCA2 profile is for a tenancy on the ***pcasys2*** appliance, and the DEFAULT profile is a copy of the PCA1 profile. The DEFAULT profile is used when you have not specified which profile to use.

In this example, the key file and fingerprint are the same in each profile, but the user OCID will be different on the two different appliances or in two different tenancies on the same appliance.

```
[DEFAULT]
user=ocid1.user.unique_ID_1
key_file=/home/username/.oci/oci_api_key.pem
tenancy=ocid1.tenancy.unique_ID_1
region=pcasys1.example.com
fingerprint=58:f8:69:13:e1:a8:51:4d:5a:a0:11:69:ca:09:48:73
[PCA1]
user=ocid1.user.unique_ID_1
key_file=/home/username/.oci/oci_api_key.pem
tenancy=ocid1.tenancy.unique_ID_1
region=pcasys1.example.com
fingerprint=58:f8:69:13:e1:a8:51:4d:5a:a0:11:69:ca:09:48:73
[PCA2]
user=ocid1.user.unique_ID_2
```

```
key_file=/home/username/.oci/oci_api_key.pem
tenancy=ocid1.tenancy.unique_ID_2
region=pcasys2.example.com
fingerprint=58:f8:69:13:e1:a8:51:4d:5a:a0:11:69:ca:09:48:73
```

> **Note:**
>
> If you do not specify a profile to use, the DEFAULT profile is used. If you do not specify a profile and do not have a DEFAULT profile, you must use the `--profile` option in your commands.

To specify a profile, set the profile name as the value of the `OCI_CLI_PROFILE` environment variable:

```
export OCI_CLI_PROFILE=PCA1
```

The `--profile` option is a global option, specified on `oci`, as in the following example:

```
$ oci --profile PCA2 iam user list
```

You must specify the same profiles in your `oci_cli_rc` file that you specified in your `config` file:

```
[DEFAULT]
cert-bundle=/home/username/.oci/pca1/ca.crt
[PCA1]
cert-bundle=/home/username/.oci/pca1/ca.crt
[PCA2]
cert-bundle=/home/username/.oci/pca2/ca.crt
```

If you have configured multiple profiles, consider creating subdirectories within the `.oci` directory to store the different API keys and external silo CA chains for each profile.

Consider creating a file of environment variables for each profile. In addition to setting `OCI_CLI_PROFILE`, set `OCI_CLI_CERT_BUNDLE` to the same path that you specified in your `oci_cli_rc` file. Set `OCI_CLI_TENANCY` to the OCID of the tenancy for this profile. Giving other compartments and resources names makes commands easier to enter and read. For example:

```
$ oci network subnet create -c $Networking --vcn-id $VCN1 ...
```

Working with API Signing Keys

If you need to use the OCI CLI or make REST API requests, you must have an RSA API signing public and private key pair in PEM format. API requests are signed with the private key, and the public key is used to verify the authenticity of the request. The private key is stored locally and the public key is uploaded to a user account. You can have a maximum of three (3) public keys per user account.

> **Important:**
>
> The API signing key pair is **not** the SSH key that you use to access compute instances.

Generating an API Key Pair

If you do not already have an existing API signing public and private key pair, we recommend that you create the key pair as part of the manual or automated configuration. To do so, use the `oci setup keys` command as shown in the Manual Configuration section or follow the prompts in the Automated Configuration section.

If you want to create a key pair independent of the OCI CLI configuration, the following sections show you how to do this on Linux, macOS, and Microsoft Windows operating systems. You can then use these keys when you configure the OCI CLI.

Using Linux or macOS

1. Generate the private key.
   - Generate the key encrypted with a passphrase:

     ```
     $ openssl genrsa -out ~/.oci/oci_api_key.pem -aes128 2048
     ```

     > **Note:**
     >
     > Use of a passphrase is strongly recommended.

   - Generate the key with no passphrase:

     ```
     $ openssl genrsa -out ~/.oci/oci_api_key.pem 2048
     ```

2. Check the permission on the private key file and change if necessary.

   The file permission should be 600 or 400 to ensure that only you can read the private key file.

3. Generate the public key from your new private key:

   ```
   $ openssl rsa -pubout -in ~/.oci/oci_api_key.pem -out ~/.oci/oci_api_key_public.pem
   ```

   This public key file can have the same permissions as the private key file or can be readable by everyone.

Using Microsoft Windows

1. Install Git Bash for Microsoft Windows.

   See https://git-scm.com/download/win.

2. Include the OpenSSL binary in your Microsoft Windows path.

   On default installations, the `openssl.exe` binary is in the following directory:

   ```
   C:\Program Files\Git\mingw64\bin
   ```

3. Generate the private key.
   - Generate the key encrypted with a passphrase:

     ```
     $ openssl genrsa -out %HOMEDRIVE%%HOMEPATH%\.oci\oci_api_key.pem -aes128 -
     passout  ^
     stdin 2048
     ```

> **Note:**
>
> Use of a passphrase is strongly recommended.

- Generate the key with no passphrase:

```
$ openssl genrsa -out %HOMEDRIVE%%HOMEPATH%\.oci\oci_api_key.pem 2048
```

4. Check the permission on the private key file and change if necessary.

   The file permission should be set so that only you can read the private key file.

5. Generate the public key from your new private key:

```
$ openssl rsa -pubout -in %HOMEDRIVE%%HOMEPATH%\.oci\oci_api_key.pem -out  ^
%HOMEDRIVE%%HOMEPATH%\.oci\oci_api_key_public.pem
```

   This public key file can have the same permissions as the private key file or can be readable by everyone.

Adding an API Public Key to a User Profile

An API signing key is a PEM-format RSA public/private key pair, at least 2048 bits.

Use the Compute Web UI to add your own API public key to your profile. You cannot use the OCI CLI until you have your API signing key pair in place. If you do not have a login and password for the Compute Web UI, contact an administrator.

A user can have a maximum of three (3) public keys added to their user account. If the user has more than one API public key, the user must specify the key's fingerprint to indicate which key they are using to sign the request.

Using the Compute Web UI

1. From a browser, log in to the Compute Web UI.
2. Navigate to the user details page.
   - If you are adding a public key to your own user account, click your user icon in the upper right of the Compute Web UI, and then click My Profile.
   - If you are adding a public key to a different user account, click Identity on the navigation menu, click Users, and then click the name of the user in the user list.
3. On the user details page, scroll to the Resources section, click API Keys, and then click Add API Key.
4. In the Add Public Key dialog, navigate to the location of the public key or paste the public key contents and then click Upload Key.

Using the OCI CLI

After you have installed and configured the OCI CLI, you can use the `api-key upload` command to upload additional keys for your user account or upload keys for another user.

1. Get the OCID of the user that needs an API signing key (`oci iam user list`).
2. Use the user API key list command to ensure that the account does not already have the maximum three API signing keys.

   Syntax:

```
$ oci iam user api-key list --user-id user_OCID
```

3. Run the API key upload command.

   Syntax:

   ```
   $ oci iam user api-key upload --user-id ocid1.user.unique_ID \
   { --key public_key | --key-file file://public_keyfile.pem }
   ```

   - ***public_key*** – an RSA public key in PEM format
   - ***public_keyfile***`.pem` – a file that contains an RSA public key in PEM format

Finding an API Public Key Fingerprint

Your public key fingerprint, which is in ***xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx*** format, can be found on your profile page in the Compute Web UI or through a terminal using the following OpenSSL commands:

Linux and macOS:

```
$ openssl rsa -pubout -outform DER -in ~/.oci/oci_api_key.pem | openssl md5 -c
```

Microsoft Windows:

```
$ openssl rsa -pubout -outform DER -in \.oci\oci_api_key.pem | openssl md5 -c
```

Deleting an API Signing Key from a User Profile

You can delete your own API signing keys, and tenancy administrators can delete API signing keys for any user in their tenancy.

Using the Compute Web UI

1. In the navigation menu, click Identity, and then click Users.
2. Click the name of the user account for which you want to delete an API signing key.
3. Scroll to the Resources section of the user details page.
4. For the API key that you want to delete, click the Actions icon (three dots) and then click Delete.

Using the OCI CLI

1. Get the following information:
   - The OCID of the user account from which you want to delete an API signing key.

     ```
     $ oci iam user list
     ```

   - The fingerprint of the API signing key that you want to delete.

     ```
     $ oci iam user api-key list --user-id user_OCID
     ```

2. Run the user API key delete command.

   Syntax:

   ```
   $ oci iam user api-key delete --user-id user_OCID --fingerprint fingerprint
   ```

Understanding Command Syntax and Finding Help

This section provides some basic information to help you as you begin using the OCI CLI, such as command syntax, how to find OCIDs, and where to get help with commands.

Command Syntax

In general, commands entered in the OCI CLI have the following syntax:

```
$ oci service type action required-parameters optional-parameters
```

For example, in the following command:

```
$ oci iam user create --name joeb --description "Product test" \
--email joeb@example.com
```

- `iam` is the service
- `user` is the resource type
- `create` is the action
- `name` and `description` are required parameters
- `email` is an optional parameter

Obtaining OCIDs

When you use the OCI CLI, the majority of commands require an OCID:

- `list` commands require the OCID of the compartment where you are looking for the resource.
- `create` commands require the OCID of the compartment where you want to create the resource.
- `get`, `update`, and `delete` commands require the OCID of the resource.
- `move` commands require the OCID of the resource and the OCID of the destination compartment.

Some commands require the OCID of a different resource. For example, creating a route table for a DRG requires the OCID of the DRG.

You can find OCIDs using the OCI CLI or the Compute Web UI. The following lists show you how to find the most commonly needed OCIDs using the OCI CLI.

Block volume service OCIDs

- Boot volume

  ```
  $ oci bv boot-volume list --availability-domain AD-1 \
  --compartment-id compartment_OCID
  ```

- Volume

  ```
  $ oci bv volume list --compartment-id compartment_OCID
  ```

- Volume backup policy

  ```
  $ oci bv volume-backup-policy list --compartment-id compartment_OCID
  ```

- Volume group

```
$ oci bv volume-group list --compartment-id compartment_OCID
```

Compute service OCIDs

- Instance

  ```
  $ oci compute instance list --compartment-id compartment_OCID
  ```

- Instance VNIC

  ```
  $ oci compute instance list-vnics --compartment-id compartment_OCID
  ```

- Volume attachment

  ```
  $ oci compute volume-attachment list --compartment-id compartment_OCID
  ```

Identity and access management service OCIDs

- Compartments within a tenancy

  ```
  $ oci iam compartment list
  ```

- Compartments within a tenancy and including the tenancy

  ```
  $ oci iam compartment list --include-root
  ```

- Compartments and all sub-compartments in a tenancy

  ```
  $ oci iam compartment list --compartment-id-in-subtree true
  ```

- Compartment including its sub-compartments

  ```
  $ oci iam compartment list --compartment-id compartment_OCID \
  --compartment-id-in-subtree
  ```

- Group

  ```
  $ oci iam group list
  ```

- Policy

  ```
  $ oci iam policy list --compartment-id compartment_OCID
  ```

- Tag namespace

  ```
  $ oci iam tag-namespace list --compartment-id compartment_OCID
  ```

- User

  ```
  $ oci iam user list
  ```

Network service OCIDs

- DHCP options

  ```
  $ oci network dhcp-options list --compartment-id compartment_OCID \
  [--vcn-id VCN_OCID]
  ```

- Route table

  ```
  $ oci network route-table list --compartment-id compartment_OCID \
  [--vcn-id VCN_OCID]
  ```

- Subnet

  ```
  $ oci network subnet list --compartment-id compartment_OCID \
  [--vcn-id VCN_OCID]
  ```

- VCN

  ```
  $ oci network vcn list --compartment-id compartment_OCID
  ```

In the Compute Web UI, an OCID Copy button is available on the details page of a resource and often also on the Actions menu for the resource in the resource list.

To more easily specify OCIDs, and to make your commands more readable, you might want to set frequently used OCIDs in environment variables. For example, you could set the tenancy to T. Compartments are the most frequently used OCIDs, and the `--compartment-id` option can be shortened to `-c`.

The following examples show listing all VCNs in the NET compartment and using the large shape instance configuration to launch an instance.

```
$ oci compute vcn list -c $NET
$ oci compute-management instance-configuration launch-compute-instance \
--instance-configuration-id $INST_CFG_LRG
```

Getting Help with Commands

You can get inline help by appending `--help`, `-h` or `-?` to a command:

- `oci --help` returns a list of commands and global command options.
- `oci` ***`service`*** `--help` returns a summary of the command reference for a service. For example:

```
$ oci compute -h
Usage: oci compute [OPTIONS] COMMAND [ARGS]...
  Compute Service CLI

Options:
  -?, -h, --help  For detailed help on any of these individual commands, enter
                  <command> --help.
Commands:
  boot-volume-attachment          Represents an attachment between a...
  capacity-reservation            A template that defines the...
  console-history                 An instance's serial console data.
  dedicated-vm-host               A dedicated virtual machine host...
  dedicated-vm-host-instance      Condensed instance data when...
  device                          Device Path corresponding to the...
  global-image-capability-schema  Global Image Capability Schema
  global-image-capability-schema-version
                                  Global Image Capability Schema...
  image                           A boot disk image for launching an...
  image-capability-schema         Image Capability Schema
  image-shape-compatibility-entry An image and shape that are...
  instance                        A compute host.
  instance-console-connection     The `InstanceConsoleConnection`...
  measured-boot-report            The measured boot report for a...
  pic                             Partner image catalog (PIC).
  shape                           A compute instance shape that can...
  vnic-attachment                 Represents an attachment between a...
  volume-attachment               A base object for all types of...
```

- `oci` ***`service resource_type`*** `--help` returns a summary of the command reference for a resource type. For example:

```
$ oci compute image -h
Usage: oci compute image [OPTIONS] COMMAND [ARGS]...
  A boot disk image for launching an instance. For more information, see
  [Overview of the Compute Service].

  To use any of the API operations, you must be authorized in an IAM policy.
  If you're not authorized, talk to an administrator. If you're an
```

```
  administrator who needs to write policies to give users access, see
  [Getting Started with Policies].

  **Warning:** Oracle recommends that you avoid using any confidential
  information when you supply string values using the API.

Options:
  -?, -h, --help  For detailed help on any of these individual commands, enter
                  <command> --help.

Commands:
  change-compartment  Moves an image into a different compartment within...
  create              Creates a boot disk image for the specified instance...
  delete              Deletes an image.
  export              Exports an image to the Oracle Cloud Infrastructure...
  get                 Gets the specified image.
  import              Imports an exported image from the Oracle Cloud...
  list                Lists a subset of images available in the specified...
  update              Updates the display name of the image.
```

- `oci` ***`service resource_type action`*** `--help` returns the complete command reference for the specified service resource action. For example, the following command displays a full description of creating a compute image, and describes all options:

  ```
  $ oci compute image create -h
  ```

For more information, see the Oracle Cloud Infrastructure CLI Command Reference.

Using JSON for Complex Command Input

Complex command input includes arrays and objects with more than one value. Complex input is passed as a block of key/value pairs in JSON format. The JSON-formatted input can be provided as a string in the command line or as a file that is referenced in the command line.

The OCI CLI supports using both JSON strings and file references in the same command line. However, if the same values are provided in a file and in a string in the same command, the string value takes precedence.

Using a JSON String

To pass a JSON block as a string in the OCI CLI command line, remove the newlines. On macOS, Linux, or UNIX, enclose the entire JSON block in single quotation marks. In Microsoft Windows command lines, enclose the JSON block in double quotation marks, and escape the double quotation marks that are within the block (`\"`).

On any operating system or shell, you might need to escape other characters such as dollar signs.

If you receive the message "Parameter '***parameter_name***' must be in JSON format," then your JSON formatting is not correct. If you copied the JSON format as described in Generating JSON Format, then recheck your command-line format, especially characters that might need to be escaped.

- macOS, Linux, or UNIX

  ```
  $ oci compute instance update --instance-id ocid1.image.unique_ID \
  --freeform-tags '{"Department":"Finance"}'
  ```

- Microsoft Windows

  ```
  > oci compute instance update --instance-id ocid1.image.unique_ID \
  --freeform-tags "{\"Department\":\"Finance\"}"
  ```

Using a JSON File

An advantage of storing complex option arguments in files is that you can easily reuse arguments that you know are in correct format. You can store the data exactly the way you copy it from output of the `--generate-param-json-input` option or of the `get` command: You do not need to remove newlines and escape certain characters as you need to do for command-line strings.

When you pass input using a JSON file, the option argument is the file name prefixed with `file://`. The file name can be the name of the file in the same directory where you are running the command, the relative path to the file, or the full path to the file.

Generating JSON Format

If you receive the message "Parameter '***parameter_name***' must be in JSON format," then your JSON formatting is not correct. The following methods will help you format the data correctly.

JSON Format for a Single Complex Type Option Value

If an option is complex type, then you can generate the JSON format for that option value by using the `--generate-param-json-input` option. The argument for the `--generate-param-json-input` option is the name of the option for which you want the JSON format, without the `--` option specifier. For example, the following command shows the JSON format to use to specify route rules (`--route-rules` option) for a route table:

```
$ oci network route-table update --generate-param-json-input route-rules
[
  {
    "cidrBlock": "string",
    "description": "string",
    "destination": "string",
    "destinationType": "string",
    "networkEntityId": "string"
  },
  {
    "cidrBlock": "string",
    "description": "string",
    "destination": "string",
    "destinationType": "string",
    "networkEntityId": "string"
  }
]
```

Values that you provide in the option argument replace any existing values. For example, if you already have an ingress rule and you want to add an egress rule, you must specify both the existing ingress rule and the new egress rule. If you specify only the rule that you want to add, any previously existing rules will be gone. Similarly, if you want to add a policy statement or add values for a defined tag, for example, you must respecify the existing statements or values that you want to keep in addition to what you want to add.

Sometimes the output contains a message about choices, such as in the following example:

```
$ oci compute instance launch --generate-param-json-input source-details
[
  "This parameter should actually be a JSON object rather than an array - pick one of the
  following object variants to use",
  {
    "bootVolumeId": "string",
```

```
    "sourceType": "bootVolume"
  },
  {
    "bootVolumeSizeInGBs": 0,
    "imageId": "string",
    "kmsKeyId": "string",
    "sourceType": "image"
  }
]
```

When you use the Compute Web UI to create an instance, you select either a Custom Image or a Boot Volume. Similarly, the message in the preceding example tells you to specify either the boot volume block or the image block, not both.

JSON Format for Values of All Options of a Command

The `--generate-full-command-json-input` option shows JSON format for values of all options of the specified command, including values that are not complex values.

```
$ oci network route-table update --generate-full-command-json-input >
route_table_update.json
```

Use the `--from-json` option to pass in your customized version of this output.

```
$ oci network route-table update --from-json file://route_table_update.json
```

JSON Format of Existing Values

If resources of this type already exist, do a `get` or `list` of the resources to see the JSON formatting and also to check the current values.

```
$ oci network route-table get --rt-id ocid1.routetable.unique_ID
```

Copy the block you want directly from this output, and then change the values as needed.

In the case where multiple values are allowed, use this method to avoid overwriting values that you want to keep. Copy the block you want from the `get` or `list` output, and then change, add, or delete values as needed.

Formatting and Filtering Command Output

By default, all command output is in JSON format:

```
$ oci iam region list
{
  "data": [
    {
      "key": "pcasysname",
      "name": "pcasysname"
    }
  ]
}
```

If you prefer, command output can be formatted as a table:

```
$ oci iam region list --output table
+------------+------------+
| key        | name       |
+------------+------------+
| pcasysname | pcasysname |
+------------+------------+
```

You can filter output using the JMESPath query option for JSON. Filtering is useful when you have a large amount of output. The `--output table` option shows a column for each property of a resource. The output table for an instance, for example, has almost 30 columns and probably overflows the width of your display.

To output only the data that you want, use the `--query` option with the `--output table` option, as shown in the following example:

```
$ oci compute image list -c ocid1.compartment.unique_ID --output table \
--query "data [*].{ImageOCID:id, OperatingSystem:\"operating-system\"}"
+-------------------------+-----------------+
| ImageOCID               | OperatingSystem |
+-------------------------+-----------------+
| ocid1.image.unique_ID_1 | OracleLinux     |
| ocid1.image.unique_ID_2 | OracleLinux     |
+-------------------------+-----------------+
```

For more information about the JMESPath query language for JSON, see JMESPath.

Using the REST API

If your user account is able to use OCI CLI on the target Private Cloud Appliance, then you can also use the Compute Enclave API. Specifically, ensure that you have completed the following steps:

1. Get the appliance domain name and the name of the tenancy where you will be working.
2. Ensure the tenancy administrator has created a user account for you.
3. Get the username and temporary password for your user account, and use those credentials to log in to the Web UI.

   You will be prompted to set a permanent password.
4. Navigate to the details page of your user account and upload your public API key.

   See Generating an API Key Pair and Adding an API Public Key to a User Profile. Your API key pair must be in PEM format.
5. Ensure the tenancy administrator has added your user to one or more user groups that grant you the authorizations that you require.

Use your preferred software development kit to manage Private Cloud Appliance resources by using REST API. See Software Development Kits and Command Line Interface.

For Compute Enclave API paths and parameters, see REST API for Oracle Private Cloud Appliance Compute Enclave.

Viewing Resource Limits

A Private Cloud Appliance has a set of service limits configured for each tenancy. A service limit is the quota or allowance set on a resource. These resource limits are listed in Service Limits in the *Oracle Private Cloud Appliance Release Notes*.

Some resource limits can be changed by the appliance administrator. The commands described in this topic show you which services currently allow resource limits to be changed, and show you what the current values are for those resource limits.

Using the Compute Web UI

1. On the dashboard, select Governance / Limits.
2. Select a service from the drop-down menu above the list.

   Resource limit definitions are shown in the list if the selected service exposes its resource limits.

   If the selected service does not expose its resource limits, see Service Limits in the *Oracle Private Cloud Appliance Release Notes*.
3. If you select a different service from the drop-down menu, also select the Refresh button to refresh the list.

Using the OCI CLI

List Limits Services

List services that support changing resource limits.

Syntax:

```
oci limits service list --compartment-id compartment_OCID
```

If a service is not listed, the resource limits for that service are given in Service Limits in the *Oracle Private Cloud Appliance Release Notes*.

List the Resource Limit Definitions of a Service

List resource limit definitions for all services that support changing resource limits. For each limit definition, show the name of the service, the name of the resource limit, and the description of the limit definition.

Syntax:

```
oci limits definition list --compartment-id compartment_OCID \
--service-name service_name --name limit_name
```

The `--service-name` option is optional and shows resource limit definitions for only that service. Specify a service that is identified by the `limits service list` command.

The `--name` option is optional and shows the limit definition for only that resource.

List the Resource Limit Values of a Service

Show the complete list of resource limits for the specified service. Specify a service that is identified by the `limits service list` command.

Syntax:

```
oci limits value list --compartment-id compartment_OCID --service-name service_name \
--name limit_name
```

The `--name` option is optional and shows the limit for only that resource.

Show Availability of a Limit Resource

Show the following for the specified compartment, service, and limit:

- The number of available resources associated with the given limit.
- The usage in the selected compartment for the given limit.

Specify a service that is identified by the `limits service list` command.

Syntax:

```
oci limits resource-availability get --compartment-id compartment_OCID \
 --service-name service_name --limit-name limit_name
```

Not all resource limits support `resource-availability get`. If the value is not available, a message is shown that resource availability is not supported for that particular service and limit.

2

Identity and Access Management

Oracle Private Cloud Appliance Identity and Access Management (IAM) enables you to control which users have what access to which cloud resources in your tenancy.

For conceptual information, see the Identity and Access Management Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

Creating and Managing Compartments

Compartments contain resources such as cloud instances, virtual cloud networks, and block volumes. Your tenancy is the root compartment where you can create cloud resources and other compartments. You can create hierarchies of compartments that are up to six levels deep. You can limit access to compartment resources to specified user groups. Most resources can be moved between compartments later if your business needs change.

The compartments that you create in your tenancy are your primary building blocks for organizing and controlling access to your cloud resources. Before you create compartments and resources, see "Organizing Resources in Compartments" in the Identity and Access Management Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

Understanding the Tenancy

A tenancy is a special compartment. The tenancy is the root compartment where you create and administer all of your cloud resources, including other compartments.

Users, groups, and identity providers are always attached directly to the tenancy, not to any compartment of the tenancy. You cannot specify a different compartment when you create a user, group, or identity provider. When you use the OCI CLI to operate on a user, group, or identity provider, the OCID of the tenancy from the `config` file is used by default.

Other resources can reside in the tenancy or in any other compartment. Operating on these resources often requires you to select the correct compartment in the Compute Web UI or specify the compartment OCID in the OCI CLI.

Use the following procedures to get the OCID of the tenancy.

Using the Compute Web UI

1. Click your user profile menu in the top right of the page.
2. Click the Tenancy option.
3. On the tenancy details page, use the Show or Copy button under OCID.

Using the OCI CLI

1. Use the `compartment list` command.

   ```
   $ oci iam compartment list
   ```

   Look for the `ocid1.tenancy.`***`unique_ID`*** OCID.

- With no options, the `compartment list` command lists all compartments that are direct child compartments of the tenancy. The tenancy is the value of the first property listed (`compartment-id`) for every compartment in the list.
- If you specify the `--include-root` option, the tenancy is listed first, and the tenancy OCID is the value of the `id` property (the value of the `compartment-id` property is `null`).

As is true for other resources, in a compartment `list` or `get`, the `compartment-id` compartment is the parent compartment of the `id` compartment.

Listing Compartments

Using the Compute Web UI

1. In the navigation menu, click Identity and then click Compartments.

   The list shows all compartments that are direct child compartments of the tenancy.
2. To view a compartment that is a subcompartment of a listed compartment, click the name of the listed compartment. On the details page for that compartment, scroll to the Resources section, and click Child Compartments.

   You might need to click the name of a compartment in the Child Compartments list, and repeat this step.

To find the compartment where a particular resource is located, navigate to a list of those resources. Above the resource list, use the Compartment drop-down menu to select the compartment.

Using the OCI CLI

Use the `--help` option to learn about the `--access-level` option and about options that are common to `list` commands such as `--lifecycle-state` and `--sort-by`.

1. To list all compartments and subcompartments in the tenancy, specify the `--compartment-id-in-subtree` option with a value of `true`.

   ```
   $ oci iam compartment list --compartment-id-in-subtree true
   ```

   Specifying the `--compartment-id` option, described in the next step, does not change this output: You cannot list just the compartment tree of a particular compartment other than the tenancy.
2. To list all compartments that are direct child compartments of another compartment, specify the OCID of that parent compartment:

   ```
   $ oci iam compartment list --compartment-id ocid1.compartment.unique_ID
   ```

   This output does not list the specified parent compartment and does not list compartments that are deeper in the hierarchy of this parent compartment. This output only lists the direct child compartments of the specified parent compartment.

   If you do not specify a parent compartment, all compartments that are direct child compartments of the tenancy are listed. To list the tenancy in addition to the direct child compartments of the tenancy, specify the `--include-root` option.
3. To list just one particular compartment, you can specify the compartment name.

   ```
   $ oci iam compartment list --name Acompartment
   ```

   The output is the same as a `get` of that compartment.

```
$ oci iam compartment get --compartment-id OCID_of_Acompartment
```

Creating a Compartment

You can create a compartment in your tenancy or in another compartment. You can create hierarchies of compartments that are up to six levels deep.

Using the Compute Web UI

1. In the navigation menu, click Identity and then click Compartments.
2. Click the Create Compartment button above the list of compartments.
3. In the Create Compartment dialog box, enter the following information:
   - *Name*: A name for this compartment. Compartment names have the following characteristics:
     - Must be unique within the tenancy.
     - Are case insensitive.
     - Can be changed later.
     - Can be no more than 100 characters.
     - Can contain only alphanumeric characters, period (.), hyphen (-), and underscore (_).
   - *Description*: A description for this compartment. This description can be no more than 400 characters and can be changed later.
   - *Create in Compartment*: The compartment in which you want to create the new compartment. The new compartment will be a sub-compartment of the selected compartment.
   - *Tagging*: (Optional) Add defined or free-form tags for this compartment as described in Adding Tags at Resource Creation. Tags can also be applied later.
4. Click the Create Compartment button on the dialog box.

   Click the name of the new compartment to view the compartment details, including tags.

Using the OCI CLI

1. Get the OCID of the compartment in which you want to create the new compartment. The new compartment will be a sub-compartment of the specified compartment.

   ```
   $ oci iam compartment list --compartment-id-in-subtree true
   ```

2. Run the compartment create command.

   Syntax:

   ```
   oci iam compartment create --compartment-id compartment_OCID \
   --name text --description "text"
   ```

   See the Compute Web UI procedure for characteristics of the name and description values. See Adding Tags at Resource Creation to add defined and free-form tags.

   Example:

   ```
   $ oci iam compartment create -c ocid1.compartment.parent_compartment_unique_ID \
   --name ProductX --description "A child compartment of compartment Products"
   {
     "data": {
       "compartment-id": "ocid1.compartment.parent_compartment_unique_ID",
   ```

```
    "defined-tags": {},
    "description": "A child compartment of compartment Products",
    "freeform-tags": {},
    "id": "ocid1.compartment.new_compartment_unique_ID",
    "inactive-status": null,
    "is-accessible": null,
    "lifecycle-state": "ACTIVE",
    "name": "ProductX",
    "time-created": "2021-10-05T22:58:23.216657+00:00"
  },
  "etag": "b212700d-45fa-46a9-90da-bcc016c587bc"
}
```


To view this output later, use the `compartment get` command.

Applying Tag Defaults

Compartments can have resources called tag defaults. Tag defaults are defined tags that are inherited by all resources and child compartments that are created after the tag default is added to the parent compartment. To add a tag default to a compartment, see Configuring Tag Defaults.

Adding Policies for Access Control

Child compartments inherit access permissions from their parent compartments. If you want the access to a new compartment to be different from the access to the parent compartment, create an access policy for the new compartment. For example, grant group DevX permission to read all resources in compartment Products and permission to manage all resources in subcompartment ProductX. Grant group DevY permission to read all resources in compartment Products and permission to manage all resources in subcompartment ProductY. Because of inheritance, group DevX will be able to read all resources in compartment ProductY, and group DevY will be able to read all resources in compartment ProductX.

For information about creating and attaching policies, see Managing Policies.

Adding Resources to a Compartment

Use either of the following methods to add resources to a compartment:

- Specify the compartment when you create the resource.
- Move the resource from a different compartment.

  See the documentation for the particular resource for information such as whether attached resources move with the moved resource.

  Check whether the moved resources have the correct tags and policies applied. You might need to manually delete and add tags and policies.

The Resources box on a compartment details page in the Compute Web UI, and the compartment `list` and `get` commands in the OCI CLI, do not show all of the resources that belong to a compartment. For resources that are not listed, go to the Compute Web UI page for that resource, such as instances, and select the compartment from the Compartment drop-down menu above the resource list. In the OCI CLI, specify the compartment OCID when you list the resources. See also Using the Compute Web UI and Using the OCI CLI.

Updating a Compartment

You can change the name and description of a compartment. You can add, change, or remove tags as described in Applying Tags to an Existing Resource. You cannot change the parent compartment. To change the parent compartment, see Moving a Compartment to a Different Compartment.

Using the Compute Web UI

1. In the navigation menu, click Identity and then click Compartments.
2. If the compartment that you want to update is not listed, navigate to the compartment that you want to update, as described in Listing Compartments.
3. For the compartment that you want to update, click the Actions menu, and click the Edit option.
4. In the Editing ***compartment_name*** Compartment dialog, make the changes.
5. Click the Save Changes button.

   Click the compartment name to view the compartment details, including tags.

Using the OCI CLI

1. Get the OCID of the compartment that you want to update.

   ```
   $ oci iam compartment list --compartment-id-in-subtree true
   ```

2. Run the compartment update command.

   Syntax:

   ```
   oci iam oci iam compartment update --compartment-id compartment_OCID \
   options_with_values_to_update
   ```

   Example:

   If you specify the `--defined-tags` or `--freeform-tags` options, then you must fully specify all defined and free-form tags that you want on this compartment, including tags that already exist on the compartment that you want to keep. Values that you provide for these tag options replace any existing values. See Working with Resource Tags. You will be prompted to confirm unless you specify the `--force` option.

   ```
   $ oci iam compartment update --compartment-id ocid1.compartment.unique_ID \
   --defined-tags '{"Product":{"LMN":"Development"}}' --freeform-tags '{"MyTag":"val-
   u"}'
   WARNING: Updates to freeform-tags and defined-tags will replace any existing values.
   Are you sure you want to continue? [y/N]: y
   ```

   The output of this command is the same as the output of the `compartment get` command.

Moving a Compartment to a Different Compartment

You can move a compartment to a different parent compartment in the same tenancy. When you move a compartment, all subcompartments of the compartment are moved. Some resources of the moved compartment are moved. You can separately move other resources as needed. See the documentation for the particular resource type for more information.

After you move a compartment to a new parent compartment, the access policies of the new parent take effect and the policies of the previous parent no longer apply. Groups who had

access to the compartment and its resources in the previous parent compartment lose their access when the compartment is moved. Groups who have access in the new parent compartment gain access to the moved compartment and its resources.

Tag defaults that are automatically applied to all resources created in the new parent are not automatically applied to the newly moved compartment and its resources. You might need to separately delete and add tag defaults to the moved compartment and delete and add tags to moved resources.

See also "Moving a Compartment to a Different Parent Compartment" in the Identity and Access Management Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

You must belong to a group that has `manage all-resources` permissions on the lowest shared parent compartment of the current compartment and the destination compartment.

To move a compartment, you must use the OCI CLI.

Using the OCI CLI

1. Get the OCID of the compartment that you want to move, and the OCID of the destination compartment.

   ```
   $ oci iam compartment list --compartment-id-in-subtree true
   ```

2. Run the compartment move command.

   Syntax:

   ```
   oci iam compartment move --compartment-id compartment_to_move_OCID \
   --target-compartment-id destination_compartment_OCID
   ```

   Use the `iam work-request get` command to check the status of the compartment move, or view the work request details in the Compute Web UI. Some resources might take longer to move than the compartment.

Deleting a Compartment

To delete a compartment, you must first move, delete, or terminate all of the resources in the compartment, including any policies that are attached to the compartment. Before you begin, check the move and delete capabilities for all resources in the compartment.

Using the Compute Web UI

1. In the navigation menu, click Identity and then click Compartments.

   The compartments in the tenancy are listed.

2. If the compartment that you want to delete is not listed, navigate to the compartment that you want to update, as described in Listing Compartments.

3. For the compartment that you want to delete, click the Actions menu, and click the Delete Compartment option.

   If the Delete Compartment option is not selectable, then you might not have permission to delete this compartment.

4. In the Delete Compartment confirmation dialog, click Delete.

   The compartment status changes to Deleting.

   In the Resources box on the compartment details page, click Work Requests and view the details of the compartment delete. When the work request is completed, the compartment

is removed from the compartments list. If the work request fails, the compartment status returns to Active.

Using the OCI CLI

1. Get the OCID of the compartment that you want to delete.

   ```
   $ oci iam compartment list --compartment-id-in-subtree true
   ```

2. Run the compartment delete command.

   Syntax:

   ```
   oci iam compartment delete --compartment-id compartment_OCID
   ```

   Use the `iam work-request get` command to check the status of the compartment delete.

Creating and Managing User Accounts

By default, the tenancy has an administrative user in an administrators group, and a policy enables the administrators group to manage the tenancy. To limit a user to managing only a subset of resources in the tenancy or another compartment, or to have less than full management access to some resources, create a user account, add the user account to one or more groups, and create one or more policies for those groups.

A user account is not automatically a member of any group. A user that is not a member of any group is visible in the tenancy but does not have access to any resources.

For conceptual information about user accounts and groups, see the Identity and Access Management Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

Creating a User

When you create a user, the user is automatically created in the tenancy. You cannot specify a different compartment for the user.

Using the Compute Web UI

1. In the navigation menu, click Identity, and then click Users.
2. Click the Create User button.
3. In the Create User dialog, enter the following information:
   - *Name*: A name for this user account. User names have the following characteristics:
     - Must be unique within the tenancy. You can create a user with the same name as a user that has been deleted.
     - Are case insensitive.
     - Cannot be changed later.
     - Must be at least two and no more than 100 characters.
     - Can contain only alphanumeric characters, period (.), hyphen (-), underscore (_), plus sign (+), and at sign (@).
   - *Description*: A description for this user, such as the full name of the person or a brief description of the account. The description has the following characteristics:
     - Must be 1-400 characters.
     - Does not need to be unique.

  - Can be changed later.
- *Email Address*: (Optional) The email address for the user. Can be updated later.
- *Password*: (Optional) To enable this user to log in to the Compute Web UI, check the box labeled "Generate a temporary password for this user."

  You can provide a password later. See Providing a Temporary Compute Web UI Password.

  > **Note:**
  >
  > Passwords for federated users are not managed through this service. See information from your federated identity provider.

- *Tagging*: (Optional) Add defined or free-form tags for this user account as described in Adding Tags at Resource Creation. Tags can also be applied later.

4. Click the Create User button on the Create User dialog.

   If you checked the box labeled "Generate a temporary password for this user," a Temporary Password for New User dialog pops up, showing the temporary password. You cannot retrieve this password again after you close this dialog. Copy the temporary password, save the password to a safe place for delivery to the user, and click the "I have made a note of the password" button.

   The details page of the new user is displayed.

   Next steps:

   - Provide the user with a temporary password so that the user can set their own permanent Compute Web UI password.
     - If you checked the box labeled "Generate a temporary password for this user," provide the temporary password that you copied from the Temporary Password for New User dialog.
     - If you did not check the box labeled "Generate a temporary password for this user," or did not save that password, follow the instructions in Providing a Temporary Compute Web UI Password to generate a temporary password for the user.
   - Add this user to at least one group. See Adding a User to a Group by Updating the User.
   - If the user wants to use the OCI CLI, see Installing the OCI CLI.

Using the OCI CLI

1. Get the following information:
   - A name and description for the user. See the Compute Web UI procedure for parameters. In the OCI CLI, a description must be provided but its value can be an empty string.
   - (Optional) The OCID of the tenancy for the user. By default, the root compartment OCID from the config file is used.
2. Run the user create command.

   Syntax:

   ```
   oci iam user create --name text --description text
   ```

See the Compute Web UI procedure for characteristics of the name and description values. See Adding Tags at Resource Creation to add defined and free-form tags.

Example:

```
$ oci iam user create --name flast --description "First Last" --email
first.last@example.com
```

The output of this command is the same as the output of the `user get` command.

Next steps:

- Provide the user with a temporary password so that the user can set their own permanent Compute Web UI password. See Providing a Temporary Compute Web UI Password.
- Add this user to at least one group. See Adding a User to a Group by Updating the User.
- If the user wants to use the OCI CLI, see Installing the OCI CLI.

Providing a Temporary Compute Web UI Password

Perform this procedure for new users and for users who forget their password. This procedure generates a temporary one-time password. When the user signs in using this password, the user is required to change the password before proceeding. The generated temporary password expires after seven (7) days.

A tenancy administrator can provide a temporary password for any user. Users must set their own permanent passwords by following the instructions in Setting Your Own Compute Web UI Password.

> **Note:**
>
> Passwords for federated users are not managed through the IAM service. See information from your federated identity provider.

Using the Compute Web UI

1. In the navigation menu, click Identity, and then click Users.

   If the user that needs a new password is in the Inactive state, see Unlocking a User.
2. For the user that needs a new password, click the Actions menu, and click the Change Password option.
3. In the Change Password dialog, click the Create Temporary Password button.

   A Password Changed dialog pops up. The New Password field contains the temporary password.
4. Copy and save this temporary password.

   You cannot retrieve this password again after you close this dialog. Copy the temporary password, and save the password to a safe place for delivery to the user.
5. Click the Close button on the dialog.
6. Deliver this temporary one-time password to the user. The user must follow the rules stated in Setting Your Own Compute Web UI Password when setting their new password.

Using the OCI CLI

1. Get the OCID of the user that needs a password (`oci iam user list`).
2. Confirm that the user is active.

   If the `lifecycle-state` of the user is `INACTIVE`, see Unlocking a User.
3. Run the command to create or reset the Compute Web UI password for the user.

   Example:

```
$ oci iam user ui-password create-or-reset --user-id ocid1.user.unique_ID
{
  "data": {
    "inactive-status": null,
    "lifecycle-state": "ACTIVE",
    "password": "N59%fP9uTq6\\",
    "time-created": "2021-10-13T22:10:49.290000+00:00",
    "user-id": "ocid1.user.unique_ID"
  }
}
```

4. Copy the `password` value from the command output and deliver this temporary one-time password to the user. The user must follow the rules stated in Setting Your Own Compute Web UI Password when setting their new password.

Setting Your Own Compute Web UI Password

Users do not require an access policy to set or change their own Compute Web UI password.

Setting Your Password

Use this procedure to set your Compute Web UI password initially, or to reset your password if you forgot your password.

Using the Compute Web UI

1. Get the temporary password that was generated for you.
2. On the login screen for the Compute Web UI, enter your user name.
3. Enter the temporary password.

   A dialog pops up that says your password has expired and you need to create a new password.
4. Click the Change my password button.
5. On the Change My Password screen, enter the temporary password in the Current Password field.
6. Enter a new password in the New Password field and again in the Confirm New Password field.

   Passwords must be at least 12 characters in length and contain at least one of each of the following: uppercase character, lowercase character, number, and symbol.
7. Click the Save Changes button.

   A dialog pops up that says your password has been successfully updated.
8. Click the Continue button.

9. Log in using your new password.

Changing Your Password

Use this procedure to change your Compute Web UI password while your current password still works.

Using the Compute Web UI

1. In the top right corner of the Compute Web UI, click your user menu.
2. Click Change My Password.
3. On the Change My Password screen, enter your current password in the Current Password field.
4. Enter a new password in the New Password field and again in the Confirm New Password field.

   Passwords must be at least 12 characters in length and contain at least one of each of the following: uppercase character, lowercase character, number, and symbol.
5. Click the Save Changes button.

   A dialog pops up that says your password has been successfully updated.
6. Click the Continue button.

Viewing User Information and Group Membership

Using the Compute Web UI

1. In the navigation menu, click Identity, and then click Users.

   The Users page shows all users of the tenancy because user accounts cannot be in different compartments. All users are in the tenancy.
2. Click the name of the user for which you want more information.
3. On the details page for that user account, scroll down to the Resources section.
4. Click the Groups resource.

   The list of groups where this user is a member is shown.
5. To see the full list of members of a group, click the name of the group in the Groups list.

   Scroll down to the Resources section for that group and click Group Members.

Using the OCI CLI

1. Get the OCID of the user account for which you want the list of groups (`oci iam user list`).
2. Run the list groups command.

   Syntax:

   ```
   oci iam user list-groups --user-id user_OCID
   ```

   The output of the `list-groups` command is the same as the output of the `group get` command for each group where this user is a member.

   The `user get` command does not show group membership.

Adding a User to a Group by Updating the User

A user must be a member of at least one group in order to have access to any resources.

Using the Compute Web UI

As an alternative to using the Users Compute Web UI page, you can use the Groups page as described in Adding a User to a Group by Updating the Group.

1. In the navigation menu, click Identity, and then click Users.
2. Click the name of the user that you want to add to a group.
3. On the details page, scroll down to the Resources section and click Groups.
4. At the top of the Groups list, click the Add User to Group button.
5. In the Add User to Group dialog, select a group from the drop-down list, and then click the OK button.

   The selected group is added to the user's Groups list.

Using the OCI CLI

1. For the OCI CLI procedure, see Adding a User to a Group by Updating the Group.
2. Use the `user list-groups` command to show the groups where this user is a member. The output of the `user list-groups` command is the same as the output of the `group get` command for each group where this user is a member.

Removing a User from a Group by Updating the User

If you remove a user from all groups, the user will not have access to any resources.

Using the Compute Web UI

As an alternative to using the Users Compute Web UI page, you can use the Groups page as described in Removing a User from a Group by Updating the Group.

1. In the navigation menu, click Identity, and then click Users.
2. Click the name of the user that you want to remove from a group.
3. Scroll to the Resources section and click Groups.
4. For the group from which you want to remove the user, click the Actions menu, and click the Remove from Group option.

   The selected group is removed from the user's Groups list.

Using the OCI CLI

1. For the OCI CLI procedure, see Removing a User from a Group by Updating the Group.
2. Use the `user list-groups` command to show the groups where this user is a member. The output of the `user list-groups` command is the same as the output of the `group get` command for each group where this user is a member.

Modifying a User

You can change a user account's description and email address. You can add, change, or remove tags as described in Applying Tags to an Existing Resource.

Using the Compute Web UI

1. In the navigation menu, click Identity, and then click Users.
2. For the user account that you want to modify, click the Actions menu, and click the Edit option.
3. In the Edit ***username*** dialog, modify the account's description, email address, or tags.
4. Click Save Changes.

Using the OCI CLI

1. Get the OCID of the user account that you want to modify (`oci iam user list`).
2. Run the user update command.

   Syntax:

   ```
   oci iam user update --user-id user_OCID [ --description desc ] \
   [ --email email ] [ --defined-tags tags ] [ --freeform-tags tags ]
   ```

   The output of this command is the same as the output of the `user get` command.

Unlocking a User

This procedure unlocks a user that is in the Inactive state. A user might be in the Inactive state after too many incorrect login attempts.

Using the Compute Web UI

1. In the navigation menu, click Identity, and then click Users.
2. For the user account that you want to unlock, click the Actions menu, and click the Unblock option.

   The user transitions from the Inactive state to the Active state.

Using the OCI CLI

1. Get the OCID of the user account that you want to unlock (`oci iam user list`).

   Confirm that the `lifecycle-state` of the user is `INACTIVE`.
2. Run the update user state command.

   Syntax:

   ```
   $ oci iam user update-user-state --user-id ocid1.user.unique_ID \
   --blocked false
   ```

   Use the `user get` command to confirm that the `lifecycle-state` of the user is `ACTIVE`.

Deleting a User

You cannot delete a user if the user is a member of any group. You cannot delete your own user.

When you delete a user, all API keys associated with that user account are also deleted.

Using the Compute Web UI

1. In the navigation menu, click Identity, and then click Users.
2. Click the name of the user that you want to delete.
3. Ensure that the user is not a member of any group.

   On the user details page, scroll down to the Resources section and click Groups. To remove this user from a group, click the Actions menu for the group in the Groups list, and click the Remove from Group option.
4. At the top of the user details page, click the Delete button.
5. On the Delete User confirmation dialog, click the Confirm button.

Using the OCI CLI

1. Get the OCID of the user account that you want to delete (`oci iam user list`).
2. Use the `user list-groups` command to ensure that the user is not a member of any group.
3. Run the user delete command.

   Syntax:

   ```
   oci iam user delete --user-id user_OCID
   ```

   Example:

   ```
   $ oci iam user delete --user-id ocid1.user.unique_ID
   Are you sure you want to delete this resource? [y/N]: y
   ```

   To delete a user without confirmation, use the `--force` option.

Creating and Managing User Groups

Access to cloud resources is granted to groups, not directly to users. A user account is not automatically a member of any group. To enable a user to do any work with cloud resources, you must add the user to a group and then create an access policy for that group. A group is therefore a set of users who have the same type of access to the same set of cloud resources. Organize users into groups according to which compartments and resources they need to access and how they need to work with those resources. A user can be a member of more than one group.

For conceptual information about user accounts and groups, see the Identity and Access Management Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

Creating a Group

When you create a group, the group is automatically created in the tenancy. You cannot specify a different compartment for the group.

Using the Compute Web UI

1. In the navigation menu, click Identity, and then click Groups.
2. Click the Create Group button.
3. In the Create Group dialog, enter the following information:
   - *Name*: A name for this group. Group names have the following characteristics:
     - Must be unique within the tenancy. You can create a group with the same name as a group that has been deleted.
     - Are case insensitive.
     - Cannot be changed later.
     - Can be no more than 100 characters.
     - Can contain only alphanumeric characters, period (.), hyphen (-), and underscore (_).
   - *Description*: A description for this group. The description has the following characteristics:
     - Must be 1-400 characters.
     - Does not need to be unique.
     - Can be changed later.
   - *Tagging*: (Optional) Add defined or free-form tags for this group as described in Adding Tags at Resource Creation. Tags can also be applied later.
4. Click the Create Group button on the Create Group dialog.

   The details page of the new group is displayed.

   Next steps:

   - Create an access policy for this group or add this group to an existing policy. A group has no permissions unless it is the subject of at least one policy. See Managing Policies.
   - Add users to this group. See Adding a User to a Group by Updating the Group.

Using the OCI CLI

1. Get the following information:
   - A name and description for the group. See the Compute Web UI procedure for limitations. In the OCI CLI, a description must be provided but its value can be an empty string.
   - (Optional) The OCID of the tenancy for the group. By default, the root compartment OCID from the config file is used.
2. Run the group create command.

   Syntax:

   ```
   oci iam group create --name text --description "text"
   ```

   See the Compute Web UI procedure for characteristics of the name and description values. See Adding Tags at Resource Creation to add defined and free-form tags.

   Example:

```
$ oci iam group create --name Product-A --description "Resource management for
Product A."
```

The output of this command is the same as the output of the `group get` command.

Next steps:

- Create an access policy for this group or add this group to an existing policy. A group has no permissions unless it is the subject of at least one policy. See Managing Policies.
- Add users to this group. See Adding a User to a Group by Updating the Group.

Viewing Group Information and Group Membership

Using the Compute Web UI

1. In the navigation menu, click Identity, and then click Groups.

   The Groups page shows all groups in the tenancy because group definitions cannot be in different compartments. All groups are in the tenancy.
2. Click the name of the group about which you want more information.
3. On the details page for that group, scroll down to the Resources section.
4. Click the Group Members resource.

   The list of users that belong to this group is shown.
5. To see the full list of groups where a user is a member, click the name of the user in the Group Members list.

   Scroll down to the Resources section for that user and click Groups.

Using the OCI CLI

1. Get the OCID of the group for which you want the list of users (`oci iam group list`).
2. Run the list users command.

   Syntax:

   ```
   oci iam group list-users --group-id group_OCID
   ```

   The output of the `list-users` command is the same as the output of the `user get` command for each user that is a member of this group.

   The `group get` command does not show member users.

Adding a User to a Group by Updating the Group

Users must be members of groups in order to have access to resources.

Using the Compute Web UI

1. In the navigation menu, click Identity, and then click Groups.
2. Click the name of the group where you want to add users.
3. On the details page, scroll down to the Resources section and click Group Members.
4. At the top of the Group Members list, click the Add User to Group button.

5. In the Add User to Group dialog, select a user from the drop-down list, and then click the OK button.

   The selected user is added to the Group Members list.

Using the OCI CLI

1. Get the following information:
   - The OCID of the group where you want to add a user (`oci iam group list`).
   - The OCID of the user that you want to add to this group (`oci iam user list`).
2. Run the group add user command.

   Syntax:

   ```
   oci iam group add-user --group-id group_OCID --user-id user_OCID
   ```

   Example:

   ```
   $ oci iam group add-user --group-id ocid1.group.unique_ID --user-id
   ocid1.user.unique_ID
   {
     "data": {
       "compartment-id": "ocid1.tenancy.unique_ID",
       "group-id": "ocid1.group.unique_ID",
       "id": "ocid1.user_group_membership.unique_ID",
       "inactive-status": null,
       "lifecycle-state": "ACTIVE",
       "time-created": null,
       "user-id": "ocid1.user.unique_ID"
     }
   }
   ```

Removing a User from a Group by Updating the Group

Using the Compute Web UI

1. In the navigation menu, click Identity, and then click Groups.
2. Click the name of the group where you want to remove a user.
3. On the details page, scroll down to the Resources section and click Group Members.
4. In the Group Members list, click the Actions menu for the user that you want to remove from the group, and click the Remove from Group option.
5. At the confirmation prompt, click OK.

   The user is removed from the group.

Using the OCI CLI

1. Get the following information:
   - The OCID of the group where you want to remove a user (`oci iam group list`).
   - The OCID of the user that you want to remove from the group (`oci iam user list`).
2. Run the group remove user command.

   Syntax:

   ```
   oci iam group remove-user --group-id group_OCID --user-id user_OCID
   ```

Modifying a Group

You can change the description for a group. You can add, change, or remove tags as described in Applying Tags to an Existing Resource.

Using the Compute Web UI

1. In the navigation menu, click Identity, and then click Groups.
2. For the group that you want to modify, click the Actions menu, and click the Edit option.
3. In the Edit ***groupname*** dialog, modify the group's description or tags.
4. Click Save Changes.

Using the OCI CLI

1. Get the OCID of the group that you want to modify (`oci iam group list`).
2. Run the group update command.

   Syntax:

   ```
   oci iam group update --group-id group_OCID [ --description desc ] \
   [ --defined-tags tags ] [ --freeform-tags tags ]
   ```

   The output of this command is the same as the output of the `group get` command.

Deleting a Group

You cannot delete a group if the group has any members.

Using the Compute Web UI

1. In the navigation menu, click Identity, and then click Groups.
2. Click the name of the group that you want to delete.
3. Ensure that the group does not have any members.

   On the group details page, scroll down to the Resources section and click Group Members. To remove a user from the group, click the Actions menu for the user in the Group Members list, and click the Remove from Group option.
4. At the top of the group details page, click the Delete button.
5. On the Delete Group confirmation dialog, click the Confirm button.

Using the OCI CLI

1. Get the OCID of the group that you want to delete (`oci iam group list`).
2. Use the `group list-users` command to ensure that the group has no members.
3. Run the group delete command.

   Syntax:

   ```
   oci iam group delete --group-id group_OCID
   ```

   Example:

   ```
   $ oci iam group delete --group-id ocid1.group.unique_ID
   Are you sure you want to delete this resource? [y/N]: y
   ```

To delete a group without confirmation, use the `--force` option.

Federating with Microsoft Active Directory

Federating enables users at your company to use the same login credentials for the Private Cloud Appliance Compute Web UI that they already use for other logins in the company. To *federate*, an administrator creates a trust relationship between the existing identity provider and Oracle Private Cloud Appliance. When this relationship is established, federated users are prompted with a *single sign-on* when accessing the Compute Web UI.

For more information, see "Federating with Identity Providers" in the Identity and Access Management Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

You can federate multiple Active Directory (AD) accounts with Oracle Private Cloud Appliance. Each federation trust is for a single AD account. To create a trust, you perform some tasks in the Oracle Private Cloud Appliance Compute Web UI and some tasks in Active Directory Federation Services (ADFS).

Before you begin federating, make sure you have completed the following tasks:

- Installed and configured Microsoft ADFS for your organization.
- Created groups in AD that will map to groups in Oracle Private Cloud Appliance.
- Created users in AD who will sign into the Oracle Private Cloud Appliance Compute Web UI.

> **Note:**
>
> Consider using a common prefix to name AD groups that you intend to map to Oracle Private Cloud Appliance. For example, use AD group names such as PCA_Administrators, PCA_NetworkAdmins, PCA_InstanceLaunchers.

Gathering Required Information from ADFS

To federate with Private Cloud Appliance, you need to have the SAML metadata document and the names of the AD groups that you want to map to Private Cloud Appliance groups.

1. Locate and download the SAML metadata document for your ADFS. The default location is:

   `https://`***`your_hostname`***`/FederationMetadata/2007-06/FederationMetadata.xml`

   You will upload this document when you create the identity provider.
2. Make a note of all the AD groups that you want to map to Private Cloud Appliance groups.

> **Important:**
>
> Ensure that you have all the Private Cloud Appliance groups configured before you add AD as an identity provider.

Verifying Identity Provider Self-Signed Certificates

> **Important:**
>
> You can skip this verification step if your ADFS certificate is signed by a known certificate authority. ADFS certificates that are signed by a known certificate authority should already exist in the Private Cloud Appliance certificate bundle.

The Private Cloud Appliance Certificate Authority (CA) is a self-signed OpenSSL generated root and intermediate x.509 certificate. This CA certificate is used to issue x.509 server/client certificates, enabling you to add outside CA trust information to the rack. If you use a self-signed certificate for ADFS, you will need to add outside CA trust information from ADFS to the management nodes on the rack.

> **Note:**
>
> If you are using the `metadataUrl` property to create or update an identity provider, then you need to add the identity provider's web server's certificate chain to the Private Cloud Appliance outside CA bundle. See your identity provider's documentation for how to find the web server's certificate chain and then perform Steps 3-8 in the following procedure.

Adding Outside CA Trust Information

1. From a browser, download the SAML metadata document for your ADFS as described in Gathering Required Information from ADFS
2. Open the file in a text or XML editor and locate the signing certificate section. For example:

```
<KeyDescriptor use="signing">
<KeyInfo>
<X509Data>
<X509Certificate>
<!--CERTIFICATE IS HERE-->
</X509Certificate>
</X509Data>
</KeyInfo>
</KeyDescriptor>
```

3. Log on to management node 1. By default, the name of management node 1 is `pcamn01`.
4. Navigate to `/etc/pca3.0/vault` and create a new directory named `customer_ca`.

> **Note:**
>
> You can use this directory for multiple files. For example, you can create a file for the identity provider certificate and one for the web server's certificate chain.

5. In the `customer_ca` directory, create a new file with extension `.pem`.

6. Copy the certificate from the FederationMetadata.xml file, which is located between the `<X509Certificate>` and `</X509Certificate>` tag set, and paste the certificate into the new `.pem` file. Be sure to include the `-----BEGIN CERTIFICATE-----` and `-----END CERTIFICATE-----` lines. For example:

   ```
   -----BEGIN CERTIFICATE-----
   CERTIFICATE CONTENT
   -----END CERTIFICATE-----
   ```

7. Save and close the file.
8. Run the following command to update the `ca_outside_bundle.crt` on all management nodes:

   ```
   python3 /usr/lib/python3.6/site-packages/pca_foundation/secret_service/
   cert_generator/cert_generator_app.py -copy_to_mns
   ```

Managing Identity Providers

This section also describes how to update your identity provider (for example, to update your metadata XML file when it expires), and how to view all identity providers, view details of an identity provider, and delete an identity provider.

Adding Active Directory as an Identity Provider

To federate with AD in Private Cloud Appliance, add AD as an identity provider. You can map account groups when you add AD as an identity provider, or you can map the account groups later.

> **Important:**
>
> Ensure that you have all the Private Cloud Appliance groups configured before you add AD as an identity provider.

Using the Compute Web UI

1. Sign in with your Private Cloud Appliance login and password.
2. Open the navigation menu, click Identity, and then click Federation.
3. On the Federation page, click the Create Identity Provider button.
4. On the Create Identity Provider dialog, provide the following information:
   a. *Display Name*

      The name that the federated users see when they choose which identity provider to use to sign in to the Compute Web UI. This name must be unique across all identity providers that you add to the tenancy and cannot be changed.

   b. *Description*

      A friendly description of the identity provider.

   c. *Authentication Contexts*

      Click Add Class Reference and select an authentication context from the list.

      When one or more values are specified, Private Cloud Appliance (the relying party), expects the identity provider to use one of the specified authentication mechanisms

when authenticating the user. The returned SAML response from the identity provider must contain an authentication statement with that authentication context class reference. If the SAML response authentication context does not match what is specified here, the Private Cloud Appliance authentication service rejects the SAML response with response code 400.

d. *Encrypt Assertion (Optional)*

When enabled, the authorization service expects encrypted assertions from the identity provider. Only the authorization service can decrypt the assertion. When not enabled, the authorization service expects SAML tokens to be unencrypted, but protected, by SSL.

e. *Force Authentication (Optional)*

When enabled, users are always asked to authenticate at their identity provider when redirected by the authorization service. When not enabled, users are not asked to re-authenticate if they already have an active login session with the identity provider.

f. *Metadata*

Upload the `FederationMetadata.xml` document from your SAML 2.0 compliant identity provider. You can drag and drop the file or you can paste the XML content.

g. *Tagging (Optional)*

Add any free-form or defined tags as described in Adding Tags at Resource Creation. Tags can also be applied later.

5. Click the Create Identity Provider button.

Your new identity provider is assigned an OCID and is displayed on the Federations page.

After the identity provider is added to your tenancy, create the group mappings between Private Cloud Appliance and Active Directory, as described in Creating Group Mappings.

Listing Identity Providers

Use this procedure to list all identity providers for a tenancy.

Using the Compute Web UI

1. Open the navigation menu, click Identity and then click Federation.

The Federation page opens with a list of all identity providers that are configured in this tenancy.

Using the OCI CLI

1. Get the following information:
   - The OCID of the tenancy (`oci iam compartment list -include-root`)
   - The protocol used for federation (`SAML2`)
2. Run the identity provider list command.

```
$ oci iam identity-provider list -c ocid1.tenancy.unique_ID \
--protocol SAML2
```

Viewing Identity Provider Details

Use this procedure to show detailed information for a specific identity provider.

The details shown for the identity provider include the OCID, authentication contexts, and settings such as the redirect URL.

Using the Compute Web UI

1. Open the navigation menu, click Identity and then click Federation.

   The identity providers that are configured in this tenancy are listed.

2. For the identity provider whose details you want to view, click the Actions menu, and then click View Details.

   The details page for that identity provider is displayed.

Using the OCI CLI

1. Get the OCID of the identity provider (`oci iam identity-provider list`)
2. Run the identity provider get command.

```
$ oci iam identity-provider get --identity-provider-id
ocid1.identityprovider.unique_ID
```

Updating an Identity Provider

Using the Compute Web UI

1. Open the navigation menu, click Identity and then click Federation.

   The identity providers that are configured in this tenancy are listed.

2. For the identity provider that you want to update, click the Actions menu, and then click Edit.
3. You can change any of the following information. For more complete descriptions, see Step 4 in Adding Active Directory as an Identity Provider. Consider the affect that some of these changes have on the federation.
   - *Description*
   - *Authentication Contexts*

     Add or delete a class reference.
   - *Encrypt Assertion*

     Enable or disable encrypted assertions from the identity provider.
   - *Force Authentication*

     Enable or disable redirect authentication from the identity provider.
   - *Metadata*

     Upload a new FederationMetadata.xml document from the identity provider.
   - *Tagging*

     Add or delete any free-form or defined tags.
4. Click the Update Identity Provider button.

Deleting an Identity Provider

If you want to remove the option for federated users to log into Private Cloud Appliance, you must delete the identity provider. Deleting the identity provider also deletes all of the associated group mappings.

Using the Compute Web UI

1. Open the navigation menu, click Identity and then click Federation.

   The identity providers that are configured in this tenancy are listed.
2. For the identity provider that you want to delete, click the Actions menu and then click Delete.
3. At the Delete Identity Provider prompt, click Confirm.

   A Success pop-up displays briefly, and then the identity provider is no longer in the Federation list.

Working with Group Mappings for an Identity Provider

When working with group mappings, keep in mind the following:

- A given AD group is mapped to a single Private Cloud Appliance group.
- Private Cloud Appliance group names cannot contain spaces and cannot be changed later. Allowed characters are letters, numerals, hyphens, periods, underscores, and plus signs (+).
- You cannot update a group mapping. You can delete the mapping and add a new one.

Creating Group Mappings

After you have created an identity provider, you must create mappings from ADFS groups to Private Cloud Appliance groups.

Repeat the following procedure for each identity provider group you want to map.

Using the Compute Web UI

1. Open the navigation menu, click Identity and then click Federation.

   The identity providers that are configured in this tenancy are listed.
2. Click the identity provider for which you want to create group mappings.

   The details page for that identity provider is displayed.
3. Scroll to the Resources section and click Group Mappings.

   The group mappings for this identity provider are listed.
4. Click the Add Mappings button.

   The Create IDP Group Mapping dialog is displayed.
5. In the Name field, enter the *exact* name of the identity provider group.
6. From the Group list, select the Private Cloud Appliance group you want to map to the identity provider group.
7. Click Create IDP Group Mapping.

The new group mapping is displayed in the list.

Viewing Group Mappings

Using the Compute Web UI

1. Open the navigation menu, click Identity and then click Federation.

   The identity providers that are configured in this tenancy are listed.
2. Click the name of the identity provider.

   The details page for that identity provider is displayed.
3. Scroll to the Resources section and click Group Mappings.

   The group mappings for this identity provider are listed.

Deleting a Group Mapping

Repeat the following procedure for each identity provider group you want to delete.

Using the Compute Web UI

1. Open the navigation menu, click Identity and then click Federation.

   The identity providers that are configured in this tenancy are listed.
2. Click the identity provider for which you want to delete a group mapping.

   The details page for that identity provider is displayed.
3. Scroll to the Resources section and click Group Mappings.

   The group mappings for this identity provider are listed.
4. For the group mapping that you want to delete, click the Actions menu and then click Delete.
5. Click Confirm when prompted.

   A Success pop-up displays briefly, and then the identity provider group mapping is no longer in the Group Mappings list.

Adding Oracle Private Cloud Appliance as a Trusted Relying Party in ADFS

To complete the federation process, you must add Private Cloud Appliance as a trusted relying party in ADFS and then add associated relying party claim rules.

1. In the Compute Web UI on the Federation page, view the following text block:

   You need the Private Cloud Appliance Federation Metadata document when setting up a trust with Microsoft Active Directory Federation Services or with other SAML 2.0-compliant identity providers. This is an XML document that describes the Private Cloud Appliance endpoint and certificate information. Click Here
2. Click "Click Here."

   A metadata XML file opens in the browser with a URL similar to:

   `https://console.`***`system-name.domain-name`***`/wsapi/rest/saml/metadata/ocid1.tenancy.`***`unique_ID`***
3. Copy the metadata XML file URL.

4. From the system installed with ADFS, open a browser window and paste the URL.
5. Save the file, making sure to use the `.xml` extension. For example, ***`my-sp-metadata`*`.xml`**.
6. Go to the AD FS Management Console and sign in to the account you want to federate.
7. Add Private Cloud Appliance as a trusted relying party.
    a. Under AD FS, right-click Relying Party Trusts and the select Add Relying Party Trust.
    b. In the Add Relying Party Trust Wizard Welcome page, select Claims Aware and then click Start.
    c. On the Select Data Source page, select "Import data about the relying party from a file."
    d. Click Browse and navigate to your ***`my-sp-metadata`*`.xml`** file and then click Open.
    e. On the Specify Display Name page, enter a display name, add any optional notes for the relying party, and then click Next.
    f. On the Choose Access Control Policy page, select the type of access you want to grant and then click Next.
    g. On the Ready to Add Trust page, review the settings, and then click Next to save your relying party trust information.
    h. On the Finish page, check "Configure claims issuance policy for this application" and then click Close.

       The Edit Claim Issuance Policy dialog appears, which you can leave open for the next section.

Adding Relying Party Claim Rules

After you add Private Cloud Appliance as a trusted relying party, you must add the claim rules so that the elements required (Name ID and groups) are added to the SAML authentication response.

To add a Name ID rule:

1. In the Edit Claim Issuance Policy dialog, click Add Rule.

   The Select Rule Template dialog is displayed.
2. For Claim rule template, select Transform an Incoming Claim and then click Next.
3. Enter the following:
    - *Claim rule name*: Enter a name for this rule. For example, `nameid`.
    - *Incoming claim type*: Select the Microsoft Windows account name.
    - *Outgoing claim type*: Select a claim type, for example, Name ID.
    - *Outgoing name ID format*: Select Persistent Identifier.
    - Select Pass through all claim values and then click Finish.

      The rule is displayed in the rules list.

The Issuance Transform Rules dialog displays the new rule.

If your AD users are in no more than 100 groups, you simply add the groups rule. However, if your AD users are in more than 100 groups, those users cannot be authenticated to use the Private Cloud Appliance Compute Web UI. For these groups, you must apply a filter to the groups rule.

To add the groups rule:

1. In the Issuance Transform Rules dialog, click Add Rule.

   The Select Rule Template dialog is displayed.
2. For Claim rule template, select Send Claims Using a Custom Rule and then click Next.
3. In the Add Transform Claim Rule Wizard, enter the following:
   a. *Claim rule name*: Enter groups.
   b. *Custom rule*: Enter the custom rule.

      ```
      c:[Type == "http://schemas.microsoft.com/ws/2008/06/identity/claims/
      windowsaccountname", Issuer == "AD AUTHORITY"] => issue(store = "Active
      Directory", types = ("https://auth.oraclecloud.com/saml/claims/groupName"),
      query = ";tokenGroups;{0}", param = c.Value);
      ```

   c. Click Finish.

The Issuance Transform Rules dialog displays the new rule.

Disable the Certificate Revocation Check

For ADFS to work with SAML, you must disable the Certificate Revocation List (CRL) checking.

1. Open Powershell on the ADFS system and enter the following command, where *TRUST_NAME* is the name of the relying party trust:

   ```
   Get-AdfsRelyingPartyTrust -Name '<TRUST_NAME>' | Set-AdfsRelyingPartyTrust -
   EncryptionCertificateRevocationCheck None -SigningCertificateRevocationCheck None
   ```

Setting Up Policies for the Groups

Configure policies to control the access the federated users have to the Private Cloud Appliance resources. For general information about policies, see "How Policies Work" in the Identity and Access Management Overview in the *Oracle Private Cloud Appliance Concepts Guide*. For specific information, see Managing Policies.

Providing Federated Users Sign-in Information

Do the following to enable a federated user to log into the Private Cloud Appliance Compute Web UI:

- Provide the user with the URL for the Compute Web UI.
- Provide the user with the name of the tenancy to which they have access.
- Ensure that you have configured the necessary group mappings.
- Ensure that you have configured the necessary policies.

> **Important:**
>
> A federated user cannot log into Private Cloud Appliance by using the OCI CLI.

Configuring Instances for Calling Services

A Private Cloud Appliance compute instance can be configured to enable applications running on the instance to call services and manage resources similar to the way Private Cloud Appliance users call services to manage resources.

The IAM service feature that enables instances to be authorized actors (or principals) to perform actions on service resources is called an *instance principal*.

Perform the following steps to set up and use an instance as a principal:

- Configure the instance firewall to enable the instance to access service endpoints. See Configuring Instance Firewalls to Allow Calling Services.
- Ensure the instance is included in a dynamic group that grants permissions to access the required resources. See Creating and Managing Dynamic Groups.

  The instance must be created in or moved to a compartment that is named in a matching rule of the dynamic group, or the instance must have a resource tag assigned that is named in a matching rule. See Writing Matching Rules.

Configuring Instance Firewalls to Allow Calling Services

This topic describes how to modify the instance firewall configuration and how to create a `systemd` service to restore the changes if the system reboots.

Modify the Firewall Configuration

As a privileged user, modify the instance firewall configuration to enable the instance to access service endpoints such as `iaas` and `identity`.

Use the `iptables` command to add the following `BareMetalInstanceServices` rules to the instance firewall:

```
iptables -I BareMetalInstanceServices 14 -p tcp -d 169.254.169.254 --dport 443 -j ACCEPT
iptables -I BareMetalInstanceServices 14 -p tcp -d 169.254.240.254 --dport 443 -j ACCEPT
```

The first entry is required for all endpoints. The second entry is required to contact the Object Storage endpoint.

Make the Configuration Changes Persistent

Use the following procedure to make these firewall configuration changes persist across instance reboots.

1. Save the updated IP tables configuration.

   ```
   iptables-save > /etc/sysconfig/iptables.rules
   ```

2. Create a script to automatically restore the current (modified) firewall configuration on reboot.

   In this example, the script is named `/sbin/restore-iptables.sh`. The following is the content of the file `/sbin/restore-iptables.sh`:

   ```
   #!/bin/sh
   /sbin/iptables-restore < /etc/sysconfig/iptables.rules
   ```

3. Set executable bit on the script.

   ```
   chmod +x /sbin/restore-iptables.sh
   ```

4. Create a `systemd oneshot` service to execute the `/sbin/restore-iptables.sh` script at boottime.

   In this example, the service is named `/etc/systemd/system/restore-iptables.service`. The following is the content of the file `/etc/systemd/system/restore-iptables.service`:

   ```
   [Unit]
   Description=Restore IP Tables
   After=cloud-final.service

   [Service]
   ExecStart=/sbin/restore-iptables.sh
   User=root
   Group=root
   Type=oneshot

   [Install]
   WantedBy=multi-user.target
   ```

5. Reload the `systemd` manager configuration, and enable the service to run at boot time.

   ```
   systemctl daemon-reload
   systemctl enable restore-iptables
   ```

Configuring Instance Certificates to Allow Calling Services

By default, Private Cloud Appliance endpoints (such as `iaas` and `identity`) offer a certificate that is signed by a CA that is specific to that appliance. By default, operating systems do not trust certificates that are signed by a CA that is specific to this Private Cloud Appliance. If the OS does not trust the certificates that are offered, attempts to use the OCI SDK or OCI CLI will fail with a `CERTIFICATE_VERIFY_FAILED` error.

Implement one of the solutions described in this topic to successfully use the OCI SDK or OCI CLI on the instance.

> **Note:**
>
> Any user who can SSH to the instance automatically inherits the privileges granted to the instance.

Option 1: Bring Your Own Certificate (BYOC)

If you have a certificate that is signed by a CA that your OS trusts, configure the Private Cloud Appliance to offer that certificate.

On a Linux OS, the following command lists CAs that are trusted by default:

```
trust list --filter=ca-anchors
```

For information about how to provide your own certificate, see "Accessing External Interfaces with Your CA Trust Chain" in the Hardware Administration chapter of the *Oracle Private Cloud Appliance Administrator Guide*.

Option 2: Specify in the SDK Code the CA Bundle to Use

This method copies the appliance-specific CA bundle to the instance, but does not verify the server's certificate (`--insecure`). To ensure security, verify the content of the retrieved bundle (`external_ca.crt`).

1. Retrieve the certificate from the `iaas` endpoint of the appliance.

   ```
   curl --insecure -sS -o external_ca.crt --noproxy "*" https://
   iaas.pca_name.domain_name/cachain
   ```

   This command could be in a script that is passed to the instance at launch time by using either the `--user-data-file` option or the `--metadata` option with a `user_data` field. The script will be run by cloud-init inside the instance during init, saving the effort of manually retrieving this certificate file on a large number of instances.
2. Verify the content of the CA bundle saved in the `external_ca.crt` file.
3. Specify the CA bundle in the Python SDK code.

   ```
   signer = oci.auth.signers.InstancePrincipalsSecurityTokenSigner(
       federation_client_cert_bundle_verify="/home/opc/external_ca.crt"
   )
   identity_client = oci.identity.IdentityClient(config={}, signer=signer)
   identity_client.base_client.session.verify = "/home/opc/external_ca.crt"
   ```

Option 3: Globally Trust the Private Cloud Appliance CA Bundle

This method is the same as the preceding method with the following difference: Instead of specifying the CA bundle in the SDK code, this method adds the CA bundle to the trust chain.

> **Note:**
>
> When the CA bundle is added to the trust chain, every application on this compute instance will trust certificates signed with the CA specified in this bundle. Consider whether this is an acceptable security risk.

1. Retrieve the certificate from the `iaas` endpoint of the appliance.

   ```
   curl --insecure -sS -o external_ca.crt --noproxy "*" https://
   iaas.pca_name.domain_name/cachain
   ```
2. Verify the content of the CA bundle saved in the `external_ca.crt` file.
3. Update the global CA trust chain.

   ```
   cp external_ca.crt /etc/pki/ca-trust/source/anchors/
   update-ca-trust extract
   ```

Steps 1 and 3 in this method could be in a script that is passed to the instance at launch time by using either the `--user-data-file` option or the `--metadata` option with a `user_data` field. The script will be run by cloud-init inside the instance during init, saving the effort of performing these steps manually on a large number of instances.

Configuring Python SDK and OCI CLI for Instance Principals

This topic describes how to enable instance principal authorization for the Python SDK, OCI CLI, or Terraform.

Enabling Instance Principal Authorization for the Python SDK

In your SDK for Python, create an `oci.auth.signers.InstancePrincipalsSecurityTokenSigner` object as shown in the following example:

```
# By default this will hit the auth service in the region returned by http://
169.254.169.254/opc/v1/instance/region on the instance.

signer = oci.auth.signers.InstancePrincipalsSecurityTokenSigner()
identity_client = oci.identity.IdentityClient(config={}, signer=signer)
```

To refresh the token without waiting, use the following command:

```
signer.refresh_security_token()
```

Enabling Instance Principal Authorization for the OCI CLI

Set the authorization option (`--auth`) for a command as shown in the following example:

```
oci os ns get --auth instance_principal
```

Alternatively, set the following environment variable:

```
OCI_CLI_AUTH=instance_principal
```

If both are set, the value set for `--auth` takes precedence over the environment variable.

Enabling Instance Principal Authorization for Terraform

Set the `auth` attribute to "InstancePrincipal" in the provider definition as shown in the following example:

```
variable "region" {}

provider "oci" {
   auth = "InstancePrincipal"
   region = "${var.region}"

}
```

When you use instance principal authorization you do not need to include the `tenancy_ocid`, `user_ocid`, `fingerprint`, and `private_key_path` attributes.

Creating and Managing Dynamic Groups

Dynamic groups are groups of compute instances that meet the criteria defined for the group. The membership of the group changes as instances newly satisfy the criteria or no longer satisfy the criteria. For example, if the members of a dynamic group are defined to be all instances in a specified compartment, the group membership changes when instances are created in or removed from that compartment.

Policies permit applications that are running on instances that are members of dynamic groups to make API calls to perform actions on service resources. Instances authenticate by using certificates that are assigned to each member instance.

Creating a Dynamic Group

When you create a dynamic group, the group is automatically created in the tenancy. You cannot specify a different compartment for the dynamic group.

Using the Compute Web UI

1. In the navigation menu, click Identity, and then click Dynamic Groups.
2. Click the Create Dynamic Group button.

3. In the Create Dynamic Group dialog, enter the following information:
   - *Name*: A name for this dynamic group. Names have the following characteristics:
     - Must be unique within the tenancy. You can create a dynamic group with the same name as a dynamic group that has been deleted.
     - Are case insensitive.
     - Cannot be changed later.
     - Can be no more than 100 characters.
     - Can contain only alphanumeric characters, period (.), hyphen (-), and underscore (_).
   - *Description*: A description for this dynamic group. The description has the following characteristics:
     - Must be 1-400 characters.
     - Does not need to be unique.
     - Can be changed later.
   - *Matching rules*: Select either "Match any rules defined below" or "Match all rules defined below."

     Matching Rule1: Enter a matching rule. See Writing Matching Rules for information about how to define a matching rule.

     Click the + Another Matching Rule button to add another matching rule.
   - *Tagging*: (Optional) Add defined or free-form tags for this group as described in Adding Tags at Resource Creation. Tags can also be applied later.
4. Click the Create Dynamic Group button on the dialog.

   The details page of the new dynamic group is displayed.
5. Create access policies for this dynamic group or add this dynamic group to existing policies. The member instances have no permissions unless the group is the subject of at least one policy. See Managing Policies.

Using the OCI CLI

1. Get the following information:
   - A name and description for the group. See the Compute Web UI procedure for limitations. In the OCI CLI, a description must be provided but its value can be an empty string.
   - OCID of the tenancy. On your user menu, click the Tenancy option.
2. Create a matching rule to specify conditions for membership in the group.

   The matching rule is a single text string. Multiple rules are comma separated, enclosed in braces, and preceded by the keyword `any` or `all`. See the syntax for a rule with multiple conditions in Writing Matching Rules.
3. Run the create dynamic group command.

   Syntax:

   ```
   oci iam dynamic-group create --name text --description "text" \
   --compartment-id tenancy_OCID --matching-rule text
   ```

   Example:

```
$ oci iam dynamic-group create --name Project-A --description "Instances for Project
A." \
--compartment-id ocid1.tenancy.unique_ID \
--matching-rule "instance.compartment.id = 'ocid1.compartment.unique_ID'"
```

The output of this command is the same as the output of the `dynamic-group get` command.

4. Create an access policy for this dynamic group or add this dynamic group to an existing policy. The group has no permissions unless it is the subject of at least one policy. See Managing Policies.

Writing Matching Rules

A rule specifies one or more conditions for membership in the group. Type the rule into the text box. To add another rule, click the +Another Matching Rule button.

A rule with a single condition has the following syntax:

```
variable = | != value
```

A rule with multiple conditions has the following syntax:

```
any | all {variable =|!= value, variable =|!= value,...}
```

variable is one of the following:

- `instance.compartment.id`. The OCID of the compartment where the instance resides.
- `instance.id`. The OCID of the instance.
- `tag.`***`tagnamespace`***`.`***`tagkey`***`.value = `***`tagvalue`***. The ***`tagnamespace`***, ***`tagkey`***, and ***`tagvalue`*** all must match to include this resource in the group.
- `tag.`***`tagnamespace`***`.`***`tagkey`***`.value`. Only the ***`tagnamespace`*** and ***`tagkey`*** must match to include this resource in the group. The value of the tag is not considered.

The following matching rule includes all instances in the specified compartment:

```
instance.compartment.id = 'ocid1.compartment.unique_ID'
```

The following matching rule includes all instances in the specified compartment except for the two instances specified with `instance.id !=`:

```
all {instance.compartment.id = 'ocid1.compartment.unique_ID',
 instance.id != 'ocid1.instance.unique_ID1', instance.id != 'ocid1.instance.unique_ID2'}
```

The following matching rule includes all instances that have a tag applied that has tag namespace Product1, key ProjectA, and value abc123:

```
tag.Product1.ProjectA.value = abc123
```

Updating a Dynamic Group

You can change the description and matching rules (members) for a dynamic group. You can add, change, or remove tags as described in Applying Tags to an Existing Resource.

Using the Compute Web UI

1. In the navigation menu, click Identity, and then click Dynamic Groups.
2. (Optional) Modify the description.

a. In the dynamic groups list, for the dynamic group that you want to modify, click the Actions menu, and click the Edit option.
b. In the Edit ***groupname*** dialog, modify the dynamic group's description or tags.
c. Click Save Changes.

3. (Optional) Modify the matching rules.
   a. In the dynamic groups list, click the name of the dynamic group that you want to modify.
   b. On the details page, click the Edit All Matching Rules button.
   c. Edit the matching rule in the text box.
   d. Click Save Changes.

Using the OCI CLI

1. Get the OCID of the dynamic group that you want to modify: `oci iam dynamic-group list`
2. Run the update dynamic group command.

   Syntax:

   ```
   oci iam dynamic-group update --dynamic-group-id dynamic_group_OCID
   ```

   Example:

   ```
   $ oci iam dynamic-group update --dynamic-group-id ocid1.dynamicGroup.unique_ID \
   --matching-rule "instance.compartment.id = 'ocid1.compartment.unique_ID'"
   ```

   The output of this command is the same as the output of the `dynamic-group get` command.

Deleting a Dynamic Group

When you delete a dynamic group, the compute instances that had been members of the group no longer have the authorizations that they had through the policies applied to the dynamic group.

Using the Compute Web UI

1. In the navigation menu, click Identity, and then click Dynamic Groups.
2. For the dynamic group that you want to delete, click the Actions menu, and click the Delete option.
3. Confirm the deletion.

Using the OCI CLI

1. Get the OCID of the dynamic group that you want to modify: `oci iam dynamic-group list`
2. Run the delete dynamic group command.

   Syntax:

   ```
   oci iam dynamic-group delete --dynamic-group-id dynamic_group_OCID
   ```

   Example:

   ```
   $ oci iam dynamic-group delete --dynamic-group-id ocid1.dynamicGroup.unique_ID \
   --force
   ```

Managing Policies

A policy is a named set of one or more policy statements. Policy statements grant permissions to users to access resources.

When designing access policies, remember the following policy characteristics:

- The policy will apply to the compartment where you attach the policy and to all subcompartments of that compartment. Permissions granted in a particular compartment, including the tenancy, are inherited by all subcompartments of that compartment.
- A user can be a member of more than one group. A group can be the subject of more than one policy. A policy can have up to 50 policy statements.
- If some users need full access to the named resources and other users only need to use the resources, you need to create multiple groups and multiple policies. A tenancy can have up to 100 policies.
- Users who have full access to resources in a subcompartment probably also need view or use access to related resources in that compartment and in parent compartments. For example, users who have access to create instances in a compartment might also need access to use tag namespaces to apply defined tags to the instances, or access to read images in a different compartment.

For conceptual information, see "How Policies Work" in the Identity and Access Management Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

Writing Policy Statements

A policy can have up to 50 policy statements. A tenancy can have up to 100 policies. Decide what you want your policy to permit, and use the information in this section to write the necessary statements.

Ensure that the groups and compartments that you plan to name in the policy statements exist. Note the name or OCID of each group and compartment that you want to use.

> **Note:**
>
> If you use names in your policy statements instead of OCIDs, the policy will still be valid if the name of the group or compartment is subsequently changed. Internally, the OCID, not the name, is used. However, the policy could be more difficult for administrators to understand if the name of the group or compartment changes.

If you plan to use a tag to apply the policy to more than one group or more than one compartment, ensure that the tag exists. Note the name of the tag namespace, the name of the key, and the value that you want to use in the policy statement.

For conceptual and reference information and examples of common statements, see "How Policies Work" in the Identity and Access Management Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

Policy Statement Syntax

Policy statements grant permissions to users or compute instances to access resources. The users or instances are also called the subject of the policy, and the permissions are also called the verb. The resource type and compartment define the set of possible resources to which the subjects are granted access. This set of resources is also called the target. Conditions can be used to narrow the subject, the target, and the operations that can be performed on the target.

The following is the policy statement syntax:

```
allow subject to permissions [resource_type] in compartment [ where conditions ]
```

Keywords `allow`, `to`, and `in` are required and are case insensitive.

subject
One or more user groups (`group`) or `any-user`, or one or more dynamic groups (`dynamic-group`). To specify more than one user group or dynamic group, use a comma between each two group names or OCIDs in a list. You cannot specify both group names and group OCIDs. You can specify the keyword `any-user` to grant permission to all users.

- `group` ***group_name***`[,`***group_name*** `…]`
- `group id` ***group_ocid***`[,`***group_ocid*** `…]`
- `any-user`

permissions

- One of `inspect`, `read`, `use`, or `manage`. For descriptions of the access that these permission aggregators grant, see "Verb" in "Policy Syntax" in the Identity and Access Management Overview in the *Oracle Private Cloud Appliance Concepts Guide*.
- One or more specific permissions, such as `INSTANCE_UPDATE`, in the following form:

  ```
  { PERMISSION_1[, PERMISSION_2]... }
  ```

  If you use this option, do not specify a ***resource_type***. The resource type is included in the permission.

To grant both a permission aggregator and one or more specific permissions for the same resource, use two policy statements.

resource_type
A single keyword that represents one of the following:

- A single resource type, such as `instances` or `volumes`.
- A family of resource types. For example, the `instance-family` resource type family includes the following resource types:
  - `app-catalog-listing`
  - `console-histories`
  - `instances`
  - `instance-console-connection`
  - `instance-images`
- `all-resources`

If you list specific permissions instead of one of the four permissions aggregator keywords, do not specify a ***resource_type***. The resource type is included in the permission.
For a list of resource types, see the table in "Resource Types" in "Policy Syntax" in the Identity and Access Management Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

compartment
A single compartment name or OCID or `tenancy`.

- `compartment` ***compartment_name***
- `compartment id` ***compartment_OCID***
- `tenancy`

To grant access in multiple compartments, use multiple statements.

condition
A predefined variable followed by an operator and a value. See Using Conditions.

Using Conditions

Conditions can be specified in a policy statement to narrow the set of users who are granted access, the set of resources to which the users are granted access, and the operations that can be performed on the resources. A condition is a predefined variable with a value that you specify. You can specify a list of conditions with AND and OR relationships. The entire condition clause must evaluate to true in order for access to be granted. For information about conditions that might unexpectedly evaluate to false, see Conditions That Are Not Applicable.

The following is the syntax of the condition clause:

```
where condition
where all|any {condition[, condition]...}
```

The syntax of ***condition*** is:

variable op 'value'

variable
See Table 2-1.

op

- `=` (equal) or `!=` (not equal) – Applies to all variables.
- `in` or `not in` – See Using Defined Tags in Conditions.

value
The ***value*** can be a fully specified string or can use the `*` wildcard. If the ***value*** is a fully specified string, enclose the value in single quotation marks. If you use `*`, enclose the value in forward slashes (`/`):

```
'BU1-ProdX'
/*Prod*/
/*ProdX/
/BU1-Prod*/
```

Condition values are case insensitive. For example, a condition with a value of BucketA also applies to bucket bucketA in the same compartment if such a bucket exists.

In the following table, variables that begin with `request` refer to the request that is being made: A user has clicked a Compute Web UI option or entered a OCI CLI command. Variables that begin with **`target`** refer to the resource that the user clicked or named in the command.

Table 2-1 Supported Predefined Variables for Conditions

Variable	Description
`request.groups.id`	The list of groups that the requesting user belongs to.
`request.operation`	The name of the operation that is being attempted.
`request.permission`	The names of the permissions that are required to perform the operation.
`target.compartment.id`	The OCID of the compartment that contains the target resource. The compartment that contains the target resource could be a child compartment of the compartment specified in the `in` clause in the policy statement.
`target.compartment.name`	The name of the compartment that contains the target resource. The compartment that contains the target resource could be a child compartment of the compartment specified in the `in` clause in the policy statement.
`target.user.id`	The OCID of the target user. This OCID is not available when the requested permission is to create the user.
`target.user.name`	The name of the target user.
`target.group.id`	The OCID of the target group. This OCID is not available when the requested permission is to create the group.
`target.group.name`	The name of the target group.
`target.group.member`	True if the requesting user belongs to the target group.
`target.policy.id`	The OCID of the target policy. This OCID is not available when the requested permission is to create the policy.
`target.policy.name`	The name of the target policy.
`target.tag-namespace.id`	The OCID of the tag namespace that the user is requesting to list, update, or delete.
`target.tag-namespace.name`	The name of the tag namespace that the user is requesting to create or update. Use commas to separate multiple names.
`request.principal.group.tag`	See Using Defined Tags in Conditions.
`target.resource.tag`	See Using Defined Tags in Conditions.
`target.resource.compartment.tag`	See Using Defined Tags in Conditions.

Example: Specify Permissions Using `request.permission`

To grant users the ability to create objects but not the ability to delete objects, you can grant `manage` access and then specify a condition that says only create and inspect access are granted:

```
allow group ObjectWriters to manage objects in compartment ABC
where any {request.permission='OBJECT_CREATE', request.permission='OBJECT_INSPECT'}
```

Example: Specify Compartments Using `target.compartment.name` and Wildcards

The following example grants users the ability to manage all resources in `virtual-network-family` in any compartment that has a name that begins with X except for compartment XYZ:

```
allow group NetworkAdmins to manage virtual-network-family in tenancy
where all {target.compartment.name=/X*/,target.compartment.name!='XYZ'}
```

Example: Nested Conditions

The following policy enables users in group BucketAdmins to either read, update, or manage retention rules for BucketA in compartment ABC:

```
allow group BucketAdmins to manage buckets in compartment ABC
where all {target.bucket.name='BucketA',
any {request.permission='BUCKET_UPDATE', request.permission='BUCKET_READ',
RETENTION_RULE_MANAGE}}
```

Because the policy is for a specific named bucket, this policy does not permit users to retrieve a list of buckets. To permit users to retrieve a list of buckets, add the following separate statement:

```
allow group BucketAdmins to inspect buckets in compartment ABC
```

See Conditions That Are Not Applicable.

Example: Apply Defined Tags

The following example enables users in groups BucketAdmins and ObjectWriters to apply tags in the StorageTags tag namespace:

```
allow group BucketAdmins,ObjectWriters to use tag-namespaces in tenancy
where target.tag-namespace.name='StorageTags'
```

Example: Edit Any Group Where You Are a Member

The following example enables all users to edit any group where they are members:

```
allow any-user to use groups in tenancy where target.group.member='true'
```

Conditions That Are Not Applicable

If a condition is not applicable to the rest of the policy statement, then that condition evaluates to false and access is not granted.

A condition is not applicable if it is testing information that is not available in the request. For example, the following policy statement grants `use` access to the resource `users`, but does not allow the requesting users to list or update users, even though those permissions are included in the `use` permission:

```
allow group GroupAdmins to use users in tenancy where target.group.name !=
'Administrators'
```

The request to list users or update a user does not include information about groups. The list users and update user requests have no value for `target.group.name`. The test fails, and the request to list or update a user is denied.

To fix this example, you could remove the `where` clause and allow only `inspect` or `read` access.

Using Defined Tags in Conditions

Certain conditions evaluate the value of a defined tag that has been applied to a user, compartment, or resource. In these conditions, the predefined variable can be called a tag variable.

Using conditions with tag variables enables you to do the following:

- Write a single policy statement that applies to multiple user groups, compartments, or resources.
- Change the permissions that are granted without changing the policy statement. Instead, to allow or revoke access, apply tags to different resources or remove tags from resources.

See Resource Tag Management for information about how to create and apply defined tags.

The general syntax of a condition that uses tag variables is the same as the syntax of a condition that uses other condition variables:

variable op 'value'

The value of each of these three parts is specialized for tags.

variable
Tag condition variables include the name of the tag namespace and the name of the key in the variable name:

base_variable_name.tag_namespace_name.tag_key_name

op
One of `=`, `!=`, `in`, or `not in`.
The `in` and `not in` operations refer to members of the set of possible values for the tag.

value
The ***value*** is a value of the defined tag. The value can be a single value or a list of values.

The following tag variables are supported:

`request.principal.group.tag`

This variable potentially grants access to multiple groups in one statement. The following statement allows any user that is a member of a group that has been tagged with tag Operations>Project>ABC to manage instance resources in compartment ProdX:

```
allow any-user to manage instance-family in compartment ProdX
where request.principal.group.tag.Operations.Project='ABC'
```

If you replace 'ABC' in the preceding statement with '*' or /*/, a user that is a member of a group that has been tagged with any value of Operations>Project could manage instance resources in compartment ProdX.

`target.resource.compartment.tag`

This variable potentially grants access to multiple compartments in one statement. The following statement allows users in group NetAdmins to use network resources in any compartment that has been tagged with either tag Operations>Project>ABC or tag Operations>Personnel>Test:

```
allow group NetAdmins to use virtual-network-family in tenancy where
any { target.resource.compartment.tag.Operations.Project='ABC',
target.resource.compartment.tag.Operations.Personnel='Test' }
```

If you replace `any` with `all` in the preceding statement, the statement allows users in group NetAdmins to use network resources in any compartment that has been tagged with both tag Operations>Project>ABC and tag Operations>Personnel>Test.

The following statement allows users in group NetAdmins to use network resources in any compartment that has been tagged with either tag Operations>Personnel>Development or tag Operations>Personnel>Test:

```
allow group NetAdmins to use virtual-network-family in tenancy where
target.resource.compartment.tag.Operations.Personnel in ('Development', 'Test')
```

```
target.resource.tag
```

This variable grants access to one or more resources of the specified type. The following statement allows group Xadmins to use any instance in compartment ProdX that is tagged with tag Operations>Project>XYZ.

```
allow group Xadmins to use instances in compartment ProdX
where target.resource.tag.Operations.Project = 'XYZ'
```

Writing Policies to Access Resources Across Tenancies

Before You Begin

You can write policies to allow tenancy access from other tenancies so you can share resources across tenancies. The administrators of both tenancies need to create special policy statements that explicitly state which resources can be accessed and shared. These special statements use the following special verbs:

- **Endorse**: This policy statement describes what work a group in a *source tenancy* can perform in other tenancies. You write the `endorse` statement for the tenancy that contains the group of users who need to work with another tenancy's resources.
- **Admit**: This policy statement describes what work a group from other tenancies can perform in a *destination tenancy*. You write the `admit` statement for the tenancy that is granting permission to access its resources. The `admit` statement identifies the group of users from the source tenancy that requires resource access in the destination tenancy.
- **Define**: This policy statement is used to assign an alias for a *source tenancy* OCID, a *source group* OCID, and a *destination tenancy* OCID. You define a *source tenancy* alias and a *source group* alias for use in `admit` policy statements. You define a *destination tenancy* alias for use in `endorse` policy statements.

  You must include a `define` statement in the same policy entity as the `endorse` or `admit` statement.

The `endorse` and `admit` statements work together. An `endorse` statement resides in the source tenancy while an `admit` statement resides in the destination tenancy. Without a corresponding statement that specifies access, a particular `endorse` or `admit` statement grants no access. *Both tenancies must agree on access and have policies that allow for access.*

In the source tenancy, you write `define` and `endorse` policy statements using the following syntax:

```
define tenancy destination-tenancy-alias as tenancy_ocid
```

```
endorse group group-name to verb resource in tenancy destination-tenancy-alias
```

In the destination tenancy, you write two `define` policy statements and an `admit` policy statement using the following syntax:

```
define tenancy source-tenancy-alias as tenancy_ocid

define group source-group-alias as group_ocid

admit group source-group-alias of tenancy source-tenancy-alias to verb resource in
compartment/tenancy
```

For conceptual and reference information and examples of common statements, see "How Policies Work" in the Identity and Access Management Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

Source Tenancy Policy Statements

An administrator for the source tenancy creates policy statements that endorses a group in their tenancy to manage resources in a destination tenancy.

You can write broad policy statements for the source tenancy as in the following examples.

- To endorse a specific group, such as `StorageAdmins`, to do anything with all object storage resources in any tenancy:

  ```
  endorse group StorageAdmins to manage object-family in any-tenancy
  ```

- To endorse a specific group, such as `DNSAdmins`, to do anything with all DNS resources in any tenancy:

  ```
  endorse group DNSAdmins to manage dns in any-tenancy
  ```

You can write a policies that reduce the scope of access for a source tenancy group. For this type of policy, the administrator for the source tenancy must define an alias for the destination tenancy OCID and reference that alias in the policy's statements, for example:

- To endorse a specific group, such as `StorageAdmins`, to manage object storage resources in a specific tenancy, such as `DestinationTenancy`, you would use the following policy statements:

  ```
  define tenancy DestinationTenancy as ocid1.tenancy.oc1..<unique_ID>
  endorse group StorageAdmins to manage object-family in tenancy DestinationTenancy
  ```

- To endorse a specific group, such as `DNSAdmins`, to manage DNS resources in a specific tenancy, such as `DestinationTenancy`, you would use the following policy statements:

  ```
  define tenancy DestinationTenancy as ocid1.tenancy.oc1..<unique_ID>
  endorse group DNSAdmins to manage dns in tenancy DestinationTenancy
  ```

The destination tenancy must also have a policy that allows the source tenancy group access to the destination tenancy and its resources. Without this corresponding policy in the destination tenancy, the source tenancy group would not be able to access the destination tenancy or its resources.

For more information on writing policy statements, see "How Policies Work" in the Identity and Access Management Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

Destination Tenancy Policy Statements

An administrator for the destination tenancy creates policy statements that allows a group in a source tenancy access to the destination tenancy and its resources.

You can write broad policy statements for the destination tenancy as in the following examples.

- To allow a specific group in a source tenancy, such as `StorageAdmins`, to do anything with all object storage resources in the destination tenancy:

```
define tenancy SourceTenancy as ocid1.tenancy.oc1..<unique_ID>
define group StorageAdmins as ocid1.group.oc1..<unique_ID>
admit group StorageAdmins of tenancy SourceTenancy to manage object-family in tenancy
```

- To allow a specific group in a source tenancy, such as `DNSAdmins`, to do anything with all DNS resources in the destination tenancy:

```
define tenancy SourceTenancy as ocid1.tenancy.oc1..<unique_ID>
define group DNSAdmins as ocid1.group.oc1..<unique_ID>
admit group DNSAdmins of tenancy SourceTenancy to manage dns in tenancy
```

You can write policies that reduces the scope of access for a source tenancy group to destination tenancy resources. For these types of policies, an administrator for the destination tenancy must define aliases for the source tenancy OCID and source group OCID, and then reference those aliases in the policy's statements, for example:

- To allow a specific group in the source tenancy, such as `StorageAdmins`, to manage object storage resources in a specific compartment, such as `SharedBuckets`, you would use the following policy statements:

```
define tenancy SourceTenancy as ocid1.tenancy.oc1..<unique_ID>
define group StorageAdmins as ocid1.group.oc1..<unique_ID>
admit group StorageAdmins of tenancy SourceTenancy to manage object-family in
compartment SharedBuckets
```

- To allow a specific group in the source tenancy, such as `DNSAdmins`, to manage DNS resources in a specific compartment, such as `SharedZomes`, you would use the following policy statements:

```
define tenancy SourceTenancy as ocid1.tenancy.oc1..<unique_ID>
define group DNSAdmins as ocid1.group.oc1..<unique_ID>
admit group DNSAdmins of tenancy SourceTenancy to manage dns in compartment
SharedZones
```

The source tenancy must also have a policy that endorses the source tenancy group's access to the destination tenancy and its resources. Without this corresponding policy in the source tenancy, the source tenancy group would not be able to access the destination tenancy or its resources.

For more information on writing policy statements, see "How Policies Work" in the Identity and Access Management Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

Creating a Policy

Before You Begin

A policy must have at least one policy statement. You cannot create an empty policy and add statements later. Decide what you want your policy to allow, and see Writing Policy Statements to design the necessary statements.

Using the Compute Web UI

1. In the navigation menu, click Identity, and then click Policies.
2. Click Create Policy.
3. In the Create Policy dialog, enter the following information:
   - *Name*: The policy name. Policy names have the following characteristics:
     - Must be unique within the tenancy.
     - Are case insensitive.

  - Cannot be changed later.
  - Can be no more than 100 characters.
  - Cannot include spaces. Can include only letters, numbers, hyphens, periods, or underscores.
- *Description*: A description for the policy. This description can be no more than 400 characters.
- *Create in Compartment*: Select the compartment where you want to attach this policy. The policy will apply to this compartment and all child compartments of this compartment.
- *Statements*: Enter a policy statement. For information about how to write policy statements, see Writing Policy Statements.

  To add a second policy statement, click the +Another Statement button. You can enter up to 50 statements. If you create more than one policy statement, you can click the X button next to a statement to delete that statement.
- *Tagging*: (Optional) Add defined or free-form tags for this policy as described in Adding Tags at Resource Creation. Tags can also be applied later.

4. Click the Create Policy button.

   The details page for the new policy is displayed. The Resources section of the page shows the policy statements.

Using the OCI CLI

1. Get the OCID of the compartment where you want to attach the policy. The policy will apply to this compartment and all child compartments of this compartment.

   ```
   $ oci iam compartment list --compartment-id-in-subtree true
   ```

2. Construct an argument for the `--statements` option.

   The value of the `--statements` option argument is an array of policy statements in JSON format. This argument can be provided as a string on the command line or in a file. For information about how to write policy statements, see Writing Policy Statements.

3. (Optional) Construct arguments for defined or free-form tags for this policy as described in Adding Tags at Resource Creation. Tags can also be applied later.

4. Run the policy create command.

   Syntax:

   ```
   oci iam policy create -c compartment_OCID --name text --description "text" \
   { --statements '["statement","statement"]' | --statements file://policy.json }
   ```

   The ***compartment_OCID*** is the compartment where you want to attach this policy. See the Compute Web UI procedure for characteristics of the name and description values. See Adding Tags at Resource Creation to add defined and free-form tags.

   This command returns the same output as the `policy get` command.

Updating a Policy

Using the Compute Web UI To Modify the Policy Description or Tags

1. In the navigation menu, click Identity, and then click Policies.

2. If the policy that you want to modify is not listed, select the correct compartment from the Compartment drop-down menu above the policies list.
3. For the policy that you want to modify, click the Actions menu for the policy, and click the Edit option.
4. Update the description or tags.

   To modify tags, see Applying Tags to an Existing Resource.
5. Click the Save Changes button.

Using the Compute Web UI To Modify the Policy Statements

1. In the navigation menu, click Identity, and then click Policies.
2. If the policy that you want to modify is not listed, select the correct compartment from the Compartment drop-down menu above the policies list.
3. Click the name of the policy that you want to modify.
4. On the policy details page, scroll to the Resources section.
5. In the Statements list, click the Configure Policy Statements button.
6. In the Edit Statements in the ***policy_name*** Policy dialog, change or add policy statements.

   To modify policy statements, see Writing Policy Statements.

   To add a policy statement, click the Another Statement button. You can enter up to 50 statements.

   If more than one policy statement exists, you can click the X button next to a statement to delete that statement.
7. Click the Submit button.

Using the OCI CLI

1. Get the policy OCID.

   ```
   $ oci iam policy list --compartment-id compartment_OCID
   ```
2. (Optional) To change or add policy statements, construct an argument for the `--statements` option.

   The value of the `--statements` option argument is an array of policy statements in JSON format. This argument can be provided as a string on the command line or in a file. For information about how to write policy statements, see Writing Policy Statements.

   The argument that you provide for the `--statements` option replaces the existing statements in the policy. Be sure to include any statements that you want to keep from the existing policy. Use the `policy get` command to view and copy current policy statements.

   If you do not specify the `--force` option, the system will display the existing statements in the policy and request that you confirm that you want to replace them.
3. (Optional) Construct arguments for defined or free-form tags for this policy as described in Adding Tags at Resource Creation.
4. Run the policy update command.

   Syntax:

   ```
   oci iam policy update --policy-id policy_OCID [ --description desc ] \
   [ --defined-tags tags ] [ --freeform-tags tags ] \
   [ --statements policy_statements --version-date "" ]
   ```

If you specify `--statements`, then you must include `--version-date ""`.

This command returns the same output as the `policy get` command.

Deleting a Policy

Using the Compute Web UI

1. In the navigation menu, click Identity, and then click Policies.
2. If the policy that you want to delete is not listed, select the correct compartment from the Compartment drop-down menu above the policies list.
3. For the policy that you want to delete, click the Actions menu, and click the Delete option.
4. In the confirmation dialog, click the Delete button.

Using the OCI CLI

1. Get the policy OCID.

   ```
   $ oci iam policy list --compartment-id compartment_OCID
   ```

2. Run the policy delete command.

   Syntax:

   ```
   oci iam policy delete --policy-id policy_OCID
   ```

   This command returns the same output as the `policy get` command.

3
Resource Tag Management

Oracle Private Cloud Appliance Tagging enables you to add metadata to resources by applying key/value pairs called defined tags or free-form tags. You can create tag defaults on compartments, which are tags that are automatically applied to all newly created resources in the tagged compartment. Uses for tags include:

- Applying access policies to resources. For example, you can change ownership of a resource to a different product group by changing a tag value on the resource rather than changing the resource access policy directly. See Using Defined Tags in Conditions).
- Filtering resource lists in the Compute Web UI.

For conceptual information, see the Tagging Overview and "How Policies Work" in the Identity and Access Management Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

Creating and Managing Tag Namespaces

Tag namespaces enable you to create collections of related tags. After you create a tag namespace, create tag key definitions within that tag namespace. See Creating and Managing Tag Key Definitions. Tag namespaces with tag key definitions must exist in the tenancy before users can apply a defined tag to a resource.

Creating a Tag Namespace

A tenancy can have at most 100 tag namespaces.

Using the Compute Web UI

1. In the navigation menu, click Governance, and then click Tag Namespaces.
2. Above the list of tag namespaces, click the Create Namespace Definitions button.
3. In the Create Namespace Definition window, enter the following information:
   - *Create in Compartment*: The compartment in which you want to create the namespace definition.
   - *Namespace Definition Name*: A name for this tag namespace. Tag namespace names have the following characteristics:
     - Must be unique within the tenancy.
     - Are case insensitive.
     - Cannot be changed later.
     - Can be no more than 100 characters.
     - Cannot contain period (.) or space characters.
   - *Description*: A description for this set of tags. This description can be no more than 256 characters.
   - *Tagging*: (Optional) Add defined or free-form tags for this tag namespace as described in Adding Tags at Resource Creation. Tags can also be applied later.

4. Click Create Namespace Definition.

   The details page for the new tag namespace definition is displayed.

Using the OCI CLI

1. Get the OCID of the compartment where you want to create the tag namespace.

   ```
   $ oci iam compartment list --compartment-id-in-subtree true
   ```

2. Run the tag namespace create command.

   Syntax:

   ```
   oci iam tag-namespace create --compartment-id compartment_OCID --name
   tag_namespace_name \
   --description "text"
   ```

   Use the following command to verify that the name you want to use is unique in the tenancy:

   ```
   $ oci iam tag-namespace list --compartment-id tenancy_OCID
   ```

   You can tag the new tag namespace during creation by adding the options described in Adding Tags at Resource Creation.

   Example:

   ```
   $ oci iam tag-namespace create --compartment-id ocid1.compartment.unique_ID --name
   Products \
   --description "Identify resources used in product development."
   ```

   This command returns the same output as the `tag-namespace get` command.

Updating a Tag Namespace

You can modify the description of a tag namespace and add or modify tags on the tag namespace.

Using the Compute Web UI

1. In the navigation menu, click Governance, and then click Tag Namespaces.
2. If the tag namespace that you want to modify is not listed, use the Compartment drop-down menu above the tag namespaces list to select the correct compartment.
3. For the namespace that you want to modify, click the Actions menu, and click the Edit option.

   The Edit dialog is displayed.

4. Update the tag namespace.

   You can modify the description of a tag namespace and add or modify tags on the tag namespace.

5. Click Update Tag Namespace.

Using the OCI CLI

1. Get the tag namespace OCID.

   ```
   $ oci iam tag-namespace list --compartment-id compartment_OCID
   ```

2. Run the tag namespace update command.

Syntax:

```
oci iam tag-namespace update --tag-namespace-id tag_namespace_OCID --description
"text"
```

To add or modify a tag on the tag namespace, add the options described in Applying Tags to an Existing Resource.

Example:

```
$ oci iam tag-namespace update --tag-namespace-id ocid1.tagnamespace.unique_ID \
--description "Identify resources used to develop different products."
```

This command returns the same output as the `tag-namespace get` command.

Retiring a Tag Namespace

When you retire a tag namespace, all tag key definitions and tags in that tag namespace are retired, and you cannot create new tag key definitions in that tag namespace. Retired tags cannot be applied to resources. However, retired tags remain applied to any resources where they were already applied and can still be used in operations such as listing, sorting, and filtering.

Using the Compute Web UI

1. In the navigation menu, click Governance, and then click Tag Namespaces.
2. If the tag namespace that you want to retire is not listed, use the Compartment drop-down menu above the tag namespaces list to select the correct compartment.
3. For the tag namespace that you want to retire, click the Actions menu, and click the Retire option.
4. At the Retire Tag Namespace confirmation prompt, click Confirm.

   The state of the tag namespace changes to Inactive. On the details page for the tag namespace, the tag key definitions are also in state Inactive.

Using the OCI CLI

1. Get the tag namespace OCID.

   ```
   $ oci iam tag-namespace list --compartment-id compartment_OCID
   ```

2. Run the tag namespace retire command.

   Syntax:

   ```
   oci iam tag-namespace retire --tag-namespace-id tag_namespace_OCID
   ```

   This command returns the same output as the `tag-namespace get` command.

Reactivating a Tag Namespace

You can reactivate a tag namespace that is retired. When you reactivate a tag namespace, you can create new tag key definitions in that tag namespace.

When you reactivate a retired tag namespace, the tag key definitions and tags are not reactivated. To use tag key definitions that were retired with the namespace, you must explicitly reactivate each tag key definition.

Using the Compute Web UI

1. In the navigation menu, click Governance, and then click Tag Namespaces.
2. If the tag namespace that you want to reactivate is not listed, use the Compartment drop-down menu above the tag namespaces list to select the correct compartment.
3. For the tag namespace that you want to reactivate, click the Actions menu, and click the Reactivate option.
4. At the Reactivate Tag Namespace confirmation prompt, click Confirm.

   The state of the tag namespace changes to Active. On the details page for the tag namespace, the tag key definitions are still shown as Inactive.

Using the OCI CLI

1. Get the tag namespace OCID.

   ```
   $ oci iam tag-namespace list --compartment-id compartment_OCID
   ```

2. Run the tag namespace reactivate command.

   Syntax:

   ```
   oci iam tag-namespace reactivate --tag-namespace-id tag_namespace_OCID
   ```

   This command returns the same output as the `tag-namespace get` command.

Moving a Tag Namespace to a Different Compartment

You can move an active or retired tag namespace and its tag key definitions to a different compartment within the same tenancy.

To move a tag namespace, you must be granted `manage tag-namespaces` access in both compartments.

To move a tag namespace, you must use the OCI CLI.

Using the OCI CLI

1. Get the following information:
   - The OCID of the compartment where you want to move the tag namespace.

     ```
     $ oci iam compartment list --compartment-id-in-subtree true
     ```

   - The tag namespace OCID.

     ```
     $ oci iam tag-namespace list --compartment-id compartment_OCID
     ```

2. Run the tag namespace move command.

   Syntax:

   ```
   oci iam tag-namespace change-compartment --compartment-id
   destination_compartment_OCID \
   --tag-namespace-id tag_namespace_OCID
   ```

   Use the `tag-namespace get` command to verify the new `compartment-id`.

Deleting a Tag Namespace

A tag namespace must be retired before it can be deleted. See Retiring a Tag Namespace.

To delete a tag namespace, all tag key definitions in that namespace must be deleted. See Deleting a Tag Key Definition.

Using the Compute Web UI

1. In the navigation menu, click Governance, and then click Tag Namespaces.
2. If the tag namespace that you want to delete is not listed, use the Compartment drop-down menu above the tag namespaces list to select the correct compartment.
3. Ensure that the tag namespace that you want to delete is retired (in state Inactive).
4. Click the name of the tag namespace that you want to delete.
5. In the Resources box on the tag namespace details page, click Tag Key Definitions.
6. Ensure that all tag key definitions are deleted.
7. On the Controls menu at the top of the details page, click the Delete option.
8. At the Delete Tag Namespace confirmation prompt, click Confirm.

   The tag namespace is removed from the Tag Namespace list, and tags in that tag namespace are removed from resources.

Using the OCI CLI

1. Get the tag namespace OCID.

   ```
   $ oci iam tag-namespace list --compartment-id compartment_OCID
   ```

2. Ensure that the tag namespace that you want to delete is retired (in state Inactive).

   ```
   $ oci iam tag-namespace get --tag-namespace-id tag_namespace_OCID
   ```

3. Ensure that all tag key definitions for this tag namespace are deleted.

   ```
   $ oci iam tag list --tag-namespace-id tag_namespace_OCID
   ```

4. Run the tag namespace delete command.

   Syntax:

   ```
   oci iam tag-namespace delete --tag-namespace-id tag_namespace_OCID
   ```

   Example:

   ```
   $ oci iam tag-namespace delete --tag-namespace-id ocid1.tagnamespace.unique_ID
   Are you sure you want to delete this resource? [y/N]: y
   {
     "opc-work-request-id": "ocid1.workrequest.unique_ID"
   }
   ```

   Use the following command to check the status of the tag namespace delete:

   ```
   $ oci iam tagging-work-request get --work-request-id ocid1.workrequest.unique_ID
   ```

   To delete a tag namespace without confirmation, use the `--force` option.

Creating and Managing Tag Key Definitions

A tag namespace contains tag key definitions. Instances of tag key definitions that are applied to resources are called *defined tags*.

A tag key definition includes a tag key name and tag value type. A tag key definition might include a value, depending on the tag value type. The tag value type specifies whether the tag user enters a value or selects a predefined value when applying a tag to a resource.

For more information about tag key definitions, including required permissions, see the Tagging Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

Creating a Tag Key Definition

Create a tag key definition within a tag namespace. A tag namespace can have at most 100 tag key definitions.

Using the Compute Web UI

1. In the navigation menu, click Governance, and then click Tag Namespaces.
2. If the tag namespace where you want to add the tag key definition is not listed, use the Compartment drop-down menu above the tag namespaces list to select the correct compartment.
3. Click the name of the tag namespace where you want to add a tag key definition.
4. In the Resources box on the tag namespace details page, click Tag Key Definitions.
5. In the Tag Key Definitions area, click the Create Tag Key Definition button.
6. In the Create Tag Key Definition window, enter the following information:
   - *Name*: The key name. Tag key names have the following characteristics:
     - Must be unique within the namespace.

       The same tag key name can be used in different tag namespaces.
     - Are case insensitive.
     - Cannot be changed later.
     - Can be no more than 100 characters.
     - Cannot contain period (.) or space characters.
   - *Description*: A description for the tag key definition. This description can be no more than 256 characters.
7. Select the Tag Value Type.
   - *Static Value*: Not populated. The user must enter a value when the tag is applied to a resource. This is the default selection.
   - *A List of Values*: A predefined list of values. Enter at least one value. The user must select one of these predefined values to apply to a resource.

     Separate multiple values with new lines. Duplicate values and blank lines are invalid. Values are case sensitive and can be no more than 256 characters.

     A value can include one of the following variables:

     `${iam.principal.name}`

     The name of the user that tagged the resource.

     `${iam.principal.type}`

     The type of principal that tagged the resource. One of: root user, IAM user, or Instance principal.

${oci.datetime}

The date and time that the tag was created.

8. Click Create Tag Key Definition.

   The details page for the new tag key definition is displayed.

Using the OCI CLI

1. Get the tag namespace OCID.

   ```
   $ oci iam tag-namespace list --compartment-id compartment_OCID
   ```

2. Run the tag key definition create command.

   The following command creates a tag key definition in which the user must enter a tag value when the tag is applied to a resource. Omitting the `--validator` option is comparable to selecting the *Static Value* value type in the Compute Web UI.

   Syntax:

   ```
   oci iam tag create --tag-namespace-id tag_namespace_OCID --name text --description
   "text"
   ```

   Example:

   ```
   $ oci iam tag create --tag-namespace-id ocid1.tagnamespace.unique_ID --name
   VolumeType \
   --description "Identify volumes by type"
   ```

   The following command creates a tag key definition with a list of values from which the user must select when applying the tag.

   Syntax:

   ```
   oci iam tag create --tag-namespace-id tag_namespace_OCID --name text --description
   "text" \
   --validator values
   ```

   The value of the `--validator` option argument is a JSON definition of the tag values. This JSON definition can be provided as a string on the command line or in a file.

   You can generate a template of the correct JSON to provide by using the `--generate-param-json-input` option with the base command that you will use to tag the resource. The argument for the `--generate-param-json-input` option is the name of the option that you will use to specify the tag values (`--validator`) without the option indicator (`--`), as shown in the following example:

   ```
   $ oci iam tag create --generate-param-json-input validator > volume_types.json
   ```

   The following is the content of the output `volume_types.json` file:

   ```
   {
     "validatorType": "ENUM",
     "values": [
       "string",
       "string"
     ]
   }
   ```

   Edit this template to provide new tag values. The value of `values` is a single string or an array of strings. The `validator-type` must be `ENUM`. Specify the result to the `--validator` option in the final command.

In the following example, the tag values are provided in an inline JSON string.

```
$ oci iam tag create --tag-namespace-id ocid1.tagnamespace.unique_ID --name
VolumeType \
--description "Identify volumes by type" \
--validator '{"validator-type": "ENUM", "values": ["typeA","typeB","typeC"]}'
```

The following example includes a variable value:

```
$ oci iam tag create --tag-namespace-id ocid1.tagnamespace.unique_ID --name Product-
XYZ \
--description "Identify resources assigned to XYZ development." \
--validator '{"validator-type": "ENUM", "values": ["Assigned by: $
{iam.principal.name}"]}'
```

In the following example, the tag values are provided in a JSON file. Use the `file://` syntax to specify a file as the option argument.

```
$ oci iam tag create  --tag-namespace-id ocid1.tagnamespace.unique_ID --name
VolumeType \
--validator file://volume_types.json
```

This command returns the same output as the `tag get` command.

Updating a Tag Key Definition

You can update the description of a tag key definition and change the tag value type and value. You cannot update a tag key definition that is retired.

If the value of the tag key definition that you are updating is a predefined list, you cannot remove or change any value that is the value of a tag default. To remove or change a tag key definition value that is the value of a tag default, first update the tag default to use a different value. See Configuring Tag Defaults.

Using the Compute Web UI

1. In the navigation menu, click Governance, and then click Tag Namespaces.
2. If the tag namespace where you want to update a tag key definition is not listed, use the Compartment drop-down menu above the tag namespaces list to select the correct compartment.
3. Click the name of the tag namespace where you want to update a tag key definition.
4. In the Resources box on the tag namespace details page, click Tag Key Definitions.
5. For the tag key definition that you want to update, click the Actions menu, and click the Edit option.
6. In the Edit Tag Key Definition dialog, you can modify the description or change the tag value type. If you choose A List of Values for the type, you must add at least one value in the Value box. Separate multiple values with new lines. Duplicate values and blank lines are invalid. Values are case sensitive and can be no more than 256 characters.
7. Click Save Changes.

Using the OCI CLI

1. Get the following information:
   - The tag namespace OCID.

     ```
     $ oci iam tag-namespace list --compartment-id compartment_OCID
     ```

- The name of the tag key.

  ```
  $ oci iam tag list --tag-namespace-id tag_namespace_OCID
  ```

2. Run the tag key definition update command.

   The following command updates only the tag key description. Values settings remain the same.

   Syntax:

   ```
   oci iam tag update --tag-namespace-id tag_namespace_OCID --tag-name text --
   description "text"
   ```

   Example:

   ```
   $ oci iam tag update --tag-namespace-id ocid1.tagnamespace.unique_ID --tag-name
   VolumeType \
   --description "Identify the type of the volume."
   ```

   The following command updates only the values.

   ```
   $ oci iam tag update --tag-namespace-id ocid1.tagnamespace.unique_ID --tag-name
   VolumeType \
   --validator file://volume_types.json
   WARNING: Updates to freeform-tags and defined-tags and validator will replace any
   existing
   values.
   Are you sure you want to continue? [y/N]: y
   ```

   See Creating a Tag Key Definition for the content of the `volume_types.json` file. Values that you provide to the `--validator` option replace any existing values. To add values or to change only some of the values, provide the complete list in this update.

   The following command updates the value type to allow the user to enter a value rather than select from a predefined list of values.

   ```
   $ oci iam tag update --tag-namespace-id ocid1.tagnamespace.unique_ID --tag-name
   VolumeType \
   --validator '{}'
   ```

   This command returns the same output as the `tag get` command.

Retiring a Tag Key Definition

When you retire a tag key definition, you cannot apply tags that are based on this tag key definition to resources. Existing tag defaults that are based on this tag key definition will not be automatically applied to newly created resources. However, the tag is not removed from resources where it was already applied. The tag still exists as metadata on those resources and you can still use the retired tag in operations such as listing, sorting, and reporting.

Using the Compute Web UI

1. In the navigation menu, click Governance, and then click Tag Namespaces.
2. If the tag namespace where you want to retire a tag key definition is not listed, use the Compartment drop-down menu above the tag namespaces list to select the correct compartment.
3. Click the name of the tag namespace where you want to retire a tag key definition.
4. In the Resources box on the tag namespace details page, click Tag Key Definitions.

5. For the tag key definition that you want to retire, click the Actions menu, and click the Retire option.
6. At the Retire Tag Key Definition confirmation prompt, click Confirm.

Using the OCI CLI

1. Get the following information:
   - The tag namespace OCID.

     ```
     $ oci iam tag-namespace list --compartment-id compartment_OCID
     ```

   - The name of the tag key.

     ```
     $ oci iam tag list --tag-namespace-id tag_namespace_OCID
     ```

2. Run the tag key definition retire command.

   Syntax:

   ```
   oci iam tag retire --tag-name text --tag-namespace-id tag_namespace_OCID
   ```

Reactivating a Tag Key Definition

When you reactivate a tag key definition, it is again available for you to apply to resources. You cannot reactivate a tag key definition if the parent tag namespace is retired.

Using the Compute Web UI

1. In the navigation menu, click Governance, and then click Tag Namespaces.
2. If the tag namespace where you want to reactivate a tag key definition is not listed, use the Compartment drop-down menu above the tag namespaces list to select the correct compartment.
3. Click the name of the tag namespace where you want to reactivate a tag key definition.
4. In the Resources box on the tag namespace details page, click Tag Key Definitions.
5. For the tag key definition that you want to reactivate, click the Actions menu, and click the Reactivate option.
6. At the Reactivate Tag Namespace confirmation prompt, click Confirm.

Using the OCI CLI

1. Get the following information:
   - The tag namespace OCID.

     ```
     $ oci iam tag-namespace list --compartment-id compartment_OCID
     ```

   - The name of the tag key.

     ```
     $ oci iam tag list --tag-namespace-id tag_namespace_OCID
     ```

2. Run the tag key definition reactivate command.

   Syntax:

   ```
   oci iam tag reactivate --tag-name text --tag-namespace-id tag_namespace_OCID
   ```

Deleting a Tag Key Definition

A tag key definition must be retired before it can be deleted. See Retiring a Tag Key Definition.

When you delete a tag key definition, tags that are based on this tag key definition are removed from all resources. Tag defaults that are based on this tag key definition are not removed from compartments.

Using the Compute Web UI

1. In the navigation menu, click Governance, and then click Tag Namespaces.
2. If the tag namespace where you want to delete a tag key definition is not listed, use the Compartment drop-down menu above the tag namespaces list to select the correct compartment.
3. Click the name of the tag namespace where you want to delete a tag key definition.
4. In the Resources box on the tag namespace details page, click Tag Key Definitions.
5. Ensure that the tag key definition that you want to delete is retired (in state Inactive).
6. For the tag key definition that you want to delete, click the Actions menu, and click the Delete option.

   You cannot restore a deleted tag key definition.
7. At the Delete Tag Key Definition prompt, click Confirm.

   The tag key definition status changes to Deleting, and all tags that are based on this tag key definition are removed from resources.

   When the tag removal process is finished, the tag key definition status changes to Deleted. You can create a new tag key definition with the same name as the deleted tag key definition.

Using the OCI CLI

1. Get the following information:
   - The tag namespace OCID.

     ```
     $ oci iam tag-namespace list --compartment-id compartment_OCID
     ```
   - The name of the tag key.

     ```
     $ oci iam tag list --tag-namespace-id tag_namespace_OCID
     ```
2. Ensure that the tag key definition that you want to delete is retired (in state Inactive).

   ```
   $ oci iam tag get --tag-namespace-id tag_namespace_OCID
   ```
3. Run the tag key definition delete command.

   Syntax:

   ```
   oci iam tag delete --tag-name text --tag-namespace-id tag_namespace_OCID
   ```

   Example:

   ```
   $ oci iam tag delete --tag-name volume-types --tag-namespace-id
   ocid1.tagnamespace.unique_ID
   Are you sure you want to delete this resource? [y/N]: y
   {
     "opc-work-request-id": "ocid1.workrequest.unique_ID"
   }
   ```

   Use the following command to check the status of the tag delete:

   ```
   $ oci iam tagging-work-request get --work-request-id ocid1.workrequest.unique_ID
   ```

   To delete a tag key definition without confirmation, use the `--force` option.

Creating OraclePCA Tags

Oracle Private Cloud Appliance uses the OraclePCA tag namespace to set attributes that are not available as OCI CLI options or API attributes. These attributes are documented under the appropriate resource, such as block storage or file systems.

When you use the OCI CLI or API, you can specify the OraclePCA tag namespace, tag key, and values for the attributes that you want to set. You do not need to first create the OraclePCA tag namespace and tag keys.

To use the Compute Web UI to set these attributes, you must first create the OraclePCA tag namespace, tag keys, and value choices.

> **Caution:**
>
> Do not delete these tag keys. Do not create this tag namespace and these keys unless you need to use the Compute Web UI to create clusters. If you create this tag namespace and these keys, create them exactly as shown here, do not modify them, and do not delete them.

The following sections describe how to create the OraclePCA tag namespace and tag key definitions.

Creating the OraclePCA Tag Namespace

1. In the navigation menu, click Governance, and then click Tag Namespaces.
2. If OraclePCA is not shown in the Tag Namespaces list, click the Create Namespace Definitions button.
3. In the Create Namespace Definition dialog, scroll down to the Tagging section and click in the Tag Namespace field.
   - If the OraclePCA tag namespace is not listed, continue with Step 4 of this procedure.
   - If the tag you need is already available, click the Cancel button in the Create Namespace Definition dialog.
4. In the Create Namespace Definition dialog, enter the following information:
   - *Create in Compartment*: The compartment in which you want to create the OraclePCA tag definitions.
   - *Namespace Definition Name*: Enter "OraclePCA".
   - *Description*: For example, "Support resource attributes that are only available on Private Cloud Appliance."
5. Click Create Namespace Definition.

   The details page for the new OraclePCA tag namespace definition is displayed.

Creating the OraclePCA Tag Key Definitions

1. Navigate to the OraclePCA tag namespace details page, scroll down to the Resources box, and click Tag Key Definitions.
2. If the tag key you need is not listed, click the Create Tag Key Definition button.

3. In the Create Tag Key Definition dialog, enter the following information for the tag key that you are creating.

   Block Volume Synchronous Write Bias Tag Key Definition

   - *Name*: Enter "logBias".
   - *Description*: For example, "Control the use of the write cache flash devices for a share or LUN."
   - *Tag Value Type*: Select "A List of Values".
   - *Values*: Enter "LATENCY", enter a newline, and enter "THROUGHPUT".

   Block Volume Secondary Cache Tag Key Definition

   - *Name*: Enter "secondaryCache".
   - *Description*: For example, "Control the use of the read cache flash devices for a share or LUN."
   - *Tag Value Type*: Select "A List of Values".
   - *Values*: Enter "ALL", newline, "METADATA", newline, and "NONE".

   File System Quota

   - *Name*: Enter "quota".
   - *Description*: For example, "The quota value in gigabytes includes the data in the file system and all snapshots created under the file system."
   - *Tag Value Type*: Select "Static Value".

   File System Database Record Size

   - *Name*: Enter "databaseRecordSize".
   - *Description*: For example, "The size of a database record in bytes."
   - *Tag Value Type*: Select "A List of Values".
   - *Values*: Enter the following values, each on a separate line: 512, 1024, 2048, 4096, 8192, 16384, 32768, 65536, 131072, 262144, 524288, 1048576.

   File System Backing Store Pool

   - *Name*: Enter "poolName".
   - *Description*: For example, "Whether to use the default pool of the attached ZFS Storage Appliance, or use a high performance pool as the backing store pool."
   - *Tag Value Type*: Select "A List of Values".
   - *Values*: Enter "PCA_POOL" and "PCA_POOL_HIGH", each on a separate line.

   Network Configuration for SR-IOV

   - *Name*: Enter "networkType".
   - *Description*: For example, "DRG and VCN network configuration required for SR-IOV."
   - *Tag Value Type*: Select "Static Value".

4. Click Create Tag Key Definition.

   The details page for the new tag key definition is displayed.

Configuring Tag Defaults

A tag default is a defined tag that is automatically applied to resources that are created in the specified compartment.

Tag defaults have the following characteristics:

- The tag default is applied to all new resources that are created in that compartment, including in child compartments.
- The tag default is not applied to resources that already existed before the tag default was created.
- Tag defaults cannot be changed by creating or editing resources. With permission to use the tag namespace, you can change the value of the tag when you create or modify a resource. To change the tag default that will be applied to all new resources in the compartment, you must update the tag default on the compartment.
- If you change the default value of the tag default, existing occurrences of that tag default on resources are not updated.
- If you change a value of a tag key definition whose value is a defined list, or if you delete a tag key definition, existing occurrences of a tag default that is based on that tag key definition are not updated. Tag default values must be separately updated.

See Creating and Managing Tag Key Definitions for information about the effect on tag defaults of retiring or deleting tag key definitions. For more information about tag defaults, see "Tag Defaults" in the Tagging Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

Creating a Tag Default

To create a tag default, specify a compartment, a tag key definition, and a value. If the value of the selected tag key definition is *Static Value*, then you can select *User-Defined Value* for the value of the tag default.

A compartment can have at most five tag defaults.

Using the Compute Web UI

1. In the navigation menu, click Identity, and then click Compartments.
2. If the compartment where you want to add a tag default is not listed, navigate to the correct compartment.

   Click the name of the top-level parent compartment and, on the compartment details page, scroll to the Child Compartments box. If necessary, click on the name of another compartment to view those child compartments.
3. Click the name of the compartment to which you want to add a tag default.
4. In the Resources section on the compartment details page, click Tag Defaults.

   Ensure that no more than four tag defaults already exist in this compartment.
5. Click the Create Tag Default button.
6. In the Tag Default dialog, select the Tag Namespace and the Tag Key.
7. For Required Tag Value Options, choose one of the following value types:
   - *Default Value*: Enter the value for this tag default. If the selected tag key has a defined list of values, then this *Default Value* must be a member of that list.

- *User-Defined Value*: Users are required to enter the value when a resource is created. Selecting *User-Defined Value* is invalid if the selected tag key definition has a predefined list of values.

8. Click Submit.

   The new tag default is displayed on the compartment details page.

Using the OCI CLI

1. Get the following information:
   - The OCID of the compartment on which you want to create the tag default.

     ```
     $ oci iam compartment list --compartment-id-in-subtree true
     ```

   - The OCID of the tag key definition.

     ```
     $ oci iam tag list --tag-namespace-id tag_namespace_OCID
     ```

2. Ensure that no more than four tag defaults already exist in this compartment.

   ```
   oci iam tag-default list --compartment-id compartment_OCID
   ```

3. Run the tag default create command.

   Syntax:

   ```
   oci iam tag-default create --compartment-id compartment_OCID \
   --tag-definition-id tag_definition_OCID --value text
   ```

   Example:

   ```
   $ oci iam tag-default create --compartment-id ocid1.compartment.unique_ID \
   --tag-definition-id ocid1.tag.unique_ID --value 789
   ```

   Depending on your shell, you might need to escape the dollar symbol to specify a variable value:

   ```
   $ oci iam tag-default create --compartment-id ocid1.compartment.unique_ID \
   --tag-definition-id ocid1.tag.unique_ID --value "Assigned by: \${iam.principal.name}"
   ```

   This command returns the same output as the `tag-default get` command.

Updating the Value of a Tag Default

Using the Compute Web UI

1. In the navigation menu, click Identity, and then click Compartments.
2. If the compartment where you want to update a tag default is not listed, navigate to the correct compartment.

   Click the name of the top-level parent compartment and, on the compartment details page, scroll to the Child Compartments box. If necessary, click on the name of another compartment to view those child compartments.
3. Click the name of the compartment that has the tag default whose value you want to change.
4. In the Resources section of the compartment details page, click Tag Defaults.
5. For the tag default that you want to change, click the Actions menu, and click the Edit option.
6. In the Tag Defaults dialog, specify the type of value you want the tag default to have:

- *Default Value*: Enter the value for this tag default. If the selected tag key has a defined list of values, then this *Default Value* must be a member of that list.
- *User-Defined Value*: Users are required to enter the value when a resource is created. Selecting *User-Defined Value* is invalid if the selected tag key definition has a predefined list of values.

7. Click Submit.

   The updated tag default is displayed on the compartment's details page.

Using the OCI CLI

1. Get the OCID of the tag default that you want to modify.

   ```
   $ oci iam tag-default list --compartment-id compartment_OCID
   ```

2. Run the tag default update command.

   Syntax:

   ```
   oci iam tag-default update --tag-default-id tag_default_OCID --value text
   ```

   This command returns the same output as the `tag-default get` command.

Deleting a Tag Default

When you delete a tag default from a compartment, existing occurrences of the tag are not removed from resources.

Using the Compute Web UI

1. In the navigation menu, click Identity, and then click Compartments.
2. If the compartment where you want to delete a tag default is not listed, navigate to the correct compartment.

   Click the name of the top-level parent compartment and, on the compartment details page, scroll to the Child Compartments box. If necessary, click on the name of another compartment to view those child compartments.
3. Click the name of the compartment that has the tag default that you want to delete.
4. In the Resources section of the compartment details page, click Tag Defaults.
5. For the tag default that you want to delete, click the Actions menu, and click the Delete option.
6. At the Delete Default Tag confirmation prompt, click Confirm.

   On the compartment details page, the state of the tag default is Deleting.

Using the OCI CLI

1. Get the OCID of the tag default that you want to delete.

   ```
   $ oci iam tag-default list --compartment-id compartment_OCID
   ```

2. Run the tag default delete command.

   Syntax:

   ```
   oci iam tag-default delete --tag-default-id tag_default_OCID
   ```

   Example:

```
$ oci iam tag-default delete --tag-default-id ocid1.tag-default.unique_ID
Are you sure you want to delete this resource? [y/N]: y
```

To delete a tag default without confirmation, use the `--force` option.

Working with Resource Tags

Tagging resources enables you to identify characteristics of resources and apply the same policies to a group of resources. For more information about policies, see Using Defined Tags in Conditions.

Note the following requirements:

- To apply, modify, or delete a tag on a resource, you must have permission to update the resource.
- To apply, modify, or delete a defined tag on a resource, you must also have `use` access on the tag namespace.
- A resource can have at most 64 defined tags and ten free-form tags.

Adding Tags at Resource Creation

Any tag defaults that are defined on a compartment are automatically added to all resources that are created in that compartment, or any child compartment of that compartment, after the tag default was defined. A tag default might require you to enter a value for the tag in order to create the resource. See Configuring Tag Defaults.

Using the Compute Web UI

1. In the Create dialog for the resource, scroll to the Tagging section.
2. Select the Tag Namespace or select None (apply a free-form tag).
   - If you selected a Tag Namespace, then select the Tag Key, and enter a value or select a value from the list.
   - If you selected None (apply a free-form tag), then enter a Tag Key and enter a value.
3. To apply another tag, click the Additional Tag button.

   You cannot specify more than one tag with the same tag namespace and the same tag key for a defined tag, or more than one tag with the same tag key for a free-form tag.
4. To review the tags on the resource, to go the details page for the new resource.

   On the resource details page, click the Tags tab to display the tags that are applied to this resource.

Using the OCI CLI

To add a tag to a resource when you create the resource, use the resource `create` or `launch` command.

1. Get the information for each tag that you want to add to the resource.
   - Get the namespace, key, and value for each defined tag that you want to add to the resource.

     Construct an argument for the `--defined-tags` option. Specify each tag namespace and tag key pair only one time.
   - Get the key and value for each free-form tag that you want to add to the resource.

Construct an argument for the `--freeform-tags` option. Specify each tag key only one time.

The value of the `--defined-tags` option argument and the `--freeform-tags` option argument is a JSON definition of the tags. This JSON definition can be provided as a string on the command line or in a file.

You can generate a template of the correct JSON to provide by using the `--generate-param-json-input` option with the base command that you will use to tag the resource. The argument for the `--generate-param-json-input` option is the name of the option that you will use to specify the tags (`--defined-tags` in this example) without the option indicator (`--`), as shown in the following example:

```
$ oci service resource create \
--generate-param-json-input defined-tags > defined_tags.json
```

The content of the output `defined_tags.json` file is:

```
{
  "tagNamespace1": {
    "tagKey1": "tagValue1",
    "tagKey2": "tagValue2"
  },
  "tagNamespace2": {
    "tagKey1": "tagValue1",
    "tagKey2": "tagValue2"
  }
}
```

If you specify `freeform-tags` instead of `defined-tags` in the preceding command, you get the following output:

```
{
  "tagKey1": "tagValue1",
  "tagKey2": "tagValue2"
}
```

Edit these templates to provide the desired tags. Specify the result in the final command as shown in the following step.

2. Run the resource create or launch command.

   If you want to add one or more defined tags, use the `--defined-tags` option. If you want to add one or more free-form tags, use the `--freeform-tags` option.

   Syntax:

   ```
   oci service resource create --compartment-id compartment_OCID \
   --defined-tags defined_tags_json --freeform-tags freeform_tags_json \
   other_resource_create_options
   ```

   Example:

   In the following example, one or more defined tags is added using a file, and a free-form tag is added using a string argument. Use the `file://` syntax to specify a file as the option argument.

   ```
   $ oci service resource create --compartment-id ocid1.compartment.unique_ID \
   --defined-tags file://defined_tags.json --freeform-tags '{"MyTag":"val-u"}' \
   other_resource_create_options
   ```

   The output of the resource `create` or `launch` command is the same as the output of the resource `get` command. The output shows the defined and free-form tags.

Applying Tags to an Existing Resource

Using the Compute Web UI

1. In the resource Edit dialog, scroll to the Tagging section.

   You can add tags, and you can modify or delete any tags that already exist.

2. To add tags, click the Additional Tag button if necessary, and select the Tag Namespace or select None (apply a free-form tag).

   - If you selected a Tag Namespace, then select a Tag Key, and enter a value or select a value from the list.
   - If you selected None (apply a free-form tag), then enter a Tag Key and enter a value.

   You cannot specify more than one tag with the same tag namespace and the same tag key for a defined tag, or more than one tag with the same tag key for a free-form tag.

3. To modify existing tags, change the selections or enter different values.
4. To delete a tag, click the trash can.
5. When you are finished adding and modifying tags, click Save Changes.
6. To review the tags on the resource, go to the details page for the resource.

   On the resource details page, click the Tags tab to display the list of tags that are applied to this resource.

Using the OCI CLI

To add tags to an existing resource, and modify or delete any tags that already exist, use the resource `update` command.

1. Get the namespace, key, and value information for each tag that you want to add to the resource.
2. Create a JSON definition of the tags that you want to apply. See Adding Tags at Resource Creation for information about how to create correct JSON.

   > **Note:**
   >
   > Any defined tags that already exist on this resource will be replaced by the `--defined-tags` argument. Any free-form tags that already exist on this resource will be replaced by the `--freeform-tags` argument. Be sure to include any existing tags that you want to keep in the new arguments for these options.

   Use the resource `get` command to show the current defined and free-form tags.

3. Run the resource update command.

   If you want to apply one or more defined tags, use the `--defined-tags` option. If you want to apply one or more free-form tags, use the `--freeform-tags` option.

   The output of the resource `update` command is the same as the output of the resource `get` command. The output shows the defined and free-form tags.

Filtering a List of Resources by Tag

Using the Compute Web UI

1. Display a list of resources.
2. Under Filter by Tag(s), click Select Tag(s).
3. Do the following in the Filter by Tag dialog:
   a. Select either Defined Tag or Free-Form Tag.
      - If you select Defined Tag, then select a Tag Namespace and a Tag Key.
      - If you selected Free-Form Tag, then enter a Tag Key.
   b. Optionally enter values in the Select Values (optional) field.
      - Leave the Select Values (optional) field blank. This option returns all resources that are tagged with the selected namespace and key (for defined tags) or returns all resources that are tagged with the specified key (for free-form tags), regardless of the tag value.
      - Enter values in the Select Values (optional) field. This option returns all resources that are tagged with any of the tag value(s) that you enter. Select a value or enter a single value in the text box. To specify multiple values for the same namespace and key, enter Enter or Return, and then enter a new value. Each value is displayed below the text box.
   c. Click Filter by Tag.

      The filter that is currently applied is displayed on the Select Tag(s) button.
4. To filter by multiple tags, click the + on the Select Tag(s) button, and repeat the previous step.
5. To remove a filter, click the filter definition on the Select Tag(s) button.

Using the OCI CLI

1. Use the `list` command for the resource to show each resource in the specified compartment.

   The information for each resource shows the existing defined tags and free-form tags.
2. Use tools for your operating system to filter the list.

4 Networking

Managing VCNs and Subnets

The VCN is the basic networking unit of the Oracle Private Cloud Appliance product. VCNs can be further divided into IP subnets, and individual VCNs can communicate with each other through various types of gateways, each type intended for a particular purpose.

> **Note:**
>
> IPv6 is not currently supported.

Creating a VCN

The VCN is the basic networking unit of the Oracle Private Cloud Appliance product. VCNs can be further divided into IP subnets. VCNs can communicate with each other through various types of gateways, each type intended for a particular purpose.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the Create Virtual Cloud Network button to open the Create Virtual Cloud Network dialog.
3. Enter the following information:
   - **Name:** Enter a descriptive name for the VCN.
   - **Compartment:** Select the compartment in which to create the VCN.
   - **CIDR Block:** Specify which CIDR range can be used within the VCN.
   - **DNS:** If you check the box to use DNS host names in this VCN, then you can either enter a DNS label or leave the field blank to let the system generate a label for you. The first character of the label must be a letter. Only use letters and numbers. Up to 15 characters are allowed.
4. Optionally, add one or more tags to this VCN resource.

   For more information about tagging, see Working with Resource Tags. If you are not sure whether to apply tags, skip this option (you can apply tags later) or ask your administrator.
5. Click the Create Virtual Cloud Network button in the dialog. The details page of the new VCN is displayed.

Using the OCI CLI

1. Gather the information that you need to run the command:

- Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)

2. Enter the `vcn create` command using at least the compartment OCID and CIDR block options.

   If you want to use DNS host names in the VCN, include the DNS label in the create command. It cannot be added later.

   Optionally, set a descriptive name for the VCN.

```
$ oci network vcn create --compartment-id compartment_OCID \
--cidr-blocks '["10.0.0.0/16"]' --dns-label vcn1 --display-name VCN1
{
  "data": {
    "cidr-block": "10.0.0.0/16",
    "cidr-blocks": [
      "10.0.0.0/16"
    ],
    "compartment-id": "ocid1.compartment.unique_ID",
    "default-dhcp-options-id": "ocid1.dhcpoptions.unique_ID",
    "default-route-table-id": "ocid1.routetable.unique_ID",
    "default-security-list-id": "ocid1.security_list.unique_ID",
    "defined-tags": {},
    "display-name": "VCN1",
    "dns-label": "vcn1",
    "freeform-tags": {},
    "id": "ocid1.vcn.unique_ID",
    "ipv6-cidr-block": null,
    "ipv6-private-cidr-block": null,
    "lifecycle-state": "PROVISIONING",
    "time-created": "2022-04-27T04:34:58.722835+00:00",
    "vcn-domain-name": "vcn1.oraclevcn.com"
  },
  "etag": "a555bf2a-0764-4389-8d72-e9a746f63a78"
}
```

Creating a Subnet

VCNs can be divided into subnets. Although it is possible to have an enormous VCN with a thousand IP addresses, it often makes sense from a performance and fault isolation standpoint to create multiple subnets within a VCN. The subnets can still communicate if configured properly.

IP subnet calculation can be a difficult task, especially when figuring out which IP addresses in the range are reserved. The wide range of allowable CIDR block addresses complicates the issue. Free subnet calculation tools available online can help, such as https://www.calculator.net/ip-subnet-calculator.html.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the name of the VCN in which you want to create a new subnet. The VCN details page is displayed.
3. In the Resources section, click Subnets.
4. Click the Create Subnet button at the top of the subnets list to open the Create Subnet dialog.
5. Enter the following information:

- **Name:** Enter a descriptive name for the subnet.
- **Create in Compartment:** Select the compartment where you want to create this subnet.
- **CIDR Block:** Specify which CIDR range can be used within the subnet.

> **Note:**
>
> Choice of CIDR range is an important subnet parameter. The range must be within the VCN CIDR block configured and, most critically, must not overlap with other subnet ranges. These addresses cannot be shared.

- **Route Table (Optional):** Select the route table to associate with this subnet. You might need to change the compartment selection. If you do not select a route table, the VCN default route table is used.
- **Private or Public Subnet:** If you select Private Subnet, instances in this subnet are not allowed to obtain a public IP address.
- **DNS Hostnames (Optional):** Check this box if you want to be able to assign a DNS hostname when you launch an instance in this subnet. If you check the box, enter a DNS label that is unique across the system.
- **DHCP Options (Optional):** Select the set of DHCP options to associate with the subnet. You might need to change the compartment selection. If you do not select a set of options, the VCN default set is used.
- **Security Lists (Optional):** If you want a security list for this subnet, click the +Add Security List button. Select a security list to associate with the subnet. You might need to change the compartment selection. If you want another security list, click the +Add Security List button and select another security list. If you do not select a security list, the VCN default security list is used.

6. Optionally, add one or more defined or free-form tags to this subnet as described in Adding Tags at Resource Creation. Tags can also be applied later.
7. Click the Create Subnet button in the dialog. The details page of the new subnet is displayed.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
   - VCN OCID (`oci network vcn list --compartment-id` ***<compartment_OCID>***)
2. Enter the `subnet create` command using at least the compartment ID, VCN ID and CIDR block options.

   If you want to use DNS host names in the subnet, include the DNS label in the create command. It cannot be added later. This option is available only if you provided a DNS label for the VCN during creation.

   This example also sets a descriptive name for the subnet. No set of DHCP options is specified, so the subnet will use the VCN default set.

```
$ oci network subnet create --compartment-id compartment_OCID \
--vcn-id vcn_OCID --cidr-block 10.0.1.0/24 --dns-label subnet1 \
--display-name NoPublicIP
```

```
{
  "data": {
    "availability-domain": "AD-1",
    "cidr-block": "10.0.1.0/24",
    "compartment-id": "ocid1.compartment.unique_ID",
    "defined-tags": {},
    "dhcp-options-id": "ocid1.dhcpoptions.unique_ID",
    "display-name": "NoPublicIP",
    "dns-label": "subnet1",
    "freeform-tags": {},
    "id": "ocid1.subnet.unique_ID",
    "ipv6-cidr-block": null,
    "ipv6-virtual-router-ip": null,
    "lifecycle-state": "PROVISIONING",
    "prohibit-internet-ingress": null,
    "prohibit-public-ip-on-vnic": true,
    "route-table-id": "ocid1.routetable.unique_ID",
    "security-list-ids": [
      "ocid1.security_list.unique_ID"
    ],
    "subnet-domain-name": "subnet1.vcn1.oraclevcn.com",
    "time-created": "2022-04-27T04:41:54.984856+00:00",
    "vcn-id": "ocid1.vcn.unique_ID",
    "virtual-router-ip": "10.0.1.1",
    "virtual-router-mac": "00:13:97:0e:8f:ff"
  },
  "etag": "30d67d2d-5e11-4b13-9607-1948c52a78f5"
}
```


Editing a Subnet

You can change the name of the subnet, the route tables and security lists used by the subnet, and DHCP options.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the name of the VCN that contains the subnet you want to edit. The VCN details page is displayed.
3. In the Subnets list in the Resources section, locate the subnet that you want to edit. In the Actions menu, click Edit to open the Edit Subnet window.
4. Make the changes to the subnet. The following properties can be edited:
   - **Name:** Change the name of the subnet.
   - **Route Table:** Select a different route table for this subnet. You might need to change the compartment selection.
   - **DHCP Options:** Select a different set of DHCP options for this subnet. You might need to change the compartment selection.
   - **Security Lists:** Select different or additional security lists for this subnet. You might need to change the compartment selection.
5. Optionally, add or delete tags for this subnet.

   For more information about tagging, see Working with Resource Tags. If you are not sure whether to apply tags, skip this option (you can apply tags later) or ask your administrator.
6. Click Save Changes. The subnet properties are updated.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
   - You might also need OCIDs for a route table, DHCP options set, or security lists.
2. Enter the `subnet update` command using the subnet OCID and the parameters you want to change.

   This example changes the DHCP options and route table for the subnet.

```
$ oci network subnet update --subnet-id ocid1.subnet.unique_ID \
--dhcp-options-id ocid1.dhcpoptions.unique_ID \
--route-table-id ocid1.routetable.unique_ID
{
  "data": {
    "availability-domain": "AD-1",
    "cidr-block": "10.0.1.0/24",
    "compartment-id": "ocid1.compartment.unique_ID,
    "defined-tags": {},
    "dhcp-options-id": "ocid1.dhcpoptions.unique_ID",
    "display-name": "NoPublicIP",
    "dns-label": "subnet1",
    "freeform-tags": {},
    "id": "ocid1.subnet.unique_ID",
    "ipv6-cidr-block": null,
    "ipv6-virtual-router-ip": null,
    "lifecycle-state": "AVAILABLE",
    "prohibit-internet-ingress": null,
    "prohibit-public-ip-on-vnic": true,
    "route-table-id": "ocid1.routetable.unique_ID",
    "security-list-ids": [
      "ocid1.securitylist.unique_ID"
    ],
    "subnet-domain-name": "subnet1.vcn1.oraclevcn.com",
    "time-created": "2022-04-27T04:41:54.984856+00:00",
    "vcn-id": "ocid1.vcn.unique_ID",
    "virtual-router-ip": "10.0.1.1",
    "virtual-router-mac": "00:13:97:0e:8f:ff"
  },
  "etag": "30d67d2d-5e11-4b13-9607-1948c52a78f5"
}
```

Deleting a Subnet

A subnet can only be deleted if it is empty. Before deleting a subnet, make sure that all compute instances and other resources have been deleted.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the name of the VCN that contains the subnet you want to delete. The VCN details page is displayed.
3. In the Subnets list in the Resources section, locate the subnet to delete. In the Actions menu, click Delete. Confirm the operation when prompted.

Using the OCI CLI

1. Get the OCID of the subnet you want to delete (`oci network subnet list -c` ***`compartment_OCID`***)
2. Enter the `subnet delete` command.

```
$ oci network subnet delete --subnet-id subnet_OCID
Are you sure you want to delete this resource? [y/N]: y
```

Terminating a VCN

A VCN can only be terminated if it is empty. Before terminating a VCN, make sure that all subnets, route tables, gateways, and other resources have been deleted.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the name of the VCN that you want to terminate. The VCN details page is displayed. Check that the Resources list is empty.
3. Click the Terminate button. Confirm the operation when prompted.

Using the OCI CLI

1. Get the OCID of the VCN you want to delete (`oci network vcn list -c` ***`compartment_OCID`***)
2. Enter the `vcn delete` command.

```
$ oci network vcn delete --vcn-id vcn_OCID
Are you sure you want to delete this resource? [y/N]: y
```

Configuring VCN Rules and Options

VCNs and their subnets have various rules and options associated with them. The main categories are the use of DHCP, route tables, and security. If you do not configure these rules and options explicitly, the system uses default values.

This section describes the parameters that are available for DHCP options, route tables, and security lists.

Working with DHCP Options

When you create a subnet, you can specify the set of DHCP options for the subnet. A set of DHCP options is a resource with an OCID. If you do not specify a set of DHCP options, the default set for the VCN is used.

A subnet can only be assigned one set of DHCP options. You can edit a set of DHCP options, create a new set, and change which set is assigned to a subnet. The assigned DHCP option set applies to all of the instances in that subnet.

For more information, see "DHCP Options" in the Virtual Networking Overview of the *Oracle Private Cloud Appliance Concepts Guide*.

Viewing a VCN's DHCP Options Sets

Every VCN has a default set of DHCP options that is named Default DHCP Options for ***VCN_name***. If you create additional sets, then you can choose which set to assign to a subnet.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the name of the VCN for which you want to list DHCP Options sets. The VCN details page is displayed.
3. Under Resources, click DHCP Options. The list of DHCP options sets is displayed.
4. The DHCP Options sets in the list are not clickable. To see the options that are defined in the set, click the Actions menu for that set and then click Edit.

Using the OCI CLI

1. Gather the information you need to run the command:
   - Compartment OCID (`oci iam compartment list`)
   - VCN OCID (`oci network vcn list --compartment-id` ***`compartment_OCID`***)
2. Run the list command.

   Use both the VCN OCID and the compartment OCID to list all DHCP Options sets that belong to the specified VCN and are in the specified compartment.

   ```
   oci network dhcp-options list --compartment-id ocid1.compartment.unique_ID \
   --vcn-id ocid1.vcn.unique_ID
   ```

   Use only the compartment OCID to list all DHCP Options sets in that compartment. The DHCP Options sets in a compartment could belong to any VCN. DHCP Options sets do not need to be in the same compartment with the VCN.

   ```
   oci network dhcp-options list --compartment-id ocid1.compartment.unique_ID
   ```
3. Use one of the following methods to show just one DHCP Options set.
   - Use the `list` command with the name of the DHCP Options set.

     ```
     oci network dhcp-options list --compartment-id ocid1.compartment.unique_ID \
     --display-name CustomDNSservers
     ```
   - Use the `get` command with the OCID for the DHCP Options set. The DHCP Options set OCID is the value of `id` property in the DHCP Options set `list` command output.

     ```
     oci network dhcp-options get --dhcp-id ocid1.dhcpoptions.unique_ID
     ```

Creating a Set of DHCP Options

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the name of the VCN for which you want to create a set of DHCP options. The VCN details page is displayed.
3. Under Resources, click DHCP Options.
4. Click the Create DHCP Options button.

5. In the Create DHCP Options dialog, enter the following information:
   - **Name**: A descriptive name for the set of options. The name doesn't have to be unique, and you can change it later.
   - **Create in Compartment:** The compartment where you want to create the set of DHCP options.
   - **DNS Type**: If you want instances in the subnet to resolve internet hostnames and hostnames of instances in the VCN, select Internet and VCN Resolver. To use a DNS server of your choice, select Custom Resolver and then enter the IP address of the DNS server. You can enter up to three DNS server IP addresses. For more information, see "Name Resolution" in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*.
   - **Search Domain**: If you want instances in the subnet to append a particular search domain when resolving DNS queries, enter that domain here. Note that the Networking service automatically sets the search domain option in certain situations. For more information, see "DHCP Options" in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*.
   - **Tagging:** For more information about tagging, see Working with Resource Tags. If you are not sure whether to apply tags, skip this option (you can apply tags later) or ask your administrator.
6. Click the Create DHCP Options button in the dialog.

   You can specify this set of options when creating or updating a subnet.

Using the OCI CLI

1. Gather the information you need to run the command:
   - The OCID of the compartment where you want to create this set of DHCP options (`oci iam compartment list`)
   - The OCID of the VCN for this set of DHCP options (`oci network vcn list --compartment-id` ***`compartment_OCID`***)
2. Construct an argument for the `--options` option.

   DHCP options are in JSON format. To see how to format the options, use the following command:

   ```
   oci network dhcp-options create --generate-param-json-input options >
   options_format.json
   ```

   Alternatively, run a `list` or `get` of an existing DHCP Options object and copy the value of the `options` property.

   Put the information for these options in the appropriate places in the format, or replace the information in the options that you copied.

   The value of the `--options` option is either a string between single quotation marks or a file specified as `file://`***`path_to_file`***`.json`.
3. Run the DHCP options create command.

   Syntax:

   ```
   oci network dhcp-options create --compartment-id <compartment_OCID> \
   --vcn-id <vcn_OCID> --options JSON_formatted_values
   ```

   Example:

```
$ oci network dhcp-options create \
--compartment-id ocid1.compartment.unique_ID --vcn-id ocid1.vcn.unique_ID \
--display-name CustomDNSservers --options \
'[{"customDnsServers": ["IP_address"], "serverType":
"CustomDnsServer","type":"DomainNameServer"},{"searchDomainNames":
["name.example.com"],"type":"SearchDomain"}]'
{
  "data": {
    "compartment-id": "ocid1.compartment.unique_ID",
    "defined-tags": {},
    "display-name": "CustomDNSservers",
    "domain-name-type": null,
    "freeform-tags": {},
    "id": "ocid1.dhcpoptions.unique_ID",
    "lifecycle-state": "PROVISIONING",
    "options": null,
    "time-created": "2022-05-04T19:29:16.763027+00:00",
    "vcn-id": "ocid1.vcn.unique_ID"
  },
  "etag": "cf50eb7e-88ff-4e1f-b129-08e2c25b3aa2"
}
```

While the new DHCP Options object is still provisioning, the specified options might not be shown. To confirm the options, use the OCID in the `id` property of the `create` output to run a `get` command:

```
$ oci network dhcp-options get --dhcp-id ocid1.dhcpoptions.unique_ID
{
  "data": {
    "compartment-id": "ocid1.compartment.unique_ID",
    "defined-tags": {},
    "display-name": "CustomDNSservers",
    "domain-name-type": null,
    "freeform-tags": {},
    "id": "ocid1.dhcpoptions.unique_ID",
    "lifecycle-state": "AVAILABLE",
    "options": [
      {
        "custom-dns-servers": [
          "IP_address"
        ],
        "server-type": "CustomDnsServer",
        "type": "DomainNameServer"
      },
      {
        "search-domain-names": [
          "name.example.com"
        ],
        "type": "SearchDomain"
      }
    ],
    "time-created": "2022-05-04T19:29:16.763027+00:00",
    "vcn-id": "ocid1.vcn.unique_ID"
  },
  "etag": "cf50eb7e-88ff-4e1f-b129-08e2c25b3aa2"
}
```

Updating a Set of DHCP Options

To update the DHCP options for the instances in a subnet, do one of the following:

- Update the DHCP Options object that is currently assigned to that subnet as described in this section.
- Update the subnet to assign a different DHCP Options object as described in Editing a Subnet.

For information about how to make the changes take effect in your instances, see "DHCP Options" in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the name of the VCN for the DHCP options that you want to edit. The VCN details page is displayed.
3. Under Resources, click DHCP Options.

   The list of DHCP options sets is displayed.
4. For the set of options that you want to change, click the Actions menu and then click Edit. See the descriptions of Name, DNS Type, and Search Domain in Creating a Set of DHCP Options.
5. Click the Save Changes button in the edit dialog.

Using the OCI CLI

1. Get the OCID of the DHCP Options object that you want to update (`oci network dhcp-options list --compartment-id` ***`compartment_OCID`***)
2. Run the DHCP options update command.

   Syntax:

   ```
   oci network dhcp-options update --dhcp-id dhcp_OCID values_to_update
   ```

   You can update the display name, domain name type, and options. Any options JSON object that you provide replaces the entire set of options. If you want to keep any of the existing options, run the `get` command with this `--dhcp-id` and copy what you want from the output `option` property to your options JSON object.

   Example:

   ```
   $ oci network dhcp-options update --dhcp-id ocid1.dhcpoptions.unique_ID \
   --options file:///home/flast/dhcp_options.json
   ```

   The output from this command is similar to the output from the `create`, `list`, and `get` commands. If you make changes to options and you do not see those changes initially, wait a few seconds and then run the `get` command.

Deleting a Set of DHCP Options

You cannot delete a set of DHCP options that is assigned to any subnet. To unassign the DHCP options set from a subnet, update the subnet to assign a different set of DHCP options. See Editing a Subnet. You cannot delete a VCN's default set of DHCP options.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.

2. Click the name of the VCN for which you want to delete a DHCP Options set. The VCN details page is displayed.
3. Under Resources, click DHCP Options.
4. For the set that you want to delete, click the Actions menu and then click Delete.
5. Confirm when prompted.

Using the OCI CLI

1. Get the OCID of the DHCP Options object that you want to delete (`oci network dhcp-options list --compartment-id` ***`compartment_OCID`***)
2. Run the DHCP options delete command.

```
$ oci network dhcp-options delete --dhcp-id ocid1.dhcpoptions.unique_ID
Are you sure you want to delete this resource? [y/N]: y
```

   To suppress this prompt, use the `--force` option.

Working with Route Tables

When you create a subnet, you specify a route table to associate with the subnet. If you don't, the VCN's default route table is used. You can change route table entries for a subnet at any time, but a subnet can only be assigned one route table at a time. The assigned route table applies to all of the instances in that subnet.

To delete a route table, it must not be associated with a subnet yet. You can't delete a VCN's default route table.

For more information, see "Route Tables" in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

Viewing a VCN's Route Tables

To see which table is assigned to a particular subnet, view the subnet. See the Route Table on the subnet details page, or see the `route-table-id` property in the subnet list or get command output.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the name of the VCN for which you want to view route tables. The VCN details page is displayed.
3. Under Resources, click Route Tables. The list of route tables is displayed.
4. Click the name of the route table to view its route rules.

Using the OCI CLI

1. Gather the information you need to run the command:
   - Compartment OCID (`oci iam compartment list`)
   - VCN OCID (`oci network vcn list --compartment-id` ***`compartment_OCID`***)
2. Run the route table list command.

   Use both the VCN OCID and the compartment OCID to list all route tables that belong to the specified VCN and are in the specified compartment.

```
oci network route-table list --compartment-id ocid1.compartment.unique_ID \
--vcn-id ocid1.vcn.unique_ID
```

Use only the compartment OCID to list all route tables in that compartment. The route tables in a compartment could belong to any VCN. Route tables do not need to be in the same compartment with the VCN.

```
oci network route-table list --compartment-id ocid1.compartment.unique_ID
```

3. Use one of the following methods to show just one route table.
   - Use the `list` command with the name of the route table.

     ```
     oci network security-list list --compartment-id ocid1.compartment.unique_ID \
     --display-name ExtRoute
     ```

   - Use the `get` command with the OCID for the route table. The route table OCID is the value of `id` property in the route table `list` command output.

     ```
     oci network route-table get --rt-id ocid1.routetable.unique_ID
     ```

Creating a Route Table

Route rules are required to send traffic outside the VCN. If you don't need to send traffic outside the VCN, you can use the default route table that was created when the VCN was created. The default route table has no rules.

Each route rule specifies a destination CIDR block and the target (the next hop) for any traffic that matches that CIDR. Before you can create a rule, you must create a target. For descriptions of target types, see "Network Gateways" in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*, and see Overview of Routing for Your VCN. To create a target, use one of the procedures in Configuring VCN Gateways.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the name of the VCN for which you want to create a route table. The VCN details page is displayed.
3. Under Resources, click Route Tables.
4. Click the Create Route Table button.
5. Enter the name and compartment.
   - **Name:** A user-friendly name for the route table. The name doesn't have to be unique, and you can change it later.
   - **Create in Compartment:** The compartment where you want to create the route table. You aren't required to create the route table in the same compartment as the VCN.
6. To add a route table rule, click Add Route Rules, and enter the following information:
   - **Target Type:** Select from the list. Possible targets are:
     - Dynamic Routing Gateway
     - Internet Gateway
     - Local Peering Gateway
     - NAT Gateway
     - Private IP
     - Service Gateway

The target is the OCID of the resource. This applies to the private IP address target also. If a gateway of the type selected is available, a list of choices for that type is presented. If no gateway of the type is available, the message None Available appears.

- **Destination Type:** Choose the destination type: either **CIDR Block** or **Service**
- **CIDR Block:** If the destination type is a CIDR block, enter the destination CIDR block for the traffic. A value of 0.0.0.0/0 means that all non-intra-VCN traffic that isn't already covered by other rules in the route table goes to the target specified in this rule.
- **Service:** If the destination type is a service, select the service from the list, which can be extensive.
- **Target Selection:** This value is the OCID of the Target Type. Click the arrow and select the target, or, if the target is a private IP address, enter the Private IP OCID.
- **Description:** An optional description of the rule.

7. Click the Create Route Table button in the dialog.

   The details page of the new route table is displayed. You can specify this route table when creating or updating a subnet.

Using the OCI CLI

1. Gather the information you need to run the command:
   - The OCID of the compartment where you want to create this route table (`oci iam compartment list`)
   - The OCID of the VCN for this route table (`oci network vcn list --compartment-id` ***`compartment_OCID`***)
2. Construct an argument for the `--route-rules` option.

   Route rules are in JSON format. To see how to format a rule, use the following command:

   ```
   oci network route-table create --generate-param-json-input route-rules >
   route_rule_format.json
   ```

   Alternatively, if a route table with route rules already exists, you can `list` or `get` that route table and copy the value of the `route-rules` property.

   Put the information for this new rule in the appropriate places in the format, or replace the information in the rule that you copied.

   The value of the `--route-rules` option is either a string between single quotation marks or a file specified as `file://`***`path_to_file`***`.json`.
3. Run the route table create command.

   If you don't specify a display name, a name is provided.

   Syntax:

   ```
   oci network route-table create --compartment-id compartment_OCID \
   --vcn-id vcn_OCID --route-rules route_rules_json
   ```

   Example:

   ```
   $ oci network route-table create --compartment-id ocid1.compartment.unique_ID \
   --vcn-id ocid1.vcn.unique_ID --display-name InternetRoute --route-rules \
   '[{"cidrBlock":"0.0.0.0/0","networkEntityId":"ocid1.internetgateway.unique_ID"}]'
   {
     "data": {
   ```

```
    "compartment-id": "ocid1.compartment.unique_ID",
    "defined-tags": {},
    "display-name": "InternetRoute",
    "freeform-tags": {},
    "id": "ocid1.routetable.unique_ID",
    "lifecycle-state": "PROVISIONING",
    "route-rules": [
      {
        "cidr-block": "0.0.0.0/0",
        "description": null,
        "destination": null,
        "destination-type": "CIDR_BLOCK",
        "network-entity-id": "ocid1.internetgateway.unique_ID"
      }
    ],
    "time-created": "2022-04-11T06:00:29.527637+00:00",
    "vcn-id": "ocid1.vcn.unique_ID"
  },
  "etag": "15dcf54f-fa85-40f6-9557-75774e73f1ce"
}
```


While the new route table is still provisioning, the `route-rules` property might be empty. To confirm the options, use the OCID in the `id` property of the `create` output to run a `get` command:

```
oci network route-table get --rt-id ocid1.routetable.unique_ID
```

Updating Rules in a Route Table

You can change the name of a route table and add, edit, or delete rules in a route table.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the name of the VCN for which you want to update the route table. The VCN details page is displayed.
3. Under Resources, click Route Tables.
4. Click the name of the route table that you want to update.
5. To change the name of the route table, click the Edit button, change the name in the dialog, and click the Save Changes button.
6. To create a route rule, click Add Route Rules and enter the information described in Creating a Route Table.
7. To edit an existing rule, click the Actions menu for that rule, and then click **Edit**.
8. To delete a rule, click the Actions menu for that rule, and then click **Delete**.

Using the OCI CLI

1. Get the OCID of the route table to update (`oci network route-table list --compartment-id compartment_OCID`)
2. If you are changing the route rules, create an argument for the `--route-rules` option. See Creating a Route Table. This argument replaces any existing route rules, so be sure to include any rules that you want to keep. Use the following command to view existing rules in this route table:

```
oci network route-table get --rt-id ocid1.routetable.unique_ID
```

3. Run this command.

   Syntax:

   ```
   oci network route-table update --rt-id route_table_OCID --route-rules
   options_to_change
   ```

   You can change the name (`--display-name`) or the rules (`--route-rules`), including the network entity ID.

   > **Note:**
   >
   > The Network Entity ID for a private IP address is the OCID of the private IP address. See the following example.

   Example:

   ```
   oci network route-table update --rt-id ocid1.routetable.unique_ID \
     --route-rules [\{"destination":"10.231.0.0/16","destination-type":"CIDR_BLOCK", \
     "network-entity-id":"ocid1.privateip.unique_ID"}]'
   {
     "data": {
       "compartment-id": "ocid1.compartment.unique_ID",
       "defined-tags": {},
       "display-name": "InternetRoute",
       "freeform-tags": {},
       "id": "ocid1.routetable.unique_ID",
       "lifecycle-state": "AVAILABLE",
       "route-rules": [
         {
           "cidr-block": "10.231.0.0/16",
           "description": "Uses the ExtG8Way",
           "destination": null,
           "destination-type": "CIDR_BLOCK",
           "network-entity-id": "ocid1.privateip.unique_ID"
         }
       ],
       "time-created": "2022-04-11T06:00:29.527637+00:00",
       "vcn-id": "ocid1.vcn.unique_ID"
     },
     "etag": "15dcf54f-fa85-40f6-9557-75774e73f1ce"
   }
   ```

Deleting a Route Table

You cannot delete a route table that is associated with a subnet. You cannot delete a VCN's default route table.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the name of the VCN for which you want to delete a route table. The VCN details page is displayed.
3. Under Resources, click Route Tables.
4. For the route table that you want to delete, click the Actions menu and click Delete.
5. Confirm when prompted.

Using the OCI CLI

1. Get the OCID of the route table to delete (`oci network route-table list --compartment-id` ***`compartment_OCID`***)
2. Run the route table delete command.

   ```
   $ oci network route-table delete --rt-id ocid1.routetable.unique_ID
   Are you sure you want to delete this resource? [y/N]: y
   ```

   To suppress this prompt, use the `--force` option.

Controlling Traffic with Security Lists

Both security lists and network security groups (NSGs) are types of virtual firewalls for your compute instances. Both security lists and NSGs define network security rules that determine which types of traffic are allowed in and out of instances (VNICs).

Security lists provide virtual firewall rules to all the VNICs in a subnet. To provide a set of firewall rules for a set of VNICs of your choice in a VCN, you can create an NSG. See Controlling Traffic with Network Security Groups.

Security lists enable you to define network security rules that apply to all VNICs in a subnet. A default security list is automatically created for each VCN. That default security list is assigned to each subnet in the VCN if you do not assign a different security list. Up to five security lists can be associated with a subnet.

If you use both security lists and NSGs, traffic in or out of a given VNIC is allowed if any rule in any applicable security list or NSG allows the traffic:

- Any rule in any security list that is associated with the VNIC's subnet
- Any rule in any NSG that the VNIC is in

For general information and a comparison of security lists and NSGs, see "Virtual Firewall" in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

Viewing a VCN's Security Lists

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the name of the VCN for which you want to view security lists. The VCN details page is displayed.
3. Under Resources, click Security Lists. The list of security lists is displayed.
4. Click the name of the security list to view its ingress and egress rules.

Using the OCI CLI

1. Gather the information you need to run the command:
   - Compartment OCID (`oci iam compartment list`)
   - VCN OCID (`oci network vcn list --compartment-id` ***`compartment_OCID`***)
2. Run the list command.

   Use both the VCN OCID and the compartment OCID to list all security lists that belong to the specified VCN and are in the specified compartment.

```
oci network security-list list --compartment-id ocid1.compartment.unique_ID \
--vcn-id ocid1.vcn.unique_ID
```

Use only the compartment OCID to list all security lists in that compartment. The security lists in a compartment could belong to any VCN. Security lists do not need to be in the same compartment with the VCN.

```
oci network security-list list --compartment-id ocid1.compartment.unique_ID
```

3. Use one of the following methods to show just one security list.
   - Use the `list` command with the name of the security list.

     ```
     oci network security-list list --compartment-id ocid1.compartment.unique_ID \
     --display-name "Custom Security List"
     ```

   - Use the `get` command with the OCID for the security list. The security list OCID is the value of `id` property in the security list `list` command output.

     ```
     oci network security-list get --security-list-id ocid1.securitylist.unique_ID
     ```

Creating a Security List

Before you create a security list, use the following command to see the security rules that are already defined in the default security list and any other security list for this VCN:

```
$ oci network security-list get --security-list-id ocid1.securitylist.unique_ID
```

A security list must have at least one rule. A security list is not required to have both ingress and egress rules.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the name of the VCN for which you want to create a security list. The VCN details page is displayed.
3. Under Resources, click Security Lists.
4. Click the Create Security List button.
5. In the Create Security List dialog, enter the following information:
   - **Name:** A descriptive name for the security list. The name does not have to be unique. The name cannot be changed later in the Console but can changed with the CLI).
   - **Create in Compartment:** The compartment where you want to create the security list.
6. Add at least one rule.

   To add one or more ingress rules, click +New Rule in the Allow Rules for Ingress box. To add one or more egress rules, click +New Rule in the Allow Rules for Egress box. Enter the following information:

   - **Stateless:** If you want the new rule to be stateless, check this box. By default, security list rules are stateful and apply to both a request and its coordinated response. For more information about stateless and stateful rules, see "Security Lists" in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*.
   - **CIDR**: The CIDR block for the ingress or egress traffic.
   - **IP Protocol:** The rule can apply to all IP protocols, or choices such as ICMP, TCP, or UDP. Select the protocol from the drop-down list.

  - **Port Range**: For some protocols, such as TCP or UDP, you can supply a source port range and destination port range.
  - **Parameter Type and Code**: For ICMP, you can select a parameter type and corresponding parameter code.
- **Description:** An optional description of the rule.

7. **Tagging:** For more information about tagging, see Working with Resource Tags. If you are not sure whether to apply tags, skip this option (you can apply tags later) or ask your administrator.
8. Click the Create Security List button in the dialog.

   The details page of the new security list is displayed. You can specify this security list when creating or updating a subnet.

Using the OCI CLI

1. Gather the information you need to run the command:
   - The OCID of the compartment where you want to create this security list (`oci iam compartment list`)
   - The OCID of the VCN for this security list (`oci network vcn list --compartment-id` ***`compartment_OCID`***)
2. Construct arguments for the `--ingress-security-rules` and `--egress-security-rules` options.

   Security rules are in JSON format. To see how to format a rule, use the following command:

   ```
   oci network security-list create --generate-param-json-input ingress-security-rules
   > ingress.json
   ```

   Use the same command with `egress-security-rules`.

   Ingress and egress security rules are the same except that ingress rules have `source` and `sourceType` properties while egress rules have `destination` and `destinationType` properties.

   The value of the `protocol` property is `all` or one of the following numbers: 1 for ICMP, 6 for TCP, or 17 for UDP.

   Alternatively, you can `list` or `get` the default security list or another security list and copy the values of the `egress-security-rules` and `ingress-security-rules` properties.

   Put the information for rules for this new security list in the appropriate places in the format, or replace the information in the rules that you copied.

   The value of both rules options is either a string between single quotation marks or a file specified as `file://`***`path_to_file`***`.json`.

   Egress and ingress rules must be in a list. If the list of egress rules or the list of ingress rules has only one item, that single rule must be enclosed in square brackets just as multiple rules would be. See the command in the next step for an example showing only one ingress rule.

   Both egress rules and ingress rules must be specified. See the command in the next step for an example showing no egress rules.
3. Run the security list create command.

   Syntax:

```
oci network security-list create --compartment-id compartment_OCID \
--vcn-id vcn_OCID --ingress-security-rules ingress_rules \
--egress-security-rules egress_rules
```

Example:

```
$ oci network security-list create --compartment-id ocid1.compartment.unique_ID \
--vcn-id ocid1.vcn.unique_ID --display-name "Limited Port Range" \
--egress-security-rules [] \
--ingress-security-rules '[{"source": "10.0.2.0/24", "protocol": "6", "isStateless":
true, \
"tcpOptions": {"destinationPortRange": {"max": 1521, "min": 1521}, \
"sourcePortRange": {"max": 1521, "min": 1521}}}]'
{
  "data": {
    "compartment-id": "ocid1.compartment.unique_ID",
    "defined-tags": {},
    "display-name": "Limited Port Range",
    "egress-security-rules": [],
    "freeform-tags": {},
    "id": "ocid1.securitylist.unique_ID",
    "ingress-security-rules": [
      {
        "description": null,
        "icmp-options": null,
        "is-stateless": true,
        "protocol": "6",
        "source": "10.0.2.0/24",
        "source-type": "CIDR_BLOCK",
        "tcp-options": {
          "destination-port-range": {
            "max": 1521,
            "min": 1521
          },
          "source-port-range": {
            "max": 1521,
            "min": 1521
          }
        },
        "udp-options": null
      }
    ],
    "lifecycle-state": "PROVISIONING",
    "time-created": "2022-05-06T02:17:10.965748+00:00",
    "vcn-id": "ocid1.vcn.unique_ID"
  },
  "etag": "30d67d2d-5e11-4b13-9607-1948c52a78f5"
}
```

Updating a Security List

You can edit the name of the security list and add, edit, or delete rules or tags in any security list, including the default security list.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the name of the VCN for which you want to update a security list. The VCN details page is displayed.
3. Under Resources, click Security Lists.

4. For the security list that you want to update, do one of the following:
   - Click the Actions menu and then click Edit to open the Edit Security List dialog. Update rules in the Allow Rules for Ingress and Allow Rules for Egress sections. To delete a rule, click the trash can icon. To add a rule, click the +New Rule button. You can also update the security list name and tags. Click the Save button on the dialog.
   - Click the Actions menu and then click View Details to open the security list details page.
     - Click the Edit button to open the Edit Security List dialog.
     - To edit only the rules, scroll to the Resources section and click either Ingress Rules or Egress Rules. To create a new rule, click the Create Security Rule button. To update a rule, click the Actions menu for that rule and then click Edit. To delete a rule, click the actions menu and then click Delete.

Using the OCI CLI

1. Get the OCID of the security list that you want to update (`oci network vcn list --compartment-id` ***`compartment_OCID`***)
2. If you want to update rules, construct arguments for the `--ingress-security-rules` and `--egress-security-rules` options as described in Creating a Security List. Arguments that you provide to these rules options overwrite any existing rules. If you want to keep some existing rules, use the following command to show the current rules, and then copy the rules that you want to keep into the new option arguments.

   ```
   $ oci network security-list get --security-list-id ocid1.securitylist.unique_ID
   ```

3. Run the security list update command.

   Example:

   ```
   oci network security-list update  \
   --security-list-id ocid1.securitylist.unique_ID \
   --ingress-security-rules file:///home/flast/ingress_rules.json

   WARNING: Updates to defined-tags and egress-security-rules and freeform-tags and
    ingress-security-rules will replace any existing values.
    Are you sure you want to continue? [y/N]: y
   ```

Delete a Security List

You cannot delete a security list that is associated with a subnet. You cannot delete a VCN's default security list.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the name of the VCN for which you want to delete a security list. The VCN details page is displayed.
3. Under Resources, click Security Lists.
4. For the security list that you want to delete, click the Actions menu and then click Delete.
5. Confirm the deletion when prompted.

Using the OCI CLI

1. Get the OCID of the security list that you want to delete (`oci network vcn list --compartment-id` ***`compartment_OCID`***)

2. Run the security list delete command.

```
$ oci network security-list delete --security-list-id ocid1.securitylist.unique_ID
Are you sure you want to delete this resource? [y/N]: y
```

   To suppress this prompt, use the `--force` option.

Controlling Traffic with Network Security Groups

Both network security groups (NSGs) and security lists are types of virtual firewalls for your compute instances. Both NSGs and security lists define network security rules that determine which types of traffic are allowed in and out of instances (VNICs).

NSGs provide virtual firewall rules for a set of VNICs of your choice in a VCN. To provide a set of firewall rules for all VNICs in a subnet, you can create a security list. See Controlling Traffic with Security Lists.

NSGs enable you to define network security rules for groups of instances, which can be in different subnets. For example, an NSG can apply to all the database servers, or to all the application servers running a certain application. Instead of applying security to a particular subnet, you create an NSG and then add the appropriate instances (VNICs) to the NSG.

When you create a VCN, a default security list is created. No default NSG is created because you must choose which VNICs to include in the group.

If you use both security lists and NSGs, traffic in or out of a given VNIC is allowed if any rule in any applicable security list or NSG allows the traffic:

- Any rule in any security list that is associated with the VNIC's subnet
- Any rule in any NSG that the VNIC is in

For general information and a comparison of security lists and NSGs, see "Virtual Firewall" in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

Creating a Network Security Group

These procedures create an NSG with no rules and no VNICs.

- To add security rules to the NSG, see Manage Rules for a Network Security Group.
- To add VNICs to the NSG, see Attaching a VNIC to a Network Security Group.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the name of the VCN for which you want to create an NSG. The VCN details page is displayed.
3. Under Resources, click Network Security Groups.
4. Click the Create Network Security Group button.
5. In the Create Network Security Group dialog, enter the following information:
   - **Name:** A descriptive name for the NSG. The name does not have to be unique, and it can be changed later.
   - **Create in Compartment:** The compartment where you want to create the NSG.

6. **Tagging:** For more information about tagging, see Working with Resource Tags. If you are not sure whether to apply tags, skip this option (you can apply tags later) or ask your administrator.
7. Click the Create Network Security Group button in the dialog.

   The details page for the new NSG is displayed. You can create security rules and select VNICs to add to the group now, or you can do these tasks later. See the procedures referenced at the beginning of this section.

Using the OCI CLI

You can add a display name and defined and free-form tags. Similarly, when you update an NSG (`oci network nsg update`), you can only update the name and tags. To add rules and VNICs, see the procedures referenced at the beginning of this section.

1. Gather the information you need to run the command:
   - The OCID of the compartment where you want to create this NSG (`oci iam compartment list`)
   - The OCID of the VCN for this NSG (`oci network vcn list --compartment-id` ***`compartment_OCID`***)
2. Run the NSG create command.

   Example:

```
$ oci network nsg create --compartment-id ocid1.compartment.unique_ID \
--vcn-id ocid1.vcn.unique_ID --display-name "Application A"
{
  "data": {
    "compartment-id": "ocid1.compartment.unique_ID",
    "defined-tags": {},
    "display-name": "Application A",
    "freeform-tags": {},
    "id": "ocid1.networksecuritygroup.unique_ID",
    "lifecycle-state": "PROVISIONING",
    "time-created": "2022-05-09T15:48:30.069904+00:00",
    "vcn-id": "ocid1.vcn.unique_ID"
  },
  "etag": "49073741-0cc7-4371-82ee-2abf4667b14d"
}
```

Viewing a VCN's Network Security Groups

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the name of the VCN for which you want to view Network Security Groups. The VCN details page is displayed.
3. Under Resources, click Network Security Groups. The list of NSGs is displayed.
4. Click the name of the NSG to view its details, including security rules and attached VNICs.

Using the OCI CLI

1. Gather the information you need to run the command:
   - Compartment OCID (`oci iam compartment list`)
   - VLAN OCID (`oci network vlan list --compartment-id` ***`compartment_OCID`***)

2. Run the NSG list command.

   Specify the compartment OCID to list all the NSGs in that compartment.

   ```
   oci network nsg list --compartment-id ocid1.compartment.unique_ID
   ```

   Specify the VLAN OCID to list all the NSGs in that VLAN.

   ```
   oci network nsg list \
   --vlan-id ocid1.networksecuritygroup.unique_ID
   ```

3. Use one of the following methods to show just one NSG.

   - Use the `list` command with the name of the NSG.

     ```
     oci network nsg list --compartment-id ocid1.compartment.unique_ID \
     --display-name "Custom NSG"
     ```

   - Use the `get` command with the OCID for the NSG. The NSG OCID is the value of `id` property in the NSG `list` command output.

     ```
     oci network nsg get --nsg-id ocid1.networksecuritygroup.unique_ID
     ```

4. The NSG security rules are not shown in the list or get command output. Use the following command to show the NSG security rules.

   ```
   oci network nsg rules list --nsg-id ocid1.networksecuritygroup.unique_ID
   ```

Manage Rules for a Network Security Group

These procedures describe how to add, update, and remove rules that are applied by an NSG.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the name of the VCN for which you want to manage rules in an NSG. The VCN details page is displayed.
3. Under Resources, click Network Security Groups.
4. In the list of NSGs, click the name of the NSG for which you want to manage rules. The NSG details page is displayed.
5. Under Resources, click Security Rules.
6. You can add new rules, and edit and delete existing rules.

   To add a rule, click the Create Security Rules button. To add one or more ingress rules, click +New Rule in the Allow Rules for Ingress box. To add one or more egress rules, click +New Rule in the Allow Rules for Egress box. Enter the following information:

   - **Stateless:** If you want the new rule to be stateless, check this box. By default, security list rules are stateful and apply to both a request and its coordinated response. For more information about stateless and stateful rules, see "Security Lists" in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*.
   - **CIDR**: The CIDR block for the ingress or egress traffic.
   - **IP Protocol:** The rule can apply to all IP protocols, or choices such as ICMP, TCP, or UDP. Select the protocol from the drop-down list.
     - **Port Range**: For some protocols, such as TCP or UDP, you can supply a source port range and destination port range.
     - **Parameter Type and Code**: For ICMP, you can select a parameter type and corresponding parameter code.

- **Description:** An optional description of the rule.

To edit a rule, click the Actions menu for the Egress or Ingress rule, click Edit, make the necessary changes, and then click Update.

To delete a rule, click the Actions menu for the Egress or Ingress rule, click Remove, and then click Confirm. While you are editing a rule, click the trash can icon to delete the rule.

Using the OCI CLI

1. Get the OCID of the NSG for which you want to manage rules (`oci network nsg list --compartment-id <compartment_OCID>`).
2. Construct an argument for the `--security-rules` option. Security rules are in JSON format. To see how to format a rule, use the following command:

   ```
   oci network nsg rules add --generate-param-json-input security-rules > nsg_rules.json
   ```

   The `--security-rules` option argument is exactly the same for the `oci network nsg rules update` command.

   Alternatively, you can list and copy the rules of an existing NSG.

   ```
   oci network nsg rules list --nsg-id ocid1.networksecuritygroup.unique_ID
   ```

   Put the information for the rules for this new or updated NSG in the appropriate places in the format output by `--generate-param-json-input`, or change the information in the rules that you copied.

   The value of the rules option is either a string between single quotation marks or a file specified as `file://path_to_file.json`.
3. Run the NSG rules add or update command.

   Add:

   The specified *security_rules* are added to any existing rules.

   ```
   oci network nsg rules add --nsg-id nsg_OCID \
   --security-rules security_rules
   ```

   Update:

   The specified *security_rules* replace any existing rules.

   ```
   oci network nsg rules update --nsg-id nsg_OCID \
   --security-rules security_rules
   ```
4. To delete one or more rules, construct a list of rule OCIDs.

   Use the following command to find the OCIDs of the rules that you want to delete:

   ```
   oci network nsg rules list --nsg-id ocid1.networksecuritygroup.unique_ID
   ```

   Run the NSG rules remove command.

   ```
   oci network nsg rules add --nsg-id ocid1.networksecuritygroup.unique_ID \
   --security-rule-ids
   '[ocid1.security_rule.unique_ID1,ocid1.security_rule.unique_ID2]'
   ```

Attaching a VNIC to a Network Security Group

An NSG has one or more VNICs. You can attach a VNIC to an NSG when you create an instance or when you create or update the VNIC. See the following procedures:

- Creating an Instance
- Creating and Attaching a Secondary VNIC
- Using the Compute Web UI to Update NSGs Only in Updating a VNIC

Do one of the following to view the list of NSGs that a VNIC is attached to:

- View the VNIC details.
  1. On the instance details page, scroll to the resources section, and click Attached VNICs.
  2. In the list, click the name of the VNIC.
  3. On the VNIC details page, scroll to the resources section, and click Network Security Groups.
- Run the following command:

```
$ oci network vnic get --vnic-id ocid1.vnic.unique_ID
```

Do one of the following to view the list of VNICs that are attached to an NSG:

- View the NSG details.
  1. On the VCN details page, scroll to the resources section, and click Network Security Groups.
  2. In the list, click the name of the NSG.
  3. On the NSG details page, scroll to the resources section, and click VNICs.
- Run the following command:

```
$ oci network nsg vnics list --nsg-id ocid1.networksecuritygroup.unique_ID
```

To change the list of NSGs that a VNIC is attached to, update the VNIC.

Deleting a Network Security Group

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the name of the VCN for which you want to delete an NSG. The VCN details page is displayed.
3. Under Resources, click Network Security Groups.
4. For the NSG that you want to delete, click the Actions menu and then click Delete.
5. Confirm the deletion when prompted.

Using the OCI CLI

1. Get the OCID of the NSG that you want to delete (`oci network nsg list --compartment-id` ***`compartment_OCID`***)
2. Run the NSG delete command..

```
$ oci network nsg delete --nsg-id ocid1.networksecuritygroup.unique_ID
Are you sure you want to delete this resource? [y/N]: y
```

To suppress this prompt, use the `--force` option.

Configuring VCN Gateways

Virtual processes communicate with other processes in a variety of ways. If two instances are in the same subnet, meaning the network portions of their IP addresses match, there is no special configuration needed to allow them to communicate. A logical switch connects source and destination at the MAC address level. Also, communication between instances in the same VCN but different subnets requires no routing configuration. Routing is only needed for traffic that is going to a destination or coming from a source external to a VCN.

When communication between two virtual processes is needed and the source and destination are in two different VCNs, then configuration of one of five different types of gateway is necessary in the source VCN. In this context, a gateway is a special type of router, connecting two different IP networks by following rules set up in a route table. (A router can be thought of as a multiport gateway, and a gateway can be thought of as a two-port router.)

When you first create a VCN, various resources are listed in the UI and available for listing with a CLI command. Some of the resources are listed automatically when you create a subnet, and others must be configured explicitly.

- Subnets. This resource gives the number of subnets created under the VCN. All other resources also display counts for the VCN.
- Route Tables. This resource gives the number of route tables. Subnets can share route tables, especially default route tables, so this count is not necessarily the same as the count of subnets, especially if there is more than one subnet for the VCN.
- Internet Gateways. This resource gives the number of internet gateways configured . Initially, there are none.
- Local Peering Gateway. This resource gives the number of local peering gateways configured. Initially, there are none.
- DHCP Options. This resource gives the number of DHCP option lists. There is at least one for the VCN by default, but more can be created.
- Security Lists. This resource gives the number of Security Lists. There is at least one set of ingress and egress rules for the VCN by default, but more can be created.
- NAT Gateways. This resource gives the number of NAT gateways configured. Initially, there are none.
- Network Security Groups. This resource gives the number of Network Security Groups configured . Initially, there are none, but you can gather existing Security Lists into Network Security Groups, where all security rules are applied at once, as needed.
- Service Gateways. This resource gives the number of service gateways configured. Initially, there are none.
- Dynamic Routing Gateways. This resource gives the number of dynamic routing gateways (DRGs) configured. Initially, there are none. Note that these gateways are not configured without the VCN, but attached to the VCN.
- Dynamic Routing Gateway Attachments. This resource gives the number of dynamic routing gateways attachments that have been configured. You must have a DRG configured to have attachments listed.

The various types of gateways are configured for very specific reasons.

- NAT Gateway. A NAT gateway is used to translate IP addresses as traffic passes from one part of an IP network to another. When used between a VCN and the on-premises data center network. the NAT address becomes the source address for traffic sent on to the

data center network. A NAT gateway allows egress to the on-premises network from a VCN. It does not allow connections to be initiated to the instances in the VCN. Although essentially one-way, return traffic is allowed for connections initiated in the VCN. Contrast NAT Gateway with the Internet Gateway, which allows connections into and out of the VCN, the NAT Gateway allows instances with public IP addresses to be reachable from outside the PCA network.

> **Note:**
>
> A VCN connected to the on-premises network with a Dynamic Routing Gateway cannot overlap with any on-premises CIDR, or other VCN CIDRs connected with a Dynamic Routing Gateway. In other words, the IP addresses used must be exclusive to the VCN.

- Internet Gateway (IGW). An IGW provides the VCN with outside access through the on-premises data center network. The source and destination must have routable, public IP addresses, and a VCN can have only one IGW.
- Local Peering Gateway (LPG). A Local Peering Gateway (LPG) is a way to connect VCNs so that elements in each VCN can communicate, even using private IP address. Peered VCNs can be in different tenancies.
- Dynamic Routing Gateway (DRG). A DRG is used to connect a VCN to the data center's IP address space. That is, outside the Oracle Private Cloud Appliance rack in the data center. The data center network can, if configured that way, pass Oracle Private Cloud Appliance traffic on to other destinations.
- Service Gateway (SG). Some services are isolated on their own network for security and performance reasons. The service gateway (SG) allows a VCN with no external access to privately access Service Network services (such as object storage) in a private subnet.

Enabling Public Connections through a NAT Gateway

A NAT gateway is used to translate IP addresses as traffic passes from one part of an IP network to another. This prevents sources and destinations from having identical IP addresses, and allows RFC 1918 private addresses used in Oracle Private Cloud Appliance traffic to communicate with on-premises data center networks. A NAT gateway is attached to a VCN at the subnet level, allowing finer control of the address translations. The NAT gateway is configured separately from the VCNs, and is not required to be in the same compartment as the VCN (but can be). However, the NAT gateway is within the VCN, and only one NAT per VCN is allowed. The NAT address becomes the source address for traffic sent on to the data center network.

Using the Compute Web UI

1. In the navigation menu, under Networking, click Virtual Cloud Networks. A list of previously configured VCNs in compartments appears. If the compartment you are creating the NAT gateway in is not in the title bar, then use the drop-down tab to select the correct compartment.
2. Click on the VCN that you are creating the NAT gateway in.
3. In the Resources menu for that VCN, click on NAT Gateways (the number of configured NAT gateways in parentheses does not matter).
4. Click on Create NAT Gateway
5. Fill in the required NAT gateway information:

- **Name:** Provide a name or description for the NAT gateway. Avoid using any of the organization's confidential information.
- **Create in Compartment:** Select the compartment in which to create the NAT Gateway.
- **Block Traffic** Choose whether to block traffic to this NAT Gateway.
  - **(Yes: Traffic Not Blocked)**: By default, the VCN uses the NAT gateway even if it is not completely configured.
  - **(No: Traffic Blocked)**: You can set the NAT gateway not see traffic until it is explicitly enabled to do so.

  For more information on NAT gateways, refer to "NAT Gateways" in the Virtual Networking Overview chapter of the *Oracle Private Cloud Appliance Concepts Guide*.
- **Tagging:** Optionally, add one or more tags to this resource. For more information about tagging, see Working with Resource Tags. If you are not sure whether to apply tags, skip this option (you can apply tags later) or ask your administrator.

6. Click Create NAT Gateway.

   The NAT Gateway is now ready for the addition of route rules or security settings.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
   - VCN OCID (`oci network vcn list --compartment-id` ***`<compartment_OCID>`***)
2. Run the `oci network nat-gateway create` command.

> **Note:**
>
> This procedure shows the minimum required parameters for this command. For information about optional parameters, run the command with the `--help` option.

Syntax (entered on a single line):

```
oci network nat-gateway create \
--compartment-id <compartment_OCID> \
--vcn-id <vcn_OCID>
```

Example:

```
oci network nat-gateway create \
 --compartment-id ocid1.compartment.….….….uniqueID \
 --vcn-id ocid1.vcn.….….….uniqueID

{
  "data": {
    "block-traffic": true,
    "compartment-id": "ocid1.compartment.….….….uniqueID",
    "defined-tags": {},
    "display-name": "natgateway20210827215953",
    "freeform-tags": {},
    "id": "ocid1.vcn.….….….uniqueID",
```

```
    "lifecycle-state": "PROVISIONING",
    "nat-ip": "10.133.80.3",
    "public-ip-id": "ocid1.publicip.AK00661530.scasg01..….….….uniqueID",
    "time-created": "2021-08-27T21:59:53.858329+00:00",
    "vcn-id": "ocid1.vcn.AK00661530.scasg01..….….….uniqueID"
  },
  "etag": "c98377e4-ae89-46cf-9c61-52aea68a3476"
}
```

The NAT Gateway is now ready for the addition of route rules or security settings. Note that the name of the gateway (`natgateway20210827215953`) is assigned automatically and not by a parameter, and that the IP address of the device (`10.133.80.3`) is also assigned automatically.

Providing Public Access through an Internet Gateway

An Internet Gateway (IGW) provides the VCN with outside access through the on-premises data center network. The IGW is configured within the VCN, so the IGW is automatically attached to the VCN in which it is configured. The source and destination must have routable, public IP addresses, and a VCN can have only one IGW. Any traffic using public IP addresses goes through the IGW. The IGW is not required to be in the same compartment as the VCN. A subnet's route table determines which public subnets can use the IGW, and the subnet security list defines the types of traffic that can use the IGW. Like a physical router, the IGW can be disabled, severing internet access no matter what permissions are established.

Using the Compute Web UI

1. In the navigation menu, under Networking, click Virtual Cloud Networks. A list of previously configured VCNs in compartments appears. If the compartment you are creating the internet gateway in is not in the title bar, then use the drop-down tab to select the correct compartment.
2. Click on the VCN that you are creating the internet gateway in.
3. In the Resources menu for that VCN, click on Internet Gateways (the number of configured internet gateways in parentheses does not matter).
4. Click on Create Internet Gateway
5. Fill in the required internet gateway information:
   - **Name:** Provide a name or description for the internet gateway. Avoid using any of the organization's confidential information.
   - **Create in Compartment:** Select the compartment in which to create the Internet Gateway.

     For more information on internet gateways, refer to the "Internet Gateways" in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*.
   - **Enabled:** Use the toggle to determine if the gateway is enabled at creation or not. The default is to enable the gateway.
     - **(Yes: Gateway Enabled**: By default, the VCN uses the gateway when created. **(No: Gateway Disabled)**: You can set the gateway not see traffic until it is explicitly enabled to do so.
   - **Tagging:** Optionally, add one or more tags to this resource.

     For more information about tagging, see Working with Resource Tags. If you are not sure whether to apply tags, skip this option (you can apply tags later) or ask your administrator.

6. Click Create Internet Gateway.

   The Internet Gateway is now ready for the addition of route rules or security settings.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
   - VCN OCID (`oci network vcn list --compartment-id` ***<compartment_OCID>***)
2. Run the `oci network internet-gateway create` command.

> **Note:**
>
> This procedure shows the minimum required parameters for this command. For information about optional parameters, run the command with the `--help` option.

Syntax (entered on a single line):

```
oci network internet-gateway create
--compartment-id <compartment_OCID>
--is-enabled <boolean: true | false>
--vcn-id <vcn_OCID>
```

Example:

```
oci network internet-gateway create \
 --compartment-id ocid1.compartment.….….….uniqueID
 --is-enabled true
 --vcn-id ocid1.vcn.….….….uniqueID

{
  "data": {
    "compartment-id": "ocid1.compartment.….….….uniqueID",
    "defined-tags": {},
    "display-name": "internetgateway20210830165014",
    "freeform-tags": {},
    "id": "ocid1.internetgateway.AK00661530.scasg01.….….….uniqueID",
    "is-enabled": true,
    "lifecycle-state": "PROVISIONING",
    "time-created": "2021-08-30T16:50:14.634466+00:00",
    "vcn-id": "ocid1.vcn.….….….uniqueID",
  },
  "etag": "c98377e4-ae89-46cf-9c61-52aea68a3476"
}
```

The Internet Gateway is now ready for the addition of route rules or security settings. The IGW is not reachable unless there is at least one route rule for the gateway in the route table. For more information about configuring route rules, see Working with Route Tables.

Disable or Enable an Internet Gateway

You can enable or disable the IGW using the Compute Web UI or the OCI CLI.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the name of the VCN that contains the IGW you want to enable or disable. The VCN details page is displayed.
3. In the Internet Gateway list in the Resources section, locate the IGW to enable or disable. The configuration details show whether the IGW is enabled or not (Yes or No).
4. Access the Edit dialog from the Actions menu, or click Edit in the upper right of the details box.
5. Change the status of the Enabled toggle to Yes or No. Click Update to change the status of the IGW.

Using the OCI CLI

1. Get the IGW OCID of the IGW you want to enable or disable (`oci network internet-gateway list -c` ***`compartment_OCID`***)
2. Enter the `internet-gateway update --is-enabled` command with the True or False boolean value.

```
$ oci network internet-gateway update --ig-id internetgateway_OCID --is-enabled
boolean
```

Use the `--force` option to override the confirmation step.

Delete an Internet Gateway

If you have previously configured an IGW, you can delete it.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the name of the VCN that contains the IGW you want to delete. The VCN details page is displayed.
3. In the Internet Gateway list in the Resources section, locate the IGW to delete. In the Actions menu, click Delete. Confirm the operation when prompted.

Using the OCI CLI

1. Get the IGW OCID of the IGW you want to delete (`oci network internet-gateway list -c` ***`compartment_OCID`***)
2. Enter the `internet-gateway delete` command.

```
$ oci network internet-gateway delete --ig-id internet-gateway_OCID
Are you sure you want to delete this resource? [y/N]: y
```

Use the `--force` option to override the confirmation step.

Connecting VCNs through a Local Peering Gateway

A Local Peering Gateway (LPG) is a way to connect VCNs so that elements in each VCN can communicate, even using private IP address. Peered VCNs can be in different tenancies. There are several other requirements for LPG configuration:

- The CIDRs for the VCNs linked by the LPG cannot overlap.
- Each peered VCN must have an LPG configured correctly, and the LPGs must be connected.
- VCN route rules must be properly configured to steer VCN subnet traffic to and from the LPGs.
- Security rules must be properly configured to allow or deny certain types VCN subnet traffic use the LPGs

Using the Compute Web UI

1. In the navigation menu, under Networking, click Virtual Cloud Networks. A list of previously configured VCNs in compartments appears. If the compartment you are creating the local peering gateway in is not in the title bar, then use the drop-down tab to select the correct compartment.
2. Click on the VCN that you are creating the local peering gateway in.
3. In the Resources menu for that VCN, click on Local Peering Gateways (the number of configured local peering gateways in parentheses does not matter).
4. Click on Create Local Peering Gateway
5. Fill in the required Local Peering gateway information:
   - **Name:** Provide a name or description for the local peering gateway. Avoid using any of the organization's confidential information.
   - **Create in Compartment:** Select the compartment in which to create the Local Peering Gateway.
   - **Tagging:** Optionally, add one or more tags to this resource.

     For more information about tagging, see Working with Resource Tags. If you are not sure whether to apply tags, skip this option (you can apply tags later) or ask your administrator.
6. Click Create Local Peering Gateway.

   The Local Peering Gateway is now ready for connecting VCNs with Establish Peering Connection, and the addition of route rules or security settings.

   For more information on local peering gateways, refer to "Local Peering Gateways" in the Virtual Networking Overview chapter of the *Oracle Private Cloud Appliance Concepts Guide*.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
   - VCN OCID (`oci network vcn list --compartment-id` ***`<compartment_OCID>`***)
2. Run the `oci network local-peering-gateway create` command.

> **Note:**
>
> This procedure shows the minimum required parameters for this command. For information about optional parameters, run the command with the `--help` option.

Syntax (entered on a single line):

```
oci network local-peering-gateway create \
--compartment-id <compartment_OCID> \
--vcn-id <vcn_OCID>
```

Example:

```
oci network local-peering-gateway create \
 --compartment-id ocid1.compartment.….….….uniqueID \
 --vcn-id ocid1.vcn.….….….uniqueID

{
  "data": {
    "compartment-id": "ocid1.compartment.….….….uniqueID",
    "defined-tags": {},
    "display-name": "localpeeringgateway20210830174050",
    "freeform-tags": {},
    "id": "ocid1.lpg.AK00661530.scasg01..….….….uniqueID",
    "is-cross-tenancy-peering": false,
    "lifecycle-state": "AVAILABLE",
    "peer-advertised-cidr": null,
    "peer-advertised-cidr-details": null,
    "peering-status": "NEW",
    "peering-status-details": null,
    "route-table-id": null,
    "time-created": "2021-08-30T17:40:50.876023+00:00",
    "vcn-id": "ocid1.vcn.….….….uniqueID"
  },
  "etag": "c98377e4-ae89-46cf-9c61-52aea68a3476"
}
```

Connecting to the On-Premises Network through a Dynamic Routing Gateway

Dynamic Routing Gateway (DRG). A DRG is the Oracle Private Cloud Appliance equivalent of a general purpose router. A DRG is used to connect a VCN to the data center's IP address space. The router is configured separately from the VCNs, at the compartment level and is not required to be in the same compartment as the VCN (but it typically is). Once configured, the DRG can be attached to more than one VCN and, like a physical router, can be attached and detached at any time, although perhaps with traffic loss. Also like a physical router, even when attached to a VCN, the DRG must have route table rules to steer traffic to the on-premises data center network's IP address space.

Create a Dynamic Routing Gateway

Using the Compute Web UI

1. In the navigation menu, under Networking, click Dynamic Routing Gateways (DRGs). A list of previously configured DRGs in compartments appears. If the compartment you are creating the dynamic routing gateway in isn't in the title bar, then use the drop-down tab to select the correct compartment.
2. Click Create Dynamic Routing Gateway.
3. Fill in the required dynamic routing gateway information:
   - **Name:** Provide a name or description for the dynamic routing gateway. Avoid using any of the organization's confidential information.

- **Create in Compartment:** Select the compartment in which to create the dynamic routing Gateway.

  For more information on dynamic routing gateways, refer to "Dynamic Routing Gateways" in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

4. Click Create Dynamic Routing Gateway.

   The Dynamic Routing Gateway is now ready for the addition of DRG attachments, such as a route table.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
2. Run the `oci network drg create` command to create the DRG and the `oci network drg-attachment update` command to attach a route table OCID to the DRG OCID.

> **Note:**
>
> This procedure shows the minimum required parameters for this command. For information about optional parameters, run the command with the `--help` option.

Syntax (entered on a single line):

```
oci network drg create
--compartment-id <compartment_OCID>
```

Example:

```
oci network drg create \
 --compartment-id ocid1.compartment.….….….uniqueID

{
  "data": {
    "compartment-id": "ocid1.compartment.….….….uniqueID",
    "defined-tags": {},
    "display-name": "drg20210830204524",
    "freeform-tags": {},
    "id": "ocid1.drg..….….….uniqueID",
    "lifecycle-state": "AVAILABLE",
    "time-created": "2021-08-30T20:45:24.236954+00:00"
  },
  "etag": "c98377e4-ae89-46cf-9c61-52aea68a3476"
}
```

> **Note:**
>
> When the DRG has been created, use the `oci network drg-attachment update` command to attach a route table to the DRG.

```
$ oci network drg-attachment update --drg-attachment-id ocid1.drgattachment.....….….
….uniqueID \
```

```
  --route-table-id ocid1.routetable....….….….uniqueID

{
  "data": {
    "compartment-id": "ocid1.compartment.….….….uniqueID",
    "defined-tags": {},
    "display-name": "drg20210830204524",
    "freeform-tags": {},
    "drg-attachment-id": "ocid1.drgattachment..….….….uniqueID",
    "lifecycle-state": "AVAILABLE",
    "route-table-id": "ocid1.routetable.….….….uniqueID",
    "time-created": "2021-08-30T20:45:24.236954+00:00"
  },
  "etag": "c98377e4-ae89-46cf-9c61-52aea68a3476"
}
```

Attach VCNs to a Dynamic Routing Gateway

You can connect many VCNs to a DRG, but each VCN can have only one DRG attached. You must still ensure the route tables and security lists allow communication.

Using the Compute Web UI

1. In the navigation menu, under Networking, click Dynamic Routing Gateways. A list of previously configured DRGs in compartments appears. If the compartment you are attaching the dynamic routing gateway to isn't in the title bar, then use the drop-down tab to select the correct compartment.
2. Click Dynamic Routing Gateway name in the list of DRGs for that compartment.
3. Click Attach to Virtual Cloud Network.
4. Click the VCN to attach the DRG to, from the list of VCNs in the drop down list. If the correct compartment isn't in the title bar, then use the drop-down tab to select the correct compartment.
5. Click Attach to DRG.
6. Repeat the process to attach the other VCNs to the DRG and connect the VCNs.

   The Dynamic Routing Gateway is attached to the selected VCN.

You can connect up to 10 VCNs to a DRG, but each VCN can have only one DRG attached. You must still ensure the route tables and security lists allow communication.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
   - VCN OCID (`oci network vcn list --compartment-id` ***<compartment_OCID>***)
   - Dynamic Routing Gateway OCID (`oci network drg-attachment --compartment-id` ***<compartment_OCID>***)
2. Run the `oci network drg-attachment create` command.

> **Note:**
>
> This procedure shows the minimum required parameters for this command. For information about optional parameters, run the command with the `--help` option.

Syntax (entered on a single line):

```
oci network drg-attachment create \
--drg-id <drg_OCID> \
--vcn-id <vcn_OCID>
```

Example:

```
oci network drg-attachment create \
 --drg-id ocid1.drg.….….….uniqueID \
 --vcn-id ocid1.vcn.….….….uniqueID

{
 "data": {
 "compartment-id": "ocid1.compartment.….….….uniqueID",
 "display-name": "drgattachment20210902221928",
 "drg-id": "ocid1.drg.….….….uniqueID",
 "id": "ocid1.drgattachment.AK00661530.scasg01..….….….uniqueID",
 "lifecycle-state": "ATTACHING",
 "route-table-id": null,
 "time-created": "2021-09-02T22:19:28.642402+00:00",
 "vcn-id": "ocid1.vcn.….….….uniqueID
 },
 "etag": "c98377e4-ae89-46cf-9c61-52aea68a3476"
}
```

Accessing Oracle Services through a Service Gateway

Some services are isolated on their own network for security and performance reasons. The service gateway (SG) allows a VCN with no external access to privately access Service Network services (such as object storage) in a private subnet. These services are reached at the infrastructure level through the management node cluster.

The feature is non-functional and implemented for compatibility purposes.

A VCN can have only one service gateway. The service gateway is automatically attached to the VCN it is created in. Services use CIDR labels, and are allowed by default.

For each enabled Service, you need a route rule with the Service object's *cidrBlock* as the rule destination and the service gateway as the rule target.

Using the Compute Web UI

1. In the navigation menu, under Networking, click Virtual Cloud Networks. A list of previously configured VCNs in compartments appears. If the compartment you are creating the service gateway in is not in the title bar, then use the drop-down tab to select the correct compartment.
2. Click on the VCN that you are creating the service gateway in.
3. In the Resources menu for that VCN, click on Service Gateways (If you are creating a service gateway for a particular VCN< the number of configured service gateways in parentheses should be zero (0)).

4. Click on Create Service Gateway
5. Fill in the required service gateway information:
   - **Name:** Provide a name or description for the service gateway. Avoid using any of the organization's confidential information.
   - **Create in Compartment:** Select the compartment in which to create the service Gateway.
   - **Services:** Select the service from the list.
   - **Tagging:** Optionally, add one or more tags to this resource.

     For more information about tagging, see Working with Resource Tags. If you are not sure whether to apply tags, skip this option (you can apply tags later) or ask your administrator.
6. Click Create Service Gateway.

   The Service Gateway is now ready for the addition of route rules or security settings.

   For more information on service gateways, refer to "Service Gateways" in the Virtual Networking Overview chapter of the *Oracle Private Cloud Appliance Concepts Guide*.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
   - VCN OCID (`oci network vcn list --compartment-id` ***<compartment_OCID>***)
2. Run the `oci network service-gateway create` command.

   Complex data types are usually handled by using the `--generate-full-command-json-input` option, or, in this case, `oci network service-gateway create --generate-param-json-input services`. This generates a sample json file to be used with this command option. The key names are pre-populated and match the command option names (converted to camelCase format, for example, compartment-id becomes compartmentId).

   The values of the keys are edited by the user before the sample file can be used as an input to this command.

   For any command option that accepts multiple values, the value of the key can be a JSON array.

   Options can still be provided on the command line. If an option exists in both the JSON document and the command line then the command line specified value will be used.

```
oci network service-gateway create
--compartment-id ocid1.compartment..….….….uniqueID
--vcn-id ocid1.vcn..….….….uniqueID
--services '[{"serviceId":"grafana"}]'

{
 "data": {
  "displayName": "servicegateway20210830204524",
  "freeform-tags": {},
  "id": "ocid1.servicegateway..….….….uniqueID",
  "maxWaitSeconds": 0,
  "routeTableId": NULL,
  "services": [
   {
   "serviceId": "grafana"
```

```
    }
    ],
   "vcnId": ""ocid1.vcn.….….….uniqueID",
   "waitForState": "PROVISIONING",
   "waitIntervalSeconds": 0
   },
  "etag": "c98377e4-ae89-46cf-9c61-52aea68a3476"
}
```


Configuring VNICs and IP Addressing

The compute nodes in the Oracle Private Cloud Appliance have physical network interface cards (NICs). When you launch a compute instance, the Networking service creates a virtual NIC (VNIC) on top of a NIC so that the instance can communicate over the network. Each instance gets a primary VNIC, and that primary VNIC gets a primary private IP address. Neither the primary VNIC nor the primary private IP address can be removed from the instance.

You can optionally attach a public IP address to the private IP address if the subnet allows a public IP address. A private IP address enables the instance to communicate with other instances on the VCN. A public IP address enables the instance to communicate with hosts outside of the VCN, on your data center network. Internet access depends on what your data center network allows. See "Public Network in Private Cloud " and "IP Addressing" in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

You can add secondary VNICs to an instance after instance launch. Each secondary VNIC also gets a private IP address, and you can optionally attach a public IP address to the private IP address if the subnet allows a public IP address. See "Virtual Network Interface Cards (VNICs) " in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

You can add secondary private IP addresses to a VNIC, and you can optionally attach a public IP address to any secondary private IP address. For information about how secondary IP addresses are used, see "About Secondary Private IPs" in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

Managing VNICs

For information about how primary and secondary VNICs are used on Private Cloud Appliance, see "Virtual Network Interface Cards (VNICs)" in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

Viewing VNIC Attachments

Using the Compute Web UI, you can only view VNIC attachments for a particular instance. Using the OCI CLI, you can view all VNIC attachments in a compartment, and you can filter the list by instance or VNIC.

Using the Compute Web UI

1. On the Dashboard, click the Compute/View Instances button.
2. Click the name of the instance for which you want to view VNIC attachments. You might need to change the compartment to find the instance you want.
3. In the Resources box on the instance details page, click Attached VNICs.

   The list of attached VNICs for that instance is displayed.

Using the OCI CLI

1. Get the information you need to run the command.
   - To list all VNIC attachments in a compartment, get the OCID of the compartment: `oci iam compartment list`
   - To list VNIC attachments only for a specific instance, get the OCID of that instance: `oci compute instance list`
   - To list VNIC attachments only for a specific VNIC, get the OCID of that VNIC: `oci compute instance list-vnics`
2. Run the VNIC attachment list command.

   Syntax:

   ```
   oci compute vnic-attachment list --compartment-id compartment_OCID
   ```

   Examples:

   The following example lists all VNIC attachments for all instances in the specified compartment:

   ```
   $ oci compute vnic-attachment list --compartment-id ocid1.compartment.uniqueID
   {
     "data": [
       {
         "availability-domain": "AD-1",
         "compartment-id": "ocid1.compartment.uniqueID",
         "display-name": "Ainstance",
         "id": "ocid1.vnicattachment.uniqueID",
         "instance-id": "ocid1.instance.uniqueID",
         "lifecycle-state": "ATTACHED",
         "nic-index": 0,
         "subnet-id": "ocid1.subnet.uniqueID",
         "time-created": "2022-05-09T15:17:39.398551+00:00",
         "vlan-id": null,
         "vlan-tag": 0,
         "vnic-id": "ocid1.vnic.uniqueID"
       },
   ...
     ]
   }
   ```

   The following example lists VNIC attachments for the specified instance:

   ```
   $ oci compute vnic-attachment list --compartment-id ocid1.compartment.uniqueID \
   --instance-id ocid1.instance.uniqueID
   ```

   The following example lists the VNIC attachment of the specified VNIC:

   ```
   $ oci compute vnic-attachment list --compartment-id ocid1.compartment.uniqueID \
   --vnic-id ocid1.vnic.uniqueID
   ```

Viewing VNICs

Use these procedures to show details of a VNIC such as resource tags, hostname label, MAC address, NSGs, private and public IP addresses, whether this VNIC is a primary or secondary VNIC, and whether source/destination checks are being skipped.

Using the Compute Web UI

1. On the Dashboard, click the Compute/View Instances button.
2. Click the name of the instance for which you want to view VNIC attachments. You might need to change the compartment to find the instance you want.
3. On the instance details page, scroll to the Resources section and click Attached VNICs.

   The list of attached VNICs for this instance is displayed.
4. Click the name of an attached VNIC to view the details page for the VNIC.

Using the OCI CLI

1. Get the information you need to run the command.
   - To list all VNICs in a compartment, get the OCID of the compartment: `oci iam compartment list`
   - To list all VNICs that are attached to a specific instance, get the OCID of that instance: `oci compute instance list`
2. Run the VNIC list command.

   Syntax:

   ```
   oci compute instance list-vnics \
   {--compartment-id compartment_OCID | --instance-id instance_OCID}
   ```

   Example:

   The following example lists all VNICs for all instances in the specified compartment:

   ```
   oci compute instance list-vnics --compartment-id ocid1.compartment.uniqueID

   {
     "data": [
       {
         "availability-domain": "AD-1",
         "compartment-id": "ocid1.compartment.uniqueID",
         "defined-tags": {
           "Oracle-Tags": {
             "CreatedBy": "flast",
             "CreatedOn": "2022-06-07T16:09:47.05Z"
           }
         },
         "display-name": "Ainstance",
         "freeform-tags": {},
         "hostname-label": "ainstance",
         "id": "ocid1.vnic.uniqueID",
         "is-primary": true,
         "lifecycle-state": "AVAILABLE",
         "mac-address": "MACaddress",
         "nsg-ids": [
           "ocid1.networksecuritygroup.uniqueID"
         ],
         "private-ip": "privateIP",
         "public-ip": "publicIP",
         "skip-source-dest-check": false,
         "subnet-id": "ocid1.subnet.uniqueID",
         "time-created": "2022-06-07T16:09:59.813530+00:00",
         "vlan-id": null
       },
   ```

```
...
  ]
}
```

The following example lists VNICs for the specified instance:

```
$ oci compute instance list-vnics --instance-id ocid1.instance.uniqueID
```

3. To view the details for a specific VNIC, use the VNIC get command.

   Use the `list-vnics` command to get the VNIC OCID.

   ```
   $ oci network vnic get --vnic-id ocid1.vnic.uniqueID
   ```

Creating and Attaching a Secondary VNIC

The number of secondary VNICs that you can add to an instance depends on the shape of the instance, as shown in Compute Shapes in the *Oracle Private Cloud Appliance Concepts Guide*.

After you perform the following Private Cloud Appliance procedure, log onto the instance to configure the instance OS to use the new interface. See Configuring the Instance OS for a Secondary VNIC.

Using the Compute Web UI

1. On the Dashboard, click the Compute/View Instances button.
2. Click the name of the instance to which you want to add a secondary VNIC. You might need to change the compartment to find the instance you want.
3. In the Resources box on the instance details page, click Attached VNICs.

   The primary VNIC and any secondary VNICs attached to the instance are displayed.
4. Click the Create VNIC Attachment button.
5. In the Subnet section of the Create VNIC Attachment dialog box, specify the subnet to use for the VNIC. You might need to select a different compartment to find the VCN and subnet that you want.

   Specifying the same subnet for this VNIC that is specified for another VNIC for this instance can introduce asymmetric routing as described in "Virtual Network Interface Cards (VNICs)" in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

   Instead of creating a VNIC in the same subnet as an existing VNIC for this instance, consider creating a secondary private IP address for the existing VNIC that is in this subnet. See Assigning a Secondary Private IP Address.
6. Specify whether to disable source/destination checks.

   By default, a VNIC looks at the source and destination listed in the header of each network packet. If the VNIC is not the source or destination, then the packet is dropped.

   If the VNIC needs to forward traffic (for example, if the VNIC needs to perform Network Address Translation), check the box to disable this source/destination check.
7. If you selected a public subnet, you can specify whether to automatically assign a public IPv4 address object to the VNIC's private IP address object.
8. (Optional) Specify the following private IP information.

- Private IP Address. An address that is within the CIDR block range assigned to the subnet and not already in use. If you do not enter an address, an IP address is automatically assigned.
- Hostname. A hostname to be used for DNS within the cloud network. This option is available only if the VCN and subnet both have DNS labels. The hostname can be up to 63 letters, numbers, and hyphens. No spaces are allowed.

9. (Optional) Add this VNIC to an NSG.

   By default, the new VNIC is not attached to any NSG. Check the box labeled Enable Network Security Groups to add this VNIC to one or more NSGs.

   a. Select an NSG from the drop-down list. You might need to change the compartment to find the NSG you want.
   b. Click the Add Another NSG button if you want to attach to another NSG.
   c. To remove an NSG from the list, click the trash can to the right of that NSG. To remove the last NSG or all NSGs, uncheck the Enable Network Security Groups box.

   See Controlling Traffic with Network Security Groups for information about NSGs.

10. Click the Create Attachment button in the dialog. The secondary VNIC is created and then displayed on the Attached VNICs list for the instance.
11. Configure the instance OS to use the secondary VNIC. See Configuring the Instance OS for a Secondary VNIC.

Using the OCI CLI

1. Get the information you need to run the command:
   - Instance OCID: `oci compute instance list`
   - Subnet OCID: `oci network subnet list`

     Specifying the same subnet for this VNIC that is specified for another VNIC for this instance can introduce asymmetric routing as described in "Virtual Network Interface Cards (VNICs)" in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

     Instead of creating a VNIC in the same subnet as an existing VNIC for this instance, consider creating a secondary private IP address for the existing VNIC that is in this subnet. See Assigning a Secondary Private IP Address.
2. Review the list of optional parameters to disable source/destination checks, explicitly specify a private IP address, assign a public IP address, specify a host name, attach to network security groups, or assign a display name.

   ```
   oci compute instance attach-vnic -h
   ```

   Use the following command to show the JSON format to use to specify a list of attached NSGs:

   ```
   oci compute instance attach-vnic --generate-param-json-input nsg-ids
   ```
3. Run the VNIC attach command.

   Syntax:

   ```
   oci compute instance attach-vnic --instance-id instance_OCID \
   --subnet-id subnet_OCID
   ```

   Example:

In this example, the newly attached VNIC gets a public IP address and a display name, and is attached to one or more NSGs.

```
$ oci compute instance attach-vnic --instance-id ocid1.instance.unique_ID \
--subnet-id ocid1.subnet.unique_ID --assign-public-ip true \
--nsg-ids file://./InstABC-nsgs.json --vnic-display-name "InstABC-Secondary-VNIC"
```

When successful, the `attach-vnic` command has no output. To confirm that the secondary VNIC attached, list VNICs for the instance. The new attached secondary VNIC is a non-primary VNIC (the value of the `is-primary` property is `false`).

```
$ oci compute instance list-vnics --instance-id ocid1.instance.unique_ID
{
  "data": [
    {
      ...
      "display-name": "InstABC-VNIC",
      ...
      "id": "ocid1.vnic.unique_ID",
      "is-primary": true,
      ...
      "time-created": "2022-06-22T22:24:31.853538+00:00",
      ...
    },
    {
      ...
      "display-name": "InstABC-Secondary-VNIC",
      ...
      "id": "ocid1.vnic.unique_ID",
      "is-primary": false,
      ...
      "nsg-ids": [
        "ocid1.networksecuritygroup.unique_ID"
      ],
      ...
      "public-ip": "publicIP",
      ...
      "time-created": "2022-06-29T18:28:44.355805+00:00",
      ...
    }
  ]
}
```

4. Configure the instance OS to use the secondary VNIC. See Configuring the Instance OS for a Secondary VNIC.

Configuring the Instance OS for a Secondary VNIC

After you create a secondary VNIC as described in Creating and Attaching a Secondary VNIC, log in to the instance to configure the instance OS to use the new VNIC.

For Linux and Microsoft Windows, you can use the following links to complete configuration.

Linux Instance OS Configuration

Follow the steps for Linux Instance OS.

Oracle Solaris Instance OS Configuration

Use the `ipadm` command to configure network interfaces persistently.

Microsoft Windows Instance OS Configuration

Follow the steps for Microsoft Windows Instance OS.

Updating a VNIC

You can update the VNIC name, the host name, and whether to disable source/destination checks. You can add the VNIC to an NSG and remove the VNIC from an NSG.

Using the Compute Web UI Edit Option

If you only want to add or remove NSGs, see Using the Compute Web UI to Update NSGs Only.

1. On the Dashboard, click the Compute/View Instances button.
2. Click the name of the instance for which you want to update a VNIC. You might need to change the compartment to find the instance you want.
3. In the Resources box on the instance details page, click Attached VNICs.

   The list of attached VNICs for that instance is displayed.
4. For the VNIC that you want to update, click the Actions menu and then click Edit.
5. In the Update VNIC dialog, update the VNIC name, the host name, whether to disable source/destination checks, or whether to attach this VNIC to an NSG or detach this VNIC from an NSG.

   See Creating and Attaching a Secondary VNIC for information about the Skip Source/Destination Check selection.

   If you change the Enable Network Security Groups box from unchecked to checked, then you must select an NSG from the drop-down list. You might need to change the compartment to find the NSG you want.

   If the Enable Network Security Groups box is already checked, then you can click the Add Another NSG button to attach to another NSG.

   If more than one NSG is already listed, you can click the trash can next to an existing NSG to detach this VNIC from that NSG. To detach the last NSG or all NSGs, uncheck the Enable Network Security Groups box.

   See Controlling Traffic with Network Security Groups for information about NSGs.
6. Click the Update VNIC button in the dialog.

Using the Compute Web UI to Update NSGs Only

Follow the steps in the preceding procedure to display the list of attached VNICs for the instance.

1. Click the name of the VNIC for which you want to change the NSGs.
2. On the VNIC details page, scroll to the resources section, and click Network Security Groups.
3. Click the Update Network Security Groups button.
4. In the Update Network Security Groups for VNIC dialog, attach this VNIC to an NSG or detach this VNIC from an NSG.

If you change the Enable Network Security Groups box from unchecked to checked, then you must select an NSG from the drop-down list. You might need to change the compartment to find the NSG you want.

If the Enable Network Security Groups box is already checked, then you can click the Add Another NSG button to attach to another NSG.

If more than one NSG is already listed, you can click the trash can to the right of an existing NSG to detach this VNIC from that NSG. To detach the last NSG or all NSGs, uncheck the Enable Network Security Groups box.

5. Click the Update Network Security Groups for VNIC button in the dialog.
6. An alternative way to detach a VNIC from an NSG is to use the Detach menu option.
   a. On the VNIC details page, scroll to the resources section, and click Network Security Groups.
   b. In the Network Security Groups list, for the NSG that you want to detach, click the Actions menu and click Detach.

Using the OCI CLI

1. Use one of the following commands to get the OCID of the VNIC that you want to update:

```
oci compute instance list-vnics
oci compute vnic-attachment list
```

2. Review the list of optional parameters to use to update the VNIC name or the host name label, change whether to disable source/destination checks, or attach or detach NSGs.

```
oci network vnic update -h
```

Use the following command to show the JSON format to use to replace the list of attached NSGs:

```
oci network vnic update --generate-param-json-input nsg-ids
```

3. Run the VNIC update command.

Syntax:

```
oci network vnic update --vnic-id vnic_OCID
```

Example:

In this example, source/destination checks are disabled, and the list of attached NSGs is replaced.

```
$ oci network vnic update --vnic-id ocid1.vnic.unique_ID \
--nsg-ids '["ocid1.networksecuritygroup.unique_ID"]' \
--skip-source-dest-check true
{
  "data": {
    "availability-domain": "AD-1",
    "compartment-id": "ocid1.compartment.unique_ID",
    "defined-tags": {
      "Oracle-Tags": {
        "CreatedBy": "flast",
        "CreatedOn": "2022-06-28T23:08:55.06Z"
      }
    },
    "display-name": "A2instance",
    "freeform-tags": {},
    "hostname-label": "a2instance",
    "id": "ocid1.vnic.unique_ID",
```

```
    "is-primary": false,
    "lifecycle-state": "AVAILABLE",
    "mac-address": "MACaddress",
    "nsg-ids": [
      "ocid1.networksecuritygroup.unique_ID"
    ],
    "private-ip": "privateIP",
    "public-ip": "publicIP",
    "skip-source-dest-check": true,
    "subnet-id": "ocid1.subnet.unique_ID",
    "time-created": "2022-06-28T23:08:55.960950+00:00",
    "vlan-id": null
  },
  "etag": "67fe1002-e72f-4cd5-9200-ea4b5721db39"
}
```

The initial command output might not show NSG updates. If your updates are not shown, use the `network vnic get` command to re-check the VNIC configuration.

Deleting a Secondary VNIC

This operation detaches and deletes the specified secondary VNIC. You cannot delete an instance's primary VNIC. When you terminate an instance, all attached VNICs (primary and secondary) are automatically detached and deleted.

Using the Compute Web UI

1. On the Dashboard, click the Compute/View Instances button.
2. Click the name of the instance for which you want to delete a VNIC. You might need to change the compartment to find the instance you want.
3. In the Resources box on the instance details page, click Attached VNICs.

   The list of attached VNICs for that instance is displayed.
4. For the VNIC that you want to delete, click the Actions menu, and then click Delete.
5. Click the Confirm button on the dialog.

   The VNIC state changes to Detached. After a few seconds, the VNIC is removed from the list.
6. Log onto the instance and delete the configuration for the IP address from the instance OS.

   Undo the configuration you did when you added the VNIC. See Configuring the Instance OS for a Secondary VNIC.

Using the OCI CLI

1. Gather the information you need to run the command:
   - Compartment OCID: `oci iam compartment list`
   - VNIC OCID: `oci compute vnic-attachment list`
2. Run the instance detach VNIC command.

```
$ oci compute instance detach-vnic \
--compartment-id ocid1.compartment.unique_ID \
--vnic-id ocid1.vnic.unique_ID
Are you sure you want to delete this resource? [y/N]: y
```

You can suppress the confirmation by using the `--force` option.

3. Log onto the instance and delete the configuration for the IP address from the instance OS.

   Undo the configuration you did when you added the VNIC. See Configuring the Instance OS for a Secondary VNIC.

Managing IP Addresses

A private IP address enables communication with resources on the VCN. Along with route rules, security rules, and gateways, a public IP address enables communication outside the VCN, including to the data center network.

All of the following are required for an instance to communicate outside the VCN:

- The instance must be in a public subnet, which is configured when the subnet is created. Private subnets cannot have a public IP address assigned to instances in the subnet.
- The instance must have a public IP address.
- The instance's VCN must have an internet gateway configured.
- The public subnet must have route table and security list entries that enable communications outside the VCN.

For information about route rules, security rules, and gateways, see Configuring VCN Rules and Options and Configuring VCN Gateways. For conceptual information, see "IP Addressing" in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

Viewing Private IP Addresses

The Compute Web UI enables you to view private and public IP addresses for a specific instance.

The OCI CLI enables you to list all private IP address objects in the tenancy or in the specified subnet or VNIC. You can also list a single private IP address object by specifying the IP address.

Using the Compute Web UI

1. On the Dashboard, click the Compute/View Instances button.
2. Click the name of the instance for which you want to view the private IP address. You might need to change the compartment to find the instance you want.
3. On the instance details page, view networking information or VNIC information.
   - Click the Networking tab. The primary private IP address and any attached public IP address are shown in the Instance Access column.
   - Scroll to the Resources section and click Attached VNICs. Click the name of the VNIC for which you want to view IP addresses.

     On the VNIC details page, scroll to the Resources section and click IP Addresses. The primary private IP address and any secondary private IP addresses, as well as any attached public IP addresses, are shown in the table.

Using the OCI CLI

1. Get the information you need to run the command:
   - Subnet OCID: `oci network subnet list`

- VNIC OCID: `oci compute instance list-vnics`

2. Run the command to list private IP address objects.

   Syntax:

   ```
   oci network private-ip list
   ```

   Examples:

   List all private IP address objects in the tenancy:

   ```
   $ oci network private-ip list
   {
     "data": [
       {
         "availability-domain": "AD-1",
         "compartment-id": "ocid1.compartment.unique_ID",
         "defined-tags": {},
         "display-name": "privateip20220705090302",
         "freeform-tags": {},
         "hostname-label": "ol8instance",
         "id": "ocid1.privateip.unique_ID",
         "ip-address": "IPaddress",
         "is-primary": true,
         "subnet-id": "ocid1.subnet.unique_ID",
         "time-created": "2022-07-05T09:03:02.025808+00:00",
         "vlan-id": null,
         "vnic-id": "ocid1.vnic.unique_ID"
       },
   ...
     ]
   }
   ```

   List all private IP address objects in the specified subnet:

   ```
   $ oci network private-ip list --subnet-id ocid1.subnet.unique_ID
   ```

   List all private IP address objects in the specified VNIC:

   ```
   $ oci network private-ip list --vnic-id ocid1.vnic.unique_ID
   ```

   List the private IP address object with the specified IP address:

   ```
   $ oci network private-ip list --ip-address IPaddress
   ```

   The output of the preceding `list` command is the same as the output from the following `get` command:

   ```
   $ oci network private-ip get --private-ip-id ocid1.privateip.unique_ID
   ```

3. Similar to the Compute Web UI instance information, the `instance list-vnics` command shows each private and public IP address in each VNIC. This command does not show OCIDs or any other information about the IP address objects. See Viewing VNICs.

Assigning a Secondary Private IP Address

When you create an instance, the instance automatically gets a VNIC, and the VNIC automatically gets a primary private IP address. You can add secondary private IP addresses to a VNIC. A VNIC can have up to 33 private IP addresses: one primary private IP address, and up to 32 secondary private IP addresses.

Creating a VNIC in the same subnet as another VNIC for the same instance can introduce asymmetric routing as described in "Virtual Network Interface Cards (VNICs)" in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*. Instead, you can create a secondary private IP address for the existing VNIC that is in the subnet that you want.

See information about secondary private IP addresses, including use cases, in "IP Addressing" in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

After you perform the following Private Cloud Appliance procedure to assign a secondary private IP address, log onto the instance to configure the instance OS to use the new IP address. See Configuring the Instance OS for a Secondary IP Address.

Moving a Secondary IP Address

In addition to adding a secondary private IP address, you can use this procedure to reassign (move) a currently assigned secondary private IP address to a different VNIC. Because the VNIC must be in the same subnet as the VNIC where the secondary private IP address is currently assigned, the new VNIC probably is attached to a different instance; as mentioned above, having two VNICs in the same subnet in the same instance can introduce asymmetric routing.

To move a secondary private IP address, see the Unassign if assigned or `--unassign-if-already-assigned` options in the following procedures.

You cannot move a VNIC's primary private IP address.

If a public IP address object is assigned to a secondary private IP address object, and you move that secondary private IP address object to another VNIC, the public IP address object moves with it.

Using the Compute Web UI

1. On the Dashboard, click the Compute/View Instances button.
2. Click the name of the instance to which you want to add a secondary private IP address. You might need to change the compartment to find the instance you want.
3. On the instance details page, scroll to the Resources section and click Attached VNICs.

   The primary VNIC and any secondary VNICs attached to the instance are displayed.
4. Click the name of the attached VNIC to which you want to add a secondary private IP address.
5. On the VNIC details page, scroll to the Resources section and click IP Addresses.
6. Click the Assign Secondary Private IP Address button.
7. In the Attach Private IP dialog, all input fields are optional.
   - **IP Address:** If you do not enter an address, an IP address from the subnet CIDR is automatically assigned.

     If you enter an address, the IP address must be within the CIDR block for the subnet. You can enter a secondary private IP address that is already assigned to another VNIC in the subnet. You cannot enter a primary private IP address.

     If you enter an IP address that is already assigned, see the following option.
   - **Unassign if assigned:** In the previous option, if you entered a secondary private IP address that is already assigned, check this button to *move* that private IP address. The address will be unassigned from the VNIC where it is currently assigned and reassigned to this VNIC.

If you entered an IP address that is already assigned and you do not check this button, this secondary private IP assignment operation fails.

- **Hostname:** Enter the hostname to be used for DNS within the cloud network. This option is available only if the VCN and subnet both have DNS labels.

8. Click the Attach IP Address button in the dialog.

   The new secondary private IP address is shown in the table.

9. Configure the new secondary private IP address in the instance. See Configuring the Instance OS for a Secondary IP Address

Using the OCI CLI

1. Get the OCID of the VNIC where you want to assign this secondary private IP address: `oci compute instance list-vnics`
2. Run the assign private IP command.

   Syntax:

   ```
   oci network vnic assign-private-ip --vnic-id vnic_OCID
   ```

   Examples:

   ```
   $ oci network vnic assign-private-ip --vnic-id ocid1.vnic.unique_ID
   {
     "data": {
       "availability-domain": "AD-1",
       "compartment-id": "ocid1.compartment.unique_ID",
       "defined-tags": {},
       "display-name": "privateip20220707213054",
       "freeform-tags": {},
       "hostname-label": null,
       "id": "ocid1.privateip.unique_ID",
       "ip-address": "IPaddress",
       "is-primary": false,
       "subnet-id": "ocid1.subnet.unique_ID",
       "time-created": "2022-07-07T21:30:54.305936+00:00",
       "vlan-id": null,
       "vnic-id": "ocid1.vnic.unique_ID"
     },
     "etag": "756b973a-c76e-4151-92ad-24fa265c8289"
   }
   ```

   In the following example, an existing private IP address is moved to a different VNIC:

   ```
   $ oci network vnic assign-private-ip --vnic-id ocid1.vnic.unique_ID \
   --ip-address IPaddress --unassign-if-already-assigned
   ```

3. Configure the new secondary private IP address in the instance. See Configuring the Instance OS for a Secondary IP Address.

Configuring the Instance OS for a Secondary IP Address

After you create a secondary private IP address on a VNIC as described in Assigning a Secondary Private IP Address, log in to the instance to configure the instance OS to use the new IP address.

Linux Instance OS Configuration

This configuration permits use of an IP address subnet, netmask, gateway, and DNS service that are entirely independent from the existing NIC. This configuration is persistent across reboots.

Create a new network interface configuration file to create a sub-interface on the existing NIC. In this example, `ens03` is the name of the existing NIC and `ifcfg-ens3:0` is the name of the new configuration file.

1. Create the network configuration file `ifcfg-ens3:0` in the `/etc/sysconfig/network-scripts/` directory to create the first sub-interface (`:0`) on the existing `ens3` NIC.

   Include the following entries in `ifcfg-ens3:0`:

   ```
   TYPE=Ethernet
   BOOTPROTO=none
   IPADDR=a.b.c.d
   PREFIX=24
   GATEWAY=
   DNS=
   NAME=ens3:0
   DEVICE=ens3:0
   ```

2. Include the appropriate `IPADDR`, `PREFIX`, `GATEWAY`, and `DNS` entries for this new sub-interface.
3. Run the following command to start the new interface:

   ```
   # ifup ens3:0
   ```

4. Run the following command to confirm that the new interface is operational:

   ```
   # ifconfig -a
   ```

See also Linux: Details about Secondary IP Addresses.

Oracle Solaris Instance OS Configuration

Use the `ipadm` command to configure network interfaces persistently.

Microsoft Windows Instance OS Configuration

See Windows: Details about Secondary IP Addresses for information about how to either:

- Create a PowerShell script.
- Use the Network and Sharing Center UI.

Updating a Secondary Private IP Address

You cannot update a VNIC's primary private IP address.

You can update the host name of a secondary private IP address object. To change the IP address, delete the secondary private IP address object as described in Deleting a Secondary Private IP Address, and create a new one as described in Assigning a Secondary Private IP Address, explicitly specifying the IP address that you want to use.

To update the host name for the primary private IP on a VNIC, update the VNIC. See Updating a VNIC.

Using the Compute Web UI

1. On the Dashboard, click the Compute/View Instances button.
2. Click the name of the instance that has the secondary private IP address object that you want to update. You might need to change the compartment to find the instance you want.
3. On the instance details page, scroll to the Resources section and click Attached VNICs.

   The primary VNIC and any secondary VNICs attached to the instance are displayed.
4. Click the name of the attached VNIC that has the secondary private IP address object that you want to update.
5. On the VNIC details page, scroll to the Resources section and click IP Addresses.
6. For the secondary private IP address object that you want to update, click the Actions menu and click Edit.
7. In the Attach Private IP dialog, update the host name.
8. Click the Attach IP Address button in the dialog.

Using the OCI CLI

1. Get the OCID of the secondary private IP address object that you want to update: `oci network private-ip list`
2. Run the private IP address update command.

   Syntax:

   ```
   oci network private-ip update --private-ip-id private_ip_OCID \
   --hostname-label newhostname
   ```

   The output is the same as the output of the `private-ip get` command.

Deleting a Secondary Private IP Address

You cannot delete a VNIC's primary private IP address.

On successful delete, the private IP address is returned to the pool of available addresses in the subnet. Any attached public IP address is again available for assignment.

Using the Compute Web UI

1. On the Dashboard, click the Compute/View Instances button.
2. Click the name of the instance for which you want to delete a secondary private IP address. You might need to change the compartment to find the instance you want.
3. On the instance details page, scroll to the Resources section and click Attached VNICs.

   The primary VNIC and any secondary VNICs attached to the instance are displayed.
4. Click the name of the attached VNIC for which you want to delete a secondary private IP address.
5. On the VNIC details page, scroll to the Resources section and click IP Addresses.
6. For the secondary private IP address that you want to delete, click the Actions menu and click Delete.

   Confirm the deletion.

7. Log onto the instance and delete the configuration for the IP address from the instance OS.

   Undo the configuration you did when you added the IP address. See Configuring the Instance OS for a Secondary IP Address.

Using the OCI CLI

1. Gather the information you need to run the command:
   - Private IP address: `oci network private-ip list`
   - VNIC OCID: `oci compute instance list-vnics`
2. Run the unassign private IP command.

   Syntax:

   ```
   oci network vnic unassign-private-ip --ip-address IPaddress --vnic-id VNIC_OCID
   ```

   Confirm the deletion, or use the `--force` option.

   The secondary private IP address object is unassigned and then deleted.
3. Log onto the instance and delete the configuration for the IP address from the instance OS.

   Undo the configuration you did when you added the IP address. See Configuring the Instance OS for a Secondary IP Address.

Viewing Public IP Addresses

The Compute Web UI enables you to view private and public IP addresses for a specific instance. See "Using the Compute Web UI" in Viewing Private IP Addresses.

The OCI CLI enables you to list public IP address objects in a specified compartment.

Using the OCI CLI

1. Get the OCID of the compartment where the instance is located: `oci iam compartment list`
2. Run the public IP list command.

   Syntax:

   ```
   oci network public-ip list --compartment-id compartment_OCID \
   --scope {region | availability_domain}
   ```

   Examples:

   List reserved public IP address objects:

   ```
   $ oci network public-ip list --compartment-id ocid1.compartment.unique_ID \
   --scope region --lifetime reserved
   {
     "data": [
       {
         "assigned-entity-id": null,
         "assigned-entity-type": "PRIVATE_IP",
         "availability-domain": null,
         "compartment-id": "ocid1.compartment.unique_ID",
         "defined-tags": {},
         "display-name": "apublicIP",
         "freeform-tags": {},
   ```

```
      "id": "ocid1.publicip.unique_ID",
      "ip-address": "IPaddress",
      "lifecycle-state": "AVAILABLE",
      "lifetime": "RESERVED",
      "private-ip-id": null,
      "public-ip-pool-id": null,
      "scope": "REGION",
      "time-created": "2022-07-06T16:36:56.860931+00:00"
    }
  ]
}
```

List the ephemeral public IP address objects that are assigned to a regional entity such as a NAT gateway:

```
$ oci network public-ip list --compartment-id ocid1.compartment.unique_ID \
--scope region --lifetime ephemeral
```

List the ephemeral public IP address objects that are assigned to primary private IP address objects:

```
$ oci network public-ip list --compartment-id ocid1.compartment.unique_ID \
--scope availability_domain --availability-domain AD-1 --lifetime ephemeral
```

Assigning an Ephemeral Public IP Address to an Instance

To assign a public IP address to an instance, you assign the public IP address object to a private IP address object.

An ephemeral public IP address is created and assigned in the same step.

An ephemeral public IP address can only be assigned to a primary private IP address: The value of the `is-primary` property of the private IP address object must be `true`. Every VNIC has one primary private IP address.

For secondary private IP addresses (the value of the `is-primary` property of the private IP address object is `false`), assign reserved public IP addresses. See Assigning a Reserved Public IP Address to an Instance.

An ephemeral public IP address cannot be unassigned and cannot be moved to a different private IP address.

An ephemeral public IP address object is deleted in the following cases:

- Its private IP address object is deleted.
- Its VNIC is detached or terminated.
- Its instance is terminated.

Using the Compute Web UI

1. On the Dashboard, click the Compute/View Instances button.
2. Click the name of the instance for which you want to assign a public IP address. You might need to change the compartment to find the instance you want.
3. On the instance details page, scroll to the Resources section and click Attached VNICs. Click the name of the VNIC for which you want to assign a public IP address.
4. On the VNIC details page, scroll to the Resources section and click IP Addresses. The primary private IP address and any secondary private IP addresses, as well as any attached public IP addresses, are shown in the table.

5. If the primary private IP address does not already have a public IP address assigned, click the Actions menu for the primary private IP address, and then click Edit Public IP,
6. In the dialog, click Ephemeral Public IP.
7. (Optional) Give the ephemeral public IP address a name.
8. Click the Reserve Public IP button in the dialog.

   You might have to refresh the page to see the new public IP address. The new public IP address shows in the IP Addresses table in Resources, in the Primary IP Information column for the VNIC, and in the Instance Access column of the Networking tab of the instance.

Using the OCI CLI

1. Get the information you need to run the command:
   - Compartment OCID: `oci iam compartment list`
   - Private IP OCID: `oci network private-ip list`
   - Public IP OCID: `oci network public-ip list`
2. Run the public IP create command.

   This command creates a new ephemeral public IP address object and assigns it to the specified private IP address object.

```
$ oci network public-ip create --compartment-id ocid1.compartment.unique_ID \
--lifetime ephemeral --private-ip-id ocid1.privateip.unique_ID
  "data": {
    "assigned-entity-id": "ocid1.privateip.unique_ID",
    "assigned-entity-type": "PRIVATE_IP",
    "availability-domain": "AD-1",
    "compartment-id": "ocid1.compartment.unique_ID",
    "defined-tags": {},
    "display-name": "publicip20220708231248",
    "freeform-tags": {},
    "id": "ocid1.publicip.unique_ID",
    "ip-address": "IPaddress",
    "lifecycle-state": "ASSIGNING",
    "lifetime": "EPHEMERAL",
    "private-ip-id": "ocid1.privateip.unique_ID",
    "public-ip-pool-id": null,
    "scope": "AVAILABILITY_DOMAIN",
    "time-created": "2022-07-08T23:12:48.610545+00:00"
  },
  "etag": "dcb8dafe-bbe4-42ff-b86a-9e1ebaf4d94c"
}
```

Reserving a Public IP Address

Use the Compute Web UI procedure in Updating a Public IP Address to create and assign a reserved public IP address in one step.

Use the following OCI CLI procedure to create a reserved public IP address that is available to assign to a private IP address object at a later time.

Using the OCI CLI

1. Get the OCID of the compartment where you want to create the IP address object: `oci iam compartment list`

2. Run the public IP create command.

   Syntax:

   ```
   oci network public-ip create --compartment-id compartment_OCID \
   --lifetime reserved
   ```

   Example:

   ```
   $ oci network public-ip create --compartment-id ocid1.compartment.unique_ID \
   --lifetime reserved --display-name apublicIP
   {
     "data": {
       "assigned-entity-id": null,
       "assigned-entity-type": "PRIVATE_IP",
       "availability-domain": null,
       "compartment-id": "ocid1.compartment.unique_ID",
       "defined-tags": {},
       "display-name": "apublicIP",
       "freeform-tags": {},
       "id": "ocid1.publicip.unique_ID",
       "ip-address": "IPaddress",
       "lifecycle-state": "PROVISIONING",
       "lifetime": "RESERVED",
       "private-ip-id": null,
       "public-ip-pool-id": null,
       "scope": "REGION",
       "time-created": "2022-07-06T16:36:56.860931+00:00"
     },
     "etag": "dcb8dafe-bbe4-42ff-b86a-9e1ebaf4d94c"
   }
   ```

Assigning a Reserved Public IP Address to an Instance

To assign a public IP address to an instance, you assign the public IP address object to a private IP address object.

An ephemeral public IP address can be assigned only to the primary private IP address of a VNIC. See Assigning an Ephemeral Public IP Address to an Instance. A reserved public IP address can be assigned to any private IP address.

See Reserving a Public IP Address to create a reserved public IP address that is available to assign to a private IP address object at a later time.

Use the procedures in Updating a Public IP Address to assign an existing reserved public IP address object to the specified private IP address object or to create and assign a reserved public IP address in one step.

A reserved public IP address object remains available for reassignment when its private IP address object is deleted, its VNIC is detached or terminated, or its instance is terminated.

Updating a Public IP Address

You can use the public IP update command to do any of the following:

- Assign an existing reserved public IP address object to a private IP address object.
- Create and assign a reserved public IP address object in one step.
- Move a reserved public IP address object to a different private IP address object.
- Unassign a reserved public IP address object from a private IP address object.

- Change the display name or tags for a public IP address object.

Using the Compute Web UI

1. On the Dashboard, click the Compute/View Instances button.
2. Click the name of the instance for which you want to assign a public IP address. You might need to change the compartment to find the instance you want.
3. On the instance details page, scroll to the Resources section and click Attached VNICs. Click the name of the VNIC for which you want to assign a public IP address.
4. On the VNIC details page, scroll to the Resources section and click IP Addresses. The primary private IP address and any secondary private IP addresses, as well as any attached public IP addresses, are shown in the table.
5. For the private IP address for which you want to add or update a public IP address, click the Actions menu and then click Edit Public IP.
6. In the Reserve Public IP dialog, click one of the following choices:
   - No public IP

     Click the Reserve Public IP button in the dialog to unassign this public IP address from this private IP address. You might have to refresh the page to see that the public IP address is no longer assigned.
   - Reserve public IP

     Click one of the following choices:
     - Reserve existing public IP
       a. Select an existing public IP address. You might need to change the compartment.
       b. Click the Reserve Public IP button in the dialog.

          If the specified public IP address object is already assigned to a different private IP address object, the public IP address object will be unassigned (moved) from the current private IP address object and reassigned to the specified private IP address object.
     - Create public IP

       Create and assign a reserved public IP address in one step.
       a. (Optional) Provide a name for the new reserved public IP address object.
       b. Select the compartment where the new reserved public IP address object will be created.
       c. Select the IP Address Source.
       d. Click the Reserve Public IP button in the dialog.

   The new reserved public IP address shows in the IP Addresses table in Resources. You might need to refresh the page to see the new public IP address.

Using the OCI CLI

1. Get the OCID of the public IP object that you want to update. See Viewing Public IP Addresses.

   If you want to assign or move the public IP object to a private IP object, get the OCID of the private IP object. See Viewing Private IP Addresses.
2. Run the public IP update command.

Syntax:

```
oci network public-ip update --public-ip-id public_ip_OCID
```

Example:

The following example updates an existing reserved public IP address object and assigns it to the specified private IP address object. If the specified public IP address object is already assigned to a different private IP address object, the public IP address object will be unassigned (moved) from the current private IP address object and reassigned to the specified private IP address object.

```
$ oci network public-ip update --public-ip-id ocid1.publicip.unique_ID \
--private-ip-id ocid1.privateip.unique_ID
{
  "data": {
    "assigned-entity-id": null,
    "assigned-entity-type": "PRIVATE_IP",
    "availability-domain": null,
    "compartment-id": "ocid1.compartment.unique_ID",
    "defined-tags": {},
    "display-name": "apublicIP",
    "freeform-tags": {},
    "id": "ocid1.publicip.unique_ID",
    "ip-address": "IPaddress",
    "lifecycle-state": "ASSIGNING",
    "lifetime": "RESERVED",
    "private-ip-id": null,
    "public-ip-pool-id": null,
    "scope": "REGION",
    "time-created": "2022-07-06T16:36:56.860931+00:00"
  },
  "etag": "dcb8dafe-bbe4-42ff-b86a-9e1ebaf4d94c"
}
```

The following example unassigns the specified reserved public IP address object and makes it available for future reassignment.

```
$ oci network public-ip update --public-ip-id ocid1.publicip.unique_ID \
--private-ip-id ""
```

Deleting a Public IP Address

An ephemeral public IP address object cannot be unassigned and cannot be directly deleted. An ephemeral public IP address object is deleted in the following cases:

- Its private IP address object is deleted.
- Its VNIC is detached or terminated.
- Its instance is terminated.

A reserved public IP address object is unassigned but remains available for reassignment when its private IP address object is deleted, its VNIC is detached or terminated, or its instance is terminated.

Use the following procedure to delete a reserved public IP address object.

Using the OCI CLI

1. Gather the information you need to run the command:
   - Compartment OCID: `oci iam compartment list`

- Public IP address OCID: `oci network public-ip list`

2. Run the public IP delete command.

   Syntax:

   ```
   oci network public-ip delete --public-ip-id public_ip_OCID
   ```

   Example:

   ```
   oci network public-ip delete --public-ip-id ocid1.publicip.unique_ID --force
   ```

Configuring SR-IOV for Virtual Networking

Single root I/O virtualization (SR-IOV) technology enables virtual machines to achieve low latency and high throughput simultaneously on 1 or more physical links. Oracle Private Cloud Appliance supports up to 84 Virtual Functions (VFs) per compute node. For more information see the SR-IOV section in the *Oracle Private Cloud Appliance Concepts Guide*.

This section describes how to configure SR-IOV networking.

1. Ensure you have the OraclePCA.networkType tag defined on the system. See "Network Configuration for SR-IOV" in Creating OraclePCA Tags.

   Setting the OraclePCA.networkType:VFIO tag enables SR-IOV functionality.

   > **Note:**
   >
   > When you update a VCN or DRG that has the OraclePCA.networkType:VFIO tag applied, that tag cannot be changed or removed from the VCN or DRG. If you want this VCN or DRG to no longer be configured for SR-IOV, then delete the VCN or DRG and create new ones that do not have the OraclePCA.networkType:VFIO tag set.

2. Create a VCN with SR-IOV functionality enabled.

   Create a VCN as described in Creating a VCN. In the Tagging section, add the OraclePCA.networkType tag with the value VFIO.

   You must create a VCN with SR-IOV support enabled: the OraclePCA.networkType tag applied with value VFIO. You cannot add SR-IOV functionality to an existing VCN.

3. If you plan to use a DRG in your SR-IOV configuration, you must create a DRG with SR-IOV functionalty. Only SR-IOV DRGs can attach to SR-IOV VCNs.

   a. Create a DRG as described in Create a Dynamic Routing Gateway. In the Tagging section, add the OraclePCA.networkType tag with the value VFIO.

      You must create a DRG with SR-IOV support enabled: the OraclePCA.networkType tag applied with value VFIO. You cannot add SR-IOV functionality to an existing DRG.

   b. Attach the SR-IOVs VCNs to the DRG as described in Attach VCNs to a Dynamic Routing Gateway.

4. Prepare an instance for SR-IOV functionality.

   a. Create and launch an instance. See Creating an Instance.

   b. Create and attach a secondary VNIC to the instance to use as the SR-IOV network interface. The primary VNIC of the instance cannot be the SR-IOV VNIC. See Assigning a Secondary Private IP Address in Configuring VNICs and IP Addressing.

c. Configure the network bond interfaces, including the secondary IP address on a SR-IOV bond port, using the `configure_vfio` script provided in the Oracle systems blog Automating SR-IOV/VFIO bond creation on Oracle Compute Cloud@Customer and Private Cloud Appliance.

Note the following when working with SR-IOV components:

- Instances configured with SR-IOV networking are non-migratable instances. These types of instances can't be live migrated. If you need to migrate these instances, you must manually shut down the instance before migration. For more information, see Migrating Instances from a Compute Node.
- You can't create these VCN components in an SR-IOV VCN:
  - Internet Gateways
  - NAT Gateways
  - Local Peering Gateways
  - Service Gateways
  - Security Lists. You can't add new entries to a default security list belonging to an SR-IOV VCN. By default, the SR-IOV VCN has open ingress and egress, with just 1 rule each.
  - DHCP Options
  - Network Security Groups
  - Route Tables. You can only add a default route with the target as an SR-IOV DRG in the default route table of an SR-IOV VCN.
  - You can't create the following objects using an SR-IOV VCN/subnet: Load Balancer, Network Load Balancer, Mount Targets, OKE clusters.

Managing Public DNS Zones

In its most basic form, DNS returns an IP address (if known) when given a string in the DNS name space for that zone. However, DNS is also the way that an IP host client application knows where to get its own configuration information using DHCP (DHCID records), go to send or receive email (MX records), and more. Without DNS, client devices would have to know the proper IP addresses not only for local servers, but for every server or application they interacted with, no matter where in the world they were located. With DNS, clients can always find the correct location of www.oracle.com or any other application.

Once you create a DNS zone inside a compartment, you cannot move the zone to another compartment.

Creating a Public DNS Zone

DNS zones are created in a compartment to associate IP addresses with portions of the DNS name space. Zones are created in a compartment using the DNS service.

Using the Compute Web UI

1. In the navigation menu, under DNS, click Zones. A list of previously configured zones in compartments appears. If the compartment you are creating the DNS zone in is not in the title bar, then use the drop-down tab to select the correct compartment.
2. Click Create Zone.

3. Fill in the required zone information:
   - **Zone Name:** Provide a name or description for the DNS zone. Avoid using any of the organization's confidential information.
   - **Compartment:** Select the compartment in which to create the DNS zone.
   - **Zone Type:** Choose the type of DNS zone you are creating.
     - **Primary**: A primary DNS zone is the original authoritative DNS zone of a portion of the DNS name space. When a DNS server hosts a primary zone, that DNS server is the Authoritative DNS Server and is considered the primary source of information in that zone.
     - **Secondary**: A secondary DNS zone is a read-only copy of a primary DNS zone or another secondary DNS zone. A secondary DNS zone is kept on a Secondary DNS Server and reduces the load on the primary DNS zone and eliminated a single point of failure risk to name resolution inside the zone.

     For more information on DNS Zones, refer to "Name Resolution" in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*.
   - **Tagging:** Optionally, add one or more tags to this resource.

     For more information about tagging, see Working with Resource Tags. If you are not sure whether to apply tags, skip this option (you can apply tags later) or ask your administrator.
4. Click Create Zone.

   The zone is now ready for the addition of zone records or for the configuration of TSIG Keys or Steering Policies.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
2. Run the `oci dns zone create` command.

   Syntax (entered on a single line):

```
oci dns zone create \
--compartment-id <compartment_OCID> \
--name <dns_zone_OCID> \
--zone-type <PRIMARY | SECONDARY>
```

   Example:

```
oci dns zone create \
 --compartment-id ocid1.compartment.….….….uniqueID \
 --name test-dns-zone \
 --zone-type PRIMARY

{
  "data": {
    "compartment-id": "ocid1.compartment.….….….uniqueID",
    "defined-tags": {},
    "external-masters": null,
    "freeform-tags": {},
    "id": "ocid1.dns-zone.….….….uniqueID",
    "is-protected": null,
    "lifecycle-state": "ACTIVE",
    "name": "test_dns_zone",
```

```
    "nameservers": [
      {
        "hostname": "ns1.example.com"
      }
    ],
    "scope": null,
    "self-uri": "https://20180115/zones/test_dns_zone",
    "serial": 1,
    "time-created": "2021-08-17T18:08:00.059867+00:00",
    "version": 1,
    "view-id": null,
    "zone-type": "PRIMARY"
  },
  "etag": "3e389cab-b3fd-4783-91c1-ede81bc132d5"
}
```


Working with Zone Records

Creating a DNS zone is only the beginning of working with DNS. The zone is essentially empty when created, except for a basic Start of Authority (SOA) and Name Server (NS) record The SOA record provides a kind of history of this DNS zone and holds information such as when it was last updated and things like that. The NS record contains the fully-qualified name of the DNS server for the zone. The NS record is very important and therefore has a high TTL, usually 24 hours (86400 seconds).

To make the name server truly useful, the zone must be rounded out and filled with the DNS records that form the basis of responses to the kinds of queries that clients make. These queries include IP addresses for parts of the domain name space, email server details, and so on.

Creating a Zone Record

The RDATA field is where the content of the zone record is entered. The format of the information varies according to the type of record you are creating. However, the data must be in one of the formats that DNS understands. For example, an A-type zone record RDATA is an IP address, and an MX record contains information on how to route email. Because of the authoritative nature of the zone records within a zone, RDATA is not editable. If DNS information in a zone changes, then the old record must be deleted and a new record created.

Using the Compute Web UI

1. In the navigation menu, under DNS, click Zones. A list of previously configured zones in compartments appears. If the compartment you are adding zone records to is not in the title bar, then use the drop-down tab to select the correct compartment.
2. Click on the name of the zone. The information screen contains general zone information such as type and compartment, OCID (which you can show in full or copy to the clipboard), and the date and time that the zone was created. The zone records that exist are also displayed, and initially there are only SOA and NS records.
3. Fill in the required zone record information:
   - **Zone Record:**Select the type of zone record you are creating from the drop-down list.
     - **A - IPv4 Address**: A host record, which is used to point a hostname to an IPv4 address. This is the most basic DNS record type.
     - You can add many other types of zone records: any types in the drop down list.

       For more information on DNS Zones, refer to "Name Resolution" in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

- **Domain (Optional)**: Type the name of the zone subdomain if used (this value is already filled in based on the zone itself: the initial dot (".") is used for adding the zone subdomain).
- **TTL**: Check this box to set your own value for the TTL of that particular record type. If you do not check this box, the default TTL value for that record type is used (for example, 300 for SOA, 86400 for NS). The valid range is from 1 to 129540 seconds (from 1 second to about a day and a half).
- **Edit RDATA**: Check this box if you wish to edit the RDATA information, such as the IP address or Target established by the zone record type. This box is only displayed for some zone record types.
- **(RDATA)**: This unlabeled field varies based on the type of zone record created. For example, you enter the 32-bit IP address that corresponds to the A-type DNS record, or Flags for a DNSKEY zone record, if that is what you are creating.
  - **A - IPv4 Address**: If you are creating an A type zone record, then the data is a properly formatted IPv4 address. This is the most basic DNS record, but there are many others.
  - The RDATA field reflects the correct information for the type of zone record selected.

4. Click Create Record.

   The zone record is now added to the zone. If you click the optional box to **Add another record**, then the screen stays at the **Create DNS Zone Record** state to make record entry more efficient.

Using the OCI CLI

There is no "`create dns zone record`" command in the CLI. Instead, the command "`oci dns zone record update`" command replaces records in the specified zone with the records specified in the request body of the command. If a specified record does not exist, then it will be created. Also, if a current record is not in the records list, it will be deleted. Care is needed, because if the record exists, then it will be updated with the record information in the body of the request. The command in this section adds an A resource record (IPv4 address and domain name) to a DNS zone named `dns-test-zone`.

1. Gather the information that you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
   - DNS zone name (`oci dns zone list --compartment_OCID` ***<compartment_OCID>***)
2. Run the `oci dns record zone update` command.

> **Note:**
>
> This procedure shows the minimum required parameters for this command. For information about optional parameters, run the command with the `--help` option.

Syntax (entered on a single line):

```
oci dns record zone update \
 --zone-name-or-id <zone_name> or <compartment_OCID> \
 --items <complex type>
```

> **Note:**
>
> DNS resource record types are provided as objects in JSON format. This is a JSON list with items of type RecordDetails. For documentation on RecordDetails please see https://docs.cloud.oracle.com/api/#/en/dns/20180115/datatypes/RecordDetails. This is a complex type whose value must be valid JSON. The value can be provided as a string on the command line or passed in as a file using the `file://path/to/file` syntax.
>
> The `--generate-param-json-input` option can be used to generate an example of the JSON which must be provided. We recommend storing this example in a file, modifying it as needed and then passing it back in via the `file://` syntax.

Example:

```
oci dns record zone update
 --zone-name-or-id <zone_name> or <compartment_OCID>
 --items
{
    "domain": "test-dns-zone.test-pca-comparment.example.com",
    "isProtected": true,
    "rdata": "10.225.15.10",
    "recordHash": "fkT4md",
    "rrsetVersion": "1",
    "rtype": "1",
    "ttl": 3600
}
```

Editing a Zone Record

There is no "`edit record`" command. You can update a group of records, and if one of the records in the list is the same except for the `rdata` for example, in effect you have updated the record.

Deleting a Zone Record

You can delete many, but not all, DNS zone records. The initial SOA and NS records, created by default when the zone is created, cannot be deleted, To delete a zone record:

Using the Compute Web UI

1. In the navigation menu, under DNS, click Zones. A list of previously configured zones in compartments appears. If the compartment you are adding zone records to is not in the title bar, then use the drop-down tab to select the correct compartment.
2. Click on the name of the zone. The information screen contains general zone information such as type and compartment, OCID (which you can show in full or copy to the clipboard), and the date and time that the zone was created. The zone records that exist are also displayed.
3. Click on the Action square with the three dots on the right side of the zone record that you are deleting.
4. Click Delete.

The zone record is deleted and removed from the list for that DNS zone.

Using the OCI CLI

To delete resource records in a zone with the CLI, use the `oci dns record rrset delete` command to delete an entire resource record set (for example, all A-type IPv4 address records for a given host name). The resource records are identified by their DNS record type (A, MX, and so on). The command in this section deletes the A resource record (IPv4 address and domain name) in a DNS zone named "dns-test-zone" for a device named "test-device-1."

1. Gather the information that you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
   - DNS zone name (`oci dns zone list --compartment_OCID` ***<compartment_OCID>***)
2. Run the `oci dns record rrset delete` command.

   Syntax: (entered on a single line):

   ```
   oci dns record rrset delete \
    --domain <domain-name> \
    --rtype <resource-record-type> \
    --zone-name-or-id <zone_name> or <compartment_OCID>
   ```

   Example:

   ```
   oci dns record rrset delete \
    --domain "test-device-1.dns-test-zone.example.com" \
    --rtype "A" \
    --zone-name-or-id "dns-test-zone"
   ```

   The record is deleted from the zone.

Editing a Public DNS Zone

You can add resource tags to an existing zone. You can also edit the `externalMasters` field of a `SECONDARY` zone.

Working with Transaction Signature Keys

A DNS transaction signature (TSIG) is a network protocol defined in RFC 2845. The main purpose of the TSIG is to allow DNS to authenticate updates to a DNS database, so that malicious users cannot change name resolution records to point to a bogus IP address instead of (for example) the IP address of a bank. TSIG uses one-way hashing and shared secret keys to provide a secure means to authenticate the endpoints of a connection for processing (or responding to) DNS update requests.

The TSIG protocol uses timestamps to prevent replay of recorded responses. Therefore, DNS servers and TSIG clients need accurate clocks to provide the timestamps. A number of extensions to the basic TSIG protocol have been made to extend the types of cryptography and hashing methods that are supported by TSIG.

To use TSIG for a DNS zone, add TSIG keys to the DNS zone. The TSIG key must be base64 encoded.

Using the Compute Web UI

1. In the navigation menu, under DNS Zones, click TSIG Keys.

2. Click Create Key.
3. Fill in the required TSIG Key information:
   - **Name:** Provide a name or description for the TSIG key. Avoid using any of the organization's confidential information.
   - **Compartment:** Select the compartment in which to create the TSIG key.
   - **Algorithm:** Choose the security algorithm for the TSIG Key you are creating, such as **hmac-sha256**.
   - **Secret Key:** Provide the base64 string encoding the binary shared secret that corresponds to the key. The maximum is 255 characters. An example key in base64 encoding is shown in RFC3874. You can provide the key in one of two ways:
     - **Select the key file**: If you provide the TSIG shared secret key this way, you can drag and drop the key file into the space provided.
     - **Paste the key**: If you provide the TSIG shared secret key this way, you can copy and paste the contents of the key file into the space provided.
   - **Tagging:** Optionally, add one or more tags to this resource. For more information about tagging, see Working with Resource Tags. If you are not sure whether to apply tags, skip this option (you can apply tags later) or ask your administrator.
4. Click Create TSIG Key.

   The TSIG key now available for use in the DNS zone between TSIG client and DNS server.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Compartment OCID (`oci iam compartment list --all`)
2. Run the `oci dns tsig-key create` command.

> **Note:**
>
> This procedure shows the minimum required parameters for this command. For information about optional parameters, run the command with the `--help` option.

Syntax (entered on a single line):

```
oci dns tsig-key create \
--algorithm <hmac-algorithm> \
--name <tsig-key-name> \
--compartment-id <compartment_OCID> \
--secret <secret-string>
```

Example:

```
oci dns tsig-key create --algorithm hmac-sha256 --name new-tsig-key \
--compartment-id ocid1.compartment.….….….uniqueID \
--secret 2o8goaon2168n(secret key string)e6um8lvd2lwdoouq46lsygak0009014
{
  "data": {
    "-self": "https://20180115/tsigKeys/new-tsig-key",
    "algorithm": "hmac-sha256",
```

```
    "compartment-id": "ocid1.compartment.….….….uniqueID",
    "defined-tags": {},
    "freeform-tags": {},
    "id": "ocid1.dns-tsig-key..….….….uniqueID",
    "lifecycle-state": "ACTIVE",
    "name": "new-tsig-key",
    "secret": "2o8goaon2168n(secret key string)e6um8lvd2lwdoouq46lsygak0009014",
    "time-created": "2021-10-29T17:50:31.219934+00:00",
    "time-updated": null
  },
  "etag": "81eb0e02-e09c-4b25-9d21-eefa9ab2aacc"
}
```

The new key appears in the list of TSIG keys for the DNS for this compartment.

Adding a TSIG Key

To add a TSIG key to an existing list of TSIG keys, simply create another key with a unique TSIG key name and a new algorithm or a new key value. To modify fields in an existing TSIG key, use the `update` command.

A TSIG key is a separate object from a DNS zone. You can have a `SECONDARY DNS` zone reference a TSIG key as part of its `ExternalMaster` definition. But creating a new key doesn't do anything for a `PRIMARY` zone.

Removing a TSIG Key

Using the Compute Web UI

1. In the navigation menu, under DNS Zones, click TSIG Keys.
2. Click on the TSIG key that you want to remove from the drop-down list of TSIG keys.
3. Click Delete from the list of actions under the Action Menu icon (three bars), or click the Delete button at the top of the display window.

   The TSIG key is removed from the list.

Using the OCI CLI

1. Gather the information for these resources:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
   - TSIG OCID (`oci dns tsig-key list --compartment-id` ***`<compartment_OCID>`***)
2. Syntax (entered on a single line):

   ```
   oci dns tsig-key delete --tsig-key-id <tsig-key_OCID>
   ```

   Example:

   ```
   oci dns tsig-key delete --tsig-key-id ocid1.dns.tsig.key.….….….uniqueID \
   Are you sure you want to delete this resource? [y/N]: y
   ```

The TSIG key is deleted from the list of TSIG keys for DNS in this compartment. Use the `--force` option to suppress the "Are you sure...?" message.

Deleting a Public DNS Zone

Using the Compute Web UI

1. In the navigation menu, under DNS Zones, click Zones.
2. Click on the zone name that you want to remove from the drop-down list of zones.
3. Click the Delete button at the top of the display window.

   The DNS zone is removed from the list.

   The DNS zone is deleted from the compartment. Use the `--force` option to suppress the "Are you sure...?" message.

Using the OCI CLI

1. Gather the information for these resources:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
   - Zone OCID (`oci dns zone list --compartment-id` ***<compartment_OCID>***)
2. Syntax (entered on a single line):

   ```
   oci dns zone delete --zone-name-or-id zone_OCID-or-zone-name>
   ```

   Example:

   ```
   oci dns zone delete --zone-id ocid1.dns.zone.….….….uniqueID \
   Are you sure you want to delete this resource? [y/N]: y
   ```

Managing Traffic with Steering Policies

DNS can do more than return an IP address (if known) when given a string in the DNS name space for that zone. DNS is also a part of a system of traffic management, where traffic is distributed among multiple servers depending on some criterion, such as location. Steering policies are a way to distribute access to a single full-qualified name across multiple servers.

For example, the same content could be available from multiple source servers, whether it is a streaming video or records from a product database. One server might be in the United States, and the other in Europe. A traffic steering policy could distribute traffic based on IP address or CIDR. Other criteria can be used for this traffic distribution, such as load balancing, which strives to keep the load on multiple servers roughly equal.

Oracle Private Cloud Appliance offers two major types of traffic steering policies based on load balancing and some value of the IP address prefix (network portion of the IP address, such as `192.168.100.0/24`).

Creating a Load Balancer Steering Policy

If you have more than one DNS server, you can distribute traffic in a load balancing fashion, based on the weight you assign to each of them.

Using the Compute Web UI

1. Open the Navigation Menu. Under DNS Zones, click Steering Policies.
2. Click Create Steering Policy.

3. Click the Load Balancer button to create a load balancer steering policy.
4. Enter the required information:
   - **Name:** Enter a name to display for the load balancer steering policy. Do not use confidential information.
   - **Policy TTL:** Enter a TTL in seconds for responses to steering policy requests. The maximum is 604800 seconds (equal to 168 hours or 7 days).
   - **Answer(s):** Supply the answer or answers to the DNS request for FILTER, WEIGHED, and LIMIT rules. You do not have to specify which condition the answers is for: that is all done by the load balancer template.
     - **Name:** Enter a name for the RData returned, such as Server1.
     - **Type:** Choose the type of resource record to return for the request from the drop-down list. Choices are items such as A (IPv4 address) or CNAME (canonical name).
     - **RData:** Enter the resource record RData that is returned that corresponds to the Type selected. For example, for Type = A, the RData would be an IPv4 address.
     - **Weight:** Enter a weight for this policy to use for load balancing. Values up to 256 are supported. The default is 10. Higher weights mean that policy answer is used more often. For example, if `dns-server1` and `dns-server2` have equal weights, DNS requests are split evenly between them. If `dns-server1` has a weight twice that of `dns-server2`, then `dns-server1` is used twice as often as `dns-server2`.
5. **Disabled:** The steering policy answer is enabled at creation by default. To disable this steering policy answer, click this toggle to change the Disabled value to TRUE.
6. Optionally, add or delete tags for this subnet resource.

   For more information about tagging, see Working with Resource Tags. If you are not sure whether to apply tags, skip this option (you can apply tags later) or ask your administrator.
7. Click Save Changes. The load balancing steering policy is created.

Using the OCI CLI

1. Gather the information you need.
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
2. Run the `oci dns steering-policy create` command with the LOAD_BALANCE parameter.

> **Note:**
>
> This procedure shows the minimum required parameters for this command. For information about optional parameters, run the command with the `--help` option. Complex types are long json strings.

Syntax (entered on a single line):***<dns_steering_policy_name>***

```
oci dns steering-policy create
--compartment-id <compartment_OCID>
--display-name <dns_steering_policy_name>
--template <LOAD_BALANCE>
```

```
--answers <complex type>
--rules <complex type>
```

Example:

```
oci dns steering-policy create
--compartment-id ocid1.compartment.….….….uniqueID
--display-name test-lb-policy-1
--template LOAD_BALANCE
--answers '[{"name": "server","pool": "server","rdata": "10.25.11.10","rtype": "A"},
{"name": "trial","pool": "trial","rdata": "10.25.11.10","rtype": "A"}]'
--rules '[{"ruleType": "FILTER","defaultAnswerData": [{"answerCondition":
"answer.isDisabled != true","shouldKeep": true}]},{"ruleType":
"WEIGHTED","defaultAnswerData":
[{"answerCondition": "answer.name == 'server'","value": 90},{"answerCondition":
"answer.name == 'trial'","value": 10}]},{"defaultCount": 1,"ruleType": "LIMIT"}]'

{
  "data": {
    "-self": "https://20180115/steeringPolicies/ocid1.dnspolicy..….….….uniqueID",
    "answers": [
      {
        "is-disabled": true,
        "name": "server",
        "pool": "server",
        "rdata": "10.25.11.10",
        "rtype": "A"
      },
      {
        "is-disabled": true,
        "name": "trial",
        "pool": "trial",
        "rdata": "10.25.11.10",
        "rtype": "A"
      }
    ],
    "compartment-id": "ocid1.compartment.….….….uniqueID",
    "defined-tags": {},
    "display-name": "lr-policy",
    "freeform-tags": {},
    "health-check-monitor-id": null,
    "id": "ocid1.dnspolicy..….….….uniqueID",
    "lifecycle-state": "ACTIVE",
    "rules": [
      {
        "cases": null,
        "default-answer-data": [
          {
            "answer-condition": "answer.isDisabled != true",
            "should-keep": true
          }
        ],
        "description": null,
        "rule-type": "FILTER"
      },
      {
        "cases": null,
        "default-answer-data": [
          {
            "answer-condition": "answer.name == 'server'",
            "value": 90
          },
          {
```

```
            "answer-condition": "answer.name == 'trial'",
            "value": 10
          }
        ],
        "description": null,
        "rule-type": "WEIGHTED"
      },
      {
        "cases": null,
        "default-count": 1,
        "description": null,
        "rule-type": "LIMIT"
      }
    ],
    "template": "LOAD_BALANCE",
    "time-created": "2021-11-03T23:36:25.392833+00:00",
    "ttl": 30
  },
  "etag": "2c63fca5-f747-487e-b2f3-0ae5d6fe939c"
}
```


The load balancer steering policy is created and available for attaching to a DNS domain.

Creating an IP Prefix Steering Policy

An IP prefix steering policy dynamically routes DNS request traffic to different servers based on the originating IP prefix (for example, `172.16.1.0/24`).

Using the Compute Web UI

1. Open the Navigation Menu. Under DNS Zones, click Manage DNS.
2. From the list of DNS resources, click Steering Policies. The steering policies for that compartment are displayed.
3. Click Create Steering Policy.
4. Select IP Prefix Steering and supply the following properties:
   - **Name:** The name for the new steering policy.
   - **Policy TTL:**The Time To Live (TTL) for responses from the steering policy, in seconds. The maximum allowed value is 604800 (equal to 168 hours or 7 days).
5. In the Answer(s) box, supply the following properties:
   - **Name:** The name for response to requests sent to the new steering policy.
   - **Type:** The type of request and response. The choices are A, AAA, or CNAME.
   - **RData:** The zone record data to return for the query. It must match the type expected by the type chosen.
   - **Pool:** Select the IP address pool to use of the policy from the drop-down list.
   - **+Add Answer:** Click this box to add more answers to the requests received by the steering policy.
   - **Disabled:** This toggle determines if the IP prefix answer is enabled at creation or not. The default is enabled.
6. In the IP Prefix Steering Rules box, supply the following properties:
   - **+Add Rule:** Click this box to add rules to the IP prefix steering policy.
   - **Order:** Use the directional arrows to order the rule in the sequence of configured rules.

- **Subnet Address:** Enter the IP subnet prefix to apply to this steering policy.
- You can add more rules to this steering policy by clicking **+Add Rule**.

7. **Tagging:** Optionally, you can add tags to the steering policy.

   For more information about tagging, see Working with Resource Tags. If you are not sure whether to apply tags, skip this option (you can apply tags later) or ask your administrator.

8. Click Save Changes. The IP prefix steering policy is created.

Using the OCI CLI

1. Gather the information you need.
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
2. Run the `oci dns steering-policy create` command with the ROUTE_BY_IP parameter.

> **Note:**
>
> This procedure shows the minimum required parameters for this command. For information about optional parameters, run the command with the `--help` option. Complex types are long json strings.

Syntax (entered on a single line):

```
oci dns steering-policy create--compartment-id <compartment_OCID>
--display-name <dns_steering_policy_name>
--template <LOAD_BALANCE>
--answers <complex type>
--rules <complex type>
```

Example:

```
oci dns steering-policy create --compartment-id ocid1.compartment..….….….uniqueID
--display-name test-ip-steering-1
--template ROUTE_BY_IP
--answers file:///root/users-stuff/ip-steering-answers.json
--rules file:///root/users-stuff/ip-steering-rules-2.json
{
  "data": {
    "-self": "https://20180115/steeringPolicies/ocid1.dnspolicy..….….….uniqueID",
    "answers": [
      {
        "is-disabled": null,
        "name": "server",
        "pool": "server",
        "rdata": "10.20.10.10",
        "rtype": "A"
      },
      {
        "is-disabled": null,
        "name": "trial",
        "pool": "trial",
        "rdata": "10.20.10.10",
        "rtype": "A"
      }
    ],
    "compartment-id": "ocid1.compartment..….….….uniqueID",
```

```
    "defined-tags": {},
    "display-name": "test-ip-steering-1",
    "freeform-tags": {},
    "health-check-monitor-id": null,
    "id": "ocid1.dnspolicy..….….….uniqueID",
    "lifecycle-state": "ACTIVE",
    "rules": [
      {
        "cases": null,
        "default-answer-data": [
          {
            "answer-condition": "answer.isDisabled != true",
            "should-keep": true
          }
        ],
        "description": null,
        "rule-type": "FILTER"
      },
      {
        "cases": [
          {
            "answer-data": [
              {
                "answer-condition": "answer.pool == 'internal'",
                "value": 1
              }
            ],
            "case-condition": "query.client.address in (subnet '10.0.3.0/24')"
          },
          {
            "answer-data": [
              {
                "answer-condition": "answer.pool == 'external'",
                "value": 1
              }
            ],
            "case-condition": null
          }
        ],
        "default-answer-data": null,
        "description": null,
        "rule-type": "PRIORITY"
      },
      {
        "cases": null,
        "default-count": 1,
        "description": null,
        "rule-type": "LIMIT"
      }
    ],
    "template": "ROUTE_BY_IP",
    "time-created": "2021-11-09T16:53:34.963177+00:00",
    "ttl": 30
  },
  "etag": "aad5bbcc-9d89-40cd-ab10-03dcc2e4ee0a"
}
```


The IP steering policy is created and available to attach to a DNS domain.

Editing a Steering Policy

Using the Compute Web UI

1. Open the Navigation Menu. Click DNS, and then click Steering Policies.
2. For the policy that you want to update, click the Actions menu, and click the Edit option.
3. Make the necessary changes on the Edit Steering Policy dialog.
4. When you have finished making changes, click the Save Changes button on the dialog.

   The details page for this steering policy is displayed with the updated information.

Using the OCI CLI

1. Get the steering policy OCID.

   Use the following command to list all of the steering policies in the specified compartment to get the OCID of the steering policy that you want to update:

   ```
   # oci dns steering-policy list --compartment-id <compartment_OCID>
   ```

2. Run the steering policy update command.

   Syntax:

   ```
   oci dns steering-policy update --steering-policy-id <steering_policy_OCID> \
   <options_with_values_to_update>
   ```

   Example:

   This example shows replacing the `answers` block of the steering policy. You can also change the display name, health check monitor, rules or rules template, TTL, and scope.

   ```
   # oci dns steering-policy update --steering-policy-id ocid1.dnspolicy.unique_ID \
   --answers file://answers.json
   ```

   This command returns the same output as the `steering-policy get` command.

Moving a Steering Policy to a Different Compartment

Using the OCI CLI

1. Get the following information:
   - The OCID of the compartment where the steering policy is currently located, and the OCID of the compartment where you want to move the steering policy.

     ```
     # oci iam compartment list --compartment-id-in-subtree true
     ```

   - The steering policy OCID.

     ```
     # oci dns steering-policy list --compartment-id <current_compartment_OCID>
     ```

2. Run the steering policy update command.

   Syntax:

   ```
   oci dns steering-policy change-compartment -c <destination_compartment_OCID> \
   --steering-policy-id <steering_policy_OCID>
   ```

   This command returns the same output as the `steering-policy get` command. Verify the new `compartment-id`.

Attaching a Domain to a Steering Policy

A steering policy must be attached to a domain for the policy to answer DNS queries for that domain. The attachment is automatically placed into the same compartment as the domain's zone.

Using the Compute Web UI

1. Open the Navigation Menu. Click DNS, and then click Steering Policies.
2. Click the name of the policy to which you want attach a domain.
3. Scroll to the Resources section and click Attached Domains.
4. In the list of attached domains, click the Add Attached Domain button.
5. In the Add Attached Domain dialog, enter the domain name and select a zone.
6. Click the Submit button.

   The new domain is added to the Attached Domains list for this steering policy.

Using the OCI CLI

1. Get the following information:
   - The steering policy OCID. Use the following command to list all of the steering policies in the specified compartment to get the OCID of the steering policy to which you want to attach a domain:

     ```
     # oci dns steering-policy list --compartment-id <current_compartment_OCID>
     ```
   - The name of the domain that you want to attach to the steering policy.
   - The OCID of the attached zone. Use the following command to list all of the zones in the specified compartment to get the OCID of the zone where the domain that you want to attach is located:

     ```
     # oci dns zone list <compartment_OCID>
     ```
2. Run the steering policy attachment create command.

   Syntax:

   ```
   oci dns steering-policy-attachment create --steering-policy-id
   <steering_policy_OCID> \
   --domain-name <domain-name> --zone-id <zone_OCID>
   ```

   The value of the `--domain-name` argument is the attached domain within the attached zone specified in the `--zone-id` argument.

   This command returns the same output as the `steering-policy-attachment get` command.

Editing an Attached Domain

Using the Compute Web UI

1. Open the Navigation Menu. Click DNS, and then click Steering Policies.
2. Click the name of the policy for which you want to edit an attached domain.
3. Scroll to the Resources section and click Attached Domains.

4. Click the name of the attached domain that you want to edit.
5. On the top of the details page for the attached domain, click the Edit button.
6. Make the necessary changes on the Edit Steering Policy Attachment dialog.
7. When you have finished making changes, click the Save Changes button on the dialog.

   The details page for this steering policy attachment is displayed with the updated information.

Using the OCI CLI

1. Get the steering policy attachment OCID.

   Use the following command to list all of the steering policy attachments in the specified compartment to get the OCID of the steering policy attachment that you want to update:

   ```
   # oci dns steering-policy-attachment list --compartment-id <compartment_OCID>
   ```

2. Run the steering policy attachment update command.

   Syntax:

   ```
   oci dns steering-policy-attachment update \
   --steering-policy-attachment-id <steering_policy_attachment_OCID>
   ```

   This command returns the same output as the `steering-policy-attachment get` command.

Deleting a Steering Policy Attachment

Using the Compute Web UI

1. Open the Navigation Menu. Click DNS, and then click Steering Policies.
2. Click the name of the policy for which you want to delete an attachment.
3. Scroll to the Resources section and click Attached Domains.
4. For the attached domain that you want to delete, click the Actions menu, click the Delete option, and confirm the deletion.

   The steering policy attachment is removed from the Attached Domains list.

Using the OCI CLI

1. Get the steering policy attachment OCID.

   Use the following command to list all of the steering policy attachments in the specified compartment to get the OCID of the steering policy attachment that you want to delete:

   ```
   # oci dns steering-policy-attachment list --compartment-id <compartment_OCID>
   ```

2. Run the steering policy attachment delete command.

   Syntax:

   ```
   oci dns steering-policy-attachment delete \
   --steering-policy-attachment-id <steering_policy_attachment_OCID>
   ```

Deleting a Steering Policy

A policy that is attached to any zones cannot be deleted. To detach a policy from a zone, see Deleting a Steering Policy Attachment.

Using the Compute Web UI

1. Open the Navigation Menu. Click DNS, and then click Steering Policies.
2. Click the name of the policy that you want to delete.
3. Scroll to the Resources section, click Attached Domains, and ensure that this policy has no attached domains.
4. Click the Delete button at the top of the steering policy details page, and confirm that you want to delete this steering policy.

   The steering policies list page is displayed.

Using the OCI CLI

1. Get the steering policy OCID.

   Use the following command to list all of the steering policies in the specified compartment to get the OCID of the steering policy that you want to delete:

   ```
   # oci dns steering-policy list --compartment-id compartment_OCID
   ```

2. Run the steering policy delete command.

   Syntax:

   ```
   oci dns steering-policy delete --steering-policy-id steering_policy_OCID
   ```

Networking Scenarios

All networking scenarios for a virtualized cloud environment are similar to scenarios for individual IP subnets connected by physical switches, routers, and gateways. In other words, virtual devices still have MAC (hardware) addresses as source and destination addresses in frames and IP addresses as source and destination addresses in packets.

Content delivery works essentially the same way as well. If the network portion of the source and destination IP addresses are in the same defined VCN subnet, delivery is through a logical switch (bridge) based on source and destination MAC frame addresses. If the network portion of the source and destination IP addresses are in different VCN subnets, then delivery is based on source and destination IP packet addresses.

You do not have to configure a logical switch for Oracle Private Cloud Appliance. The MAC addresses are known to all the other entities connected to the logical switch. It is assumed that if you place two or more VMs in the same VCN subnet, it is okay if they communicate. If instance isolation is the goal, then establish separate subnets or VCNs for them.

Logical Routers

Traffic between different VCN subnets is handled by at least one logical router, by definition. Devices that use IP packet addresses to determine forwarding steps, while using different MAC frame addresses on the different subnets they link, are called routers. In the case where the routers attach to the internet, these virtual devices are internet gateways (IGWs).

In cases where the source or destination IP address rules include IP network address translation (NAT), the packets are handled by a NAT gateway of some type (there are several types of NAT gateways). If some form of NAT is used, these are NAT gateways (NATGW or NGW).

In many cases of virtualized cloud networking, routing is very simple and can be handled by a small, static routing table that has a handful of destinations. More complex virtual environments require more complex logical routers, containing dynamic information that is updated periodically.

Logical routers inspect the packet's IP addresses. The IP addresses, source and destination, are looked up in a route table, called the *route rule table* in the Oracle Private Cloud Appliance, and forwards the packet to the next hop (another router) or destination (for local delivery) if the IP address rule allows this. If the route rules do not apply to the IP addresses, the packet is silently dropped and does not generate an error message (this is a security feature to prevent blocked probes from gathering information). However, this lack of error messages means that the route rules must be configured very carefully.

One IP address is special when it comes to route rules. This is the IPv4 address `0.0.0.0/0`, which essentially matches any IP address at all. In some documents, the `0.0.0.0/0` notation is called an IP address CIDR block, but the same universal matching is true no matter what it is called.

For example, the following route rule allows a packet sent from inside the VCN to any IP address at all to reach a gateway to the internet:

```
Destination                                                    Target
0.0.0.0/0                                  Internet Gateway    vcn-20210714-0910
```

Route rules do more than determine the destination of a packet. Route rules also form the basis for network firewalls.

Using Firewalls

Internet access is convenient, but brings concerns over vulnerability and security. Firewalls exist to limit the free passage of traffic between network elements and secure the network. A firewall, logical or physical, examines the traffic flow from a particular source to a particular destination and permits or blocks the packets based on the configured security rules in the route rule table.

Firewalls should be configured not only to allow or block traffic from or to external sources, but should also be configured to validate the traffic passing from subnet to subnet within the same VCN. Threats could be coming from external sources, but also from compromised instances within the network.

Use of Network Segmentation

It is tempting to configure a virtual network as one big entity, with everything easily reachable by everything else. But this makes it relatively easy for attackers to compromise the network: once they are in, they are in everywhere. It is much better to use segmentation for the network and group resources and data into the various segments.

In the Oracle Private Cloud Appliance, segments are essentially network security groups.

Typically, you group data and resources based on similarity or data sensitivity. For example, you can establish a group that examines all traffic received from the data center. Based on your security rules, this traffic can then be passed to a group of application servers, and then onto the database servers.

With this approach, firewalls between the groups secure the application and database servers from any compromised components of the data center.

Use of Tunneling

One complication of virtualized cloud networking is that there is no central authority to assign IP addresses to VCN subnets. Nothing stops one hypervisor from assigning, for example, IP address `192.168.1.6` to VM-1 in Subnet-1 of VCN-1, while another hypervisor assigns the same IP address `192.168.1.6` to VM-7 in Subnet-1 of VCN-1. Yet, if various tables are configured correctly, they can still communicate.

In order to effectively hide these network address complications, the Oracle Private Cloud Appliance moves traffic between network components such as logical routers through IP tunnels. However, this use of tunnels, a common IP network practice, does not change the network scenarios. Instances still have IP addresses, and these addresses are still assigned to a particular subnet and the subnets are assigned to virtual network interface cards (VNICs). All traffic from the various running instances under the same a hypervisor is bonded together into an IP tunnel for transport to the next device between source and destination.

Use of Virtual Cloud Networks

It is easy to say that resources are gathered into one or more VCNs, which are private cloud networks running in a tenancy. But there is a lot more to VCNs than declaring a group of resources and creating a boundary for a VCN.

Planning your VCNs is a critical part of any deployment. VCNs serve as a foundation to structure your application servers, databases, and any other services provided. VCNs should take into account any needs such as redundancy, high availability, scalability, security, and more.

This section details the critical parts of a VCN.

IP Address Ranges

When planning VCNs, the first decision to make is which IP address CIDR block to use.

> **Note:**
>
> To help with the calculation of CIDR blocks, a good resource is: IP Address Guide for CIDR

The VCN network address range should be any VLSM between /16 and /30. This covers virtual networks for between 4 available IP addresses (/30) to 65,536 available IP addresses (/16), although the highest and lowest IP address are not useful for endpoint devices.

The size of the CIDR block chosen for the VCN is of critical importance. If the size is too large, then IP addresses are wasted that could be used in other places in the network. If the size is too small, the solution does not scale because there are not enough IP addresses for the VCN. You can always create another VCN and peer them together, but this is a complication that can be avoided through careful planning.

There are some important points about VCN CIDR blocks:

- VCNs should use one of the RFC1918 private address ranges: `10.0.0.0/8`, `172.16.0.0/12`, and `192.168.0.0/16`.
- VCNs use a contiguous IPv4 CIDR block. IPv6 is not supported.

- You cannot change the VCN and subnet sizes (such as `10.100.0.0/21`) after their creation. The IP address range must not overlap with any VCN you want to peer with.
- Subnets within a VCN must not overlap with each other.
- Remember that the highest and lowest IP addresses in a range are reserved for network functions.

IP Subnets

A subnet is a subdivision of a VCN that uses a continuous IP address range as established by CIDR. Subnets group resources by IP address.

From the VCN perspective, subnets can be public or private. In the context of Oracle Private Cloud Appliance, private VCN addresses must be changed using NAT before they can interact with other VCNs that might be using the same private address space. Public VCN addresses allow for connectivity to the data center network and are accessible from outside the rack.

It is, however, possible to have an RFC1918 address space subnetted into both public IP address spaces (which allows external data center connectivity) and private IP address spaces (which connect within the Oracle Private Cloud Appliance rack).

For example, it is possible for a VCN to do this with an IP CIDR block:

```
Subnet Name                    Subnet Access                  IP
CIDR Block

Public_Subnet_01               Public
10.100.0.0/24
Private_Subnet_01              Private
10.100.1.0/24
```

`Public_Subnet_01` has 256 IP addresses, from `10.100.0.0` to `10.100.0.255` (these two addresses are not often used for devices and have special uses in most IP networks). The netmask is `255.255.255.0`

Private_Subnet_01 also has 256 IP addresses, from `10.100.1.0` to `10.100.1.255` (again, the low and high addresses are not often useful for devices). The netmask is `255.255.255.0`.

Note that the two address ranges do not overlap. (If the public range was `10.100.0.0/23`, then 256 devices would overlap with `10.100.1.0/24`.)

Route Tables

A VCN uses a route table to hold the rules governing the sending and receiving of traffic by an instance. A VCN could be allowed or prevented form reaching destinations like:

- The global public internet
- An on-premise network, such as the date center network
- A peer VCN

Whenever a VCN is created, a virtual router with a default route table is also created.

For better security, it is preferable to create a dedicated route table for every subnet rather than use the default rules, which might not be adequate for the level of security needed. Dedicated route tables can more effectively manage the route rules that each subnet needs. For example, if the subnet is allowed to send traffic out onto the global public internet, then the route table for that subnet needs a rule for routing traffic to an internet gateway with the "universal destination" `0.0.0.0/0`.

> **Note:**
>
> The CIDR block `0.0.0.0/0` matches all addresses in the IPv4 address space. It means "any IPv4 address with any subnet mask."

A subnet not needing global public network access should not include that rule in the route table. In addition:

- Traffic with a source and destination within a VCN is not governed by any route table rules.
- If rules overlap (usually regarding different CIDR block or subnets), then the more specific rule applies (often seen as "the longest match").
- If there is no rule that applies to the traffic flow, then the traffic is silently dropped (no error message sent).

Every rule in the route table has a target. The targets are the functional network nodes for the common components of any network:

- Dynamic Routing Gateway (DRG: the route table rules are not statically configured, but use a routing protocol such as BGP to change them).
- Internet Gateway (IG) to connect to the global public internet.
- NAT gateway (NATG) for IP address translation.
- Service Gateway (SG) to reach a variety of services in other subnets.
- Local Peering Gateway (LPG) to connect to peer VCNs.
- Private IP address, which routes traffic to a specific instance inside the VCN.

As an example, a VCN could contain a route table with the following rules:

```
Subnet Name                          Route Table                          Target

Public_Subnet_01                     Public_Subnet_01_Route_Table         IG
Private_Subnet_01                    Private_Subnet_01_Route_Table        NAT
Gateway
```

Route tables are the foundation of VCN security.

Security Lists and Network Security Groups

It might seem odd that an on-premises network needs firewall-like security. But security is important in every context, and not all threats come from outside an organization.

Oracle Private Cloud Appliance offers two security networking mechanisms that act like firewalls to control traffic at the packet level:

- Security lists, which are used like physical firewalls for subnets.
- Network security groups (NSGs), which act as firewalls for groups of instances across subnets.

You use a security list to define the rules that apply to all inbound (ingress) and outbound (egress) traffic of a subnet. You can associate up to five security lists per subnet. In the same way as route tables, there are default security lists and dedicated security lists. For better control and management, you should always use dedicated security lists for each subnet.

In contrast to security lists, NSGs let you build rules for groups of instances, even if the instances are in different subnets. For example, the same NSG can apply to all the database servers, or all the application servers running a certain application. Instead of applying security to a particular subnet, you create an NSG and then add the appropriate instances to the NSG.

There is no requirement to use either security lists or NSGs. You can use security lists without establishing NSGs, or create NSGs without creating any security lists. However, if you use both security lists and NSGs, the rules that apply to a VNIC are the union (both sets) of the rules that are in the security list for the VNIC and the rules specific to that VNIC from the NSG.

When creating a VCN, a default security list with three ingress rules and one egress rule is created. The default rules are stateful, which means that they know what connections are and that a request is followed by a response and that the rules should take this into consideration. In contrast, stateless rules are applied to every packet regardless of situation.

The default ingress and egress security rules look like this if you display them:. First, the ingress rules (for a `10.0.0.0/16` CIDR block):

```
Stateless Source       Source Type  IP Protocol Source Port Range Destination Port
Range  Type and

 Code

No        10.0.0.0/16  CIDR Block
ICMP                                              3
No        0.0.0.0/0    CIDR Block
ICMP                                              3, 4
No        0.0.0.0/0    CIDR Block   TCP         22 - 22            22 -
22
```

These rules, line by line, can be read as:

- There is a stateful rule that apples to traffic originating from the VCN `10.0.0.0/16` CIDR block using the ICMP protocol Type = 3 message format (Destination Unreachable) and all Codes. In other words, this rule allows devices in the `10.0.0.0/16` CIDR block to pass Destination Unreachable messages with any code (such as Net or Host Unreachable) back to the sender (another instance in the VCN subnet).
- There is a stateful rule that apples to traffic originating from any IP address (`0.0.0.0/0` CIDR block) using the ICMP protocol Type = 3 message format (Destination Unreachable) and a Code = 4 (message must be fragmented, but the Do Not Fragment (DF) bit is set in the packet: these are Path MTU Discovery messages). In other words, this rule allows devices to notify sources that the DF bit is set on packets that need to be fragmented because the content exceeds the MTU size established for this VCN subnet.
- There is a stateful rule that apples to traffic originating from any IP address (`0.0.0.0/0` CIDR block) using the connection-oriented TCP protocol and with the Source or Destination Port = 22 (SSH). In other words, this rule allows SSH for this VCN subnet.

This last rule makes allows you to create a new VCN and subnet, launch a Linux instance, and then use SSH to connect to that instance without writing any new security list rules.

> **Important:**
>
> The default ingress security list does not include a rule to allow Remote Desktop Protocol (RDP) access. If you're using Windows images, make sure to add a stateful ingress rule for TCP traffic on destination port 3389 from authorized source IP addresses and any source port. See To enable RDP access for more information.

The single egress rule is very simple:

```
Stateless Source       Source Type  IP Protocol Source Port Range Destination Port
Range  Type and

 Code

No        0.0.0.0/0    CIDR Block   All
```

This can be read as: "There is a stateful rule that allows packets with any source address and for any IP protocol at all to leave the VCN." This basically says that anything can leave the VCN subnet without a problem.

The default security list comes with no stateless rules. However, you can always add or remove rules from the default security list.

To use these default rules in an NSG, let's give the default ingress and egress rules distinctive names, such as `Ingress_Security_List_Subnet01` and `Egress_Security_List_Subnet01`. Before you can add these rules to an NSG, you must first create the NSG. To create an NSG, you must give it a name and assign it to an existing compartment. You can also add tags, but these can be added later.

Network Gateway Example: Internet Gateway

VCNs are the basic networking unit of the Oracle Private Cloud Appliance, and can communicate with other processes through various types of gateways used for a particular purpose.

In this example, access to a VCN from outside the rack is established through an Internet gateway (IGW). All steps to produce a working IGW are detailed.

This example sets up an IGW to allow access to web servers on a public IP subnet running inside an instance. It also adds an ingress rule to the default security list to allow outside access to the web servers on the public subnet. Then this example allows ingress connections for HTTPS connections on TCP port 443, the standard port for HTTP encrypted traffic.

Without this ingress rule, inbound HTTPS connections are not allowed. You should make the new rule `stateful`, which is the default and allows a reply to an HTTPS request without creating an explicit rule for responses.

Next, this example adds the existing IGW target to the route rules. The route rules are added to the default route table for the VCN, or a new route table created for reaching the IGW specifically. The route rule uses CIDR block 0.0.0.0/0. This means that all traffic not already covered by other rules in the route table goes to the IGW target specified in this new rule.

This example includes steps to enable or disable the IGW, and how to delete it.

Overview

There are three major operations for establishing and using an IGW. Each step has its own set of prerequisites and can usually be configured with the Compute Web UI or the OCI CLI. If both methods are available, both methods are presented.

The three activities used to configure and operate an IGW are:

- Set up an IGW
- Create or update a route table to include a rule for an IGW
- Update the Security List or NSG

Set Up an Internet Gateway

The first thing that needs to be done is to configure the IGW.

Basic Internet Gateway Configuration

There are certain considerations that need to be assessed before setting up and using an IGW:

- There are public subnets in the VCN that need internet access. (Only public subnets can use the IGW successfully.)
- Because the default condition is to deny access, the types of ingress and egress internet traffic that are allowed must have been determined. These include ingress HTTPS connections, ingress ICMP pings, or other types of traffic. The IGW primarily responds to ingress network protocol requests.

For the basic configuration of an IGW, see Providing Public Access through an Internet Gateway. You must perform this initial configuration before proceeding.

Once the IGW has been created, there are two additional steps needed to make the IGW work properly with route tables and security list for the VCN or instance port Network Security Groups (NSGs). First, route table entries for the subnet must be configured to direct authorized traffic to the proper gateway destination. Second, the VCN containing to IGW must have the correct security rules to prevent unauthorized access and yet allow users to access resources they need.

Establish the Route Table Entries

Each public subnet that needs to use the internet gateway needs to update the subnet's route table entry to include a route to the IGW.

Each route rule specifies a destination CIDR block and the target (the next hop) for any traffic that matches that CIDR. Before you can create a rule, you must create a target for the rule, in this case, an IGW.

This example adds the existing IGW target to the route rules of the default route table for the VCN. You can also create a new route table for reaching the IGW specifically, but that is not done here. The route rule uses CIDR block 0.0.0.0/0 so that all traffic not covered by other rules in the route table goes to the IGW target specified in this new rule.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the name of the VCN for which you want to create a route table. The VCN details page is displayed.
3. Under Resources, click Route Tables.
4. Go to the details page of the Default Route Table and click the Add Route Rule button.
5. Click +New Rule, and enter the following information for this example:
   - **Target Type:** Select `Internet Gateway` from the list.
   - **CIDR Block**: Enter 0.0.0.0/0 as the destination CIDR block for the traffic.
   - **Target:** The target is the IGW. Click the arrow and select the target IGW. You might need to change the compartment just above the arrow.

- **Description:** An optional description of the rule, such as "New rule for IGW."

6. Click the Create Route Table Rule button in the dialog.

   The details page of the edited default route table is displayed. Because the subnet was set up to use the default route table, the resources in the subnet can now use the internet gateway.

Using the OCI CLI

1. Gather the information you need to run the command:
   - The OCID of the compartment where you want to create this route table (`oci iam compartment list`)
   - The OCID of the VCN for this route table (`oci network vcn list --compartment-id` ***`compartment_OCID`***)
2. Construct an argument for the `--route-rules` option.

   Route rules are in JSON format. To see how to format a rule, use the following command:

   ```
   oci network route-table update --generate-param-json-input route-rules >
   route_rule_format.json
   ```

   Example (put the following content into the `IGW_route_rule.json` file):

   ```
   [
     {
       "cidr-block": "0.0.0.0/0",
       "description": null,
       "destination": null,
       "destination-type": "CIDR_BLOCK",
       "network-entity-id": "ocid1.internetgateway.unique_ID"
    }
   ]
   ```

3. Run the route table update command.

   Syntax:

   ```
   oci network route-table update --compartment-id compartment_OCID \
   --vcn-id vcn_OCID --route-rules file:///home/flast/IGW_route_rule.json
   ```

While the new route table is still provisioning, the `route-rules` property might be empty. To confirm the options, use the OCID in the `id` property of the `create` output to run a `get` command:

```
oci network route-table get --rt-id ocid1.routetable.unique_ID
```

Once the route table rule has been added, the IGW is now reachable from the subnet and VCN.

Establish the Internet Gateway Security Rules

Once the IGW has been created, the correct security setting must be established to prevent unauthorized access to the gateway. For example, all outside HTTPS access should only be allowed to access port 443 which is the default port of secure web page access. Without this explicit rule, the standard port is not reachable.

This section uses security lists to accomplish this goal, but a similar result can be achieved using security rules in a Network Security Group (NSG), which is what Oracle recommends.

For more information about security lists and NSGs, see the Virtual Networking Overview.

> **Important:**
>
> If you have configured the public subnet to use the default security list, remember that the default includes several rules to enable basic access, such as ingress SSH and egress access to all destinations. Oracle recommends that you become familiar with this basic access set of rules. If you do not use the default security list, make sure that basic access is still provided either in the customized security rules or in an NSG containing those modified rules.

This example adds an ingress rule to the default security list to allow ingress connections for HTTPS connections on TCP port 443, the standard port for HTTP encrypted traffic.

Without this ingress rule, inbound HTTPS connections are not allowed. You should make the new rule `stateful`, which allows a reply to an HTTPS request without creating an explicit rule for responses.

For information about creating a new security list instead of modifying the default or adding a rule to an existing security list, see Creating a Security List.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Virtual Cloud Networks.
2. Click the name of the VCN for which you want to add the rule to a security list. The VCN details page is displayed.
3. Under Resources, click Security Lists.
4. For the security list that you want to add the rule, click the Actions menu and then click Edit to open the Edit Security List dialog. Update rules in the Allow Rules for Ingress and Allow Rules for Egress sections.
5. To add a new rule, in the Allow Rules sections, click the +New Rule button. You can also update the security list name and tags.
6. When you are done, click the Save Changes button on the dialog.

For the HTTPS example using TCP port 443 ingress rule, enter the following information:

- **Stateless:** To allow for a response to the incoming HTTPS request, the new rule should be stateful. Make sure that the stateless box is unchecked. For more information about stateless and stateful rules, see "Security Lists" in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*.
- **CIDR**: The CIDR block for the example 0.0.0.0/0, which applies the rule to all IP source addresses.
- **IP Protocol:** Select the TCP protocol from the drop-down list.
- **Port Range**:
  - Source Port Range: Leave blank.
  - Destination Port Range: Enter 443.
- **Description:** An optional description of the rule, such as "Allow stateful traffic for HTTPS on TCP port 443."

Click Save Changes to save your new rule. You can always edit the new rule at any time.

Using the OCI CLI

1. Get the OCID of the default security list of the VCN that you want to update (`oci network vcn list --compartment-id` ***`compartment_OCID`***)
2. To update rules, construct arguments for the `--ingress-security-rules` and `--egress-security-rules` options as described in Creating a Security List. Arguments that you provide to these rules options overwrite any existing rules. If you want to keep some existing rules, use the following command to show the current rules, and then copy the rules that you want to keep into the new option arguments.

   ```
   $ oci network security-list get --security-list-id ocid1.securitylist.unique_ID
   ```

   Example (put the following content in the file `IGW_ingress_rule.json`):

   ```
   [
     {
       "description": null,
       "icmp-options": null,
       "is-stateless": false,
       "protocol": "6",
       "source": "0.0.0.0/0",
       "source-type": "CIDR_BLOCK",
       "tcp-options": {
         "destination-port-range": {
           "max": 22,
           "min": 22
         },
         "source-port-range": null
       },
       "udp-options": null
     },
     {
       "description": null,
       "icmp-options": null,
       "is-stateless": false,
       "protocol": "6",
       "source": "0.0.0.0/0",
       "source-type": "CIDR_BLOCK",
       "tcp-options": {
         "destination-port-range": {
           "max": 443,
           "min": 443
         },
         "source-port-range": null
       },
       "udp-options": null
     }
   ]
   ```

3. Run the security list update command to add the rule for HTTPS and TCP port 443 traffic.

   Example:

   ```
   oci network security-list update  \
   --security-list-id ocid1.securitylist.unique_ID \
   --ingress-security-rules file:///home/flast/IGW_ingress_rule.json

   WARNING: Updates to defined-tags and egress-security-rules and freeform-tags and
    ingress-security-rules will replace any existing values.
    Are you sure you want to continue? [y/N]: y
   ```

5

Load Balancing

Load balancing is the method of sharing a workload equally among servers. It prevents clients from overwhelming certain servers.

The Load Balancer service provides automated traffic distribution from one entry point to multiple servers reachable from your virtual cloud network (VCN). The service offers a load balancer with your choice of a public or private IP address.

There are two types of load balancers:

- Load Balancer as a Service (LBaaS): This type of load balancer operates at all protocol layers, including the application. When the term "load balancer" (LB) appears without qualification, the statement refers to LBaaS.
- Network Load Balancer (NLB): This type of load balancer operates on protocol layers below the application itself, at the Network Layer. The term "network load balancer" (NLB) always refers to a network load balancer, not to LBaaS.

The verb "load balancing" refers to the actions of both LBs and NLBs. The term "load balancer" can refer to both LBs and NLBs. When you need to be specific, use LB and NLB.

A load balancer (both LBs and NLBs) can be either private or public.

- Private: A private load balancer is isolated from the network outside the Oracle Private Cloud Appliance. A private load balancer is assigned a private IP address from the address block of the specified private subnet. This private IP address is used as a front end for incoming internal VCN traffic, to balance that traffic across all backend servers.

  For a private LB you need a VCN with at least one private subnet. The subnet must have security rules that allow the intended traffic. The backend servers must be reachable from the selected VCN.
- Public: A public load balancer accepts traffic from a network location outside of the appliance. A public load balancer can be assigned a public IP address from a public subnet of a VCN that has a NAT gateway and an internet gateway (IGW) configured, or you can select the public IP address from a list. This public IP address is used as the entry point for incoming traffic, to balance that traffic across all backend servers. You can associate the public IP address with a friendly DNS name through any DNS provider.

  For a public LB, you need a VCN with at least one public subnet. The subnet must have security rules that allow the intended traffic. The backend servers must be reachable from the selected VCN.

  You can select a public IP address from a list, or you can allow the system to assign an IP address.

  For a public LB, create a NAT gateway as described in Enabling Public Connections through a NAT Gateway.

Load Balancer as a Service

LBaaS on the Oracle Private Cloud Appliance provides automated traffic distribution from one entry point to multiple servers reachable from the virtual cloud network (VCN). The service

implements either a private or public load balancer (LB), and supports provisioned bandwidth and various load balancing policies.

For more general information about LBaaS, see the Load Balancing Overview chapter in the *Oracle Private Cloud Appliance Concepts Guide*.

Managing a Load Balancer

This section describes how to create, view details, update, and delete a load balancer (LB).

- Creating a Load Balancer
- Viewing Load Balancer Details
- Editing a Load Balancer
- Deleting a Load Balancer

Creating a Load Balancer

This topic describes how to create a load balancer (LB).

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the Create Load Balancer button to open the Create Load Balancer dialog.
3. Enter the following information:
   - **Name:** Enter a descriptive name for the LB. The name does not need to be unique, and you can change it.
   - **Create in Compartment:** Select the compartment in which to create the LB. The LB does not have to be in the same compartment as the VCN or backend set. If you aren't sure which compartment to use, create the LB in the same compartment as the VCN.
   - Choose **Visibility Type:**
     - **Public Load Balancer**. The Select Public IP menu is shown. Select a public IP from the list. You might need to change the compartment above the menu. If the menu displays None Available or if you do not select a public IP from the list, a public IP is automatically assigned from the configured public IP range. You can use the assigned public IP address as a front end for incoming traffic.
     - **Private Load Balancer**. The LB receives a private IP address from the selected subnet. You can use the assigned private IP address as a front end for internal incoming VCN traffic.

     See Load Balancing for more information about private and public load balancers.
   - **Subnet:** Select the names of the VCN and Subnet for the LB. You might need to change the compartment above the menus.
   - **Network Security Group:** (Optional) By default, the LB is not attached to any NSG. Select the box labeled Enable Network Security Group to add this LB to one or more NSGs.
     a. Select an NSG from the drop-down list. You might need to change the compartment to find the NSG that you want.
     b. Click the Add Another NSG button if you want to attach to another NSG.

c. To remove an NSG from the list, click the trash can to the right of that NSG. To remove the last NSG or all NSGs, uncheck the Enable Network Security Groups box.

- **Tagging:** (Optional) Add defined or free-form tags for this LB as described in Adding Tags at Resource Creation. Tags can also be applied later.

4. Select the Create Load Balancer button in the dialog. The details page of the new LB is displayed.

Next Steps: On the LB details page, scroll down to the Resources section and select resources to create to complete the configuration.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Compartment OCID: `oci iam compartment list`
   - At least one subnet OCID: `oci network subnet list`
2. Run the create LB command.

   Syntax:

   The following shows only the required parameters. Use the `-h` option to get information about optional parameters such as backend sets and listeners.

   ```
   oci lb load-balancer create --compartment-id compartment_OCID \
   --display-name load-balancer-name --shape-name 400Mbps \
   --subnet-ids file://subnet_OCIDs.json
   ```

   Example:

   The following example creates a private LB with a fixed bandwidth of 400 Mbps.

   The `--is-private` option value is `false` by default. If `--is-private` is omitted, a public IP address is assigned from one of the specified subnets if available. If `--is-private true` is specified, a private IP address is assigned from one of the specified subnets. See Load Balancing for more information about private and public load balancers.

   ```
   --shape-details '{"maximumBandwidthInMbps": 400, "minimumBandwidthInMbps": 400}'
   ```

   The bandwidth cannot be changed after the LB is created.

   ```
   $ oci lb load-balancer create --compartment-id ocid1.compartment.unique_ID \
   --display-name Private_LB1 --shape-name 400Mbps \
   --subnet-ids '["ocid1.subnet.unique_ID1", "ocid1.subnet.unique_ID2"]'

   {
     "data": {
       "backend-sets": {},
       "certificates": {},
       "compartment-id": "ocid1.compartment.unique_ID",
       "defined-tags": {
         "Oracle-Tags": {
           "CreatedBy": "auser",
           "CreatedOn": "2025-01-28T23:12:58.28Z"
         },
       "display-name": "Private_LB1",
       "freeform-tags": null,
       "hostnames": {},
       "id": "ocid1.loadbalancer.unique_ID",
       "ip-addresses": [
         {
   ```

```
        "ip-address": "IP_address",
        "is-public": false,
        "reserved-ip": null
      }
    ],
    "is-private": true,
    "lifecycle-state": "ACTIVE",
    "listeners": {},
    "network-security-group-ids": null,
    "path-route-sets": {},
    "routing-policies": null,
    "rule-sets": {},
    "shape-details": null,
    "shape-name": "400Mbps",
    "ssl-cipher-suites": {},
    "subnet-ids": [
      "ocid1.subnet.unique_ID1",
      "ocid1.subnet.unique_ID2"
      ],
    "system-tags": null,
    "time-created": "2025-01-28T23:12:58.000001+00:00"
  },
"etag": "00c648d7-b654-4583-b7bf-k5oed55"
}
```


This output is the same as the output of the `oci lb load-balancer get` command.

Next Steps: If you did not create all the resources needed for the LB in the `load-balancer create` command, complete the LB configuration by adding resources using their separate commands, such as `listener create`. For a list of commands, see `oci lb -h`.

Viewing Load Balancer Details

This topic describes how to display a list of load balancers (LBs) and view their details.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. If necessary, select a different compartment from the compartment menu above the LB list.
3. Select the name of the LB to go to its details page.

   Alternatively, for the LB for which you want to see the details, select the Actions menu and select the View details option.

Using the OCI CLI

1. Get the LB OCID: `oci lb load-balancer list`
2. Run the get LB command.

   Syntax:

   `oci lb load-balancer get --load-balancer-id load-balancer_OCID`

   The details of all the resources that have been created, such as backend sets, certificates, and listeners are included in the output.

Editing a Load Balancer

You can change the load balancer (LB) name and tags.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. For the LB that you want to edit, select the Actions menu, and select the Edit option to open the Edit Load Balancer dialog.
3. Make your changes and select the Update Load Balancer button to update the LB properties.

To add or update related resources such as backend sets or listeners, go to the LB details page, scroll down to the Resources section, and select the resource that you want to add or edit.

Using the OCI CLI

1. Get the LB OCID: `oci lb load-balancer list`
2. Run the update LB command.

   Example:

```
$ oci lb load-balancer update \
 --load-balancer-id ocid1.loadbalancer.unique_ID \
 --display-name new_lb_name
```

If you did not add resources such as backend sets or listeners when you created the LB, add them by using their separate command, such as `oci lb listener create`. If you did add resources when you created the LB, update them by using their separate command, such as `oci lb listener update`.

Deleting a Load Balancer

This topic describes how to delete a load balancer (LB) and remove it from service.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. For the LB that you want to delete, select the Actions menu, and select the Terminate option.
3. Confirm the operation when prompted.

Using the OCI CLI

1. Get the LB OCID: `oci lb load-balancer list`
2. Run the delete LB command.

   Syntax:

```
$ oci lb load-balancer delete --force --load-balancer-id loadbalancer_OCID
```

Cipher Suites

This section describes how to use cipher suites with a load balancer (LB) to determine the security, compatibility, and speed of HTTPS traffic.

- Creating a Load Balancer SSL Cipher Suite
- Viewing a Load Balancer Cipher Suite Details

- Editing a Load Balancer Cipher Suite
- Deleting a Load Balancer Cipher Suite

Creating a Load Balancer SSL Cipher Suite

A load balancer (LB) uses a cipher suite to secure Transport Layer Security (TLS) or Secure Socket Layer (SSL) network connections. The cipher suite defines a list of security algorithms that the LB uses to negotiate with peers exchanging information with the LB. The cipher suites used affect the security level, performance, and compatibility of data traffic.

Oracle has created a series of predefined cipher suites that you can use when you create an SSL configuration. If the predefined cipher suites don't meet requirements, you can create custom cipher suites.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to create the SSL cipher suite.
3. On the LBdetails page, scroll to the Resources section and select Cipher Suites.
4. Select the Create Cipher Suite button.
5. In the Load Balancer SSL Cipher Suite dialog, give the LB SSL cipher suite a name.

> **Note:**
>
> The name of a user-defined cipher suite can't be the same as any of Oracle's predefined or reserved SSL cipher suite names.

6. Check the boxes of the cipher suite components to be part of the SSL cipher suite.
7. Select the Create Cipher Suite button in the dialog.

   To check the result, select the cipher suite name in the Cipher Suites list in the Resources section of the LB details page.

Using the OCI CLI

1. Get the LB OCID: `oci lb load-balancer list`
2. Run the create SSL cipher suite command.

   Syntax:

   ```
   oci lb ssl-cipher-suite create --ciphers ssl_ciphers \
   --load-balancer-id load-balancer_OCID --name ssl_cipher_suite_name
   ```

   Option values:

   • ***ssl_ciphers*** - A list of SSL ciphers the load balancer must support for HTTPS or SSL connections.

   • ***load-balancer_OCID*** - The OCID of the associated load balancer.

   • ***ssl_cipher_suite_name*** - A user-friendly name for the SSL cipher suite. The name must be unique and cannot be changed.

   Example:

```
$ oci lb ssl-cipher-suite create --ciphers ["ECDHE-RSA-AES256-GCM-SHA384", \
"ECDHE-ECDSA-AES256-GCM-SHA384","ECDHE-RSA-AES128-GCM-SHA256"] \
--load-balancer-id ocid1.loadbalancer.uniqueID
--name my_ssl_cipher_suite

{
  "opc-work-request-id": "ocid1.workrequest.oc1.pca.uniqueID"
}
```

To see the cipher suite details, use the `oci lb ssl-cipher-suite list` command to list all the cipher suites associated with the specified LB, and then use the `oci lb ssl-cipher-suite get` command to view the SSL cipher suite details as shown in Viewing a Load Balancer Cipher Suite Details.

Viewing a Load Balancer Cipher Suite Details

This topic describes how to view a list of the SSL cipher suites associated with a load balancer (LB) and how to view the details of a specific cipher suite.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to view existing cipher suites.
3. On the LB details page, scroll to the Resources section and select Cipher Suites. The list of cipher suites is shown.
4. To view the details page for a cipher suite, either select the name of the cipher suite in the list, or select the Actions menu and then select the View Details option.

Using the OCI CLI

1. Gather the information you need to run the command:
   - LB OCID: `oci lb load-balancer list`
   - Name of the cipher suite: `oci lb ssl-cipher-suite list`
2. Run the get cipher suite command to view the details of the cipher suite that you are interested in.

   Syntax:

   Use the following command to list all cipher suites associated with the LB:

   ```
   oci lb ssl-cipher-suite list --load-balancer-id load-balancer_OCID
   ```

   Use the following command to show the details of the named cipher suite:

   ```
   oci lb ssl-cipher-suite get --load-balancer-id load-balancer_OCID --name cipher-suite-name
   ```

   Example:

   ```
   $ oci lb ssl-cipher-suite get --load-balancer-id ocid1.loadbalancer.uniqueID \
   --name "my_ssl_cipher_suite"

   {
     "data": {
       "ciphers": [
         "ECDHE-RSA-AES256-GCM-SHA384",
         "ECDHE-ECDSA-AES256-GCM-SHA384",
         "ECDHE-RSA-AES128-GCM-SHA256"
   ```

```
    ],
    "name": "my_ssl_cipher_suite"
  }
}
```

Editing a Load Balancer Cipher Suite

This topic describes how to edit SSL cipher suites associated with a load balancer (LB) to add or remove ciphers. The name of the cipher suite cannot be changed.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to edit existing cipher suites.
3. On the LB details page, scroll to the Resources section and select Cipher Suites. The list of cipher suites is shown.
4. Use one of the following methods to update a cipher suite:
   - Select the name of the cipher suite. On the cipher suite details page, select the Edit button.
   - Select the Actions menu for the cipher suite and select Edit.

Using the OCI CLI

1. Gather the information you need to run the command:
   - LB OCID: `oci lb load-balancer list`
   - Name of the cipher suite: `oci lb ssl-cipher-suite list`
2. Run the update SSL cipher suite command to edit the ciphers of the cipher suite.

   Syntax:

   ```
   oci lb ssl-cipher-suite update --load-balancer-id load-balancer_OCID \
   --name cipher-suite-name --ciphers list_of_ciphers
   ```

   Example:

   ```
   $ oci lb ssl-cipher-suite update --load-balancer-id  ocid1.loadbalancer.uniqueID \
   --name "my_ssl_cipher_suite" \
   --ciphers ["ECDHE-RSA-AES256-GCM-SHA384","ECDHE-ECDSA-AES256-GCM-SHA384"]

   {
     "opc-work-request-id": "ocid1.workrequest.oc1.pca.uniqueID"
   }
   ```

   Use the `ssl-cipher-suite get` command to verify that the cipher suite is updated. See Viewing a Load Balancer Cipher Suite Details.

Deleting a Load Balancer Cipher Suite

This topic describes how to delete an SSL cipher suite associated with a load balancer (LB).

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to delete cipher suites.

3. On the LB details page, scroll to the Resources section and select Cipher Suites. The list of cipher suites is shown.
4. Use one of the following methods to delete a cipher suite:
   - Select the name of the cipher suite. On the cipher suite details page, select the Delete button.
   - Select the Actions menu for the cipher suite and select the Delete button.

Using the OCI CLI

1. Gather the information you need to run the command:
   - LB OCID: `oci lb load-balancer list`
   - Name of the cipher suite: `oci lb ssl-cipher-suite list`
2. Run the delete cipher suite command.

   Syntax:

   ```
   oci lb ssl-cipher-suite delete --load-balancer-id load-balancer_OCID \
   --name cipher-suite-name
   ```

   Example:

   ```
   $ oci lb ssl-cipher-suite delete --load-balancer-id ocid1.loadbalancer.uniqueID \
   --name "my_ssl_cipher_suite" --force

   {
     "opc-work-request-id": "ocid1.workrequest.oc1.pca.uniqueID"
   }
   ```

   Use the `ssl-cipher-suite list` command to verify that the cipher suite is deleted. See Viewing a Load Balancer Cipher Suite Details.

SSL Certificates

This section describes how to use secure socket layer (SSL) certificates with a load balancer (LB).

- Adding a Load Balancer Certificate
- Viewing a Load Balancer Certificate
- Deleting a Load Balancer Certificate

Adding a Load Balancer Certificate

This topic describes how to add a public SSL certificate to use with a load balancer (LB).

Optionally, you can also provide a certificate for a Certificate Authority (CA) or configure a private key.

> **Note:**
>
> You can use a custom, self-signed SSL certificate. However, for production environments, Oracle recommends that you use a CA-issued SSL certificate, which reduces the risk of a man-in-the-middle attack.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to add the certificate.
3. On the LB details page, scroll to the Resources section and select Certificates.
4. Select the Create Certificate button.
5. Enter the following information in the Load Balancer Create Certificate dialog.
   - **Name:** Enter a descriptive name for the certificate bundle. The name must be unique and cannot be changed. The name can include only alphanumeric characters, dashes, and underscores. The name cannot contain spaces.
   - **Public certificate:** Either upload the certificate `.pem` file, or paste the content from the `.pem` file directly into the dialog box using drag and drop.
   - **Certificate Authority:** Click the Enable certificate authority box if you are also using a certificate authority (CA) certificate. Either upload the CA certificate `.pem` file, or paste the content from the `.pem` file directly into the dialog box using drag and drop.
   - **Private Key:** Click the Enable private key box if you are also using a private key certificate. Either upload the private key `.pem` file, or paste the content from the `.pem` file directly into the dialog box using drag and drop.
6. Select the Create Certificate button in the dialog.

Using the OCI CLI

1. Get the LB OCID: `oci lb load-balancer list`
2. Run the create certificate command.

   Only the certificate name and the LB OCID are always required for the `create` command, but you probably also need to provide the private key file and public certificate, and you might need to provide the CA certificate and passphrase. See the example below and use the `-h` option for more information.

   Syntax:

   ```
   oci lb certificate create --certicate-name certificate-name \
   --load-balancer-id load-balancer_OCID
   ```

   Option values:

   - ***certificate-name*** - A user-friendly name for the certificate bundle.
   - ***load-balancer_OCID*** - The OCID of the load balancer associated with the backend set and servers.

   Example:

   ```
   $ oci lb certificate create --certificate-name example-certificate \
   --load-balancer-id ocid1.loadbalancer.unique_ID \
   --ca-certificate-file CA_cert_file \
   --public-certificate-file pub_cert_file \
   --private-key-file priv_key_file --passphrase "passphrase"

   {
     "opc-work-request-id": "ocid1.workrequest.oc1.pca.unique_ID"
   }
   ```

   Option values:

- ***CA_cert_file*** - The Certificate Authority certificate, or any interim certificate, that you received from your SSL certificate provider.
- ***pub_cert_file*** - The public certificate, in PEM format, that you received from your SSL certificate provider.
- ***priv_key_file*** - The SSL private key for your certificate, in PEM format.
- ***passphrase*** - A passphrase for encrypted private keys. This is needed only if you created your certificate with a passphrase.

To view the certificate details, use `oci lb certificate list` with the LB OCID to list all certificates associated with the specified LB, and find the certificate with ***certificate-name***. In the following example, the certificate content is truncated.

```
$ oci lb certificate list --load-balancer-id ocid1.loadbalancer.unique_ID

{
  "data": [
    {
      "ca-certificate": "-----BEGIN CERTIFICATE-----
...
-----END CERTIFICATE-----",
      "certificate-name": "example-certificate",
      "passphrase": "passphrase"
      "private-key": "-----BEGIN RSA PRIVATE KEY-----
...
-----END RSA PRIVATE KEY-----",
      "public-certificate": "-----BEGIN CERTIFICATE-----
...
-----END CERTIFICATE-----"
    }
  ]
}
```

Viewing a Load Balancer Certificate

This topic describes how to view an SSL certificate that is used with a load balancer (LB).

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to view the certificate.
3. On the LB details page, scroll to the Resources section and select Certificates.
4. The details of all configured certificates are displayed.

Using the OCI CLI

1. Get the LB OCID: `oci lb load-balancer list`
2. Run the list certificate command.

   Syntax:

   `oci lb certificate list --load-balancer-id load-balancer_OCID`

   Example:

   ```
   $ oci lb certificate list --load-balancer-id ocid1.loadbalancer.uniqueID

   {
   ```

```
 "data": [
   {
     "ca-certificate": null,
     "certificate-name": "CERT1-lb1-w",
     "public-certificate": "-----BEGIN CERTIFICATE-----
...-----END CERTIFICATE-----"
   }
  ]
}
```

Deleting a Load Balancer Certificate

This topic describes how to delete an SSL certificate that is used with a load balancer (LB). You cannot change an LB SSL certificate. To change a certificate, delete the certificate and create a new certificate.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to delete a certificate.
3. On the LB details page, scroll to the Resources section and select Certificates.
4. All the details of the configured certificates are displayed.
5. For the certificate that you want to delete, select the Actions menu and select Delete.
6. Confirm to delete the named certificate.

Using the OCI CLI

1. Gather the information you need to run the command:
   - LB OCID: `oci lb load-balancer list`
   - Name of the certificate: `oci lb certificate list`
2. Run the delete certificate command.

   Syntax:

   ```
   oci lb certificate delete --certificate-name certificate_name \
   --load-balancer-id load-balancer_OCID
   ```

Backend Sets

This section describes how to use backend sets to create logical entities consisting of a load balancing policy, health check policy, and a list of backend servers for a load balancer (LB).

- Creating a Load Balancer Backend Set
- Viewing Load Balancer Backend Set Details
- Editing a Load Balancer Backend Set
- Deleting a Load Balancer Backend Set

Creating a Load Balancer Backend Set

This topic describes how to create a backend set for a load balancer (LB). The backend set is a group of servers to which traffic is load balanced. Using the OCI CLI or Compute Enclave API, you can create backend servers when you create the backend set. Using the Compute Web UI, you must add backend servers after the backend set is created.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Click the name of the LB for which you want to create the backend set.
3. On the LB details page, scroll to the Resources section, and select Backend Sets.
4. Select the Create Backend Set button.
5. Enter the following information:
   - **Name:** Enter a descriptive name for the LB backend set. The name must be unique and cannot be changed.
   - **Traffic Distribution Policy:** Select one of the following policies for the backend set:
     - **Weighted Round Robin:** Traffic is balanced in a "next turn" fashion, with some servers having a preference.
     - **Least Connections:** Traffic is balanced based on the server with the fewest current connections.
     - **IP Hash:** Traffic is balanced based on a hash of several fields in the IP header.

     For more information, see "Load Balancing Policies" in "Frontend Configuration" in the Load Balancing Overview chapter of the *Oracle Private Cloud Appliance Concepts Guide*.
   - **SSL:** Associate an SSL certificate with the backend set.
     - **Use SSL:** When you check the Use SSL box, a drop-down list of certificates appears.
     - **Certificates:** Select a certificate from the list.
     - **Verify peer certificate:** Check this box to enable peer certificate verification.
   - **Health Checking:** Enter the health checking parameters to use to test the health of backend servers. All of these values are optional (the parameters have default values) except for protocol.
     - **Protocol:** Select the protocol to use: HTTP or TCP. Choose the protocol that matches your application or service.
     - **Port:** Enter the backend server port against which to run health checks.
     - **Interval in Milliseconds:** Specify how often to run health checks, in milliseconds. Enter a number between 1 and 1,800,000.
     - **Timeout in Milliseconds:** Specify the maximum time to wait for a reply to a health check, in milliseconds. Enter a number between 1 and 600,000.
     - **Number of Retries:** Enter the number of times to retry the health check before the server is considered unhealthy.
     - **Status Code:** (HTTP only) Specify the HTTP status code a healthy server must return.
     - **URL Path:** (HTTP only) Specify a URL endpoint against which to run the health check.

     For more information, including how to diagnose misconfigurations, see "Load Balancer Health Checks" in "Backend Configuration" in the Load Balancing Overview chapter of the *Oracle Private Cloud Appliance Concepts Guide*.
6. Click the Create Backend Set button in the dialog.

To check the configuration, select the backend set name in the Backend Sets list in the Resources section of the LB details page.

Using the OCI CLI

1. Get the LB OCID: `oci lb load-balancer list`
2. Construct an argument for the `--backends` option.

   The `--backends` option is a list of backend server definitions in the following JSON format. For brevity, only one list item is shown in the following output:

   ```
   $ oci lb backend-set create --generate-param-json-input backends
   [
     {
       "backup": true,
       "drain": true,
       "ipAddress": "string",
       "offline": true,
       "port": 0,
       "weight": 0
     }
   ]
   ```

3. Run the create backend set command.

   Syntax:

   ```
   oci lb backend-set create --health-checker-protocol [HTTP | TCP] \
   --load-balancer-id load-balancer_OCID \
   --name backend-set-name --policy load-balancer-policy
   ```

   For possible values of ***load-balancer-policy***, use `oci lb policy list`. See also Load Balancer Policies.

   Example:

   The create backend set command has many options. Use the `-h` option to learn about them.

   This example creates a set of backend servers for the backend set. The backends are defined in a file named backendsSet1. You can also create backend servers later.

   ```
   oci lb backend-set create --health-checker-protocol TCP \
   --load-balancer-id ocid1.loadbalancer.unique_ID --name BackendSet1 \
   --policy LEAST_CONNECTIONS --backends file://./backendsSet1 \
   --health-checker-port 22 --health-checker-return-code 200

   {
     "opc-work-request-id": "ocid1.workrequest.unique_ID"
   }
   ```

   To view the backend set details, see Viewing Load Balancer Backend Set Details.

Viewing Load Balancer Backend Set Details

This topic describes how to view a list of backend sets of a load balancer (LB) and how to view the configuration information and the list of servers in a backend set.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to view the backend set details.

3. On the LB details page, scroll to the Resources section and select Backend Sets. The list of backend sets for this LB is shown.
4. Select the name of the backend set that you are interested in.
5. On the backend set details page, the Backend Set Information tab shows the load balancing policy and the overall health of the servers in the set. Select the Backend Set Configuration tab to see health checker and SSL configuration details. Scroll to the Resources section to view the list of backend servers.

Using the OCI CLI

1. Gather the information you need to run the command:
   - LB OCID: `oci lb load-balancer list`
   - Backend set name: `oci lb backend-set list`
2. Display the list of backend sets for an LB.

```
$ oci lb backend-set list --load-balancer-id ocid1.loadbalancer.unique_ID
```

3. Display the details of a backend set, including configuration and backend servers.

```
$ oci lb backend-set get --load-balancer-id ocid1.loadbalancer.unique_ID \
--backend-set-name BackendSet1
```

Editing a Load Balancer Backend Set

This topic describes how to change load balancer (LB) backend set properties, such as the health checker protocol used.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to modify a backend set.
3. On the LB details page, scroll to the Resources section and select Backend Sets.
4. For the backend set that you want to modify, select the Actions menu, and select the Edit option.
5. In the Edit Load Balancer Backend Set dialog, make your changes.
6. Select the Update Load Balancer Backend Set button in the dialog.

Using the OCI CLI

1. Gather the information you need to run the command:
   - LB OCID: `oci lb load-balancer list`
   - Backend set name: `oci lb backend-set list`
2. Run the update backend set command.

   Syntax:

```
oci lb backend-set update --load-balancer-id loadbalancer_OCID \
--backend-set-name backendset_name --backends list_of_server_definitions \
--health-checker-protocol [HTTP | TCP] \
--policy load-balancer-policy
```

   Example:

```
$ oci lb backend-set update --load-balancer-id ocid1.loadbalancer.uniqueID \
--backend-set-name BackendSet1 --backends file://./backendsSet2 \
--health-checker-protocol HTTP --policy ROUND_ROBIN
WARNING: Updates to backends and health-checker and ssl-configuration and session-
persistence-
  configuration and lb-cookie-session-persistence-configuration will replace any
existing values.
  Are you sure you want to continue? [y/N]: y

{
  "opc-work-request-id": "ocid1.workrequest.xxx.loadbalancer.uniqueID"
}
```

Use the `backend-set get` command to verify that the backend set is updated. See Viewing Load Balancer Backend Set Details.

Deleting a Load Balancer Backend Set

This topic describes how to delete a load balancer (LB) backend set and remove it from service.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to delete a backend set.
3. On the LB details page, scroll to the Resources section and select Backend Sets.
4. For the backend set that you want to delete, select the Actions menu, and select Terminate.
5. Confirm the operation when prompted.

Using the OCI CLI

1. Gather the information you need to run the command:
   - LB OCID: `oci lb load-balancer list --compartment-id compartment_OCID`
   - Backend set name: `oci lb backend-set list`
2. Run the delete backend set command.

```
$ oci lb backend-set delete --load-balancer-id ocid1.loadbalancer.unique_ID \
--backend-set-name BackendSet1 --force
{
  "opc-work-request-id": "ocid1.workrequest.unique_ID"
}
```

Backend Servers

This section describes how to manage backend servers for use with a load balancer.

- Creating a Load Balancer Backend Server
- Viewing Load Balancer Backend Server Details
- Editing a Load Balancer Backend Server
- Deleting a Load Balancer Backend Server

Creating a Load Balancer Backend Server

This topic describes how to create a backend server to add to a backend set. A backend set is a group of backend servers to which traffic is load balanced.

If you create a backend server in a VCN that is not the same as the load balancer (LB) VCN, then you must set up a Local Peering Gateway to enable communication between the LB and the backend server. See Connecting VCNs through a Local Peering Gateway.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to create a backend server.
3. On the LB details page, scroll to the Resources section, and select Backend Sets.
4. In the list of backend sets, select the name of the backend set for which you want to create a backend server.
5. On the details page of the backend set, select the Create Backend button.
6. Enter the following information:
   - **Computed Instances**
     - **Instance:** If you select Computed Instances, the IP Address area presents a drop-down list of instances. Select one of these instances for a backend server. You can change the compartment above the list.
     - **Port:** The server port to load balance.
     - **Weight:** The load balancing policy weight assigned to the server. Backend servers with a higher weight receive a larger proportion of incoming traffic. For example, a server with weight 3 receives 3 times the number of new connections as a server with weight 1. For more information about load balancing policies, see "Load Balancing Policies" in "Frontend Configuration" in the Load Balancing Overview chapter of the *Oracle Private Cloud Appliance Concepts Guide*.
     - **Security Rules:** To enable load balancer traffic, you must add ingress and egress security rules to the corresponding subnets.

       If you select Configure Manually, then when you are finished adding backends, go to the VCN and create or update a security list to add security rules, and ensure that the applicable subnet is using that security list. See Controlling Traffic with Security Lists.

       If you select Configure Automatically, a table of egress rules is shown and a table of ingress rules is shown. Each table lists the name of the security list, the name of the subnet, and the rule: the CIDR block and port to allow egress or ingress traffic. Select the button to the right of the rule to enable or disable that rule for each backend that you added.
   - **IP Addresses**
     - **IP Address:** If you select IP Addresses, the IP Address area presents a text field where you must enter the IP address of the instance that you want to use as a backend server.
     - **Port:** The server port to load balance.
     - **Weight:** See the description in "Computed Instances."

To add another backend, select the Add IP Address button. You can select an instance or you can specify an IP address that is already a member of the backend set if you specify a different port.

7. Select the Submit button on the dialog to create the backend servers. The new backends appear in the list for the backend set.

Using the OCI CLI

1. Gather the information you need to run the command:
   - LB OCID: `oci lb load-balancer list`
   - Backend set name: `oci lb backend-set list`
2. Run the create backend server command.

   Syntax:

   ```
   oci lb backend create --load-balancer-id load-balancer_OCID \
   --backend-set-name backend-set-name \
   --ip-address backend-svr-ip-addr --port port-number
   ```

   Option values:

   - ***backend-set-name*** - The name of the backend set in which to add the backend server.
   - ***backend-svr-ip-addr*** - The IP address of the compute instance to add as a backend server.
   - ***port-number*** - The port to load balance on the backend server.

   Example:

   ```
   $ oci lb backend create --load-balancer-id ocid1.loadbalancer.unique_ID \
   --backend-set-name example_backend_set \
   --ip-address 10.0.0.3 --port 8080 --weight 3

   {
     "opc-work-request-id": "ocid1.workrequest.unique_ID"
   }
   ```

   The `--weight` option specifies the load balancing policy weight assigned to the server. Backend servers with a higher weight receive a larger proportion of incoming traffic. For example, a server with weight 3 receives 3 times the number of new connections as a server with weight 1. For more information on load balancing policies, see "Load Balancing Policies" in "Frontend Configuration" in the Load Balancing Overview chapter of the *Oracle Private Cloud Appliance Concepts Guide*.

   You can also set backup, drain, offline, and maximum connections. See the `-h` option for more information.

   To create multiple backends in one command, use the `--backends` option with the `backend-set create` command as described in Creating a Load Balancer Backend Set .

   To view the newly-added backend, use the `backend get` command as shown in Viewing Load Balancer Backend Server Details.

   ```
   $ oci lb backend get --load-balancer-id ocid1.loadbalancer.unique_ID \
   --backend-set-name example_backend_set --backend-name 10.0.0.3:8080
   ```

Viewing Load Balancer Backend Server Details

This topic describes how to view a list of the backend servers in a load balancer (LB) backend set and how to view the configuration details of a specific backend.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to list backend servers.
3. On the LB details page, scroll to the Resources section and select Backend Sets to see the list of backend sets for this LB.
4. Select the name of a backend set to see the list of backends in that backend set. For each backend in the list, the IP address, port, weight, and other attributes are shown.

Using the OCI CLI

1. Gather the information you need to run the command:
   - LB OCID: `oci lb load-balancer list`
   - Backend set name: `oci lb backend-set list`
2. Display the list of backends for a backend set.

```
$ oci lb backend list --load-balancer-id ocid1.loadbalancer.unique_ID \
--backend-set-name backend-set-name
```

3. Display the details of a specific backend.

```
$ oci lb backend get --load-balancer-id ocid1.loadbalancer.unique_ID \
--backend-set-name example_backend_set --backend-name 10.0.0.3:8080

{
  "data": [
    {
      "backup": false,
      "drain": false,
      "ip-address": "10.0.0.3",
      "name": "10.0.0.3:8080",
      "offline": false,
      "port": 8080,
      "weight": 1
    }
  ]
}
```

Editing a Load Balancer Backend Server

This topic describes how to change the weight, drain, offline, and backup attributes of a load balancer (LB) backend server.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to edit a backend server.
3. On the LB details page, scroll to the Resources section and select Backend Sets.
4. Select the name of the backend set that contains the backend server that you want to edit.

5. On the backend set details page, scroll to the Resources section.
6. For the backend server that you want to edit, select the Actions menu and select Edit.
7. On the Update Backends dialog, enter a new weight and enable or disable the drain, offline, or backup attributes.
8. Select the Submit button to update the LB backend server.

Using the OCI CLI

1. Gather the information you need to run the command:
   - LB OCID: `oci lb load-balancer list`
   - Backend set name: `oci lb backend-set list`
   - Backend server name: `oci lb backend list`
2. Run the update backend command.

```
oci lb backend update --load-balancer-id ocid1.loadbalancer.unique_ID \
--backend-set-name example_backend_set --backend-name 10.0.0.3:8080 \
--backup false --drain false --offline false --weight 3
```

   The `--backup`, `--drain`, `--offline`, and `--weight` options are required; They do not have default values. You can also specify `--max-connections`. Use the `-h` option to see descriptions of these options. See also Creating a Load Balancer Backend Server for a description of the weight value.

   To view the result of the backend update, use the `backend get` command as shown in Viewing Load Balancer Backend Server Details.

Deleting a Load Balancer Backend Server

This topic describes how to delete a load balancer (LB) backend server from a backend set and remove the backend from service.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to delete a backend server.
3. On the LB details page, scroll to the Resources section and select Backend Sets.
4. Select the name of the backend set that contains the backend server that you want to delete.
5. On the backend set details page, scroll to the Resources section.
6. For the backend server that you want to delete from the set, select the Actions menu and select Terminate.
7. Confirm the operation when prompted.

Using the OCI CLI

1. Gather the information you need to run the command:
   - LB OCID: `oci lb load-balancer list`
   - Backend set name: `oci lb backend-set list`
   - Backend server name: `oci lb backend list`

2. Run the delete backend command.

```
$ oci lb backend delete --load-balancer-id ocid1.loadbalancer.unique_ID \
--backend-set-name example_backend_set --backend-name 10.0.0.3:8080 --force
```

   To verify that the backend is deleted, use the `backend list` command.

Virtual Hostnames

This section describes how to use virtual hostnames with a load balancer for one or more listeners.

- Creating a Load Balancer Virtual Hostname
- Viewing Load Balancer Virtual Hostnames
- Editing a Load Balancer Virtual Hostname
- Deleting a Load Balancer Virtual Hostname

Creating a Load Balancer Virtual Hostname

A virtual hostname is associated with a load balancer (LB) and used by one or more listeners. Hostnames associated with a listener correspond to the backend set of that listener. The backend set routes traffic to specific backends which host different applications.

Virtual hostnames simplify the construction of the hostnames associated with listeners and backend servers because virtual hostnames can use wild card asterisks (*) at the start or end of the hostname. Listeners detect a hostname pattern that matches the virtual hostname patterns created.

> **Note:**
>
> The asterisk (*) doesn't have to be used in a virtual hostname. However, when used, the asterisk can only be added at the beginning or ending of a virtual hostname. Traffic sent to ***app.example.com*** is load balanced by a listener and backend server set when configured as ***app.example.com*** (exact), ****example.com*** (wild card at start), or ***app.example**** (wild card at end).

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to create the virtual hostname.
3. On the LB details page, scroll down to the Resources section and select Hostnames.
4. Select the Create Hostname button to open the Load Balancer Create Hostname dialog.
5. Enter the following information:
   - **Name:** A name for the LB virtual hostname.
   - **Hostname:** The virtual hostname.
6. Select the Create Hostname button in the dialog.

Using the OCI CLI

1. Get the LB OCID: `oci lb load-balancer list`

2. Run the create virtual hostname command.

   Syntax:

   ```
   oci lb hostname create --load-balancer-id load-balancer_OCID \
   --hostname virtual-hostname --name virtual-hostname-friendly-name
   ```

   Example:

   ```
   $ oci lb hostname create --load-balancer-id ocid1.loadbalancer.unique_ID \
   --hostname *example.com --name my_virtual_hostname

   {
     "opc-work-request-id": "ocid1.workrequest.oc1.pca.unique_ID"
   }
   ```

   To verify that the hostname was created, see Viewing Load Balancer Virtual Hostnames .

Viewing Load Balancer Virtual Hostnames

This topic describes how to view the virtual hostnames associated with a load balancer (LB).

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to view the virtual hostnames.
3. On the LB details page, scroll to the Resources section, and select Hostnames.

Using the OCI CLI

1. Get the LB OCID: `oci lb load-balancer list`
2. Display the list of virtual hostnames for an LB.

   ```
   $ oci lb hostname list --load-balancer-id ocid1.loadbalancer.unique_ID

   {
     "data": [
       {
         "hostname": "app.example.com",
         "name": "match-exact-hostname"
       },
       {
         "hostname": "*example.com",
         "name": "my_virtual_hostname"
       },
       {
         "hostname": "app.example*",
         "name": "suffix-match-hostname"
       }
     ]
   }
   ```

3. Display the details of a specific virtual hostname.

   ```
   $ oci lb hostname get --load-balancer-id ocid1.loadbalancer.unique_ID \
   --name "my_virtual_hostname"
   ```

Editing a Load Balancer Virtual Hostname

This topic describes how to change the virtual hostname associated with a load balancer (LB).

Virtual hostnames can use wild card asterisks (*) at the start or end of the hostname. Listeners detect a hostname pattern that matches the virtual hostname patterns created.

> **Note:**
>
> The asterisk (*) does not have to be used in a virtual hostname. However, when used, the asterisk can only be added at the beginning or ending of a virtual hostname. Traffic sent to ***app.example.com*** is load balanced by a listener and backend server set when configured as ***app.example.com*** (exact), ****example.com*** (wild card at start), or ***app.example**** (wild card at end).

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to edit the virtual hostnames.
3. On the LB details page, scroll to the Resources section and select Hostnames.
4. For the hostname that you want to edit, select the Actions menu, and select Edit.
5. Edit the virtual hostname.

   To change the name of the virtual hostname, delete the virtual hostname and create a new one.
6. Select the Save Changes button in the dialog.

Using the OCI CLI

1. Gather the information you need to run the command:
   - LB OCID: `oci lb load-balancer list`
   - Name of the hostname resource: `$ oci lb hostname list`
2. Run the update virtual hostname command.

   ```
   $ oci lb hostname update --load-balancer-id ocid1.loadbalancer.unique_ID \
   --name "my_virtual_hostname" --hostname "*example.net"
   ```

   To verify that the hostname was updated, see Viewing Load Balancer Virtual Hostnames .

Deleting a Load Balancer Virtual Hostname

This topic describes how to delete a virtual hostname associated with a load balancer (LB).

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to delete a virtual hostname.
3. On the LB details page, scroll to the Resources section, and select Hostnames.
4. For the hostname that you want to delete, select the Actions menu, and select Delete.
5. Confirm the deletion.

Using the OCI CLI

1. Gather the information you need to run the command:

- LB OCID: `oci lb load-balancer list`
- Name of the hostname resource: `$ oci lb hostname list`

2. Run the virtual hostname delete command.

```
$ oci lb hostname delete --load-balancer-id ocid1.loadbalancer.unique_ID \
--name "my_virtual_hostname" --force
```

Path Route Sets

You can apply a set of path routes to a load balancer (LB) to determine the appropriate destination backend set for incoming URIs.

Some applications have multiple endpoints or content types, each distinguished by a unique URI path such as `/admin/`, `/data/`, `/video/`, or `/cgi/`. Each rule in a path route set names a backend set, a partial URI to match, and the pattern match type.

Path route rules route traffic to the correct backend set without requiring multiple listeners or LBs.

A path route set includes all path route rules that define the data routing for a particular listener. You can have at most one path route set for each listener. You can specify at most 20 path route rules for each path route set.

Path route rules apply only to HTTP, HTTP/2, and HTTPS requests. Path route rules do not apply to TCP requests.

Path route rule URL strings have the following restrictions:

- You can't use asterisks in path route strings.
- You can't use regular expressions.
- Path route string matching is case-insensitive. For example, both "data" and "DATA" match.

Browsers often add an ending slash to the path in a request. You might want to configure a rule with a URL string that includes the trailing slash and a second rule with a URL string that does not include the trailing slash (for example,`/admin` and `/admin/`).

A path route rule pattern match type is one of the following:

- EXACT_MATCH: The path string must match the incoming URI path exactly.
- FORCE_LONGEST_PREFIX_MATCH: The path string must match longest ("best") match of the beginning portion of the incoming URI path.
- PREFIX_MATCH: The path string must match the beginning portion of the incoming URI path.
- SUFFIX_MATCH: The path string must match the ending portion of the incoming URI path.

For more information about match types and priority of matches, see "Path Route Sets" in "Request Routing" in "Frontend Configuration" in the Load Balancing Overview chapter of the *Oracle Private Cloud Appliance Concepts Guide*.

Creating a Path Route Set

This topic describes how to create a path route set to route URIs to load balancer (LB) backend sets. To create a path route set, a backend set must already exist.

See Path Route Sets for information about URI patterns, pattern match types, and path route set limits.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of LB for which you want to create the path route set.
3. On the LB details page, scroll to the Resources section, and select Path Route Sets.
4. Select the Create Path Route Set button to open the Create Path Route Set dialog.
5. Enter the following information:
   - **Name:** Enter a descriptive name for the Path Route Set.
   - **Path Route Rules:** Enter the following information:
     - **Match Style:** Select Exact Match, Force Longest Prefix Match, Prefix Match, or Suffix Match. The match style must be able to match the URL string entered next.
     - **URL String:** Enter the pattern to match.
     - **Backend Set:** Select the name of the backend set from the drop-down list.

     To create another rule, select the New Rule button.
6. Select the Create Path Route Set button in the dialog.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Load balancer OCID: `oci lb load-balancer list`
   - Backend set name: `oci lb backend-set list`
2. Construct an argument for the `--path-routes` option.

   The `--path-routes` option is a list of path route rules in the following JSON format, where the `path` is the pattern to match and the `pathMatchType` is Exact Match, Force Longest Prefix Match, Prefix Match, or Suffix Match. For brevity, only one list item is shown in the following output:

```
$ oci lb path-route-set create --generate-param-json-input path-routes
[
  {
    "backendSetName": "string",
    "path": "string",
    "pathMatchType": {
      "matchType": "string"
    }
  }
]
```

3. Run the create path route set command.

   Syntax:

```
oci lb path-route-set create --load-balancer-id load-balancer_OCID \
--name name-of-path-route-set --path-routes list-of-path-route-rules
```

   Example:

```
$ oci lb path-route-set create --load-balancer-id ocid1.load-balancer.unique_ID \
--name PathRouteSet1 --path-routes file://./PathRouteSet1Rules.json
```

```
{
  "opc-work-request-id": "ocid1.workrequest.1X49XC30ZP.unique_ID"
}
```

To view the new path route set, use the `path-route-set get` command as shown in Viewing a Path Route Set Details.

Viewing a Path Route Set Details

This topic describes how to view a list of path route sets for a load balancer (LB) and how to view the details (route rules) of a specific path route set.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to view path route sets.
3. On the LB details page, scroll to the Resources section and select Path Route Sets to see the list of backend sets for this LB.
4. Select the name of a Path Route Set to display the details.

Using the OCI CLI

1. Get the LB OCID: `oci lb load-balancer list`
2. Display the list of path route sets for the specified LB.

```
$ oci lb path-route-set list --load-balancer-id ocid1.loadbalancer.unique_ID
```

3. Display the details of a specific path route set.

```
$ oci lb path-route-set get --load-balancer-id "ocid1.load-balancer.unique_ID" \
--path-route-set-name PathRouteSet1

"data": {
  "name": "PathRouteSet1",
  "pathRoutes": [
    {
      "path": "/new/",
      "pathMatchType": {
        "matchType": "EXACT_MATCH"
      },
      "backendSetName": "BackendSet1"
    }
  ]
}
```

Editing a Path Route Set

This topic describes how to edit a path route set to update the path route rules.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to update the rules of a path route set.
3. On the LB details page, scroll to the Resources section and select Path Route Sets.
4. For the backend set that you want to modify, select the Actions menu and select the Edit option.

5. Make changes to the path route rules or add or delete rules.
6. Select the Save Changes button to update the path route set.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Load balancer OCID: `oci lb load-balancer list`
   - Path route set name: `oci lb path-route-set list`
2. Construct an argument for the `--path-routes` option.

   See Creating a Path Route Set.
3. Run the update path route set command.

   Syntax:

   ```
   oci lb path-route-set update --load-balancer-id load-balancer_OCID \
   --path-route-set-name name-of-path-route-set \
   --path-routes list-of-path-route-rules
   ```

   Example:

   ```
   $ oci lb path-route-set update --load-balancer-id ocid1.load-balancer.unique_ID \
   --path-route-set-name PathRouteSet1 \
   --path-routes file://./PathRouteSet1Rules.json

   WARNING: Updates to path-routes will replace any existing values. Are you sure you
   want to continue? [y/N]: y
   {
     "opc-work-request-id": "ocid1.workrequest.1749XC302P.unique_ID"
   }
   ```

   To view the updated path route set, use the `path-route-set get` command as shown in Viewing a Path Route Set Details.

Deleting a Path Route Set

This topic describes how to delete a load balancer (LB) path route set.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to delete the path route set.
3. On the LB details page, scroll to the Resources section and select Path Route Sets.
4. For the backend set that you want to delete, select the Actions menu and select Delete.
5. Confirm the Path Route Set deletion.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Load balancer OCID: `oci lb load-balancer list`
   - Path route set name: `oci lb path-route-set list`
2. Run the delete path route set command.

```
$ oci lb path-route-set delete --load-balancer-id ocid1.load-balancer.unique_ID \
--path-route-set-name PathRouteSet1 --force
```

Listeners

This section describes how to use listeners to check for incoming traffic on the load balancer (LB) IP address.

- Creating a Load Balancer Listener
- Editing a Load Balancer Listener
- Deleting a Load Balancer Listener

Creating a Load Balancer Listener

This topic describes how to configure a listener for a load balancer (LB). The listener checks for incoming traffic on the LB IP address. Configure at least one listener per traffic type: HTTP, HTTP/2, HTTPS, and TCP.

Prerequisites

1. Ensure that your VCN's security rules allow the listener to accept traffic.
2. Know the traffic protocols that the LB accepts.
   - Using the Compute Web UI: On the LB details page, scroll to the Resources section, and select the Create Listener button. The supported traffic protocols are shown on the Protocol drop-down list.
   - Using the OCI CLI: `oci lb protocol list -c compartment_OCID`
3. If the LB accepts HTTPS traffic, and you plan to select HTTPS, create a certificate and cipher suite for use in SSL configuration. See Adding a Load Balancer Certificate and Creating a Load Balancer SSL Cipher Suite .
4. At least one backend set must exist for this LB.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to create the listener.
3. On the LB details page, scroll to the Resources section, and select Listeners.
4. Select the Create Listener button.
5. Enter the following information. Name, protocol, port, and backend set are required. Other parameters, such as hostnames, path route sets, SSL certificate, and cipher suites can be added later.
   - **Name:** Enter a descriptive name for the Listener. The name must be unique and cannot be changed.
   - **Protocol:** Select the protocol to listen for (HTTP, HTTP/2, HTTPS, TCP) from the drop-down list.

     If you select HTTP/2, HTTPS, or TCP, an SSL section appears at the bottom of the dialog, after Idle Timeout in Seconds.
   - **Port:** A default port value, depending on the protocol you selected, is preselected. Use the up or down arrows to change the port value, or enter a value between 1 and 65,535.

- **Backend Set:** Select a backend set from the list.
- **Hostnames:** Select a hostname from the list.
- **Path Route Set:** Select a path route set from the list.
- **Idle Timeout in Seconds:** Use the up or down arrows to change the idle timeout value, or enter a value greater than or equal to 1.
- **SSL:** This item appears if you select HTTP/2, HTTPS, or TCP for protocol.
  - **Use SSL:** For HTTP/2 and HTTPS protocols, this box is prechecked and cannot be unchecked. For TCP protocol, this box is not checked. If you check this box, the following parameters appear, just as they do if you selected HTTP/2 or HTTPS protocol.

    Checking this box enables SSL handling for this listener. The following settings are required to associate an SSL certificate bundle with the listener to enable SSL handling.
  - **Certificates:** Select a certificate from the list.
  - **Verify peer certificate:** Check this box to enable peer certificate verification.
  - **TLS Version:** Select at least one TLS version.
  - **Cipher Suite:** Select a cipher suite from the list. The cipher suite details are shown below the list.

6. Select the Create Listener button in the dialog.

Using the OCI CLI

1. Get the LB OCID: `oci lb load-balancer list`
2. Run the create listener command.

   Syntax:

   ```
   oci lb listener create --default-backend-set-name backend-set-name \
   --load-balancer-id load-balancer_OCID \
   --name listener-name --port listener-port \
   --protocol listener-protocol
   ```

   Option values:

   • ***backend-set-name*** - The name of the associated backend set.

   • ***load-balancer_OCID*** - The OCID of the load balancer on which to add a listener.

   • ***listener_name*** - A user-friendly name for the listener. It must be unique and it can't be changed.

   • ***listener-port*** - The communication port number for the listener.

   • ***listener-protocol*** - The protocol on which the listener accepts connection requests.

   Example:

   This example shows only required parameters. Other parameters, such as hostnames, path route sets, SSL certificate, and cipher suites can be added later. Use the `-h` option for more information.

   ```
   $ oci lb listener create --default-backend-set-name PublicLB1-Backend-Set \
   --load-balancer-id ocid1.loadbalancer.unique_ID \
   --name LB1-Listener --port 80 --protocol HTTP

   {
   ```

```
 "opc-work-request-id": "ocid1.workrequest.oc1.pca.loadbalancer.unique_ID"
}
```

To view the listener details, use the `load-balancer get` command. In the following example, most of the command output is omitted to show only the listener details:

```
$ oci lb load-balancer get --load-balancer-id ocid1.loadbalancer.unique_ID
{
  "data": {
    ...
    "listeners": {
      "LB1_Listener": {
        "connection-configuration": {
          "backend-tcp-proxy-protocol-version": null,
          "idle-timeout": 60
          },
        "default-backend-set-name": "PublicLB1-Backend-Set",
        "hostname-names": null,
        "name": "LB1_Listener",
        "path-route-set-name": null,
        "port": 80,
        "protocol": "HTTP",
        "routing-policy-name": null,
        "rule-set-names": null,
        "ssl-configuration": null
        }
      },
    ...
    },
  "etag": "9326dbb5-d842-4975-9cfb-ced7717e92d6"
}
```

Editing a Load Balancer Listener

This topic describes how to change load balancer (LB) listener properties, such as the listener communication port used.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to modify a backend set.
3. On the LB details page, scroll to the Resources section and select Listeners.
4. For the LB listener that you want to edit, select the Actions menu, and select the Edit option.
5. In the Edit Listener dialog, you can change anything except the name of the listener.
6. Select the Save Changes button to update the LB Listener properties.

Using the OCI CLI

1. Get the LB OCID: `oci lb load-balancer list`
2. Run the update listener command.

   Syntax:

   ```
   oci lb listener update --default-backend-set-name default-backendset-name \
   --listener-name listener-name --load-balancer-id loadbalancer_OCID\
   --port port-integer --protocol protocol-text
   ```

Option values:

- ***default-backendset-name*** - The name of the associated backend set.
- ***listener-name*** - The name of the listener to update.
- ***loadbalancer_OCID*** - The OCID of the load balancer associated with the listener to update.
- ***port-integer*** - The communication port for the listener.
- ***protocol-text*** - The protocol on which the listener accepts connection requests.

Example:

```
$ oci lb listener update --default-backend-set-name PublicLB1-Backend-Set \
--listener-name LB1_Listener --load-balancer-id ocid1.loadbalancer.unique_ID \
--port 80 --protocol HTTP
```

To view the listener details, use `load-balancer get`. See Creating a Load Balancer Listener .

Deleting a Load Balancer Listener

This topic describes how to delete a load balancer (LB) listener.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to delete a listener.
3. On the LB details page, scroll to the Resources section and select Listeners.
4. For the listener that you want to delete, select the Actions menu, and select the Delete option.
5. Confirm the operation when prompted.

Using the OCI CLI

1. Get the LB OCID: `oci lb load-balancer list`
2. Run the delete listener command.

```
$ oci lb listener delete --load-balancer-id ocid1.loadbalancer.unique_ID \
--listener-name LB1-Listener --force
```

Health Checks

A load balancer (LB) backend set health check is a test to confirm the availability of LB backend servers. A health check can be a request or a connection attempt. The LB applies the health check policy, based on a configured time interval, to monitor the backend server set. If a server fails the health check, then the LB takes the server temporarily out of the balancing rotation. If the server later passes a subsequent health check, then the LB returns that backend server to the balancing rotation.

The health status of the specified backend set server is reported by the primary and standby load balancers.

Health checks are configured when you create a backend set. See Creating a Load Balancer Backend Set . This section describes how to view and update health check configuration.

For more information, including how to diagnose misconfigurations, see "Load Balancer Health Checks" in "Backend Configuration" in the Load Balancing Overview chapter of the *Oracle Private Cloud Appliance Concepts Guide*.

Viewing Health Status and Health Check Configuration

This topic describes how to view the overall health of a load balancer (LB) and backend set of servers and the specific health checker configuration values set for the backend set.

Overall Health

- Critical
- Warning
- Incomplete
- Pending
- OK

Health Checker Configuration Parameters

- Protocol - HTTP or TCP
- Port - The backend server port against which to run the health check
- Interval In Milliseconds - Time between health checks
- Timeout In Milliseconds - Maximum time to wait for the health check response
- Number of Retries - The number of retries to attempt before a backend server is considered "unhealthy"
- Status Code (HTTP only) - The code a healthy backend server should return
- URL Path (HTTP only, Optional) - The path against which to run the health check

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to view health status and configuration.

   On the LB details page, the Overall Health of the LB is shown in the second column of the Load Balancer Information tab.
3. On the LB details page, scroll to the Resources section and select Backend Sets.
4. Select the name of the backend set for which you want to view health status and configuration.

   On the backend set details page, the Overall Health of the servers in the backend set is shown in the first column of the Backend Set Information tab.
5. On the backend set details page, select the Backend Set Configuration tab.

   All health check configuration parameters listed at the beginning of this section are shown in the Health Checker Configuration column.
6. On the backend set details page, scroll to the Resources section and select Backends.

   The overall health of each backend server is shown in the Overall Health column in the middle of the table.

Using the OCI CLI

1. Gather the information you need to run the command:

- Compartment OCID: `oci iam compartment list`
- LB OCID: `oci lb load-balancer list`
- Backend set name: `oci lb backend-set list`

2. List the health statuses for all load balancers in the specified compartment.

```
$ oci lb load-balancer-health list --compartment-id compartment_OCID
```

3. Show the health status for the specified load balancer.

```
$ oci lb load-balancer-health get --load-balancer-id loadbalancer_OCID
```

4. Show the health status for the specified backend set.

```
$ oci lb backend-set-health get --load-balancer-id loadbalancer_OCID \
--backend-set-name backendset_name
```

5. Show the health status of the specified backend server.

```
$ oci lb backend-health get --load-balancer-id loadbalancer_OCID \
--backend-set-name backendset_name --backend-name backend_name
```

The ***backend_name*** is the IP address and port of the backend server to retrieve the health status for, such as 10.0.0.3:8080.

6. Show the health check policy information for the specified load balancer and backend set.

```
$ oci lb health-checker get --load-balancer-id loadbalancer_OCID \
--backend-set-name backendset_name
```

All health check configuration parameters listed at the beginning of this section are shown.

Editing Backend Set Health Check Configuration

This topic describes how to change load balancer (LB) backend set health check configuration.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Load Balancers.
2. Select the name of the LB for which you want to edit backend set health check parameters.
3. On the LB details page, scroll to the Resources section and select Backend Sets.
4. For the backend set that you want to modify, select the Actions menu, and select the Edit option.

   In the Edit Load Balancer Backend Set dialog, you can modify all health check configuration parameters listed at the beginning of Viewing Health Status and Health Check Configuration.
5. Select Update Backend Set to save the changes.

Using the OCI CLI

1. Gather the information you need to run the command:
   - LB OCID: `oci lb load-balancer list`
   - Backend set name: `oci lb backend-set list`
2. Update the health check policy for the specified load balancer and backend set.

   Syntax:

```
ci lb health-checker update --load-balancer-id loadbalancer_OCID \
--backend-set-name backendset_name --interval-in-millis integer \
--port integer --protocol [HTTPS | TCP] --retries integer \
--return-code integer --timeout-in-millis integer
```

Example:

```
$ oci lb health-checker update --load-balancer-id ocid1.loadbalancer.uniqueID \
--backend-set-name BackendSet1 --interval-in-millis 10000 \
--port 8080 --protocol HTTPS --retries 3 --return-code 200 \
--timeout-in-millis 3000 --url-path /healthcheck
```

Network Load Balancers

A network load balancer (NLB) on the Oracle Private Cloud Appliance provides automated traffic distribution from one entry point to multiple servers reachable from the virtual cloud network (VCN). NLBs, like LBs, offer a choice of using a public or private IP address and various load balancing policies.

For more general information about NLBs, see the Network Load Balancing Overview chapter in the *Oracle Private Cloud Appliance Concepts Guide*.

Managing a Network Load Balancer

This section describes how to create, update, view details, and delete a network load balancer (NLB).

- Creating a Network Load Balancer
- Editing a Network Load Balancer
- Viewing Network Load Balancer Details
- Deleting a Network Load Balancer

Creating a Network Load Balancer

This topic describes how to creat a network load balancer (NLB).

Using the Compute Web UI

1. Open the Navigation Menu, select Networking, and select Network Load Balancers.
2. Select the Create Network Load Balancer button to open the Create Network Load Balancer dialog.
3. Enter the following information:
   - **Name:** Enter a descriptive name for the NLB. The name does not need to be unique, and you can change it.
   - **Create in Compartment:** Select the compartment in which to create the NLB.
   - **Visibility Type:** Select either Public or Private Load Balancer.
     - If you select Private Load Balancer, the NLB receives a private IP address from the selected subnet.
     - If you select Public Load Balancer, the Select Public IP menu is shown. Select a public IP from the list. You might need to change the compartment above the menu. If the menu displays None Available or if you do not select a public IP from the list, a public IP is automatically assigned from the configured public IP range.

See Load Balancing for more information about private and public load balancers.

- **Subnet:** Select the name of the VCN and Subnet for the NLB. You might need to change the compartment above the menus.
- **Network Security Group:** (Optional) By default, the NLB is not attached to any NSG. Select the box labeled Enable Network Security Group to add this NLB to one or more NSGs.
  a. Select an NSG from the drop-down list. You might need to change the compartment to find the NSG you want.
  b. Click the Add Another NSG button if you want to attach to another NSG.
  c. To remove an NSG from the list, click the trash can to the right of that NSG. To remove the last NSG or all NSGs, uncheck the Enable Network Security Groups box.

4. **Tagging:** (Optional) Add defined or free-form tags for this NLB as described in Adding Tags at Resource Creation. Tags can also be applied later.
5. Select the Create Network Load Balancer button in the dialog. The details page of the new NLB is displayed.

Next Steps: On the NLB details page, scroll down to the Resources section and select resources to create to complete the configuration.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Compartment OCID (`oci iam compartment list`)
   - At least one subnet OCID (`oci network subnet list`)
2. Enter the create NLB command.

   Syntax:

   The following shows only the required parameters. Use `oci nlb network-load-balancer create -h` to get information about optional parameters such as backend sets and listeners.

```
oci nlb network-load-balancer create --compartment-id compartment_OCID \
--display-name name-of-network-load-balancer \
--subnet-id subnet_OCID
```

   Example:

   The following example creates a private NLB (the `--is-private` option value is `true` by default), and a private IP address is assigned from the specified subnet. See Load Balancing for more information about private and public load balancers.

```
$ oci nlb network-load-balancer create --compartment-id ocid1.compartment.unique_ID \
  --display-name Private_NLB1 --subnet-id ocid1.subnet.unique_ID

{
  "data": {
    "backend-sets": {},
    "compartment-id": "ocid1.tenancy.unique_ID",
    "defined-tags": {
      "Oracle-Tags": {
        "CreatedBy": "auser",
        "CreatedOn": "2025-01-28T23:12:58.28Z"
      },
    "display-name": "Private_NLB1",
```

```
    "freeform-tags": null,
    "hostnames": {},
    "id": "ocid1.networkloadbalancer.unique_ID",
    "ip-addresses": [
    {
      "ip-address": 10.10.1.16,
      "is-public"; false,
      "reserved-ip": null
    }
    ],
    "is-preserve-source-destination": false,
    "is-private": true,
    "lifecycle-details": null,
    "lifecycle-state": "ACTIVE",
    "listeners": {},
    "network-security-group-ids": null,
    "nlb-ip-version": "IPV4",
    "subnet-id":"ocid1.subnet.unique_ID",
    "system-tags": null,
    "time-created": "2025-01-28T23:12:58.000001+00:00",
    "time-updated": null
  },
"etag": "00c648d7-b654-4583-dbdb-k5oed55"
}
```


This output is the same as the output of the `oci nlb network-load-balancer get` command.

Next Steps: Complete the NLB configuration by adding resources using their separate commands, such as `oci nlb listener create`. For a list of commands, see `oci nlb -h`.

Editing a Network Load Balancer

You can change the network load balancer (NLB) name and tags.

Using the Compute Web UI

To add or update related resources such as backend sets or listeners, go to the NLB details page, scroll down to the resources section, and select the resource that you want to add or edit.

1. Open the Navigation menu, select Networking, and select Network Load Balancers.
2. For the NLB that you want to edit, select the Actions menu, and select the Edit option to open the Edit Network Load Balancer dialog.
3. Make your changes and select the Update Network Load Balancer button to update the NLB properties.

Using the OCI CLI

If you did not add resources such as backend sets or listeners when you created the NLB, add them by using their separate command, such as `oci nlb listener create`. If you did add resources when you created the NLB, update them by using their separate command, such as `oci nlb listener update`.

1. Get the NLB OCID (`oci nlb network-load-balancer list`)
2. Run the update NLB command.

   Example:

```
$ oci nlb network-load-balancer update \
 --network-load-balancer-id ocid1.networkloadbalancer.unique_ID \
 --display-name new_nlb_name
```

Viewing Network Load Balancer Details

You can view a list of existing network load balancers and view their details.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Network Load Balancers.
2. If necessary, select a different compartment from the compartment menu above the NLB list.
3. Select the name of the NLB to go to its details page.

   Alternatively, for the NLB for which you want to see the details, select the Actions menu and select the View details option.

Using the OCI CLI

1. Get the NLB OCID (`oci nlb network-load-balancer list`)
2. Run the get NLB command.

   Syntax:

   ```
   oci nlb network-load-balancer get --network-load-balancer-id networkloadbalancer_OCID
   ```

   The details of all the resources that have been created, such as backend sets, certificates, and listeners are included in the output.

Deleting a Network Load Balancer

You can delete a network load balancer (NLB) to remove it from service.

Using the Compute Web UI

1. Open the Navigation menu, select Networking, and select Network Load Balancers.
2. For the NLB that you want to delete, select the Actions menu, and select the Terminate option.
3. Confirm the operation when prompted.

Using the OCI CLI

1. Get the NLB OCID (`oci nlb network-load-balancer list`)
2. Run the delete NLB command.

   Syntax:

   ```
   $ oci nlb network-load-balancer delete --force \
   --network-load-balancer-id networkloadbalancer_OCID
   ```

Network Load Balancer Backend Sets

This section describes how to use backend sets to create logical entities consisting of a network load balancing policy, health check policy, and a list of backend servers for a Network Load Balancer resource.

Creating a Network Load Balancer Backend Set

You can create a backend set for an existing network load balancer. The backend set is a group of servers to which network traffic is load balanced. You can create backend servers *after* you create the backend set, or at the same time. This topic creates only the backend set.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Network Load Balancers.
2. Click the name of the Network Load Balancer for which you want to create the network load balancer backend set.
3. Click Backend Sets under Resources.
4. Any existing backend sets are listed, otherwise the list says **No data available**. To create a backend set, click Create Backend Set.
5. Enter the following:
   - **Name:** Specify a friendly name for the backend set. It must be unique within the network load balancer, and cannot be changed. Valid backend set names include only alphanumeric characters, dashes, and underscores. Backend set names cannot contain spaces. Avoid entering confidential information.
   - **Load Balancing Policy:** The IP Hash policy uses an incoming request's source IP address as a hashing key to route "non-sticky" traffic to the same backend server. The load balancer routes requests from the same client to the same backend server as long as that server is available. This policy honors server weight settings when establishing the initial connection. Select one of the following load balancing policies:
     - **5-Tuple hash:** This policy distributes incoming traffic based on 5-Tuple (source IP and port, destination IP and port, protocol) IP Hash.
     - **3-Tuple hash:** This policy ensures that requests from a particular client are always directed to the same backend server based on 3-Tuple (source IP, destination IP, protocol) IP Hash.
     - **2-Tuple hash:** This policy routes incoming traffic to the same backend server based on 2-Tuple (Source/Destination) IP Hash.
   - **Source Header Preservation:** The default value cannot be changed.
   - **IP Protocol Verion:** The network load balancer listener and backend set must use the same IP protocol version. Accepted values are: IPV4.
   - **Health Check:** Specify the parameters to confirm the health of backend servers in the set:
     - **Protocol:** Enter the protocol: TCP or HTTP. HTTP is valid for NLB health checks. When using TCP as the protocol, you can optionally provide the request data and the response data.
     - **Port:** Specify the backend server port against which to run the health check. You can enter the value '0' to have the health check use the backend server's traffic port.

- **Interval in MS:** Specify how frequently to run the health check in milliseconds. The default value is 10000 (10 seconds).
- **Timeout in MS:** Specify the maximum time in milliseconds to wait for a reply to a health check. A health check is successful only if a reply returns within this timeout period. The default is 3000 (3 seconds).
- **Number of Retries:** Specify the number of retries to attempt before a backend server is considered "unhealthy." This number also applies when recovering a server to the "healthy" state. The default is 3.
- **Status Code:** Specify the status code a healthy backend server must return.
- **URL Path (URI):** Specify a URL endpoint against which to run the health check.

6. Click the Create Backend Set button in the dialog. To display the details of the new backend set, view the backend set.

Using the OCI CLI

1. Get the NLB OCID (`oci nlb network-load-balancer list`)
2. Run the create backend set command.

   Syntax:

   ```
   oci nlb backend-set create --health-checker health-checker-parameters \
     --name backend-set-name --network-load-balancer-id network-load-balancer_OCID \
      --policy network-load-balancer-policy
   ```

   Where:

   • ***health-checker-parameters*** is the set of parameters associated with the health checker for this backend set.

   - **Protocol:** Enter the protocol: TCP. Configure your health check protocol to match your application or service. When using TCP as the protocol, you can optionally provide the request data and the response data.
   - **Port:** Specify the backend server port against which to run the health check. You can enter the value '0' to have the health check use the backend server's traffic port.
   - **Interval in MS:** Specify how frequently to run the health check in milliseconds. The default value is 10000 (10 seconds).
   - **Timeout in MS:** Specify the maximum time in milliseconds to wait for a reply to a health check. A health check is successful only if a reply returns within this timeout period. The default is 3000 (3 seconds).
   - **Number of Retries:** Specify the number of retries to attempt before a backend server is considered "unhealthy." This number also applies when recovering a server to the "healthy" state. The default is 3.
   - **Status Code:** Specify the status code a healthy backend server must return.
   - **URL Path (URI):** Specify a URL endpoint against which to run the health check.

   • ***backend-set-name*** is the name specific to the backend set.

   • ***network-load-balancer_OCID*** is the OCID of the NLB.

   • ***network-load-balancer-policy*** is the policy associated with the load balancer.

   Example:

   ```
   oci nlb backend-set create --health-checker '{"interval-in-
   milliseconds":10000,"port": 22,
   ```

```
"protocol": "TCP", "retries": 3, "timeoutInMillis": 3000}' --name PrivTCP_NLB1BESet
--network-load-balancer-id ocid1.networkloadbalancer.unique_ID \
--policy "TWO_TUPLE"

{
  "opc-work-request-id": "ocid1.workrequest.unique_ID"
}
```

The command returns a work request OCID. To view the backend set details, use `oci nlb backend-set list` with the NLB OCID to list all backend sets associated with the specified NLB, and then use `oci lb backend-set get` with the NLB OCID and backend set name to view the backend set details.

```
$ oci nlb backend-set list --network-load-balancer-id
ocid1.networkloadbalancer.unique_ID
$ oci nlb backend-set get --backend-set-name PrivTCP_NLB1BESet \
--network-load-balancer-id ocid1.networkloadbalancer.unique_ID
```

Viewing Network Load Balancer Backend Set Details

You can view a list of the backend sets of an existing network load balancer and view their details.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Network Load Balancers.
2. Click the name of the network load balancer (NLB) for which you want to list the existing backend set details.
3. Click the Backend Sets under Resources to display a list of any available backend sets.
4. If the NLB backend set exists, you can view its details by clicking on the backend set name or under the Action (three vertical dots) pull-down menu.

Using the OCI CLI

1. Gather the information you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
   - NLB OCID (`oci nlb network-load-balancer list --compartment-id` ***`compartment_OCID`***)
2. Run the `get` command.

   Use the backend set name and NLB OCID to view the details for the NLB backend set in the specified compartment.

> **Note:**
>
> For information about optional parameters, run the command with the `--help` option.

Syntax (entered on a single line):

```
oci nlb backend-set get --backend-set-name backend_set_name \
  --network-load-balancer-id network-load-balancer_OCID
```

Where:

• ***backend-set-name*** is the name specific to the backend set.

• ***network-load-balancer_OCID*** is the OCID of the load balancer associated with the backend set.

Example:

```
$ oci nlb backend-set get --backend-set-name PrivTCP_NLB1BESet \
  --network-load-balancer-id ocid1.networkloadbalancer..….….uniqueID

{
  "data": {
    "items": [
    {
      "backends": [],
      "health-checker": {
        "interval-in-millis": 10000,
        "port": 22,
        "protocol": "TCP",
        "request-data": "123",
        "response-body-regex": ".*",
        "response-data": "123",
        "retries": 3,
        "return-code": 200,
        "timeout-in-millis": 3000,
        "url-path": "/"
      },
      "ip-version": "IPV4",
      "is-preserve-source": false,
      "name": "PrivTCP_NLB1BESet",
      "policy": "TWO_TUPLE"
      }
    ]
  }
}
```

Editing a Network Load Balancer Backend Set

You can change network load balancer (NLB) backend set properties, such as the health checker parameters used.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Network Load Balancers.
2. Click the name of the network load balancer (NLB) for which you want to edit the backend set information.
3. Under Resources, click Backend Sets.
4. Select the name of the NLB backend set that you want to edit.
5. In the Actions list, click Edit to open the Edit Network Load Balancer Backend Set window.
6. Make allowable changes in the pop-up window.
7. Click Update Network Load Balancer Backend Set to update the NLB Backend Set properties.

Using the OCI CLI

1. Gather the information you need to run the command:

- Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
- NLB OCID (`oci nlb network-load-balancer list --compartment-id` ***`compartment_OCID`***)

2. Run the `update` command, which returns a work request ID.

> **Note:**
>
> For information about optional parameters, run the command with the `--help` option.

Syntax (entered on a single line):

```
oci nlb backend-set update --health-checker health-checker-parameters \
  --name backend-set-name --network-load-balancer-id network-load-balancer_OCID \
   --policy network-load-balancer-policy
```

Where:

• ***health-checker-parameters*** is the set of parameters associated with the health checker for this backend set.

- **Protocol:** Enter the protocol: TCP. Configure your health check protocol to match your application or service. When using TCP as the protocol, you can optionally provide the request data and the response data.
- **Port:** Specify the backend server port against which to run the health check. You can enter the value '0' to have the health check use the backend server's traffic port.
- **Internal in MS:** Specify how frequently to run the health check in milliseconds. The default value is 10000 (10 seconds).
- **Timeout in MS:** Specify the maximum time in milliseconds to wait for a reply to a health check. A health check is successful only if a reply returns within this timeout period. The default is 3000 (3 seconds).
- **Number of Retries:** Specify the number of retries to attempt before a backend server is considered "unhealthy." This number also applies when recovering a server to the "healthy" state. The default is 3.
- **Status Code:** Specify the status code a healthy backend server must return.
- **URL Path (URI):** Specify a URL endpoint against which to run the health check.
- **Response Body Regex:** Provide a regular expression for parsing the response body from the backend server.

• ***backend-set-name*** is the name specific to the backend set.

• ***network-load-balancer_OCID*** is the OCID of the NLB.

• ***network-load-balancer-policy*** is the policy associated with the load balancer.

Example (change policy to TWO_TUPLE):

```
$ oci nlb backend-set update --health-checker '{"intervalInMillis":10000,"port": 22,
\
  "protocol": "TCP", "retries":3, "timeoutInMillis": 3000}' --backend-set-name
PrivTCP_BackEndSet_1 \
  --network-load-balancer-id $Priv_NLB1 --policy "TWO_TUPLE"
```

```
WARNING: Updates to backends and health-checker and ssl-configuration and session-
persistence-
  configuration and lb-cookie-session-persistence-configuration will replace any
existing values.
  Are you sure you want to continue? [y/N]: y

{
  "opc-work-request-id": "ocid1.workrequest.xxx.networkloadbalancer......unique_ID"
}
```

> **Note:**
>
> The command returns a work request ID. To see the backend set results, you must list backend sets associated with the specified NLB and verify that the backend set parameters are changed. Use the ***oci nlb backend-set list*** command to view the backend set details.

```
oci nlb backend-set list --network-load-balancer-id ocid1.networkloadbalancer..….
….uniqueID

{
  "data": {
    "items": [
    {
      "backends": [],
      "health-checker": {
        "interval-in-millis": 10000,
        "port": 22,
        "protocol": "TCP",
        "request-data": "123",
        "response-body-regex": ".*",
        "response-data": "123",
        "retries": 3,
        "return-code": 200,
        "timeout-in-millis": 3000,
        "url-path": "/"
      },
      "ip-version": "IPV4",
      "is-preserve-source": false,
      "name": "PrivTCP_BackEndSet_1",
      "policy": "TWO_TUPLE"
      }
    ]
  }
}
```

Deleting a Network Load Balancer Backend Set

You can delete a network load balancer (NLB) Backend Set and remove it from service.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Network Load Balancers.
2. Click the name of the network load balancer (NLB) for which you want to delete the backend set.
3. Under Resources, click Backend Sets.

4. Select the name of the NLB backend set you want to delete.
5. Under the Actions column, click the three-dots pull-down menu, and select Delete.
6. Confirm the operation when prompted.

Using the OCI CLI

1. Gather the information you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
   - NLB OCID (`oci nlb network-load-balancer list --compartment-id` ***`compartment_OCID`***)
2. Run the `delete` command, which returns a work request ID.

   Syntax (entered on a single line):

```
$ oci nlb backend-set delete --backend-set-name HTTP_BckEndSet \
  --network-load-balancer-id ocid1.networkloadbalancer....….….….uniqueID
Are you sure you want to delete this resource? [y/N]: y
{
  "opc-work-request-id": "ocid1.workrequest.xxx.networkloadbalancer-.....….
….uniqueID"
}
```

> **Note:**
>
> To delete the LB backend set without verification, run the command with the `--force` option.

Network Load Balancer Backend Servers

This section describes how to manage backend servers for use with a network load balancer.

- Creating a Network Load Balancer Backend
- Viewing a Network Load Balancer Backend Details
- Editing a Network Load Balancer Backend
- Deleting a Network Load Balancer Backend

Creating a Network Load Balancer Backend

You must create backend servers (compute instances) for an existing network load balancer. The backend is one of a group of servers in a backend set to which network traffic is load balanced. You can create backends *after* you create the backend set, or at the same time. This topic creates only the backend.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Network Load Balancers.
2. Click the name of the Network Load Balancer for which you want to create the network load balancer backend.
3. Click Backendset under Resources.

4. Any existing backends are listed, otherwise the list says **No data available**. To create a backend server, click on the hyperlink of the BackendSet to add the backend.
5. Click Create Backend.
6. Select the way you are adding backends. Possible values are:
   - **Computed Instances** Backends are added by instance.
   - **IP Addresses** Backends are added by IP address.
7. For **Computed Instances**:
   - **Instance:** Enter the name of the backend.
   - **Port:** Enter 22 (TCP).
   - **Name:** Leave blank to take the default (ipaddress:port#).
   - **Weight:** Enter a weight in the range 1 to 100.
   - **Security Rules:** Select the security rules of the backend. Possible values are:
     - **Configure Manually** You manually configure security rules for the backend.
     - **Configure Automatically** The system automatically configures security rules for the backend.
8. For **IP Addresses**:
   - **IP Address:** Enter the IP address of the backend.
   - **Port:** Enter the port number.
   - **Name:** Enter the name of the backend.
   - **Weight:** Enter a weight in the range 1 to 100.
9. Click the Submit button in the dialog. To display the details of the new backend, view the backend.

Using the OCI CLI

1. Gather the information you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
   - NLB OCID (`oci nlb load-balancer list --compartment-id` ***`compartment_OCID`***)
2. Run the backend create command.

   Use the NLB OCID and backend set name to create the backend set for the LB in the specified compartment.

   > **Note:**
   >
   > For information about optional parameters, run the command with the `--help` option.

   Syntax (entered on a single line):

```
oci nlb backend create --backend-set-name [backend-set-name] --ip-address [text]\
  --network-load-balancer-id [network-load-balancer_OCID] \
  --port [port-number-integer]
```

Where:

• ***backend-set-name*** is the name specific to the backend set where the backend is to be added.

• ***ip-address*** is the IP address for the backend to be added.

• ***network-load-balancer_OCID*** is the OCID of the load balancer associated with the backend set and servers.

• ***port-number-integer*** is the port number associated with the backend.

Example:

```
oci nlb backend create --backend-set-name PrivTCP_NLB1BESet \
  --ip-address 10.10.1.13 \
  --network-load-balancer-id ocid1.networkloadbalancer..….….uniqueID \
  --port 22

{
  "opc-work-request-id": "ocid1.workrequest.....….….….uniqueID "
}
```

> **Note:**
>
> The command returns a work request ID. To see the backend set results, you must list backends associated with the specified NLB and verify that the backend created is listed. Use the ***oci nlb backend list*** command to view the backend details.

```
oci nlb backend list --network-load-balancer-id ocid1.networkloadbalancer.....….….
….uniqueID

{
  "data": {
    "items": [
    {
      "ip-address": "10.10.1.13",
      "is-backup": false,
      "is-drain": false,
      "is-offline": false,
      "name": "10.10.1.13:22",
      "port": 22,
      "target-id": "ocid",
      "weight": 1
      },
      {
      "ip-address": "10.10.1.14",
      "is-backup": false,
      "is-drain": false,
      "is-offline": false,
      "name": "10.10.1.14:22",
      "port": 22,
      "target-id": "ocid",
      "weight": 3
      },
    ]
  }
}
```

Viewing a Network Load Balancer Backend Details

You can view a list of the backends in an existing network load balancer backend set and view their details.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Network Load Balancers.
2. Click the name of the network load balancer (NLB) to which you want to list existing backends.
3. Under Resources, click the name of the existing Backend Set to view its details, such as IP address, port, and weight.

Using the OCI CLI

1. Gather the information you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
   - NLB OCID (`oci nlb network-load-balancer list --compartment-id` ***`compartment_OCID`***)
2. Run the list command.

   Use the NLB OCID and backend set name to list the backend details for the backend set.

> **Note:**
>
> For information about optional parameters, run the command with the `--help` option.

Syntax (entered on a single line):

```
oci nlb backend list --backend-set-name backend-set-name \
--load-balancer-id network-load-balancer_OCID
```

Where:

• ***backend-set-name*** is the name specific to the backend set where the backends are located.

• ***network-load-balancer_OCID*** is the OCID of the load balancer associated with the backends.

Example:

```
$ oci nlb backend list --backend-set-name PrivLB1_BckEndSet  \
--load-balancer-id ocid1.loadbalancer.....….….….uniqueID

{
  "data": {
    "items": [
    {
      "ip-address": "10.10.1.13",
      "is-backup": false,
      "is-drain": false,
```

```
      "is-offline": false,
      "name": "10.10.1.13:22",
      "port": 22,
      "target-id": "ocid",
      "weight": 1
    },
    {
      "ip-address": "10.10.1.14",
      "is-backup": false,
      "is-drain": false,
      "is-offline": false,
      "name": "10.10.1.14:22",
      "port": 22,
      "target-id": "ocid",
      "weight": 3
    },
    {
      "ip-address": "10.10.1.2",
      "is-backup": false,
      "is-drain": false,
      "is-offline": false,
      "name": "10.10.1.2:22",
      "port": 22,
      "target-id": "ocid",
      "weight": 1
    },
    {
      "ip-address": "10.10.1.3",
      "is-backup": false,
      "is-drain": false,
      "is-offline": false,
      "name": "10.10.1.3:22",
      "port": 22,
      "target-id": "ocid",
      "weight": 1
    }
    ]
  }
}
```


Editing a Network Load Balancer Backend

You can change some properties of a network load balancer (NLB) backend that's a member of a backend set.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Network Load Balancers.
2. Click the name of the network load balancer for which you want to edit backend properties.
3. Under Resources, click Backend Sets.
4. Click the name of the Backend Set with the backend that you want to edit.
5. Select the name of the NLB backend that you want to edit.
6. In the Actions list, click Edit to open the Edit Network Load Balancer Backend window.
7. Make allowable changes in the pop-up window.
8. Click Update Network Load Balancer Backend to update the NLB backend properties.

Using the OCI CLI

1. Gather the information you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
   - NLB OCID (`oci nlb network-load-balancer list --compartment-id` ***`compartment_OCID`***)
2. Run the update command, which returns a work request ID.

> **Note:**
>
> For information about optional parameters, run the command with the `--help` option.

Syntax (entered on a single line):

```
oci nlb backend update --backend-name backend-name \
--backend-set-name backend-set-name \
--network-load-balancer-id networkloadbalancer_OCID [PARAMETERS TO CHANGE]
```

> **Note:**
>
> The **is-drain**, **is-offline**, and **is-backup** parameters are ignored, but must be set to false.

Where:

• **backend-name** is the name of the backend associated with the backend set being edited.

• **backend-set-name** is the name of the backend set associated with the backend being edited.

• **networkloadbalancer_OCID** is the OCID of the load balancer associated with the backend set and backend.

Options:

• **max-wait-seconds** The maximum time to wait for the work request to reach the state defined by **wait-for-state**. Defaults to 1200 seconds.

• **wait-for-state** This operation asynchronously creates, modifies or deletes a resource and uses a work request to track the progress of the operation. Accepted values are: ACCEPTED, CANCELED, CANCELING, FAILED, IN_PROGRESS, SUCCEEDED.

• **wait-interval-seconds** Check every **--wait-interval-seconds** to see whether the work request has reached the state defined by **--wait-for-state**. Defaults to 30 seconds.

• **weight** The network load balancing policy weight assigned to the backend. Backend servers with a higher weight receive a larger proportion of incoming traffic.

Example (change the backend weight to 3):

```
$ oci nlb backend update --backend-name 10.0.0.3:8080 --backend-set-name example-
backend-set \
```

```
  --network-load-balancer-id ocid1.networkloadbalancer....….….….uniqueID \
  --weight 3

{
  "opc-work-request-id": "ocid1.workrequest.xxx....….….….uniqueID"
}
```

> **Note:**
>
> To see the backend update results, use the backend **get** command for that backend.

```
$ oci nlb backend get --backend-name 172.16.0.151:22 --backend-set-name
example_backend_set \
--network-load-balancer-id ocid1.networkloadbalancer....….….….uniqueID

{
  "data": {
    "ip-address": "172.16.0.151:222",
    "is-backup": false,
    "is-drain": false,
    "is-offline": false,
    "name": "172.16.0.151:222",
    "port": 22,
    "target-id": "ocid",
    "weight": 3
  },
}
```

Deleting a Network Load Balancer Backend

You can delete a network load balancer (NLB) Backend from a Backend Set and remove it from service.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Network Load Balancers.
2. Click the name of the network load balancer for which you want to delete the backend server.
3. Under Resources, click Backend Sets.
4. Click the name of the Backend Set that contains the backend you want to delete in order to view the backend set details.
5. Select the name of the backend you want to delete.
6. Under the Actions column, click the three-dots pull-down menu, and select Delete.
7. Confirm the operation when prompted.

Using the OCI CLI

1. Gather the information you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
   - NLB OCID (`oci nlb network-load-balancer list --compartment-id` ***`compartment_OCID`***)

2. Run the `nlb backend delete` command.

   Syntax (entered on a single line):

   ```
   $ oci nlb backend delete --backend-name backend-name --backend-set-name backend-set-name \
   --network-load-balancer-id network-load-balancer_OCID
   ```

   > **Note:**
   >
   > To delete the NLB backend server without verification, run the command with the `--force` option.

   Where:

   • **backend-name** is the name of the backend associated with the backend set.

   • **backend-set-name** is the name of the backend set associated with the backend being deleted.

   • **network-load-balancer_OCID** is the OCID of the network load balancer associated with the backend set and backend.

   Example:

   ```
   $ oci nlb backend delete --backend-name 172.16.0.154:80 --backend-set-name
   PubLB1_BckEndSet \
   --network-load-balancer-id ocid1.networkloadbalancer....….….….uniqueID
   Are you sure you want to delete this resource? [y/N]: y
   ```

Network Load Balancer Listeners

This section describes how to use listeners to check for incoming traffic on the network load balancer's IP address.

- Creating a Network Load Balancer Listener
- Editing a Network Load Balancer Listener
- Deleting a Network Load Balancer Listener

Creating a Network Load Balancer Listener

A listener waits for traffic to arrive for an IP address and distributes the traffic to the backend set servers of the NLB. Configure at least one listener for each traffic type. Ensure that your VCN's security rules allow the listener to accept traffic.

Using the Compute Web UI

1. On the Navigation menu. select Networking, and select Network Load Balancers.
2. In the list, select the name of the Network Load Balancer for which you want to create the listener.
3. On the network load balancer's details page, scroll to the Resources section, and select Listeners.
4. On the Listeners list, select the Create Listener button.
5. Enter the following information:

- **Name:** Enter a descriptive name for the Listener.
- **Protocol:** Select TCP from the drop-down list.
- **Port:** Enter the port you want to use. For TCP, the default port value 22 is preselected. Use the up or down arrows to change the port value, or enter a value between 1 and 65,535.
- **Backend Set:** Select the backend set for the listener from the pull-down list. If the value is None Available, then you haven't yet created any NLB backend sets and must do so before this parameter can be configured.
- **IP Version:** The default IP Version 4 is preselected.

6. Click the Create Listener button in the dialog. To display the details of the listener, you must view the details for the network load balancer.

Using the OCI CLI

1. Gather the information you need to run the command:
   - Compartment OCID (`oci iam compartment list`)
   - NLB OCID (`oci nlb network-load-balancer list`)
   - NLB accepted protocol list (`oci nlb protocol list`)
2. Run the create listener command.

   Syntax:

   ```
   oci nlb listener create --default-backend-set-name backend-set-name \
   --network-load-balancer-id network-load-balancer_OCID \
   --name listener-name --port listener-port \
   --protocol listener-protocol
   ```

   Where:

   • ***backend-set-name*** is the name of the associated backend set.

   • ***network-load-balancer_OCID*** is the OCID of the load balancer on which to add a listener.

   • ***listener-name*** is a user-friendly name for the listener. It must be unique and it cannot be changed.

   • ***listener-port*** is the communication port integer for the listener.

   • ***listener-protocol*** is the protocol on which the listener accepts connection requests.

   For information about optional parameters, use `oci nlb listener create -h`.

   Example:

   ```
   $ oci nlb listener create --default-backend-set-name PublicNLB1-Backend-Set \
   --network-load-balancer-id ocid1.networkloadbalancer.uniqueID \
   --name LB1-Listener --port 22 --protocol TCP

   {
    "opc-work-request-id": "ocid1.workrequest.oc1.pca.networkloadbalancer.uniqueID"
   }
   ```

   To view the listener details, use `oci nlb load-balancer get`:

   ```
   $ oci nlb listener get --listener-name PrivNLB_TCPListener \
   --network-load-balancer-id ocid1.networkloadbalancer.uniqueID
   {
   ```

```
  "data": {
    "default-backend-set-name": "PrivNLB_TCPListen",
    "ip-version": "IPV4",
    "name": "PrivNLB_TCPListener",
    "port": 22,
    "protocol": "TCP"
  },
}
```

Editing a Network Load Balancer Listener

You can change some network load balancer (NLB) listener properties.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Network Load Balancers.
2. Click on the Network Load Balancer for which you want to view listeners.
3. Under Resources, click Listeners.
4. Select the name of the NLB listener that you want to edit.
5. In the Actions list, click Edit to open the Edit Network Load Balancer Listener window.
6. Make allowable changes to the listener in the pop-up window.
7. Click Update Network Load Balancer Listener to update the NLB Listener properties.

Using the OCI CLI

1. Get the NLB OCID (`oci nlb network-load-balancer list`)
2. Run the update listener command.

   Syntax:

   ```
   oci nlb listener update --default-backend-set-name default-backendset-name \
   --listener-name listener-name --network-load-balancer-id networkloadbalancer_OCID\
   --port port-integer --protocol protocol-text
   ```

   Where:

   • **default-backendset-name** is the name of the associated backend set.

   • **listener-name** is the name of the listener to update. Example: example_listener

   • **networkloadbalancer_OCID** is the OCID of the load balancer associated with the listener to update

   • **port-integer** is the communication port for the listener. Example: 22.

   • **protocol-text** is the protocol on which the listener accepts connection requests. Example: TCP

   Example (change listener port to 222):

   ```
   $ oci nlb listener update --default-backend-set-name PublicLB1-Backend-Set \
     --listener-name NLB1_Listener --network-load-balancer-id \
     ocid1.networkloadbalancer.unique_ID \
     --port 222 --protocol TCP

   {
     "opc-work-request-id": "ocid1.workrequest.unique_ID"
   }
   ```

To view the listener details, use `oci nlb listener get`:

```
$ oci nlb listener get --listener-name NLB1_Listener \
--network-load-balancer-id  --network-load-balancer-id
ocid1.networkloadbalancer.unique_ID
{
  "data": {
    "default-backend-set-name": "PrivNLB_TCPListen",
    "ip-version": "IPV4",
    "name": "PrivNLB_TCPListener",
    "port": 22,
    "protocol": "TCP"
  },
}
```

Deleting a Network Load Balancer Listener

You can delete a network load balancer (NLB) Listener and remove it from service.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Network Load Balancers.
2. Click the name of the network load balancer (NLB) or which you want to list existing listeners.
3. Under Resources, click Listeners.
4. Select the name of the NLB listener you want to delete.
5. Under the Actions column, click the three-dots pull-down menu, and select Delete.
6. Confirm the operation when prompted.

Using the OCI CLI

1. Gather the information you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
   - NLB OCID (`oci nlb network-load-balancer list --compartment-id` ***`compartment_OCID`***)
2. Run the `delete` command.

   Syntax (entered on a single line):

   ```
   $ oci nlb listener delete --listener-name listener-name-text \
     --network-load-balancer-id ocid1.networkloadbalancer.....….….….uniqueID
   Are you sure you want to delete this resource? [y/N]: y
   ```

> **Note:**
>
> To delete the NLB listener without verification, run the command with the `--force` option.

Network Load Balancer Health Checks

A network load balancer (NLB) backend set health check is a test to confirm the availability of NLB backend servers. A health check can be a request or a connection attempt. The NLB applies the health check policy, based on a configured time interval, to monitor the backend server set. If a server fails the health check, then the NLB takes the server temporarily out of the balancing rotation. If the server later passes a subsequent health check, then the NLB returns the backend server to the balancing rotation.

The health status of the specified backend set server is reported by the primary and standby network load balancers.

You can perform the following network load balancer (NLB) health checks:

- Viewing Health Checker Status for All Network Load Balancers
- Viewing a Network Load Balancer Health Checker Status
- Viewing Network Load Balancer Health Checker Policy
- Editing Network Load Balancer Health Check Parameters
- Viewing Health of a Network Load Balancer Backend Set
- Viewing Health of a Network Load Balancer Backend Server

For general information about NLB, see the *Oracle Private Cloud Appliance Concepts Guide*.

Viewing Health Checker Status for All Network Load Balancers

You can view network load balancer (NLB) backend set health status indicators to report on the general health of your network load balancers and their resources.

The NLB provides health status indicators that use your health check policies to report on the general health of your NLBs and their components. You can see health status indicators and summaries in the NLBs, backend sets, and backend servers.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Network Load Balancers.
2. Click the name of the NLB for which you want to view health checker status.

Using the OCI CLI

1. Gather the information you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
2. Run the `network-load-balancer list` command, which lists the status of all NLBs in the compartment.

> **Note:**
>
> For information about optional parameters, run the command with the `--help` option.

Syntax (entered on a single line):

```
oci nlb network-load-balancer-health list --compartment-id compartment_OCID \
[OPTIONS]
```

Where [OPTIONS] are:

- **--all** Fetches all pages of results. If you provide this option, then you cannot provide the --limit option.
- **--from-json [text]** Provide input to this command as a JSON document from a file using the file://path-to/file syntax.
- **--limit [integer]** For list pagination. The maximum number of results per page or items to return, in a paginated “List” call.
- **--page [text]** The page token representing the page from which to start retrieving results.
- **--page-size [integer]** When fetching results, the number of results to fetch per call. Only valid when used with --all or --limit, and ignored otherwise.
- **--sort-by [text]** The field to sort by. Only one sort order can be provided. The default order for timeCreated is descending. The default order for displayName is ascending. If no value is specified, then timeCreated is the default. Accepted values are: displayName, timeCreated.
- **--sort-order [text]** The sort order to use, either ‘asc’ (ascending) or ‘desc’ (descending). Accepted values are: ASC, DESC.

Examples:

```
$ oci nlb network-load-balancer-health list --compartment-id ocid1.tenancy.....….….
….uniqueID

{
  "data": {
    "items": [
    {
      "network-load-balancer-id": "ocid1.networkloadbalancer.....….….….uniqueID2",
      "status": "UNKNOWN"
    },
    {
      "network-load-balancer-id": "ocid1.networkloadbalancer.....….….….uniqueID1",
      "status": "OK"
    }
    ]
  }
}

$ oci nlb network-load-balancer-health list --compartment-id ocid1.tenancy.....….….
….uniqueID \
    --sort-order ASC

{
  "data": {
    "items": [
    {
      "network-load-balancer-id": "ocid1.networkloadbalancer.....….….….uniqueID1",
      "status": "OK"
    },
    {
      "network-load-balancer-id": "ocid1.networkloadbalancer.....….….….uniqueID2",
      "status": "UNKNOWN"
```

```
    }
    ]
  }
}
```

Viewing a Network Load Balancer Health Checker Status

You can view the status (OK, warning, critical, unknown) of the health of the backend servers in an existing network load balancer (NLB).

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Network Load Balancers.
2. Click the name of the NLB for which you want to view existing load balancer backend set health parameters.
3. Under Resources, click Backend Sets.
4. Click the name of the Backend Set to view its health check details, such as OK or Critical, among others.

Using the OCI CLI

1. Gather the information you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
   - NLB OCID (`oci nlb network-load-balancer list --compartment-id` ***`compartment_OCID`***)
2. Run the **network-load-balancer-health get** command.

   Use the NLB OCID to list the details for the NLB in the specified compartment.

> **Note:**
>
> For information about optional parameters, run the command with the `--help` option.

Syntax (entered on a single line):

```
$ oci nlb network-load-balancer-health get --network-load-balancer-id \
  network-load-balancer_OCID
```

Where:

• **network-load-balancer_OCID** is the OCID of the network load balancer associated with the backend set.

Example:

```
$ oci nlb network-load-balancer-health get --network-load-balancer-id \
  ocid1.networkloadbalancer.....….….….uniqueID

{
 "data": {
 "critical-state-backend-names": [],
 "status": "OK",
 "total-backend-count": 2,
```

```
 "unknown-state-backend-names": [],
 "warning-state-backend-names": []
 }
}
```

Viewing Network Load Balancer Health Checker Policy

You can view the health checker policy parameters used by a network load balancer (NLB) to check backend set health.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Network Load Balancers.
2. Click the name of the NLB for which you want to view heath check policy.
3. Under Resources, click Backend Sets.
4. To view the health check parameters of a backend set, you can:
   a. Click View Details under the Actions column (three dots) pull down menu.
   b. Click the backend set to view its details.
5. Click Backend Set Configuration on the Backend-Set Details page under the backend-set name for which to you want to view the health check policy.

Using the OCI CLI

1. Gather the information you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
   - NLB OCID (`oci nlb network-load-balancer list --compartment-id` ***`compartment_OCID`***)
2. Run the health-checker `get` command and view the `health-checker` parameters.

   Use the NLB OCID to list the details for the backend set health checker policy in the specified compartment.

> **Note:**
>
> For information about optional parameters, run the command with the `--help` option.

Syntax (entered on a single line):

```
oci nlb health-checker get --backend-set-name backend-set-name-text \
 --network-load-balancer-id network-load-balancer_OCID
```

Where:

• ***backend-set-name*** is the name specific to the backend set.

• ***network-load-balancer_OCID*** is the OCID of the load balancer associated with the backend set.

Example:

```
$ oci nlb health-checker get --backend-set-name BckEndSet \
  --network-load-balancer-id ocid1.networkloadbalancer.. . .unique-id

{
  "data": {
    "health-checker": {
      "interval-in-millis": 10000,
      "port": 22,
      "protocol": "TCP",
      "request-data": null,
      "response-body-regex": ".*",
      "response-data": null,
      "retries": 3,
      "return-code": 200,
      "timeout-in-millis": 3000,
      "url-path": "/"
      },
  },
}
```

Editing Network Load Balancer Health Check Parameters

You can change network load balancer (NLB) backend server set health check properties, such as the health check interval.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Network Load Balancers.
2. Click the name of the NLB for which you want to edit existing backend set health check parameters.
3. Under Resources, click Backend Sets
4. To edit the health check parameters of a backend set, you can:
   a. Click Edit under the Actions column (three dots) pull down menu.
   b. Click the backend set to view its details, then click Edit to make allowable changes.
5. Click Save to save the changes.

Using the OCI CLI

1. Gather the information you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
   - NLB OCID (`oci nlb network-load-balancer list --compartment-id` ***`compartment_OCID`***)
2. Run the `nlb health-checker update` command.

> **Note:**
>
> For information about optional parameters, run the command with the `--help` option.

Syntax (entered on a single line):

```
oci nlb health-checker update --backend-set-name name-of-backend-set \
 --interval-in-millis integer-in-millis --max-wait-seconds wait-in-seconds \
 --network-load-balancer-id networkloadbalancer_OCID --port port-integer \
 --protocol protocol-text --response-body-regex expression-text \
 --retries retries-integer --return-code integer \
 --timeout-in-millis integer-in-millis --url-path text \
 --wait-for-state text --wait-interval-secondx integer
```

> **Note:**
>
> The **update** command returns a work request ID. To see the command results, use the NLB backend set **get** command.

Where:

- **name-of-backend-set** is the name of the backend set associated with the health check policy to be edited. Example: example_backend_set.
- **networkloadbalancer_OCID** is the OCID of the network load balancer associated with the backend set health status to be edited.

OPTIONS:

--from-json [text] provides input to this command as a JSON document from a file using the **file://path-to/file** syntax.

--if-match [text] optimistic concurrency control is desired, in the PUT or DELETE call for a resource, set the if-match parameter to the value of the etag from a previous GET or POST response for that resource.

--interval-in-millis [integer-in-millis] is the interval between health checks, in milliseconds. Example: 10000 (10 seconds).

--max-wait-seconds [wait-in-seconds] is the maximum time to wait for the work request to reach the state defined by **--wait-for-state**. Defaults to 1200 seconds.

--port [port-integer] is the backend server port against which to run the health check. Example: 22.

--protocol [protocol-text] is the protocol the health check must use. Example: TCP.

--response-body-regex [expression-text] is a regular expression for parsing the response body from the backend server. Example: ^((?!false).|\s)*$

--retries [retries-integer] is the number of retries to attempt before a backend server is considered “unhealthy”. This number also applies when recovering a server to the “healthy” state. Example: 3

--return-code [integer] is the status code a healthy backend server should return. Example: 200.

--timeout-in-millis [integer-in-millis] is the maximum time in milliseconds to wait for a reply to a health check. A health check is successful only if a reply returns within this timeout period. Example:3000

--url-path [text] is the path against which to run the health check. Example: /healthcheck

--wait-for-state [text] asynchronously creates, modifies, or deletes a resource and uses a work request to track the progress of the operation. Specify this option to perform the action and then wait until the work request reaches a certain state. Accepted values are: ACCEPTED, CANCELED, CANCELING, FAILED, IN_PROGRESS, SUCCEEDED

--wait-interval-seconds [integer] checks every --wait-interval-seconds to see whether the work request has reached the state defined by --wait-for-state. Defaults to 30 seconds.

Example updating health-checker internal to 10000 milliseconds (10 seconds):

```
$ oci nlb health-checker update --backend-set-name PrivTCP_BackEndSet \
  --interval-in-millis 10000 --load-balancer-id ocid1.loadbalancer.....….….
….uniqueID

{
"opc-work-request-id": "ocid1.workrequest.....….….….uniqueID"
}

$ oci nlb backend-set get --backend-set-name PrivTCP_BackEndSet \
  --network-load-balancer-id ocid1.loadbalancer.. . .unique-id

{
  "data": {
    "backends": [
    {
      "ip-address": "10.10.1.2",
      "is-backup": false,
      "is-drain": false,
      "is-offline": false,
      "name": "nlbserver1",
      "port": 22,
      "target-id": "ocid",
      "weight": 1
    },
    {
      "ip-address": "10.10.2.3",
      "is-backup": false,
      "is-drain": false,
      "is-offline": false,
      "name": "nlbserver4",
      "port": 22,
      "target-id": "ocid",
      "weight": 1
    },
    ],
    "health-checker": {
      "interval-in-millis": 10000,
      "port": 22,
      "protocol": "TCP",
      "request-data": null,
      "response-body-regex": ".*",
      "response-data": null,
      "retries": 3,
      "return-code": 200,
      "timeout-in-millis": 3000,
      "url-path": "/"
      },
    "ip-version": "IPV4",
    "is-preserve-source": false,
    "name": "PrivTCP_BackEndSet",
    "policy": "THREE_TUPLE"
  },
}
```

Viewing Health of a Network Load Balancer Backend Set

You can view backend set health status for a network load balancer (NLB).

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Network Load Balancers.
2. Click the name of the NLB for which you want to view backend set health status.
3. Under Resources, click Backend Sets
4. Click the Backend Set to view the backend set health status for an NLB.

Using the OCI CLI

1. Gather the information you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
   - NLB OCID (`oci nlb network-load-balancer list --compartment-id` ***`compartment_OCID`***)
2. Run the `backend-set-health get` command.

> **Note:**
>
> For information about optional parameters, run the command with the `--help` option.

Syntax (entered on a single line):

```
oci nlb backend-set-health get --backend-set-name backend_set_name \
  --network-load-balancer-id network-load-balancer_OCID [OPTIONS]
```

Where:

• ***backend-set-name*** is the name specific to the backend set.

• ***network-load-balancer_OCID*** is the OCID of the network load balancer associated with the backend set.

Where [OPTIONS] are:

• **--from-json [text]** Provide input to this command as a JSON document from a file using the file://path-to/file syntax.

Example:

```
$ oci nlb backend-set-health get --backend-set-name PubTCP_BackEndSet_1 \
    --network-load-balancer-id ocid1.networkloadbalancer.. . .unique-id

{
  "data": {
    "critical-state-backend-names": [],
    "status": "OK",
    "total-backend-count": 4,
    "unknown-state-backend-names": [],
    "warning-state-backend-names": []
  },
}
```

Viewing Health of a Network Load Balancer Backend Server

You can view backend server health status for backend in a network load balancer (NLB).

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Network Load Balancers.
2. Click the name of the NLB for which you want to view backend set health status.
3. Under Resources, click Backend Sets
4. Click the Backend Set to view the backend health status for a backend in the NLB.

Using the OCI CLI

1. Gather the information you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
   - NLB OCID (`oci nlb network-load-balancer list --compartment-id` ***`compartment_OCID`***)
2. Run the `backend-health get` command.

> **Note:**
>
> For information about optional parameters, run the command with the `--help` option.

Syntax (entered on a single line):

```
oci nlb backend-health get --backend-name backend_name \
  --backend-set-name backend_set_name \
  --network-load-balancer-id network-load-balancer_OCID [OPTIONS]
```

Where:

• ***backend-name*** is the name specific to the backend, such as 10.10.1.13:22.

• ***backend-set-name*** is the name specific to the backend set.

• ***network-load-balancer_OCID*** is the OCID of the network load balancer associated with the backend set.

Where [OPTIONS] are:

• **--from-json [text]** Provide input to this command as a JSON document from a file using the file://path-to/file syntax.

Example:

```
$ oci nlb backend-health get --backend-name 10.10.1.13:22 \
    --backend-set-name PubTCP_BackEndSet_1 \
    --network-load-balancer-id ocid1.networkloadbalancer.. . .unique-id

{
  "data": {
    "health-check-results": [
    {
```

```
      "health-check-status": "OK",
      "timestamp": "2023-10-04T12:37:54.934773+00:00"
    },
    {
      "health-check-status": "OK",
      "timestamp": "2023-10-04T12:37:54.934777+00:00"
    }
    ],
  "status": "OK"
}
```


Viewing Network Load Balancer Work Request Errors

Many of the configuration steps used to create and configure network load balancer (NLB) operation do not take effect immediately. In these cases, the request initiates an asynchronous work flow known as a work request to carry out the operation.

Because of the asynchronous nature of work request fulfillment, it is not always obvious that a configuration step has failed with an error. The failed step is often not revealed until the next step, dependent on the success of the first, is attempted,

You can view NLB work request status indicators to find out if a work request has failed with an error. Using these methods, you can check the progress of each operation, whether or not it resulted in a failed state, which step it failed on, and the reason for the failure.

Using the Compute Web UI

1. Open the Navigation Menu. Under Networking, click Network Load Balancers.
2. Click the name of the NLB for which you want to view work request errors.
3. In the Resources list, click on Work Requests.
4. From the listing of work requests, for each work request you can view:
   - Type of work request
   - State of the work request (Succeeded, Failed, and so on)
   - Start and finish time stamp
5. From the View Details page of each work request, you can view:
   - General information about the work request, such as type
   - OCID of the work request
   - Error detail of the work request failed (nothing is displayed for non-failure status)
   - Start and finish time stamp

Using the OCI CLI

1. Gather the information you need to run the command:
   - Compartment OCID (`oci iam compartment list --compartment-id-in-subtree true`)
2. Run the `nlb work-request list` command, which lists the OCIDs for all work requests for all the NLBs in the compartment.

> **Note:**
>
> For information about optional parameters, run the command with the `--help` option.

Syntax (entered on a single line):

```
oci nlb work-request-errors list --work-request-id workrequest_OCID \
  --compartment-id tenancy _OCID [OPTIONS]
```

Where [OPTIONS] are:

- **--all** Fetches all pages of results. If you provide this option, then you cannot provide the --limit option.
- **--from-json [text]** Provide input to this command as a JSON document from a file using the file://path-to/file syntax.
- **--limit [integer]** For list pagination. The maximum number of results per page or items to return, in a paginated “List” call.
- **--page [text]** The page token representing the page from which to start retrieving results.
- **--page-size [integer]** When fetching results, the number of results to fetch per call. Only valid when used with --all or --limit, and ignored otherwise.

Example: (Attempt to create an NLB with an invalid Subnet ID, see that the work request has failed, and display the error details):

```
$ oci nlb network-load-balancer create --display-name Priv_NLB1  \
  --compartment-id ocid1.tenancy....….….….uniqueID --subnet-id ocid1.subnet....….….
….uniqueID \
  --is-private true

{
  "data": {
    "backend-sets": {},
    "compartment-id": "ocid1.tenancy.1....….….….uniqueID",
    "defined-tags": {},
    "display-name": "Priv_NLB1",
    "freeform-tags": {},
    "id": "ocid1.networkloadbalancer.....….….….uniqueID",
    "ip-addresses": null,
    "is-preserve-source-destination": false,
    "is-private": true,
    "lifecycle-details": null,
    "lifecycle-state": "CREATING",
    "listeners": {},
    "network-security-group-ids": null,
    "nlb-ip-version": "IPV4",
    "subnet-id": "ocid1.subnet.....….….….uniqueID",
    "system-tags": null,
    "time-created": "2023-10-18T12:58:34.000001+00:00",
    "time-updated": null
    },
  "opc-work-request-id": "ocid1.workrequest.....….….….uniqueID"
}

$ oci nlb work-request get --work-request-id ocid1.workrequest.....….….….uniqueID
```

```
{
  "data": {
    "compartment-id": "ocid1.tenancy.....….….….uniqueID",
    "id": "ocid1.workrequest.1.....….….….uniqueID",
    "operation-type": "CREATE_NETWORK_LOAD_BALANCER",
    "percent-complete": 100.0,
    "resources": [
    {
      "action-type": "CREATED",
      "entity-type": "NetworkLoadbalancer",
      "entity-uri": null,
      "identifier": "ocid1.networkloadbalancer.1.....….….….uniqueID"
    }
    ],
    "status": "FAILED",
    "time-accepted": "2023-10-18T12:58:34.701224+00:00",
    "time-finished": "2023-10-18T12:58:34.977166+00:00",
    "time-started": "2023-10-18T12:58:34.715807+00:00"
  }
}

$ oci nlb work-request-error list --compartment-id ocid1.tenancy.....….….….uniqueID \
  --work-request-id ocid1.workrequest.....….….….uniqueID

{
  "data": {
    "items": [
    {
      "code": "INTERNAL_ERROR",
      "message": "Error occurred. Network response code: 404.Error Message:
        'No Subnet was found'",
      "timestamp": "2023-10-18T12:58:34.948927+00:00"
    }
    ]
  }
}
```

6
Compute Images

An image is a required resource for creating an instance. Private Cloud Appliance provides some images and enables you to use other images that you create:

- **Platform images:** Private Cloud Appliance includes some images such as Oracle Linux and Oracle Solaris images. These included images are called platform images. Platform images are available in every compartment of every tenancy. Platform images do *not* need to be downloaded or imported by a Compute Enclave user in order to access them to create an instance.

  To make platform Images available in the Compute Enclave, a Service Enclave administrator must import images as described in Providing Platform Images in the *Oracle Private Cloud Appliance Administrator Guide*.

  > **Note:**
  >
  > Only the three most recently published versions of each major distribution are listed in the Compute Web UI and by the `compute image list` command. After an upgrade or patch, older versions might no longer be listed, but they are still accessible. To use an older image version, use the OCI CLI and specify the OCID of the image. One way to get the OCID of an older image is from the output of the `compute instance get` command of an instance that is using the image.

- **Custom images from existing instances:** You can create a custom image of the boot disk of an instance. Instances that you launch from such an image include the customizations, configuration, and software installed on the instance when you created the image. See Creating an Image from an Instance.
- **Bring your own image:** You can use your own versions of operating systems to create instances if the Private Cloud Appliance hardware supports it. See Bring Your Own Image (BYOI).

> **Note:**
>
> Images for Private Cloud Appliance must have paravirtualized network devices and boot volumes. Otherwise, image import will fail.

For conceptual information and important limitations, refer to "Compute Images" in the Compute Instance Concepts chapter of the *Oracle Private Cloud Appliance Concepts Guide*.

Initial User Account for Platform Images

After you launch an instance from a platform image, you initially connect to the instance using `ssh` with initial user account `opc`.

The SSH connection is authenticated using your SSH key pair that is used during instance launch. For more information, see Connecting to a Compute Instance.

Listing Images and Viewing Details

In both the Compute Web UI and the OCI CLI, platform images are listed first, followed by custom images. The list of platform images includes the three most recently published versions of each major distribution. Any older versions that were previously listed are still available by specifying the image OCID.

Using the Compute Web UI

1. On the Dashboard, click the Compute/View Images button.
2. Use the Compartment drop-down menu above the image list to select the compartment where you want to list images.
3. To see details of an image, click the name of the image in the list.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Compartment OCID (`oci iam compartment list`)
2. Run the image list command.

   Syntax:

   ```
   oci compute image list --compartment-id compartment_OCID
   ```

   The newest three published versions of each platform image are listed, and all custom images in the specified compartment are listed.

   Example:

   Platform images do not show a compartment OCID because platform images are available in all compartments.

   ```
   oci compute image list --compartment-id ocid1.compartment.unique_ID
   {
     "data": [
       {
         "agent-features": null,
         "base-image-id": null,
         "billable-size-in-gbs": null,
         "compartment-id": "",
         "create-image-allowed": true,
         "defined-tags": {},
         "display-name": "uln-pca-Oracle-Linux-8-2022.08.29_0.oci",
         "freeform-tags": {},
         "id": "ocid1.image.unique_ID",
         "launch-mode": "PARAVIRTUALIZED",
         "launch-options": {
           "boot-volume-type": "PARAVIRTUALIZED",
           "firmware": "UEFI_64",
           "is-consistent-volume-naming-enabled": false,
           "is-pv-encryption-in-transit-enabled": false,
           "network-type": "PARAVIRTUALIZED",
           "remote-data-volume-type": "PARAVIRTUALIZED"
         },
         "lifecycle-state": "AVAILABLE",
         "listing-type": null,
   ```

```
      "operating-system": "OracleLinux",
      "operating-system-version": "8",
      "size-in-mbs": 47694,
      "time-created": "2022-10-28T20:02:38.833966+00:00"
    },
...
    {
      "agent-features": null,
      "base-image-id": "ocid1.bootvolume.unique_ID",
      "billable-size-in-gbs": null,
      "compartment-id": "ocid1.compartment.unique_ID",
      "create-image-allowed": true,
      "defined-tags": {},
      "display-name": "Sales Team Image",
      "freeform-tags": {},
      "id": "ocid1.image.unique_ID",
      "launch-mode": "PARAVIRTUALIZED",
      "launch-options": {
        "boot-volume-type": "PARAVIRTUALIZED",
        "firmware": "BIOS",
        "is-consistent-volume-naming-enabled": false,
        "is-pv-encryption-in-transit-enabled": false,
        "network-type": "PARAVIRTUALIZED",
        "remote-data-volume-type": "PARAVIRTUALIZED"
      },
      "lifecycle-state": "AVAILABLE",
      "listing-type": null,
      "operating-system": "CUSTOM",
      "operating-system-version": "CUSTOM",
      "size-in-mbs": 51200,
      "time-created": "2021-09-17T18:26:03.221604+00:00"
    },
}
```

3. To see this information for just one image, run the image get command with the OCID of the image. You can use the image list command to get the OCID of the image.

   Syntax:

   ```
   oci compute image get --image-id ocid1.image.unique_ID
   ```

Managing Custom Images

This section describes the following operations that can be performed on custom images:

- Edit and delete
- Move to a different compartment
- Share across tenancies
- Upload, import, and export in the following ways:
  - Upload an image from a local file system to an Object Storage bucket
  - Import an image from an Object Storage bucket to a compartment
  - Export an image to an Object Storage bucket
  - Export an image to a URL
  - Import an image from a URL

To create a custom image, see Creating an Image from an Instance and Bring Your Own Image (BYOI).

Updating the Image Name

You can only edit custom images. You cannot change the name of a platform image.

Using the Compute Web UI

1. On the Dashboard, click the Compute/View Images button.
2. Use the Compartment drop-down menu above the image list to select the compartment where you want to list images.
3. Use one of the following methods to open the Update Image dialog.
   - For the image that you want to update, click the Actions menu, and click the Edit option.
   - Click the name of the image that you want to update. On the image details page, click the Controls menu and click Edit Details.
4. In the Update Image dialog, modify the image name.

   The image name does not need to be unique.
5. Click the Save Changes button.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Image OCID (`oci compute image list`)
2. Run the image update command.

   Syntax:

   ```
   $ oci compute image update --image-id ocid1.image.unique_ID --display-name new_name
   ```

Moving an Image to a Different Compartment

To move an image, you must use the OCI CLI.

Using the OCI CLI

1. Get the following information:
   - The OCID of the current compartment, and the OCID of the destination compartment:

     ```
     # oci iam compartment list --compartment-id-in-subtree true
     ```
   - The OCID of the image that you want to move:

     ```
     # oci compute image list --compartment-id current_compartment_OCID
     ```
2. Run the image change compartment command.

   Syntax:

   ```
   oci compute image change-compartment \
   --compartment-id destination_compartment_OCID \
   --image-id image_OCID
   ```

Deleting an Image

Using the Compute Web UI

1. On the Dashboard, click the Compute/View Images button.
2. Use the Compartment drop-down menu above the image list to select the compartment where you want to list images.
3. For the custom image that you want to delete, click the Actions menu, then click Delete image.

   The image is deleted.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Image OCID (`oci compute image list`)
2. Run the image delete command.

   Syntax:

   ```
   oci compute image delete --image-id image_OCID
   ```

   Example:

   ```
   $ oci compute image delete --image-id ocid1.image.unique_ID
   Are you sure you want to delete this resource? [y/N]: y
   {
     "etag": "bbb9a3df-8f9d-47df-a419-f9d2de912b57",
     "opc-work-request-id": "ocid1.workrequest.unique_ID"
   }
   ```

Uploading an Image to an Object Storage Bucket

Advantages of storing an image in an Object Storage bucket are that you can implement object versioning or pre-authenticated requests as described in Managing Object Versioning and Using Pre-Authenticated Requests.

1. Create an Object Storage bucket as described in Creating a Bucket.
2. Upload an image from a local file system to the bucket. See Uploading an Object.
3. Import the image from the Object Storage bucket to a compartment so that the image is available to select when you launch an instance. See Importing an Image from an Object Storage Bucket.

Importing an Image from an Object Storage Bucket

You can import an image into a compartment from an Object Storage bucket.

Alternatively, you can import an image from a URL as described in Importing an Image from a URL.

Before You Begin

Ensure you have read access to the Object Storage bucket, and that the bucket contains the image that you want to use. See Managing Object Storage Buckets.

If the bucket does not contain the image that you want to use, perform the procedure described in Uploading an Image to an Object Storage Bucket.

Using the Compute Web UI

1. Go to the Custom Images page.
   - On the Dashboard, click Compute/View Images. In the menu on the left side of the Images page, click Custom Images.
   - In the navigation menu, click Compute, then click Custom Images.
2. On the Custom Images page, click the Import Image button.
3. In the Import Image dialog, enter the following information:
   - **Name:** Enter a descriptive name for the image.
   - **Create in Compartment:** Select the compartment where the image will be placed.
   - **Source Type:** Select the Import from an Object Storage Bucket option.
   - **Bucket:** Select a bucket. You might need to change the compartment to locate the bucket.
   - **Object Name:** Select the name of an image object from the list.
   - **Image Type:** Select one of the following options based on the type of image you are importing.
     - **VMDK:** Virtual machine disk file format (`.vmdk`), used for virtual machine disk images.
     - **QCOW2:** For disk image files (`.qcow2`) used by QEMU copy on write.
     - **OCI:** For Oracle Cloud Infrastructure files with a QCOW2 image and OCI metadata (`.oci`).
   - **Launch Mode:** Paravirtualized is the default and cannot be changed.
   - **Tagging:** Optionally, add one or more tags to this image as described in Adding Tags at Resource Creation. Tags can also be applied later.
4. Click the Import Image button in the dialog.

   The imported image appears in the Custom Images list for the compartment, with a state of Importing. To track the progress of the operation, view the associated work request.

   When the import completes successfully, the image state changes to Available, and the image can be used to launch instances as described in Creating an Instance.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Compartment OCID:

     ```
     oci iam compartment list --compartment-id-in-subtree true
     ```
   - Object Storage bucket name:

     ```
     oci os bucket list -c compartment_OCID
     ```
   - Name of the image object in the bucket:

     ```
     oci os object list --bucket-name bucket_name
     ```
   - Object Storage namespace. See Obtaining the Object Storage Namespace.

- Image type. If the image is a VMDK or QCOW2 image, you must specify that type as the argument to the `--source-image-type` option. If you do not specify the `--source-image-type` option, the image is assumed to be an OCI file: a QCOW2 image and OCI metadata. If `--source-image-type` is not specified and the image type is not OCI, the import will fail.

2. Run the image import from object command.

   Syntax:

   ```
   oci compute image import from-object \
   --compartment-id compartment_OCID --bucket-name bucket_name \
   --name bucket_image_object_name --namespace namespace
   ```

   You can specify the `--display-name` option to give the imported image a custom name. The name does not need to be unique, and you can change it later. You cannot use a platform image name as a custom image name.

> **Important:**
>
> If you are importing a Microsoft Windows image, specify the `--operating-system` option and include the case-insensitive string "Windows" in the value to ensure optimal performance of the instance.
>
> If you specify the `--operating-system` option and this is *not* a Microsoft Windows image, make sure the value does not contain the case-insensitive string "Windows".

Importing an Image from a URL

You can import an image into a compartment by specifying the URL of the image file.

Alternatively, you can import an image from an Object Storage bucket as described in Importing an Image from an Object Storage Bucket.

Before You Begin

Get the URL that you need for this procedure. Ensure that the URL is accessible from your tenancy.

Using the Compute Web UI

1. On the Dashboard, click Compute/View Images. In the menu on the left side of the Images page, click Custom Images.
2. On the Custom Images page, click the Import Image button.
3. In the Import Image dialog, enter the following information:
   - **Name:** Enter a descriptive name for the image.
   - **Create in Compartment:** Select the compartment where the image will be placed.
   - **Source Type:** Select the Import from an Object Storage URL option.
   - **Object Storage URL:** Enter the URL of the image. The URL does not need to be an Object Storage URL. It can be any URL that provides access to the image.
   - **Image Type:** Select one of the following options based on the type of image you are importing.

  - **VMDK:** Virtual machine disk file format (`.vmdk`), used for virtual machine disk images.
  - **QCOW2:** For disk image files (`.qcow2`) used by QEMU copy on write.
  - **OCI:** For Oracle Cloud Infrastructure files with a QCOW2 image and OCI metadata (`.oci`).
- **Launch Mode:** Paravirtualized is the default and cannot be changed.
- **Tagging:** Optionally, add one or more tags to this image as described in Adding Tags at Resource Creation. Tags can also be applied later.

4. Click the Import Image button in the dialog.

   The imported image appears in the Custom Images list for the compartment, with a state of Importing. To track the progress of the operation, view the associated work request.

   When the import completes successfully, the image state changes to Available, and the image can be used to launch instances as described in Creating an Instance.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Compartment OCID:

     ```
     oci iam compartment list --compartment-id-in-subtree true
     ```
   - The URL for the image. The URL does not need to be an Object Storage URL. It can be any URL that can be accessed from your tenancy.
   - Image type. If the image is a VMDK or QCOW2 image, you must specify that type as the argument to the `--source-image-type` option. If you do not specify the `--source-image-type` option, the image is assumed to be an OCI file: a QCOW2 image and OCI metadata. If `--source-image-type` is not specified and the image type is not OCI, the import will fail.
2. Run the image import from object URI command.

   Syntax:

   ```
   oci compute image import from-object-uri \
   --compartment-id compartment_OCID --uri URL_for_image
   ```

   You can specify the `--display-name` option to give the imported image a custom name. The name does not need to be unique, and you can change it later. You cannot use a platform image name as a custom image name.

   > **Important:**
   >
   > If you are importing a Microsoft Windows image, specify the `--operating-system` option and include the case-insensitive string "Windows" in the value to ensure optimal performance of the instance.
   >
   > If you specify the `--operating-system` option and this is *not* a Microsoft Windows image, make sure the value does not contain the case-insensitive string "Windows".

   Example:

```
$ oci compute image import from-object-uri \
--compartment-id compartment_OCID \
--uri http://fqdn_or_ip_address/compute_images/uln-pca-Oracle-
Linux-8-2022.02.25_0.oci \
--display-name "Oracle Linux 8 2-25-22"
{
  "data": {
    "agent-features": null,
    "base-image-id": null,
    "billable-size-in-gbs": null,
    "compartment-id": "ocid1.compartment.unique_ID",
    "create-image-allowed": true,
    "defined-tags": {},
    "display-name": "Oracle Linux 8 2-25-22",
    "freeform-tags": {},
    "id": "ocid1.image.unique_ID",
    "launch-mode": "PARAVIRTUALIZED",
    "launch-options": null,
    "lifecycle-state": "IMPORTING",
    "listing-type": null,
    "operating-system": "UNAVAILABLE",
    "operating-system-version": "UNAVAILABLE",
    "size-in-mbs": 0,
    "time-created": "2022-04-19T20:44:35.163119+00:00"
  },
  "etag": "ab0c6265-c671-4ccb-a9b1-279d9437ba87",
  "opc-work-request-id": "ocid1.workrequest.unique_ID"
}
```

Exporting an Image to an Object Storage Bucket

You can export an image to an Object Storage bucket. You need write access to the bucket.

Alternatively, you can export an image to a object storage URL as described in Exporting an Image to a URL.

Exported images are a copy of the boot volume and metadata when the image was created.

Using the Compute Web UI

1. On the Dashboard, click Compute/View Images. In the menu on the left side of the Images page, click Custom Images.
2. Click the name of the custom image that you want to export. If the image that you want to export is not listed, use the compartment menu above the image list.
3. On the image details page, click the Controls menu and then click the Export Image option.
4. In the Export Custom Image dialog, enter the following information:
   - **Export Destination:** Select the Export to an Object Storage Bucket option.
   - **Bucket:** Select a bucket. If necessary, use the menu to select a different compartment.
   - **Object Name:** Enter a name for the exported image.
   - **Export Format:** Select one of the following options based on the type of image you are exporting.
     - **VMDK:** Virtual machine disk file format (`.vmdk`), used for virtual machine disk images.
     - **QCOW2:** For disk image files (`.qcow2`) used by QEMU copy on write.

- **OCI:** For Oracle Cloud Infrastructure files with a QCOW2 image and OCI metadata (`.oci`).

5. Click the Create Export button in the dialog.

   The image state changes to Exporting. Exporting a custom image copies the data to the Object Storage location that you specified. To track the progress of the operation, view the associated work request.

   You can still launch instances while the image is exporting, but you cannot delete the image until the export has finished.

   When the export is complete, the image state changes to Available. If the image state changes to Available, but you do not see the exported image in the Object Storage location you specified, the export failed, and you need to go through the steps again to export the image.

Using the OCI CLI

1. Ensure that a bucket is available.

   See Listing Buckets and Creating a Bucket.

2. Gather the information that you need to run the command:
   - Object Storage bucket name. See Step 1.
   - Image OCID (`oci compute image list`)
   - Object Storage namespace. See Obtaining the Object Storage Namespace.
   - The name the exported image file will have in object storage. This is the same parameter as `my-object` in the URL example in the following "Export to a URL" procedure.
   - The format of the exported image: OCI, QCOW2, or VMDK. See the preceding Compute Web UI procedure for descriptions.

     If the image format is not OCI, then you must use the `--export-format` option to specify the image format. If `--export-format` is not specified, the image is exported in OCI format.

3. Run the image export to object command.

   Syntax:

```
oci compute image export to-object --bucket-name bucketname \
--image-id image_OCID --namespace namespace \
--name exported_image_object_name
--export-format VMDK
```

Exporting an Image to a URL

You can export an image to a object store URL. You need write access to the export location.

Alternatively, you can export an image to an Object Storage bucket as described in Exporting an Image to an Object Storage Bucket.

Exported images are a copy of the boot volume and metadata when the image was created.

Using the Compute Web UI

1. On the Dashboard, click Compute/View Images. In the menu on the left side of the Images page, click Custom Images.

2. Click the name of the custom image that you want to export. If the image that you want to export is not listed, use the compartment menu above the image list.
3. On the image details page, click the Controls menu and then click the Export Image option.
4. In the Export Custom Image dialog, enter the following information:
   - **Export Destination:** Select the Export to an Object Storage URL option.
   - **Object Storage URL:** Enter the URL where you want to export the image. The URL must be an Object Storage URL.
   - **Export Format:** Select one of the following options based on the type of image you are exporting.
     - **VMDK:** Virtual machine disk file format (`.vmdk`), used for virtual machine disk images.
     - **QCOW2:** For disk image files (`.qcow2`) used by QEMU copy on write.
     - **OCI:** For Oracle Cloud Infrastructure files with a QCOW2 image and OCI metadata (`.oci`).
5. Click the Export Image button in the dialog.

   The image state changes to Exporting. Exporting a custom image copies the data to the Object Storage location that you specified. To track the progress of the operation, view the associated work request.

   You can still launch instances while the image is exporting, but you cannot delete the image until the export has finished.

   When the export is complete, the image state changes to Available. If the image state changes to Available, but you do not see the exported image in the Object Storage location you specified, the export failed, and you need to go through the steps again to export the image.

Using the OCI CLI

1. Ensure that a bucket with a pre-authenticated request is available, and that you have the request URL. See:
   - Creating a Bucket
   - Creating a Pre-Authenticated Request for All Objects in a Bucket
   - Constructing the Pre-Authenticated Request URL
2. Gather the information that you need to run the command:
   - Image OCID (`oci compute image list`)
   - URI. See Step 1.
   - The format of the exported image: OCI, QCOW2, or VMDK. See the preceding Compute Web UI procedure for descriptions.

     If the image format is not OCI, then you must use the `--export-format` option to specify the image format. If `--export-format` is not specified, the image is exported in OCI format.
3. Run the image export to object URI command.

   Syntax:

   ```
   oci compute image export to-object-uri --image-id image_OCID \
   --uri URL_to_export_to
   ```

Example:

```
oci compute image export to-object-uri \
--image-id ocid1.image.unique_ID \
--uri https://objectstorage.mypca01.example.com/p/MrxLFkKlFkIlNDhvhcZnrjbUAlsoeah/n/
mynamespace/b/my-bucket/o/my-object
{
  "data": {
    "agent-features": null,
    "base-image-id": null,
    "compartment-id": "ocid1.tenancy.unique_ID",
    "create-image-allowed": true,
    "defined-tags": null,
    "display-name": "PCA OL8 Image",
    "freeform-tags": null,
    "id": "ocid1.image.unique_ID",
    "launch-mode": "PARAVIRTUALIZED",
    "launch-options": {
      "boot-volume-type": "PARAVIRTUALIZED",
      "firmware": "UEFI_64",
      "is-consistent-volume-naming-enabled": false,
      "is-pv-encryption-in-transit-enabled": false,
      "network-type": "PARAVIRTUALIZED",
      "remote-data-volume-type": "PARAVIRTUALIZED"
    },
    "lifecycle-state": "EXPORTING",
    "listing-type": null,
    "operating-system": "OracleLinux",
    "operating-system-version": "8",
    "size-in-mbs": 47694,
    "time-created": "2022-01-18T16:29:13.114742+00:00"
  },
  "etag": "5d24f645-b446-42f2-a777-112457f0cafe",
  "opc-work-request-id": "ocid1.workrequest.unique_ID"
}
```

Sharing Custom Images Across Tenancies

You can use image import and export to share custom images across tenancies so that you don't need to recreate the image manually in each tenancy. You create the image in one of the tenancies, and then export the image, making it available for import in additional tenancies.

These are the high-level tasks:

1. Export the image to an Object Storage bucket. See Exporting an Image to an Object Storage Bucket.
2. Create a pre-authenticated request with read-only access for the image in the bucket. See Using Pre-Authenticated Requests.
3. In the destination tenancy, import the image. Use the pre-authenticated request URL as the Object Storage URL. See Importing an Image from a URL.

Creating an Image from an Instance

You can create a custom image of a compute instance's boot disk and use that custom image to launch other compute instances. Instances that you launch from this image include the customizations, configuration, and software that were installed on the boot disk when you created the image.

Custom images do not include the data from any attached block volumes.

Custom images inherit the compatible shapes that are set by default from the base image. For additional details to consider, refer to "Custom Images Created From Instances" in the Compute Instance Concepts chapter in the *Oracle Private Cloud Appliance Concepts Guide*.

The instance you use to create this custom image must be in the Stopped state.

Once the new custom image reaches the Available state, you can use it to launch new instances. See Creating an Instance.

Using the Compute Web UI

1. Click Dashboard, and click the Compute/View Instances button.
2. Select the compartment where the source instance is located.
3. Click the name of the instance that you want to use as the basis for the custom image.
4. Ensure that the instance is in the Stopped state.

   On the details page for the instance, click the Controls menu, and then click Stop.

   Wait for the instance status to change to Stopped. The status is displayed above the icon of the object.
5. Click Controls and then click Create Custom Image.
6. In the Create Image From Instance dialog, enter the following information:
   - **Name:** Replace the name with the name you want for the image.
   - **Create in Compartment:** Select the compartment where the image will be stored.
7. Click the Create Custom Image button in the dialog.

   The status of the instance changes to Creating Image.
8. Monitor the image creation progress.

   The time required to create the custom image depends on the size of the instance's boot volume.

   To monitor the progress, on the navigation menu, click Compute and then click Custom Images. Select the correct compartment, and click on the image name in the list. On the details page for the image, under Resources click Work Requests.
9. (Optional) Restart the instance.

   When the instance status changes from Creating Image to Stopped, you can restart the instance.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - The OCID of the compartment where the new image will be located (`oci iam compartment list`)
   - The OCID of the instance that will provide the boot volume image as the basis for the custom image (`oci compute instance list`)
   - Display name for the new image
2. Ensure that the instance is in the Stopped state.

   List the instance as described in Listing Images and Viewing Details. If `lifecycle-state` is not STOPPED, stop the instance as described in Stopping, Starting, and Resetting an Instance.

3. Run the image create command.

   Syntax:

   ```
   oci compute image create --compartment-id compartment_OCID \
   --instance-id base_instance_OCID --display-name display_name
   ```

   Example:

   ```
   $ oci compute image create \
   --compartment-id ocid1.compartment.unique_ID \
   --instance-id ocid1.instance.unique_ID \
   --display-name "Oracle Linux 8"
   {
     "data": {
       "agent-features": null,
       "base-image-id": "ocid1.bootvolume.unique_ID",
       "billable-size-in-gbs": null,
       "compartment-id": "ocid1.compartment.unique_ID",
       "create-image-allowed": true,
       "defined-tags": {},
       "display-name": "Oracle Linux 8",
       "freeform-tags": {},
       "id": "ocid1.image.unique_ID",
       "launch-mode": "PARAVIRTUALIZED",
       "launch-options": null,
       "lifecycle-state": "PROVISIONING",
       "listing-type": null,
       "operating-system": "Custom",
       "operating-system-version": "Custom",
       "size-in-mbs": 0,
       "time-created": "2022-02-17T18:26:03.221604+00:00"
     },
     "etag": "3c0e56a0-b58c-486b-b659-9f5b13f377ee",
     "opc-work-request-id": "ocid1.workrequest.unique_ID"
   }
   ```

4. (Optional) Restart the instance.

   When the instance `lifecycle-state` returns to STOPPED, you can restart the instance as described in Stopping, Starting, and Resetting an Instance.

Bring Your Own Image (BYOI)

The Bring Your Own Image (BYOI) feature enables you to bring your own versions of operating systems to the appliance as long as the underlying hardware supports it. The Private Cloud Appliance services do not depend on the OS you run.

> **Important:**
>
> You must comply with all licensing requirements when you upload and start instances based on OS images that you supply.

For more conceptual information, refer to "Bring Your Own Image (BYOI)" in the Compute Instance Concepts chapter in the *Oracle Private Cloud Appliance Concepts Guide*.

Importing Custom Linux Images

Preparing Linux VMs for Import

Before you import a custom Linux image, you must prepare the image to ensure that instances launched from the image can boot correctly and that network connections will work.

Perform these steps.

1. Review the requirements.

   Refer to "Linux Source Image Requirements" in the Compute Instance Concepts chapter of the *Oracle Private Cloud Appliance Concepts Guide*.

2. Create a backup of the root volume.
3. If the VM has remotely attached storage, such as NFS or block volumes, configure any services that rely on this storage to start manually. Remotely attached storage is not available the first time an imported instance boots on the appliance.
4. Ensure that all network interfaces use DHCP, and that the MAC address and IP addresses are not hardcoded. See your system documentation for steps to perform network configuration for your system.
5. Stop the VM.
6. Clone the stopped VM as a VMDK or QCOW2 file, and then export the image from your virtualization environment.

   Refer to the tools documentation for your virtualization environment.

Importing a Linux Image

After you prepare a Linux image for import, follow these steps to import the image:

1. Upload the image file to an Object Storage bucket.

   Ensure that you select a bucket where you have read and write access. See Exporting an Image to an Object Storage Bucket.

2. Import the image from the bucket to your tenancy.

   See Importing an Image from an Object Storage Bucket

3. Complete the post-import tasks.

   See Post-Import Tasks for Linux Images.

Post-Import Tasks for Linux Images

After you import a custom Linux image, perform these steps.

1. Use the imported image to launch an instance.

   For the image source, select Custom Images, and then select the image that you imported. See Creating an Instance.

2. If the instance requires any remotely attached storage, such as block volumes, create and attach the storage.

   See Creating and Attaching Block Volumes.

3. Create and attach any required secondary VNICs.

   See Configuring VNICs and IP Addressing.

4. Test that all applications are working as expected.
5. Reconfigure any services that were set to start manually.

Importing Custom Microsoft Windows Images

The Compute service enables you to import Microsoft Windows images and use them to launch instances. You can import images that you create from Microsoft Windows systems that are running on your on-premises physical or virtual machines (VMs).

Perform the procedures in this section to prepare, create, export, import, and perform post-import tasks.

Preparing Microsoft Windows Systems for Import

The configuration described in this section is required so that Compute instances that are launched from the Microsoft Windows system image can boot correctly and network connections will work.

> **Important:**
>
> The system drive configuration where the Microsoft Windows source system is installed will be imported to the image. All partitions on the drive will follow through the imported image. Any other drives will not be imported, and you must re-create them on the instance after they are launched from the image. You will then need to manually move the data on the non-system drives to storage on the instance.

You can perform this configuration on the running source system or after you have launched the Compute instance.

- **Preparing the Source System Prior to Creating the Image**. This is the recommended method.
- **Preparing the Compute Instance After Instance Launch**. If you have concerns about modifying the live source system, you can use this method. If you use this method, your Compute instance is not initially viable. After you launch your Compute instance, connect to the VNC console and use the VNC window to make the changes described in Preparing the Source System Prior to Creating the Image.

Preparing the Source System Prior to Creating the Image

1. Review the requirements.

   Refer to "Microsoft Windows Source Image Requirements" in the Compute Instance Concepts chapter of the *Oracle Private Cloud Appliance Concepts Guide*.

2. Follow your organization's security guidelines to ensure that the Microsoft Windows system is secured. This can include, but is not limited to the following tasks:
   - Install the latest security updates for the operating system and installed applications.
   - Enable the firewall, and configure it so that you only enable the rules that are needed.
   - Disable unnecessary privileged accounts.

- Use strong passwords for all accounts.

3. Configure Remote Desktop Protocol (RDP) access to the image.
   a. Enable Remote Desktop connections to the image. See Enabling Remote Desktop Protocol Access.
   b. Modify the Microsoft Windows Firewall inbound port rule to allow RDP access for both Private and Public network location types. When you import the image, the Microsoft Windows Network Location Awareness service will identify the network connection as a Public network type.
4. Determine whether the current Microsoft Windows license type is a volume license by running the following command in PowerShell:

   ```
   Get-CimInstance -ClassName SoftwareLicensingProduct | where {$_.PartialProductKey} |
   select ProductKeyChannel
   ```

   If the license is not a volume license, after you import the image, you will update the license type.
5. If you plan to use this custom image to launch more than one instance, create a generalized image of the boot disk. A generalized image is cleaned of computer-specific information, such as unique identifiers. When you create instances from a generalized image, the unique identifiers are regenerated. This prevents two instances that are created from the same image from colliding on the same identifiers.
6. Create a backup of the root volume.
7. If the system has remotely attached storage, such as NFS or block volumes, configure any services that rely on this storage to start manually. Remotely attached storage is not available the first time an instance that was created from a custom image boots on Oracle Private Cloud Appliance.
8. Ensure that all network interfaces use DHCP, and that the MAC address and IP addresses are not hardcoded. See your system documentation for steps to perform network configuration for your system.
9. Install the Oracle VirtIO Drivers for Microsoft Windows.
   a. Downloading the Oracle VirtIO Drivers for Microsoft Windows
   b. Installing the Oracle VirtIO Drivers for Microsoft Windows
10. Perform the Creating and Exporting an Image procedure unless you already followed the Preparing the Compute Instance After Instance Launch procedure.

Creating and Exporting an Image

1. Stop the system.
2. Clone the stopped system as a VMDK or QCOW2 file. Refer to the tools documentation for your system.
3. Export the image from your physical system or virtualization environment.
4. Perform the Importing a Microsoft Windows Image procedure to import the image into Oracle Private Cloud Appliance.

Preparing the Compute Instance After Instance Launch

1. Perform as many of the Preparing the Source System Prior to Creating the Image steps as you are comfortable performing.
2. Perform the Creating and Exporting an Image procedure.

After importing the image, do *not* perform the Post-Import Tasks for Microsoft Windows Images procedure.

3. Use the imported image to launch an instance.

   For the image source, select Custom Images, and then select the image that you imported. See Creating an Instance.

4. Connect to the console as described in Remotely Troubleshooting an Instance by Using a Console Connection.

5. Perform the Preparing the Source System Prior to Creating the Image procedure.

6. Perform the Post-Import Tasks for Microsoft Windows Images procedure.

Downloading the Oracle VirtIO Drivers for Microsoft Windows

The Oracle VirtIO Drivers for Microsoft Windows are paravirtualized drivers for Microsoft Windows instances. These drivers improve performance for network and block (disk) devices on Microsoft Windows instances and resolve common issues.

Download the Oracle VirtIO Drivers for Microsoft Windows from the Oracle Software Delivery Cloud website or from My Oracle Support (MOS).

Download the Oracle VirtIO Drivers for Microsoft Windows from Oracle Software Delivery Cloud

1. Sign in to the Oracle Software Delivery Cloud site.
2. In the All Categories list, select Release.
3. Type Oracle Linux 7.9 in the search box and click Search.
4. Click "REL: Oracle Linux 7.9.0.0.0" to add it to your cart.
5. At the top right of the page, to the right of your cart, click Continue.
6. In the Platforms/Languages list, select x86 64 bit. Click Continue.
7. Review and accept the license agreement (click "I reviewed and accept the Oracle License Agreement."). Click Continue.
8. Click the `V1009702-01.zip` file name to the left of "Oracle VirtIO Drivers Version for Microsoft Windows 1.1.7, 67.9 MB".
9. Follow the prompts to save the `V1009702-01.zip` file.

Download the Oracle VirtIO Drivers for Microsoft Windows from MOS

1. Sign in to My Oracle Support.
2. Click the Patches & Updates tab.
3. In the Patch Search pane, in the Patch Name or Number field, enter `27637937`. Click the Search button.
4. From the search results table, click the Patch Name to the left of "Oracle VirtIO driver version 1.1.7" for Release 7.9.0.0.0.

   A more detailed description of the patch is shown.

5. In the box, click the Download button.
6. In the File Download window, follow the prompts to save the `p27637937_79000_MSWIN-x86-64.zip` file.

Installing the Oracle VirtIO Drivers for Microsoft Windows

To install the Oracle VirtIO Drivers for Microsoft Windows, configure Microsoft Windows policies and then run the installation program.

Configuring Policies for Device Installation

Configure Microsoft Windows policies to allow the installation of the Oracle VirtIO Drivers for Microsoft Windows, if these policies are not already configured.

1. Go to the Microsoft Windows system on which you want to install the Oracle VirtIO Drivers for Microsoft Windows.
2. From the Start menu, select Run.
3. Enter `gpedit.msc` and then click OK.

   The Local Group Policy Editor is displayed.
4. From the Console Tree, display the list of Device Installation Restrictions as follows:
   a. Expand Computer Configuration, and then expand Administrative Templates.
   b. Expand System, and then expand Device Installation.
   c. Select Device Installation Restrictions.
5. Edit the policy settings so that no device installation restrictions are configured.
6. Close the Local Group Policy Editor.
7. Restart the Microsoft Windows system.

After performing one of the procedures described in Downloading the Oracle VirtIO Drivers for Microsoft Windows, the Microsoft Windows system should have a copy of the Oracle VirtIO Drivers for Microsoft Windows installation program, `Setup.exe`.

You can use a graphical user interface (GUI) to install the drivers, or use the command line to install the drivers by using a response file that you previously created.

The Oracle VirtIO Drivers for Microsoft Windows are installed in the following directories:

- On 32-bit systems: `C:\Program Files\Oracle Corporation\Oracle Windows VirtIO Drivers`
- On 64-bit systems: `C:\Program Files (x86)\Oracle Corporation\Oracle Windows VirtIO Drivers`

Installing the Oracle VirtIO Drivers for Microsoft Windows by Using the GUI

This procedure installs the drivers on a single Microsoft Windows system. You can optionally record your responses for use on other systems.

1. Run the `Setup.exe` driver installation program.
   - To install the drivers on only this system, double-click the `Setup.exe` file.
   - To record a response file for use on other systems, start the `Setup.exe` installer from the command line.
     a. Open a command-line window.
     b. Navigate to the directory where the `Setup.exe` file is located.
     c. Run `Setup.exe -r` to start the installer and create a response file.

2. If prompted, select Yes in the User Account Control dialog to allow the installer to proceed.

   The Welcome window is displayed.

3. Click Next.

   The "Start to install Oracle VirtIO Drivers for Microsoft Windows Release 2.0" window is displayed with information about your selection.

4. Click Install to start the installation.

   The installer copies the Oracle VirtIO Drivers for Microsoft Windows files and installs the drivers on the system.

5. Once the installation completes, click Finish.

   The system is restarted.

Installing the Oracle VirtIO Drivers for Microsoft Windows by Using an Existing Response File

This procedure uses a response file that was created in the Installing the Oracle VirtIO Drivers for Microsoft Windows by Using the GUI procedure.

1. Locate the response file, `setup.iss`, in the `C:\Windows` directory.
2. Copy the response file to the same directory where the Oracle VirtIO Drivers for Microsoft Windows installation program, `Setup.exe`, is located.

   Alternatively, you can specify the location of the response file at the command line.

3. Open a command-line window.
4. Run `Setup.exe -s` to install the drivers by using the response file.

   The following additional options to the `Setup.exe -s` command are available:

   - `-f1c:`***`path_to`***`\setup.iss` to specify the location of the `setup.iss` response file.
   - `-f2c:`***`path_to`***`\setup.log` to specify the location of the `setup.log` log file.

     By default, log files are written to the `C:\Windows` directory.

Importing a Microsoft Windows Image

After you prepare a Microsoft Windows image for import, follow these steps to import the image:

1. Upload the image file to an Object Storage bucket.

   Ensure that you select a bucket where you have read and write access. See Exporting an Image to an Object Storage Bucket.

2. Import the image from the bucket to your tenancy.

   See Importing an Image from an Object Storage Bucket and Importing an Image from a URL. Use the OCI CLI procedure and specify the `--operating-system` option. Make sure the value of the `--operating-system` option includes the case-insensitive string "Windows".

3. Complete the post-import tasks.

   See Post-Import Tasks for Microsoft Windows Images.

Post-Import Tasks for Microsoft Windows Images

After you import a custom Microsoft Windows image, perform these steps.

1. Use the imported image to launch an instance.

   For the image source, select Custom Images, and then select the image that you imported. See Creating an Instance.

2. Enable Remote Desktop Protocol (RDP) access to the Compute instance.

   See Enabling Remote Desktop Protocol Access.

3. Connect to the instance using RDP.

   See Connecting with an RDP Client.

4. If the instance requires any remotely attached storage, such as block volumes, create and attach the storage.

   See Creating and Attaching Block Volumes.

5. Create and attach any required secondary VNICs.

   See Configuring VNICs and IP Addressing.

6. Test that all applications are working as expected.

7. Reconfigure any services that were set to start manually.

8. Configure your instance to use the Network Time Protocol (NTP).

To avoid performing this post-launch configuration every time you launch an instance using this custom image, consider creating a new image from the fully configured instance. See Creating an Image from an Instance.

7

Compute Instance Deployment

This chapter describes how to create (launch), update, and delete (terminate) a compute instance, and how to stop, start, or reset an instance.

This chapter also describes how to create and use an instance configuration. With an instance configuration, you can create a single instance or pool of instances quickly. You can create an instance configuration from an existing instance to replicate that instance more quickly.

You can attach instances to a pool or detach instances from a pool manually, or you can configure autoscaling to automatically grow or shrink the pool on a predefined schedule.

Finally this chapter describes how to connect to a compute instance and how to back up an instance and restore the instance from backup.

Tutorial – Launching Your First Instance

In this tutorial you'll learn the basic features of Oracle Private Cloud Appliance by performing some guided steps to launch and connect to an instance. After your instance is up and running, this tutorial steps you through creating and attaching a block volume to your instance.

This tutorial also includes optional instructions for deleting all the resources you create.

Task Flow to Launch an Instance

No.	Task	Links
1.	Review the prerequisites.	Prerequisites
2.	Log into the appliance.	Log into Oracle Private Cloud Appliance
3.	Create a compartment for your resources.	Create a Compartment
4.	Create a Virtual Cloud Network (VCN).	Create a Virtual Cloud Network (VCN)
5.	Create a subnet in the VCN.	Create a Subnet
6.	Configure additional network parameters to enable instance connectivity.	Create an Internet Gateway and Configure Route Rules
7.	Launch an instance	Launch an Instance
8.	Get the instance IP address.	Get the Instance IP Address
9.	Connect to your instance.	Connect to Your Instance
10.	Add Storage to your instance.	Add a Block Volume Attach the Block Volume to an Instance
11.	(Optional) Clean up your resources.	(Optional) Clean Up Resources

Prerequisites

To perform this tutorial, ensure that you have the following items.

- The URL for your Private Cloud Appliance.

  For example, `https://console.pca_name.example.com` where ***pca_name*** is the name of your Private Cloud Appliance and ***example.com*** is your domain.
- A Private Cloud Appliance user account and password.
- The name of your tenancy.
- An SSH key pair. If you need to create a key pair for this tutorial, see Managing Key Pairs.
- The virtual IP address (VIP) or hostname of the Private Cloud Appliance management nodes.

If you don't have these items, ask your Service Enclave administrator.

Log into Oracle Private Cloud Appliance

1. In a browser, enter the URL for your Private Cloud Appliance.
2. If necessary, click Change Tenant, enter the name of your tenancy, and click Continue.
3. Enter your user name and password, and then click Sign In.

   If this is the first time you've logged in, you are prompted to change your password.

For more information about using the Compute Web UI, see Using the Compute Web UI.

What's Next

Continue to Create a Compartment.

Create a Compartment

Compartments help you organize and control access to your resources. A compartment is a collection of resources (such as cloud networks, compute instances, and block volumes) that can be accessed only by those groups that have been given permission by an administrator in your organization.

In a production environment, the compartment for the instance you plan to create might already exist, and you could use it instead of creating a new compartment. However, in this tutorial, you create a new compartment to learn how to do it, and to provide an empty compartment from which you can create your cloud network.

In this tutorial, you use one compartment for all of your resources. However, when you are ready to create a production environment you can separate resources into different compartments. For example, you might place all instances in one compartment and all networking resources in another compartment.

For more information about compartments, refer to these resources:

- For conceptual information, see "Organizing Resources in Compartments" in the Identity and Access Management Overview in the *Oracle Private Cloud Appliance Concepts Guide*.
- For step-by-step instructions to manage compartments, see Creating and Managing Compartments.

Using the Compute Web UI

1. In the Navigation Menu, click Identity and then click Compartments.
2. On the Compartments page, click the Create Compartment button.
3. In the Create Compartment dialog, enter the following details:
   - **Name:** Enter Sandbox.
   - **Description:** Enter a description for the compartment.
   - **Create in Compartment:** Select the compartment in which to create this new compartment.
4. Click the Create Compartment button in the dialog.

   The new compartment is displayed on the Compartments page.

What's Next

Continue to Create a Virtual Cloud Network (VCN).

Create a Virtual Cloud Network (VCN)

Before you can launch an instance, you need a virtual cloud network (VCN) and a subnet.

A VCN is a software-defined equivalent of a traditional network, with firewall rules and various types of communication gateways.

In a production environment, a VCN that you can use for the instance might already exist, and you could use it instead of creating a new VCN. However, in this tutorial, you create a new VCN to learn how to do it.

> **Important:**
>
> This tutorial creates a simple cloud network to make it easy to launch an instance for learning purposes. When you create your production instances, ensure that you create appropriate security lists and route table rules to restrict network traffic to your instances.

For more information about VCNs, refer to these resources:

- For conceptual information, see "Virtual Cloud Network" in the Virtual Networking Overview in the *Oracle Private Cloud Appliance Concepts Guide*.
- For step-by-step instructions to manage VCNs, see Managing VCNs and Subnets.

Using the Compute Web UI

1. Click Dashboard, and click the Networking/View Virtual Cloud Networks button.
2. On the VCNs page, click the Create Virtual Cloud Network button.
3. In the Create Virtual Cloud Network dialog, enter the following information:
   - **Name:** Enter a descriptive name for the cloud network.
   - **Create in Compartment:** Select the Sandbox compartment.
   - **CIDR Block:** Enter a valid CIDR block for the VCN. For example 10.0.0.0/16.

- **Use DNS hostnames in this VCN:** Indicate whether you want to use DNS host names in the VCN.
- **DNS Label:** If you selected to use DNS, enter a DNS label or leave the field blank to let the system generate a DNS name for you.
- **Tagging:** Leave blank. This tutorial does not use tags.

4. Click the Create Virtual Cloud Network button in the dialog.

What's Next

Continue to Create a Subnet.

Create a Subnet

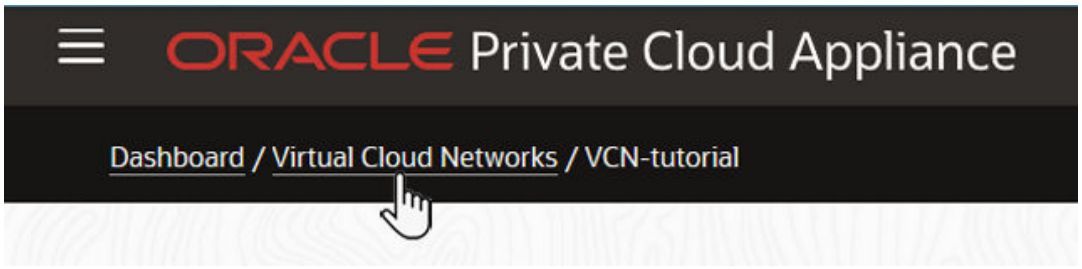
A subnet is a subdivision of your VCN. The subnet directs traffic according to a route table.

For this tutorial, you'll access the instance over the internet using its public IP address, so your route table will direct traffic to an internet gateway. The subnet also uses a security list to control traffic in and out of the instance.

Using the Compute Web UI

1. Return to the Virtual Cloud Networks page.

   A quick way to do this is to click the name of page in the breadcrumb that is in the top banner. For example:



   If the VCNs page is not in your breadcrumb, Click Dashboard, and click the Networking/ View Virtual Cloud Networks button.

2. On the VCNs list, click the name of the VCN you just created.

   The details page for that VCN is displayed.

3. Scroll down to the Resources panel, click Subnets, and click the Create Subnet button.
4. In the Create Subnet dialog, enter the following information:
   - **Name:** Enter a descriptive name for your subnet.
   - **Create in Compartment:** Select the Sandbox compartment.
   - **CIDR Block:** Enter a valid CIDR block for the subnet. The value must be within the VCN's CIDR block. For example, 10.0.0.0/24.
   - **Route Table:** For this tutorial, select the default route table.
   - **Subnet Access:** For this tutorial, select Public Subnet to allow public IP addresses for instances in the subnet.

- **Use DNS Hostnames in this Subnet:** For this tutorial, leave this unselected.
- **DHCP Options:** Leave this unselected.
- **Security Lists:** Click Add Security List and select the default security list.
- **Tagging:** Leave blank. This tutorial does not use tags.

5. Click the Create Subnet button in the dialog.

What's Next

Continue to Create an Internet Gateway and Configure Route Rules.

Create an Internet Gateway and Configure Route Rules

An internet gateway is an optional virtual router you can add to your VCN to enable access to your data center network.

The gateway supports connections initiated from within the VCN (egress) and connections initiated from the internet (ingress).

Security list rules control the types of traffic allowed in and out of resources in that subnet. Make sure to allow only the desired types of internet traffic.

Each public subnet that needs to use the internet gateway must have a route table rule that specifies the gateway as the target.

Using the Compute Web UI

1. Navigate to your VCN's details page.
2. In the Resources panel, select Internet Gateways.
3. Click Create Internet Gateway.
4. Enter the required information:
   - **Name:** Enter a descriptive name for your internet gateway.
   - **Create in Compartment:** Select the Sandbox compartment.
   - **Enabled:** Select whether you want this internet gateway to be enabled upon creation.
   - **Tagging:** Leave blank. This tutorial does not use tags.
5. Click the Create button on the Create Internet Gateway dialog.
6. Under Resources, click Route Tables.
7. Click the name of the default route table.
8. Scroll down to the Resources panel and click the Add Route Rules button.
9. On the Create Route Table Rule dialog, enter the required information:
   - **Target Type:** From the drop-down menu, select Internet Gateway.
   - **CIDR Block:** Enter: `0.0.0.0/0` (which means that all non-intra-VCN traffic that is not already covered by other rules in the route table will go to the target specified in this rule).
   - **Internet Gateway:** From the drop-down menu, select the name of the Internet Gateway that you created.
   - **Description:** An optional description of the rule.
10. Click the Create Route Table Rule button.

What's Next

Continue to Launch an Instance.

Launch an Instance

In this task, launch an instance with an image and a basic shape.

A compute instance is a virtual machine (VM), which is an independent computing environment that runs on top of physical hardware. The virtualization makes it possible to run multiple compute instances that are isolated from each other.

A shape describes the instance resources such as the number of CPUs, amount of memory, and network resources. In a production environment, you would select a shape that best suits workload and application requirements for the instance.

For more information about instances, refer to these resources:

- For conceptual information, including descriptions of standard and flexible shapes, see Compute Instance Concepts in the *Oracle Private Cloud Appliance Concepts Guide*.
- For step-by-step instructions to manage instances, see Working with Instances.

Before You Begin

Ensure that you have performed these tasks:

- Create a Compartment
- Create a Virtual Cloud Network (VCN)
- Create a Subnet
- Create an Internet Gateway and Configure Route Rules

Using the Compute Web UI

1. Click Dashboard, and click the Compute/View Instances button.
2. On the Instances page, click the Create Instance button.
3. In the Launch Instance dialog, enter the following information:
   - **Name:** Enter a descriptive name for your compute instance.
   - **Create in Compartment:** Select the Sandbox compartment.
   - **Fault Domain:** Leave the default set to "Automatically select the best fault domain."
   - **Source Image:**
     - **Source Type:** Select Platform Image.
     - **List of images:** Select the Oracle Linux 8 image.
   - **Shape:** Select one of the smaller shapes such as VM.PCAStandard1.1.
   - **Boot Volume:** Leave the check box empty so that the default boot volume size is created.
   - **Subnet:**
     - **VCN:** Select the VCN you created.
     - **Subnet:** Select the subnet you created.
   - **Public IP Address:** Ensure the check box is checked so that a public IP address is assigned to the instance.

- **Private IP Address:** Leave the field blank.
- **Hostname:** You can leave this field blank or enter a hostname.
- **SSH Keys:** Do one of the following to provide your public SSH key:
  - Click inside the Drag and Drop box to open a file browser and select the file.
  - Drag the file from your file browser listing and drop the file on the Drag and Drop box.
  - Select the Paste the public key(s) button, copy your public SSH key text, and paste the text into the field.
- **Initialization Script:** Leave this area as is.
- **Network Security Group:** Leave the check box unchecked.
- **Tagging:** Leave blank. This tutorial does not use tags.

4. Click the Launch Instance button in the dialog.
5. Monitor the state of the instance.

   The state is displayed above the icon of the object. For example:

   Your instance begins in the Provisioning state. Once the instance is in the Running state, you can connect to it.

What's Next

Continue to Get the Instance IP Address.

Get the Instance IP Address

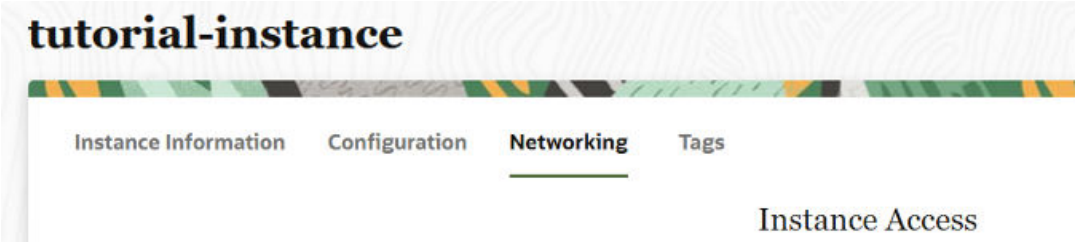
You can connect to the instance using SSH with the instance IP address.

Using the Compute Web UI

1. Navigate to the details page of your instance.

   Click Dashboard, and click the Compute/View Instances button. In the Instances list, click the name of your instance.
2. Select the Networking tab.

   The tabs are displayed at the top of the details panel:



3. Under Instance Access on the Networking tab, note the Public IP Address.

What's Next

Continue to Connect to Your Instance.

Connect to Your Instance

In most cases, you connect to a running instance using a Secure Shell (SSH) connection. Some instances support authenticating your connection with a password. This tutorial assumes you used one of the images provided on the appliance, which creates an instance that authenticates your SSH connection with an SSH key pair.

For the system that you will be connecting from, most Linux and other UNIX operating systems include an SSH client by default.

Microsoft Windows 10 and Microsoft Windows Server 2019 systems should include the OpenSSH client, which you'll need if you created your instance using the SSH keys generated by Oracle Cloud Infrastructure.

For other Microsoft Windows versions, you can download a free SSH client called PuTTY from http://www.putty.org.

Before You Begin

- Get the public IP address of your instance, as described in Get the Instance IP Address.
- Get the path to your private key file.
- Get the valid user name.

  The user name is configured in the image used to launch the instance. If you launched an instance using one of the platform images that is provided on the appliance, the default user is `opc`. See Initial User Account for Platform Images.

Perform one of the following tasks based on the type of system you are connecting from:

- Connect from a UNIX System
- Connect Using PuTTY

Connect from a UNIX System

Perform this procedure on your UNIX system.

1. Open a terminal window.
2. Use the `ssh` command to connect to your instance.

   Syntax:

   ```
   ssh -i private_key_pathname username@public-ip-address
   ```

- ***private_key_pathname*** is the full path name of the file that contains the private key associated with the instance you want to access.
- ***username*** is the default user name for the instance. For this tutorial, `opc` is the user name.
- ***public-ip-address*** is your instance IP address.

Example:

```
$ ssh -i /home/flast/.ssh/id_rsa opc@192.0.2.1
```

3. If asked whether you want to continue connecting, type `yes`.

   You are now logged in to your instance.

What's Next

Continue to Add a Block Volume.

Connect Using PuTTY

This connection method is commonly performed from Microsoft Windows systems.

Use this procedure if the instance uses a key pair that you created using the PuTTY Key Generator. See Creating an SSH Key Pair Using PuTTY Key Generator.

1. Open PuTTY.
2. In the Category pane (on the left), select Session and enter the following:
   - **Host Name (or IP address):** ***username***`@`***public-ip-address***
     - ***username*** is the default user name for the instance. For this tutorial, the user name is `opc`.
     - ***public-ip-address*** is your instance IP address.
   - **Port:** 22
   - **Connection type:** SSH
3. In the Category pane, expand Window, and then select Translation.
4. In the Remote character set drop-down list, select UTF-8. The default locale setting on Linux-based instances is UTF-8, and this setting configures PuTTY to use the same locale.
5. In the Category pane, expand Connection, expand SSH, and then click Auth.
6. Click Browse, and then select your `.ppk` private key file.
7. Click Open to start the session.

   If this is your first time connecting to the instance, you might see a message that the server's host key is not cached in the registry. Click Yes or Accept to continue the connection.

> **Tip:**
>
> If the connection fails, you might need to update your PuTTY proxy configuration.

What's Next

Continue to Add a Block Volume.

Add a Block Volume

You can use block volumes to expand the storage capacity of your compute instances.

After a block volume is created, you attach the volume to one or more instances. You can use the volume like a regular hard drive.

Using the Compute Web UI

1. Click Dashboard, and click the Block Storage/View Block Volumes button.
2. Click the Create Block Volume button.
3. In the Create Block Volume dialog, enter the following information:
   - **Name:** Enter a descriptive name for your block volume.
   - **Create in Compartment:** Select the Sandbox compartment.
   - **Size:** Leave the default size (1024 GB).
   - **Backup Policy:** Do not select a backup policy.
   - **Tags:** Leave blank. This tutorial does not use tags.
4. Click Create Block Volume button in the dialog.
5. Monitor the state of the new block volume.

   The state is displayed above the icon for the object. You can also scroll down to the Resources section and check the Work Request.

   Initially, the block volume is in the Provisioning state. When the volume changes to the Available state, you can attach it to your instance.

What's Next

Continue to Attach the Block Volume to an Instance.

Attach the Block Volume to an Instance

Using the Compute Web UI

1. Click Dashboard, and click the Compute/View Instances button.
2. Ensure that the Sandbox compartment is selected at the top of the page.
3. In the Instances list, click the name of your instance to view its details.
4. Scroll down to the Resources panel, and click Attached Block Volumes.
5. Click the Attach Block Volume button.
6. In the Attach Block Volume dialog, enter the following information:
   - **Select from Compartment:** Select the Sandbox compartment.
   - **Block Volume:** Select the block volume you created.
   - **Access:** Select Read/Write.
7. Click the Attach to Instance button in the dialog.

   The attachment process takes about a minute. The volume is ready when the Attachment State for the volume is Attached.

   If your block volume isn't displayed, reload the web page.

When a block volume is initially attached to an instance, the instance sees the volume as a new disk. To make the volume available to the instance OS, you need to give the volume a file system and mount it to the OS.

To learn about the block volume and how to make it available to the instance OS, refer to these sections outside of this tutorial:

- Find Your Volume in the Instance
- Configuring Volumes to Automatically Mount (Linux Instances)

What's Next

Continue to (Optional) Clean Up Resources.

(Optional) Clean Up Resources

After you've finished with the resources you created for this tutorial, you can delete and release the resources you don't intend to continue working with.

Detach and Delete the Block Volume

> **Caution:**
>
> You cannot undo a termination. Any data on a volume will be permanently deleted once the volume is terminated.

Using the Compute Web UI

1. Click Dashboard, and click the Compute/View Instances button.
2. Select the Sandbox compartment.
3. Click the name of your instance.
4. In the Resources panel, click Attached Block Volumes.
5. Find your volume, click the Actions menu, and then click Detach. Confirm the detachment in the dialog box.

   You might need to refresh the web page to see that the block volume is no longer attached.
6. Click Dashboard, and click the Block Storage/View Block Volumes button.
7. Select the Sandbox compartment.
8. Find your volume, click the Actions menu, and then click Terminate. Confirm the termination in the dialog box.

What's Next

Continue to Terminate the Instance.

Terminate the Instance

You can permanently terminate (delete) instances that you no longer need. Any attached VNICs and volumes are automatically detached when the instance terminates. Eventually, the

instance's public and private IP addresses are released and become available for other instances.

Using the Compute Web UI

1. Click Dashboard, and click the Compute/View Instances button.
2. Select the Sandbox compartment.
3. Find the instance you created, click the Actions menu, and click Terminate.
4. In the Confirm Instances termination dialog box, move the "Permanently delete the attached boot volume" selector to the right. Click the Confirm button.

   Moving the selector to the right results in the boot volume being permanently deleted, which is appropriate for this tutorial.

   In production, you can leave the selector in the left position to preserve the boot volume for use with another instance. This is convenient when you want to reuse a configured OS or data on the boot volume.

What's Next

Continue to Delete the Subnet, Internet Gateway, and VCN.

Delete the Subnet, Internet Gateway, and VCN

Using the Compute Web UI

1. Click Dashboard, and click the Networking/View Virtual Cloud Networks button.
2. Select the Sandbox compartment.
3. Click the name of your VCN.
4. Under Resources, click Route Tables.
5. Click the name of the route table.
6. For the route rule you created, click the Actions menu, click Delete, and confirm the deletion.

   The route rule is deleted.
7. In the breadcrumb path at the top of the page, click the name of your VCN.

   The VCN details page is displayed.
8. Under Resources, click Internet Gateways.
9. For the internet gateway that you created, click the Actions menu, click Delete, and confirm the deletion.

   The internet gateway is deleted.
10. Under Resources, click Subnets.
11. For the subnet you created, click the Actions menu, click Delete, and confirm the deletion.

    The subnet is deleted.
12. At the top of the VCN details page, click the Terminate button and confirm the termination.

    The VCN is deleted.

What's Next

Continue to Delete the Compartment.

Delete the Compartment

You must remove all resources from a compartment before you can delete it, otherwise, the delete action fails and the compartment returns to an Active state.

Using the Compute Web UI

1. In the navigation menu, click Identity, then click Compartments.
2. For the Sandbox compartment, click the Actions menu, and then click Delete.
3. Confirm the deletion in the dialog box.

Working with Instances

You can create compute instances as needed to meet your compute and application requirements. After you create an instance, you can access it securely from your computer, restart it, attach and detach volumes, and terminate it.

For general information about instances, see Compute Instance Concepts in the *Oracle Private Cloud Appliance Concepts Guide*.

Creating an Instance

See Tutorial – Launching Your First Instance for information about input you need to create an instance.

The following is the minimum information that you must provide to create an instance using the Compute Web UI:

- A name for the instance
- The compartment where you want to create the instance
- An image or boot volume
- A shape
- A subnet
- A public SSH key

  To log in to the instance, users need either an SSH key or a password, depending on how the image was built. If the instance will require SSH keys for authentication, you must provide the public key when you create the instance. You cannot provide the public SSH key after the instance is created.

To create an instance using the OCI CLI, you need the same information as listed above for the Compute Web UI except that you do not need an instance name. If you do not provide a name for the instance, the default name will be `instance`***`YYYYMMDDhhmmss`***, where ***`YYYYMMDDhhmmss`*** is the creation date and time.

To modify launch options, use the OCI CLI. You cannot modify launch options or boot volume VPUs per GB after the instance is created.

An alternative way to create an instance is to create an instance configuration and use that configuration to launch an instance, as described in Using an Instance Configuration to Launch an Instance. When you use an instance configuration to create an instance, you can specify `blockVolumes` and `secondaryVnics`, as shown in the OCI CLI procedure in Creating an Instance Configuration by Entering Configuration Values.

Using the Compute Web UI

1. Create or get the following resources and information:
   - An image or boot volume and the compartment where the image or boot volume is located
   - A virtual cloud network (VCN) and subnet and the compartment where the VCN and subnet are located
   - A public Secure Shell (SSH) key if users will connect to the instance using SSH
2. On the Dashboard, do one of the following to open the Create Instance dialog:
   - Click the Create Instance button in the top left corner of the Dashboard.
   - In the Compute block, click Instances. At the top of the instances list, click the Create Instance button.
   - If you want to create the instance from an existing custom image, then in the Compute block, click Custom Images. For the image that you want to use to create the instance, click the Actions menu and click the Create Instance From Image option. You might have to change the compartment at the top of the image list to see the image you want.

     When you use the Create Instance From Image option, the image name is already entered in the Create Instance dialog and you cannot change it. You do not need to enter any of the information described in "Source Image" in the following step.
3. In the Create Instance dialog, enter the following information:
   - *Name*: Enter a name for the instance. Instance names have the following characteristics:
     - Can be changed after the instance is created.
     - Do not need to be unique.
     - Can contain only alphanumeric characters and the hyphen (-) character.
     - Can be a maximum of 63 characters.
   - *Create in Compartment*: Select the compartment where you want to create the instance.
   - *Fault Domain*: (Optional) Select a fault domain. By default, the system automatically selects the best fault domain for the instance when the instance is created. If you specify a fault domain, and the requested fault domain cannot accommodate the instance, instance launch fails. The fault domain can be changed after the instance is created.
   - *Source Image*: Select an image or boot volume.
     a. Select the Source Type: Platform Image, Custom Image, or Boot Volume.
     b. If you selected Custom Image or Boot Volume, select the compartment where the image or boot volume that you want to use is located.
     c. Select an image or boot volume from the list.

        If you selected Platform Image, you see a tabular list with columns Operating System, OS Version, and Image Build (the date the image was built). You can use the drop-down menu arrow to the right of the OS Version to select a different version. For example, for the Oracle Linux operating system, you can use the drop-down menu to select 9, 8, or 7.9.

If you selected Custom Image, you see a tabular list with columns Name, Operating System, and OS Version. You can use the arrows in the column headings to sort the list. You can filter the list by using the Operating System drop-down menu above the list of images.

If you selected Boot Volume, you see a tabular list with columns Name, Size (GB), and Created (the date the boot volume was created). You can use the arrows in the column headings to sort the list. In the Boot Volume section (after the Shape section), you can customize the boot volume size.

If the list is too long to fit in one view, use the arrow buttons to view another page of the list.

To use a platform image that was previously available but is no longer listed, use the OCI CLI to create the instance and specify the OCID of the image.

The source image cannot be changed after the instance is created.

- *Shape*: Select a shape. For a description of each compute instance shape, see Compute Shapes in the *Oracle Private Cloud Appliance Concepts Guide*.

  If you select a standard shape, the amount of memory and number of OCPUs are displayed. These numbers match the numbers shown for this shape in the table in the *Oracle Private Cloud Appliance Concepts Guide*.

  If you select a flexible shape, you must specify the number of OCPUs you want and you can specify the total amount of memory you want. The default value for gigabytes of memory is 16 times the number you specify for OCPUs. Click inside each value field to see the minimum and maximum allowed values.

  If the system includes an optional GPU expansion and you plan to use the instance for GPU-accelerated workloads, select a dedicated GPU shape. You can select 1-4 GPUs, and the instance is allocated 27 OCPUs and 200GB memory per GPU. For driver installation, see known issue "GPU Drivers Not Included in Oracle Linux Platform Images".

  The shape and shape configuration can be changed after the instance is created.

- *Boot Volume*: (Optional) Check the box to specify a custom boot volume size or volume performance setting.

  – **Boot volume size (GB)**: The default boot volume size for the selected image is shown. To specify a larger size, enter an integer number of gigabytes up to 16384 (16 TB) or use the increment and decrement arrows. You cannot enter a value smaller than the default size.

    If you specify a custom boot volume size, you need to extend the partition to take advantage of the larger size. Oracle Linux platform images include the `oci-utils` package. Use the `oci-growfs` command from that package to extend the root partition and then grow the file system. For other operating systems or for custom images, follow the instructions for that operating system.

  – **Boot volume performance (VPUs)**: Use the increment and decrement arrows to toggle between balanced performance (10 VPUs/GB) and high performance (20 VPUs/GB). For more information, see "Block Volume Performance Options" in the Block Volume Storage Overview chapter in the *Oracle Private Cloud Appliance Concepts Guide*. Before you specify high performance, check with an appliance administrator to verify that a high performance pool is available.

    If you specify the high performance option and a high performance pool exists but the specified image does not exist in the high performance pool (the high performance pool was created after the image was imported), the specified image is copied from the capacity pool to the high performance pool. This operation can

take 20-30 minutes, depending on the image size, network configuration, and load on the appliance.

This copy operation is a one-time operation for each image. Future requests to create an instance specifying this image and the high performance pool will not incur this image copy delay.

You will receive an error message from the image copy if the image is larger than 200 gigabytes. Platform images (see "Latest Features" in the Oracle Private Cloud Appliance Release Notes) are not larger than 200 gigabytes.

- *Subnet*: Select a subnet.
  a. Select a VCN from the list. You might need to change the compartment to the compartment where the VCN is located.
  b. Select a subnet.
- *Public IP Address*: To connect to the instance using SSH, check the Assign Public IP box to have a public IP address assigned to the instance. This box is checked by default if you specified a public subnet. If you do not check this box, or if you uncheck this box, and then want to assign a public IP address later, see Assigning an Ephemeral Public IP Address to an Instance for instructions.
- *Private IP Address*: (Optional) Specify an available private IP address from the subnet's CIDR. By default, a private IP address is automatically assigned.
- *Hostname*: (Optional) Enter a hostname if you are using DNS within the cloud network. The hostname must be unique across all VNICs in the subnet.

  By default, the instance name is used for the hostname. The hostname can also be configured in the OS after the instance is created.

  If this is a UNIX instance, see Creating a Mount Target and Mounting File Systems on UNIX-Based Instances for more information about setting the host name correctly for mounting file systems.
- *SSH Keys*: To connect to the instance using SSH, provide a public SSH key.

  > **Note:**
  >
  > You cannot provide this SSH key after the instance is created.

- *Initialization Script*: (Optional) Provide an initialization script. This is a file of data to be used for custom instance initialization.
- *Network Security Group*: (Optional) By default, the new instance is not attached to any NSG. Check the box labeled Enable Network Security Group to add the primary VNIC for this instance to one or more NSGs.
  a. Select an NSG from the drop-down list. You might need to change the compartment to find the NSG you want.
  b. Click the Add Another NSG button if you want to attach to another NSG.
  c. To remove an NSG from the list, click the trash can to the right of that NSG. To remove the last NSG or all NSGs, uncheck the Enable Network Security Groups box.

  To update NSG attachments for this instance later, see Updating a VNIC.

  See Controlling Traffic with Network Security Groups for information about NSGs.

- *Instance Options*: Check the box to disable Legacy Instance Metadata Service Endpoints. By default, legacy (`/v1`) Instance Metadata Service (IMDS) routes are enabled. If you have upgraded your applications to use `/v2` endpoints, check this box to disable `/v1` endpoints. For more information about the Instance Metadata Service, see Retrieving Instance Metadata from Within the Instance. For more information about upgrading your applications, see Upgrading to IMDS Version 2 Endpoints.
- *Availability configuration*: (Optional) Specify how to handle this instance during compute node maintenance.
  - "Let Oracle Cloud Infrastructure choose the best migration option" is selected by default to allow the system to choose the best option to handle this instance during compute node maintenance. This best option typically is live migration to a healthy compute node. You cannot change this setting. If this instance should not be live migrated, for example live migration is not supported for instances in a Microsoft Windows cluster, then set the PCA_no_lm free-form tag to True to prevent live migration for this instance.
  - "Restore instance lifecycle state after infrastructure maintenance" is selected by default to specify that running instances should be automatically restarted after a maintenance operation such as live migration. If this box is not checked, the instance is recovered in the stopped state. For more information, see Configuring the Compute Service for High Availability in the *Oracle Private Cloud Appliance Administrator Guide*.
- *Tagging*: (Optional) Add defined or free-form tags for this instance as described in Adding Tags at Resource Creation. Tags can also be applied later.

4. Click the Create Instance button in the dialog.

   On success, the instance details page is displayed. On the Configuration tab, the Shape Configuration column shows the shape, the number of OCPUs, the network bandwidth, and the total memory. On the Networking tab, the VNIC column shows the VCN and subnet, and the Instance Access column shows the primary private IP address and any assigned public IP address.

   To check the status of the instance launch, scroll to the Resources section and click Work Request(s).

   If instance launch fails because of resource constraints, try remedies such as the following:

   - Specify a different fault domain or do not specify any fault domain and allow the system to choose.
   - Specify a less resource-intensive shape.
   - Stop an instance that you do not need currently.
   - Terminate an instance that you no longer need.

   If the status of the work request is Failed, and no reason is given for the failure, the cause of the failure might be temporary. If no reason is given for the failure, wait a short time and then retry the instance create.

Using the OCI CLI

1. Create or get the following resources and information:
   - The OCID of the compartment where you want to create the instance: `oci iam compartment list`
   - The name of the shape for this instance. Use the following command to list the available shapes and their characteristics. Use the OCID of the compartment where you want to create the instance. To list only shapes that are compatible with the image

that you plan to use, specify the image OCID. See also Compute Shapes in the *Oracle Private Cloud Appliance Concepts Guide*.

```
$ oci compute shape list --compartment-id compartment_OCID --image-id image_OCID
```

If you specify the flexible shape, `VM.PCAStandard1.Flex`, then you must also specify the shape configuration, as shown in the following example. You must provide a value for `ocpus`. The `memoryInGBs` property is optional; the default value in GBs is 16 times the number of `ocpus`.

```
--shape-config '{"ocpus": 32, "memoryInGBs": 512}'
```

If you specify a standard shape, do not specify `--shape-config`. The number of OCPUs and amount of memory are set to the values shown for this shape in "Standard Shapes" in Compute Shapes in the *Oracle Private Cloud Appliance Concepts Guide*.

The shape and shape configuration can be changed after the instance is created.

- The OCID of the subnet where the VNIC that is attached to this instance will be created: `oci compute vnic-attachment list`
- If you provide a value for the `--hostname-label` option, see the description of Hostname in the preceding Compute Web UI procedure.
- One of the following to specify either an image or a boot volume.
  - The OCID of the image used to boot the instance: `oci compute image list`

    > **Note:**
    >
    > Do not choose an image that has "`-OKE-`" in its display name. The "`-OKE-`" images can only be used with Kubernetes Engine. See the Oracle Private Cloud Appliance Kubernetes Engine user guide.

  - The OCID of the boot volume used to boot the instance: `oci compute boot-volume-attachment list`
- A public Secure Shell (SSH) key to connect to the instance using SSH.

  > **Note:**
  >
  > You cannot provide this SSH key after the instance is created.

For a complete list of required and optional parameters, use the following command:

```
$ oci compute instance launch -h
```

You might want to specify the `--availability-config` option to set a `recoveryAction` value to be used after a compute node maintenance operation. The value of `recoveryAction` can be `RESTORE_INSTANCE` or `STOP_INSTANCE`. For the meaning of these settings, see Configuring the Compute Service for High Availability in the *Oracle Private Cloud Appliance Administrator Guide*.

```
--availability-config '{"isLiveMigrationPreferred": true, "recoveryAction":
"RESTORE_INSTANCE"}'
```

The settings shown in this example are the default values (a `null` value of `isLiveMigrationPreferred` behaves the same as `true`). If you explicitly set `isLiveMigrationPreferred`, you must set it to `true`. If this instance should not be live migrated, for example live migration is not supported for instances in a Microsoft Windows cluster, then set the `PCA_no_lm` free-form tag to `True` to prevent live migration for this instance.

See the Compute Web UI procedure for characteristics of the `--display-name` and `--hostname-label` values. See Adding Tags at Resource Creation to add defined and free-form tags.

2. Construct an argument for the `--source-details` option.

   The `--source-details` argument can be a JSON file or a command-line string. Use the following command to show the correct format of the JSON properties and values:

   ```
   $ oci compute instance launch --generate-param-json-input source-details
   ```

   For information about `bootVolumeSizeInGBs`, see "Boot volume size" in the preceding Compute Web UI procedure.

   For information about `bootVolumeVpusPerGB`, see "High Performance" in the preceding Compute Web UI procedure.

   > **Note:**
   >
   > When you later `list` or `get` this instance, the value of `bootVolumeVpusPerGB` is `null` because this boot volume property is not stored in the instance object after the instance is launched. To check the value after instance launch, use the `bv boot-volume list` or `get` command and check the value of `vpus-per-gb`.

3. (Optional) Construct an argument for the `--launch-options` option.

   Only the `firmware` property can be changed. The default value is BIOS. You can alternatively specify UEFI_64. If you do not provide a correct value for `firmware`, the instance might not launch properly. You cannot update the value of the `firmware` property with the `instance update` command.

   The following shows the default values:

   ```
   {
     "bootVolumeType": "PARAVIRTUALIZED",
     "firmware": "BIOS",
     "isConsistentVolumeNamingEnabled": false,
     "is-pv-encryption-in-transit-enabled": false,
     "networkType": "PARAVIRTUALIZED",
     "remoteDataVolumeType": "PARAVIRTUALIZED"
   }
   ```

   To change the value of the `firmware` property, specify the following option:

   ```
   --launch-options file://launch_options.json
   ```

   Where the following is the content of the `launch_options.json` file:

   ```
   {
     "bootVolumeType": "PARAVIRTUALIZED",
     "firmware": "UEFI_64",
     "isConsistentVolumeNamingEnabled": false,
   ```

```
  "is-pv-encryption-in-transit-enabled": false,
  "networkType": "PARAVIRTUALIZED",
  "remoteDataVolumeType": "PARAVIRTUALIZED"
}
```

4. (Optional) Construct an argument for the `--metadata` or `--extended-metadata` option.

   Custom user data can be attached to the instance by using the `--metadata` and `--extended-metadata` options. Metadata key/value pairs are string/string maps in JSON format. Extended metadata can be nested JSON objects.

   The combined size of the metadata and extended metadata can be a maximum of 32,000 bytes. See "Metadata Key Limits" in the Compute Instance Concepts chapter of the *Oracle Private Cloud Appliance Concepts Guide* for more information about metadata limits.

   SSH keys can be provided in the value of the `ssh_authorized_keys` metadata key or in the file argument of the `--ssh-authorized-keys-file` option, as shown in the example in the next step. This value must be a valid public key in OpenSSH format. Use a newline character to separate multiple keys.

   User data can be provided in the `user_data` metadata key or in the file argument of the `--user-data-file` option. This value is data that cloud-init can use to run custom scripts or provide custom cloud-init configuration. For Linux instances with cloud-init configured, the `user_data` value is a Base64-encoded string of cloud-init user data. For more information, see cloud-init user data formats.

5. Run the instance launch command.

   Syntax:

   ```
   oci compute instance launch --availability-domain AD-1 \
   --compartment-id compartment_OCID --shape shape --subnet-id subnet_OCID \
   --source-details file://image_info.json
   ```

   Example:

   If you are using a public subnet, a public IP address is assigned by default, or you can set the `--assign-public-ip` option value to `true`. If you need to assign a public IP address later, see Assigning an Ephemeral Public IP Address to an Instance for instructions.

   If you have upgraded your applications to use `/v2` Instance Metadata Service (IMDS) endpoints, use the `--instance-options` option to set `areLegacyImdsEndpointsDisabled` to `true`. By default, legacy (`/v1`) Instance Metadata Service routes are enabled. For more information about the Instance Metadata Service, see Retrieving Instance Metadata from Within the Instance. For more information about upgrading your applications, see Upgrading to IMDS Version 2 Endpoints.

   ```
   $ oci compute instance launch --availability-domain AD-1 \
   --compartment-id ocid1.compartment.unique_ID --display-name ops1 \
   --shape VM.PCAStandard1.16 --subnet-id ocid1.subnet.unique_ID --source-details \
   '{"bootVolumeSizeInGBs":100,"bootVolumeVpusPerGB":20,"imageId":"ocid1.image.unique_ID
   ","sourceType":"image"}' \
   --assign-public-ip true --ssh-authorized-keys-file ./.ssh/
   name_of_public_SSH_key_file \
   --instance-options '{"areLegacyImdsEndpointsDisabled": true}'
   {
     "data": {
       "agent-config": null,
       "availability-config": {
         "is-live-migration-preferred": null,
         "recovery-action": "RESTORE_INSTANCE"
       },
       "availability-domain": "AD-1",
   ```

```
    "capacity-reservation-id": null,
    "compartment-id": "ocid1.compartment.unique_ID",
    "dedicated-vm-host-id": null,
    "defined-tags": {},
    "display-name": "ops1",
    "extended-metadata": null,
    "fault-domain": "FAULT-DOMAIN-1",
    "freeform-tags": {},
    "id": "ocid1.instance.unique_ID",
    "image-id": "ocid1.image.unique_ID",
    "instance-options": {
      "are-legacy-imds-endpoints-disabled": true
    },
    "ipxe-script": null,
    "launch-mode": "PARAVIRTUALIZED",
    "launch-options": {
      "boot-volume-type": "PARAVIRTUALIZED",
      "firmware": "BIOS",
      "is-consistent-volume-naming-enabled": false,
      "is-pv-encryption-in-transit-enabled": false,
      "network-type": "PARAVIRTUALIZED",
      "remote-data-volume-type": "PARAVIRTUALIZED"
    },
    "lifecycle-state": "PROVISIONING",
    "metadata": {
      "ssh_authorized_keys": "public_SSH_key"
    },
    "platform-config": null,
    "preemptible-instance-config": null,
    "region": "region_name",
    "shape": "VM.PCAStandard1.16",
    "shape-config": {
      "baseline-ocpu-utilization": null,
      "gpu-description": null,
      "gpus": null,
      "local-disk-description": null,
      "local-disks": null,
      "local-disks-total-size-in-gbs": null,
      "max-vnic-attachments": 16,
      "memory-in-gbs": 256.0,
      "networking-bandwidth-in-gbps": 24.6,
      "ocpus": 16.0,
      "processor-description": null
    },
    "source-details": {
      "boot-volume-size-in-gbs": 100,
      "bootVolumeVpusPerGB": 20,
      "image-id": "ocid1.image.unique_ID",
      "kms-key-id": null,
      "source-type": "image"
    },
    "system-tags": null,
    "time-created": "2021-09-22T20:20:04.715304+00:00",
    "time-maintenance-reboot-due": null
  },
  "etag": "92180faa-3660-446c-9559-c12a6e6111f9",
  "opc-work-request-id": "ocid1.workrequest.unique_ID"
}
```


Use the `work-requests work-request get` command to monitor the status of the instance launch:

```
$ oci work-requests work-request get --work-request-id ocid1.workrequest.unique_ID
```

If the status of the work request is Failed, and no reason is given for the failure, the cause of the failure might be temporary. If no reason is given for the failure, wait a short time and then retry the instance create.

Retrieving Instance Metadata from Within the Instance

The Instance Metadata Service (IMDS) serves information about a running instance to users who are logged in to that instance. IMDS also provides information to cloud-init that you can use for various system initialization tasks.

> **Note:**
>
> To access IMDS metadata, use an instance image that is provided by Oracle.

The IMDS metadata includes instance information such as the following:

- The SSH public key that enables users to log in to the instance
- Instance attached VNICs, VNIC IDs
- Instance CIDR blocks

In general, the IMDS instance metadata includes the following:

- The same information that you see on the details page of an instance in the Compute Web UI and in the output of the instance `get` command in the OCI CLI.
- Custom information that you add to an instance by using the `--metadata`, `--extended-metadata`, `--ssh-authorized-keys-file`, and `--user-data-file` options of the instance `launch` command. This metadata cannot be updated after instance launch. For a user logged into the instance, the instance metadata is read-only.

Upgrading to IMDS Version 2 Endpoints

The Instance Metadata Service is available in two versions: version 1 and version 2.

> **Important:**
>
> To increase the security of metadata requests, upgrade all applications to use the IMDS version 2 endpoints, if supported by the image. Then disable use of IMDS version 1 endpoints.

IMDS version 2 endpoints (IMDSv2) are supported on the Oracle Linux images listed in "Platform Images" in the "Feature Updates" chapter of the Oracle Private Cloud Appliance Release Notes. Other platform images and most other images do not support IMDSv2.

For each instance, perform the following steps to upgrade to IMDSv2:

1. Identify applications that are making IMDSv1 requests.

   For example, cloud-init makes requests to `/v#` instance endpoints.

2. Migrate the identified applications to support IMDSv2 endpoints.

When you use `/v2` endpoints, you must include the "Authorization: Bearer Oracle" header. See the examples in Retrieving IMDS Instance Metadata.

3. Disable IMDSv1 endpoints.

   Perform one of the following steps, as described in Creating an Instance.

   - On the details page of an instance, under Instance Details, check the value of Legacy Instance Metadata Service Endpoints. If the value of Legacy Instance Metadata Service Endpoints is Enabled, click Edit on the Controls menu, and check the box for Legacy Instance Metadata Service Endpoints Disabled.
   - In the output from `instance list` or `instance get`, under `instance-options`, check the value of `are-legacy-imds-endpoints-disabled`. If the value of `are-legacy-imds-endpoints-disabled` is `null` or `false`, use the `instance update` command to specify the following option:

     ```
     --instance-options '{"areLegacyImdsEndpointsDisabled": true}'
     ```

   Future requests to legacy (`v1`) endpoints will be rejected with a 404 not found error.

Retrieving IMDS Instance Metadata

To retrieve the IMDS instance metadata, follow these steps:

1. Log in to the instance.
2. Use a cURL command to retrieve the metadata information from the HTTP endpoint.

   Information is provided through an HTTP endpoint that listens on 169.254.169.254. If an instance has multiple VNICs, you must send the request using the primary VNIC.

   Use the `instance` command to retrieve the instance metadata. Use the `vnics` command to retrieve the VNIC data.

   If you are using `/v2` endpoints, as shown in the following examples, then you must include the "Authorization: Bearer Oracle" header.

   Example: Instance Metadata

   ```
   $ curl -H "Authorization: Bearer Oracle" -L http://169.254.169.254/opc/v2/instance/
   {
       "availabilityDomain": "AD-1",
       "faultDomain": "FAULT-DOMAIN-1",
       "compartmentId": "ocid1.compartment.unique_ID",
       "displayName": "dev1",
       "hostname": "hostname",
       "id": "ocid1.instance.unique_ID",
       "image": "ocid1.image.unique_ID",
       "metadata": {
           "ssh_authorized_keys": "public_SSH_key"
       },
       "region": "PCA",
       "canonicalRegionName": "PCA",
       "ociAdName": "PCA",
       "regionInfo": null,
       "shape": "VM.PCAStandard1.1",
       "state": "RUNNING",
       "timeCreated": 1634943279000,
       "agentConfig": null
   }
   ```

   To retrieve a single value, specify the key name as shown in the following example.

   Example: VNIC Metadata

```
$ curl -H "Authorization: Bearer Oracle" -L http://169.254.169.254/opc/v2/vnics/
[
    {
        "vnicId": "ocid1.vnic.unique_ID",
        "privateIp": "privateIp",
        "vlanTag": 0,
        "macAddr": "00:13:97:9f:16:32",
        "virtualRouterIp": "virtualRouterIp",
        "subnetCidrBlock": "subnetCidrBlock"
    }
]
```

You can view all of the data for one of multiple VNICs by specifying the array index for that VNIC data, or you can retrieve a single value for that specified VNIC:

```
$ curl -H "Authorization: Bearer Oracle" -L http://169.254.169.254/opc/v2/vnics/0/
privateIp
privateIp
```

Updating an Instance

In addition to updating the properties of an instance, you might want to attach additional block volumes or secondary VNICs. See Creating and Attaching Block Volumes and Creating and Attaching a Secondary VNIC. You can also specify block volumes and secondary VNICs when you use an instance configuration to create an instance.

If you did not add a public IP address when you created the instance, and you want to assign a public IP address now, see Assigning an Ephemeral Public IP Address to an Instance for instructions.

You can update the display name, fault domain, shape, instance options, availability configuration, and tags of an instance. By using the OCI CLI, you can also update the instance metadata. See the descriptions of these properties in Creating an Instance.

If the updated instance cannot run because of resource constraints, see the suggested remedies in Creating an Instance.

Using the Compute Web UI

1. In the Compute block on the Dashboard, click Instances.
2. If the instance that you want to update is not listed, use the Compartment drop-down menu above the instances list to select the correct compartment.
3. For the instance that you want to update, click the Actions menu, and click the Edit option.
4. In the Edit *instance_name* dialog, make the changes.

   When you update an instance by using the Compute Web UI, you can change the following:

   - Name of the instance
   - Fault domain

     When you change the fault domain of a stopped instance, the new fault domain is set in the instance properties. When you change the fault domain of a running instance, the instance is stopped, moved, and started on a new compute node in the new fault domain. See Stopping, Starting, and Resetting an Instance for how to prepare for an instance to stop. Stopping and starting an instance can take up to five minutes.

If you specify a fault domain, and the requested fault domain cannot accommodate the instance, instance restart fails; the instance remains stopped. The new fault domain specification remains in the instance properties.

- Shape

  When you change the shape of a stopped instance, the new shape and shape configuration are set in the instance properties. When you change the shape of a running instance, the instance is stopped, reconfigured, and restarted. See Stopping, Starting, and Resetting an Instance for how to prepare for an instance to stop. Stopping and starting an instance can take up to five minutes.

  If you select the flexible shape, VM.PCAStandard1.Flex, you must specify the number of OCPUs you want and you can specify the total amount of memory you want. The default value for GBs of memory is 16 times the number you specify for OCPUs. Click inside each value field to see the minimum and maximum allowed values.

  If the specified shape and shape configuration cannot be accommodated in the fault domain, instance restart fails; the instance remains stopped. The new shape and shape configuration remain in the instance properties.

- Instance Options

  If you have upgraded your applications to use `/v2` Instance Metadata Service (IMDS) endpoints, check this box to disable `/v1` endpoints. For more information about the Instance Metadata Service, see Retrieving Instance Metadata from Within the Instance.

- Availability configuration

  This sets whether an instance that has been stopped by the Compute service is restarted when resources become available (the default) or remains stopped. See Configuring the Compute Service for High Availability in the *Oracle Private Cloud Appliance Administrator Guide*.

- Tags

5. Click the Save Changes button.

   If you changed the fault domain, shape, or shape configuration of a running instance, you must confirm that you understand that the instance will be rebooted. For those changes, the instance will be stopped, reconfigured, and restarted.

Using the OCI CLI

1. Get the OCID of the instance that you want to update: `oci compute instance list`

   If you want to change the fault domain of the instance, get the OCID of the fault domain:

   ```
   $ oci iam fault-domain list --compartment-id compartment_OCID \
   --availability-domain AD-1
   ```

   If you want to change the shape of the instance, get the name of the shape:

   ```
   $ oci compute shape list --compartment-id compartment_OCID --image-id image_OCID
   ```

2. Run the instance update command.

   Syntax:

   ```
   oci compute instance update --instance-id instance_OCID \
   options_with_values_to_update
   ```

   For descriptions of instance properties that you can change, enter the following command and scroll to Optional Parameters:

```
$ oci compute instance update -h
```

Use `--instance-options` if you need to disable IMDSv1 endpoints for this instance. See Retrieving Instance Metadata from Within the Instance.

You might want to specify the `--availability-config` option to set a `recoveryAction` value to be used after a compute node maintenance operation. For the meaning of this setting, see Configuring the Compute Service for High Availability in the *Oracle Private Cloud Appliance Administrator Guide*. For how to set this property and default values, see Creating an Instance. If you are using Terraform, see "Terraform Instance Update Requires Explicit Live Migration Setting" in Compute Service Issues in Known Issues and Workarounds in the Oracle Private Cloud Appliance Release Notes.

See the Compute Web UI procedure for more information about changing the fault domain or shape.

If you specify the flexible shape, `VM.PCAStandard1.Flex`, then you must also specify `--shape-config` You must provide a value for `ocpus`. The `memoryInGBs` property is optional; the default value in GBs is 16 times the number of `ocpus`.

If you specify a standard shape, do not specify `--shape-config`. The number of OCPUs and amount of memory are set to the values shown for this shape in "Standard Shapes" in Compute Shapes in the *Oracle Private Cloud Appliance Concepts Guide*.

Example:

```
$ oci compute instance update --instance-id ocid1.instance.unique_ID \
--shape VM.PCAStandard1.Flex --shape-config '{"ocpus": 16, "memoryInGBs": 512}'
```

Moving an Instance to a Different Compartment

You can move an instance to a different compartment within the same tenancy.

To move an instance, you must use the OCI CLI.

Using the OCI CLI

1. Get the following information:
   - The OCID of the destination compartment: `oci iam compartment list`
   - The OCID of the instance: `oci compute instance list`
2. Run the instance change compartment command.

   Syntax:

   ```
   oci compute instance change-compartment \
   --compartment-id destination_compartment_OCID \
   --instance-id instance_OCID
   ```

Stopping, Starting, and Resetting an Instance

You can perform the following power actions on an instance: Start, stop, soft stop, reset, and soft reset. The following note applies to all of the stop and reset actions.

> **Important:**
>
> For soft stop and soft reset, any application that is running on the instance that takes more than 15 minutes to shut down could be improperly stopped, resulting in data corruption. For stop and reset, any application that is running on the instance will be immediately stopped, possibly resulting in data corruption. To avoid stopping the instance while applications are running, manually shut down the instance by using the commands available in the instance OS.
>
> After the instance is shut down from the OS, then stop, soft stop, reset, or soft reset the instance from the appliance.

- START. Power on the instance. The Compute service attempts to restart the instance in the same fault domain that it was in when it was stopped, or in the currently specified fault domain if the fault domain was updated while the instance was stopped. If the instance start operation fails, the instance remains stopped.

  If the start operation fails because of resource constraints, you could specify a different fault domain for the instance (see Updating an Instance), change the configuration of the instance, or stop, reconfigure, or terminate other instances.

  If the start operation fails, and no reason is given for the failure, the cause of the failure might be temporary. If no reason is given for the failure, wait a short time and then retry the instance start.

  Starting an instance can take up to five minutes.
- STOP. Power off the instance. Compute resources are released and the instance is disconnected and unassigned from the compute node. See the important note at the beginning of this section.

  Stopping an instance can take up to five minutes.

  An instance that is Stopped cannot be migrated to a different compute node. See "Migrating Instances from a Compute Node" in Hardware Administration in the Oracle Private Cloud Appliance Administrator Guide.
- SOFTSTOP. Gracefully shut down the instance by sending a shutdown command to the operating system. After waiting 15 minutes for the OS to shut down, the instance is powered off. See also the description of STOP and the important note at the beginning of this section.
- RESET. Power off the instance and then power it back on. See the descriptions of STOP and START and the important note at the beginning of this section.
- SOFTRESET. Gracefully reboot the instance by sending a shutdown command to the operating system. After waiting 15 minutes for the OS to shut down, the instance is powered off and then powered back on. See the important note at the beginning of this section.

Using the Compute Web UI

1. In the Compute block on the Dashboard, click Instances.

   If the instance that you want to manage is not listed, use the Compartment drop-down menu above the instances list to select the correct compartment.
2. For the instance that you want to manage, click the Actions menu, and click the Start, Stop, or Reset option.

Alternatively, in the instance list, click the name of the instance to display the details page for that instance. Click the Controls menu, and click the Start, Stop, or Reset option.

3. If applicable, enable force stop or force reboot options.

   By default, clicking Stop performs Soft Stop and clicking Reset performs Soft Reset. To stop or reset the instance immediately, enable the Force option on the confirmation dialog.

   - On the "Stop the instance named *instance_name*" dialog, enable the "Force stop the instance by immediately powering off" option.
   - On the "Reboot the instance named *instance_name*" dialog, enable the "Force reboot the instance by immediately powering off, then powering back on" option.

4. Click the Start Instance, Stop Instance, or Reboot Instance button on the confirmation dialog.

   In the Resources section of the instance details page, click Work Request(s) to check the status of the instance start, stop, or reboot.

Using the OCI CLI

1. Get the OCID of the instance that you want to stop, start, or reset: `oci compute instance list`
2. Run the `START`, `STOP`, `RESET`, `SOFTSTOP`, or `SOFTRESET` command.

   Syntax:

   ```
   oci compute instance action --instance-id instance_OCID \
   --action {START | STOP | RESET | SOFTSTOP | SOFTRESET}
   ```

   For descriptions of these actions, enter:

   ```
   $ oci compute instance action -h
   ```

   Example:

   ```
   $ oci compute instance action --instance-id ocid1.instance.unique_ID --action RESET
   ```

   If you need more information about the instance state change, see the logs as described in "Accessing System Logs" in Status and Health Monitoring in the Oracle Private Cloud Appliance Administrator Guide.

Terminating an Instance

By default, the boot volume of the instance is preserved when you terminate the instance. You can attach the boot volume to a different instance as a data volume, or use it to launch a new instance. You have the option to permanently delete the attached boot volume when you terminate the instance. If you preserve the attached boot volume and then later decide to permanently delete the volume, see Deleting a Boot Volume.

Using the Compute Web UI

1. In the Compute block on the Dashboard, click Instances.

   If the instance that you want to terminate is not listed, use the Compartment drop-down menu above the instances list to select the correct compartment.

2. For the instance that you want to terminate, click the Actions menu, and click the Terminate option.

   Alternatively, in the instance list, click the name of the instance to display the details page for that instance. Click the Controls menu, and click the Terminate option.

3. On the confirm instance termination dialog, choose whether to permanently delete the attached boot volume.

   By default, the option to permanently delete the attached boot volume is not enabled. The boot volume will be preserved after the instance is terminated. To permanently delete the boot volume at instance termination, enable the permanently delete the attached boot volume option in the confirm instance termination dialog.

4. On the confirm instance termination dialog, click the Confirm button.

   Click Work Request(s) in the Resources box to check the status of the instance terminate. After the TerminateInstance operation is 100% complete and in state Succeeded, the instance remains visible in the instance list in state Terminated for at least 24 hours, up to 24.5 hours. No further action is needed to terminate the instance.

Using the OCI CLI

1. Get the OCID of the instance that you want to terminate: `oci compute instance list`
2. Run the instance terminate command.

   To permanently delete the attached boot volume when the instance is terminated, specify the following option on the instance terminate command: `--preserve-boot-volume false`

   Example:

   ```
   $ oci compute instance terminate --instance-id ocid1.instance.unique_ID
   ```

   Use the `work-requests work-request get` command to check the status of the instance terminate. After the `TerminateInstance` operation is `percent-complete 100.0` and `status SUCCEEDED`, the instance remains visible in instance `list` or `get` in `lifecycle-state TERMINATED` for at least 24 hours, up to 24.5 hours. No further action is needed to terminate the instance.

Working with Instance Configurations

An instance configuration contains settings that are used to create a compute instance. Instance configurations enable you to consistently create instances with the same configuration without re-typing the configuration values using the OCI CLI or re-entering the configuration in the Compute Web UI. You can use an instance configuration to create a single instance or to create an instance pool.

Creating an Instance Configuration

You can create an instance configuration from an existing instance (a template instance) or by entering the individual configuration settings.

Creating an Instance Configuration from an Instance

These procedures describe how to create an instance configuration by using the configuration information from an existing compute instance (a template instance).

Note the following when you use a template instance to create an instance configuration:

- The new instance configuration does not include any information from the boot volume of the template instance. For example, installed applications, binaries, and files that are on the template instance are not included in the instance configuration.

Use the following procedure to create an instance configuration that includes the custom setup from the template instance:

1. Create a custom image from the template instance. See Creating an Image from an Instance.
2. Use the custom image to create a new instance. See Creating an Instance.
3. Use the instance that you created in the preceding step as the template to create the instance configuration.

- Instances with replicated boot volumes cannot be used as template instances for an instance configuration. For example, instances that are included in a disaster recovery configuration cannot be used as template instances for an instance configuration.
- The instance configuration does not include the contents of any block volumes that are attached to the template instance.
- Instances that are created from this new instance configuration are placed in the same compartment as the template instance.

To include block volumes or secondary VNICs in the configuration, or to specify the compartment where new instances will be created, create the instance configuration as described in Creating an Instance Configuration by Entering Configuration Values.

Using the Compute Web UI

1. On the Dashboard, click the Compute/View Instances button.
2. If the instance that you want to use to create the new instance configuration is not listed, use the Compartment drop-down menu above the instances list to select the correct compartment.
3. Click the name of the instance that you want to use to create the new instance configuration.
4. On the instance details page, click the Controls menu, and then click the Create Instance Configuration option.
5. In the Create Instance Configuration dialog, enter the following information:
   - *Name*: Enter a name for the instance configuration.
   - *Compartment*: Select the compartment where this instance configuration will be created.
   - *Tagging*: (Optional) Add defined or free-form tags for this instance as described in Adding Tags at Resource Creation. Tags can also be applied later.
6. Click the Create Instance Configuration button in the dialog.

Using the OCI CLI

1. Get the following information:
   - The OCID of the compartment where you want to create this instance configuration.
   - The OCID of the instance to use to create the instance configuration.
2. Run the instance configuration create command.

   Syntax:

```
oci compute-management instance-configuration create-from-instance \
--compartment-id compartment_OCID --instance-id ocid1.instance.unique_ID \
--display-name IC_name
```

The specified compartment is where this instance configuration will be created.

The specified display name is the name of the instance configuration. If you do not provide a value for the `--display-name` option, the default name of the instance configuration is `instanceconfiguration`***`YYYYMMDDhhmmss`***, where ***`YYYYMMDDhhmmss`*** is the creation date and time.

The output of this command is the same as the output of the `instance-configuration get` command.

Creating an Instance Configuration by Entering Configuration Values

These procedures describe how to create an instance configuration by entering values for individual instance configuration settings in the Compute Web UI or in command line options.

Using the Compute Web UI

1. In the navigation menu, click Compute, and then click Instance Configurations.
2. Click the Create Instance Configuration button.
3. In the Create Instance Configuration dialog, enter the following information:
   - *Name*: Enter a name for the instance configuration.
   - *Create in compartment*: Select the compartment where you want this instance configuration to be created.
   - *Compartment to create instances in*: Select the compartment where you want the instances that are created using this instance configuration to be created.
   - *Fault Domain*: (Optional) You can select a fault domain. By default, the system automatically selects the best fault domain for instances created using this instance configuration. If you specify a fault domain, and the requested fault domain cannot accommodate the instance, instance launch fails. See more information about fault domains in Creating an Instance.
   - *Source Image*: Select an image or boot volume.
     a. Select the Source Type: Platform Image, Custom Image, or Boot Volume.
     b. If you selected Custom Image or Boot Volume, select the compartment where the image or boot volume that you want to use is located.
     c. Select an image or boot volume from the list.

        If you selected Platform Image, you see a tabular list with columns Operating System, OS Version, and Image Build (the date the image was built). You can use the drop-down menu arrow to the right of the OS Version to select a different version. For example, for the Oracle Linux operating system, you can use the drop-down menu to select 9, 8, or 7.9.

        If you selected Custom Image, you see a tabular list with columns Name, Operating System, and OS Version. You can use the arrows in the column headings to sort the list. You can filter the list by using the Operating System drop-down menu above the list of images.

        If you selected Boot Volume, you see a tabular list with columns Name, Size (GB), and Created (the date the boot volume was created). You can use the arrows in the column headings to sort the list. In the Boot Volume section (after the Shape section), you can customize the boot volume size.

        If the list is too long to fit in one view, use the arrow buttons to view another page of the list.

To use a platform image that was previously available but is no longer listed, use the OCI CLI to create the instance and specify the OCID of the image.

- *Shape*: Select a shape. For a description of each compute instance shape, see Compute Shapes in the *Oracle Private Cloud Appliance Concepts Guide*.

  If you select a standard shape, the amount of memory and number of OCPUs are displayed. These numbers match the numbers shown for this shape in the table in the *Oracle Private Cloud Appliance Concepts Guide*.

  If you select the flexible shape, VM.PCAStandard1.Flex, you must specify the number of OCPUs you want and you can specify the total amount of memory you want. The default value for gigabytes of memory is 16 times the number you specify for OCPUs. Click inside each value field to see the minimum and maximum allowed values.

- *Boot Volume*: (Optional) Check the box to specify a custom boot volume size or volume performance setting.

  - **Boot volume size (GB)**: The default boot volume size for the selected image is shown. To specify a larger size, enter an integer number of gigabytes up to 16384 (16 TB) or use the increment and decrement arrows. You cannot enter a value smaller than the default size.

    If you specify a custom boot volume size, you need to extend the partition to take advantage of the larger size. Oracle Linux platform images include the `oci-utils` package. Use the `oci-growfs` command from that package to extend the root partition and then grow the file system. For other operating systems or for custom images, follow the instructions for that operating system.

  - **Boot volume performance (VPUs)**: Use the increment and decrement arrows to toggle between balanced performance (10 VPUs/GB) and high performance (20 VPUs/GB). For more information, see "Block Volume Performance Options" in the Block Volume Storage Overview chapter in the *Oracle Private Cloud Appliance Concepts Guide*.

- *Subnet*: Select a subnet.

  a. Select a VCN from the list. You might need to change the compartment to the compartment where the VCN is located.

  b. Select a subnet.

- *Public IP Address*: To use SSH to connect to instances created with this instance configuration, check the Assign Public IP box to have a public IP address assigned to the instances. This box is checked by default if you specified a public subnet. If you do not check this box, or if you clear this box, and then want to assign a public IP address later, see Assigning an Ephemeral Public IP Address to an Instance for instructions.

- *Secondary VNICs*: (Optional) Check the Create Additional VNIC box to create secondary VNICs for instances created with this instance configuration, For descriptions of the information requested here, see Creating and Attaching a Secondary VNIC.

- *Private IP Address*: (Optional) Specify an available private IP address from the subnet's CIDR. By default, a private IP address is automatically assigned. Because the private IP address must be unique for each instance, do not specify a private IP address if you are going to use this instance configuration to create an instance pool.

- *DNS Record*: (Optional) Check the Assign a private DNS record box to assign a DNS record to instances created with this instance configuration,

- *Hostname*: (Optional) Enter a hostname if you are using DNS within the cloud network. The hostname must be unique across all VNICs in the subnet. Do not specify a host name if you are going to use this instance configuration to create an instance pool.

  By default, the instance name is used for the hostname. The hostname can also be configured in the OS after the instance is created.

  If this is a UNIX instance, see Creating a Mount Target and Mounting File Systems on UNIX-Based Instances for more information about setting the host name correctly for mounting file systems.

- *SSH Keys*: To connect to the instance using SSH, provide a public SSH key.

  > **Note:**
  >
  > You cannot provide this SSH key after the instance is created.

- *Network Security Group*: (Optional) By default, instances are not attached to any NSG. Check the Enable Network Security Group box to add the primary VNIC for this instance to one or more NSGs.

  a. Select an NSG from the drop-down list. You might need to change the compartment to find the NSG you want.

  b. Click the Add Network Security Group button to attach to another NSG.

  c. To remove an NSG from the list, click the trash can to the right of that NSG. To remove the last NSG or all NSGs, clear the Enable Network Security Groups box.

  See Controlling Traffic with Network Security Groups for information about NSGs.

- *Instance Options*: Check the box to disable Legacy Instance Metadata Service Endpoints. By default, legacy (`/v1`) Instance Metadata Service (IMDS) routes are enabled. If you have upgraded your applications to use `/v2` endpoints, check this box to disable `/v1` endpoints. For more information about the Instance Metadata Service, see Retrieving Instance Metadata from Within the Instance. For more information about upgrading your applications, see Upgrading to IMDS Version 2 Endpoints.

- *Availability configuration*: (Optional) By default, the system automatically selects the best instance availability option during a maintenance operation such as live migration. Check the "Restore instance lifecycle state after infrastructure maintenance" box to specify that running instances should be automatically restarted after a maintenance event. If this box is not checked, the instance is recovered in the stopped state. For more information, see Configuring the Compute Service for High Availability in the *Oracle Private Cloud Appliance Administrator Guide*.

- *Tagging*: (Optional) Add defined or free-form tags for this instance as described in Adding Tags at Resource Creation. Tags can also be applied later.

4. Click the Create Instance Configuration button in the dialog.

Using the OCI CLI

1. Get the following information:

   - The OCID of the compartment where you want to create this instance configuration.
   - The OCID of the compartment where you want instances that use this instance configuration to be created.

- The OCID of the image or boot volume for instances that use this instance configuration.
- The name of the shape for instances that use this instance configuration.
- The OCID of the subnet for instances that use this instance configuration.

2. Create the configuration file that is input to the configuration create command.

   The configuration file is a JSON file of property/value pairs.

   - The following command shows the correct syntax of the configuration file and names of properties:

     ```
     $ oci compute-management instance-configuration create \
     --generate-param-json-input instance-details > instance_details.json
     ```

     You do not need all of the data that is output by this command. Copy just the information you need, being careful to keep each property in its correct context.

     If you omit the fault domain specification, the system automatically selects the best fault domain. If you specify only a single fault domain, all instances will be placed in only that fault domain.

     If a fault domain that you specify does not have enough resources, instances could fail to launch:

     – When you launch a single instance (Using an Instance Configuration to Launch an Instance), and you specify a fault domain in the instance configuration, only that specified fault domain will be used to launch the instance. Resource constraints could cause the instance launch to fail.

     – When you create instances in a pool, fault domains specified in the placement configuration override fault domains specified in the instance configuration. See Creating an Instance Pool for more information.

     You can specify secondary VNICs and subnets. If you specify a hostname label for a secondary VNIC, the specified hostname label must be unique across all VNICs in the subnet. If you provide a value for the `hostnameLabel` property, you must also set the value of `assignPrivateDnsRecord` to `true`.

     – If the specified hostname label is already in use in the subnet, instance launch (Using an Instance Configuration to Launch an Instance) will fail with the error "Hostname *`hostname`* already in use for the subnet."

     – The `hostnameLabel` property is ignored when you use the instance configuration to create a pool of instances. By default, the instance name is used for the hostname.

     If you omit the `assignPublicIp` property, a public IP address is assigned by default if you specify a public subnet. If you set this property to `false` and then decide to assign a public IP address later, see Assigning an Ephemeral Public IP Address to an Instance for instructions.

     If users will use `ssh` to connect to the instance, specify the SSH public key as the value of the `ssh_authorized_keys` property in the `metadata` block. You cannot add the SSH public key after the instance is created.

     The `displayName` property is used for the instance name when you use the `launch-compute-instance` command as described in Using an Instance Configuration to Launch an Instance. If you do not provide a value for the `displayName` property, the default name of instances will be `instance`***`YYYYMMDDhhmmss`***, where ***`YYYYMMDDhhmmss`*** is the creation date and time.

The `displayName` property is ignored when you create instances in a pool as described in Creating an Instance Pool.

- The following command shows which properties are required to create an instance:

  ```
  $ oci compute instance launch -h
  ```

  Scroll to the Required Parameters section. Optional parameters are described below the required parameters.

The names of the properties in the configuration file are similar to, but different from, the names of the `instance launch` options. Also, some properties are organized into groups of properties, such as `createVnicDetails`, `shapeConfig`, and `sourceDetails`, as shown in the following example configuration file:

```
{
  "instanceType": "compute",
  "launchDetails": {
    "availabilityDomain": "AD-1",
    "compartmentId": "compartment_OCID",
    "createVnicDetails": {
      "assignPublicIp": true,
      "freeformTags": {
        "ConfigType": "Configuration for an XYZ instance."
      },
      "subnetId": "subnet_OCID"
    },
    "displayName": "instance_name",
    "instanceOptions": {
      "areLegacyImdsEndpointsDisabled": true
    },
    "metadata": {
      "ssh_authorized_keys": "public_SSH_key"
    },
    "shape": "shape_name",
    "shapeConfig": {
      "memoryInGBs": 512,
      "ocpus": 32
    },
    "sourceDetails": {
      "bootVolumeSizeInGBs": 100,
      "bootVolumeVpusPerGB": 20,
      "imageId": "image_OCID",
      "sourceType": "image"
    }
  }
}
```

Use `instanceOptions` if you need to disable IMDSv1 endpoints for this instance. See Retrieving Instance Metadata from Within the Instance.

If you specify the flexible shape, `VM.PCAStandard1.Flex`, then you must also specify the shape configuration, as shown in the preceding example. You must provide a value for `ocpus`. The `memoryInGBs` property is optional; the default value in gigabytes is 16 times the number of `ocpus`.

If you specify a standard shape, do not specify `shapeConfig`.

For information about `bootVolumeSizeInGBs`, see "Boot volume size" in the preceding Compute Web UI procedure.

For information about `bootVolumeVpusPerGB`, see "High Performance" in the preceding Compute Web UI procedure. When instances are launched, the value of

`bootVolumeVpusPerGB` is `null` because this boot volume property is not stored in the instance object after the instance is launched. To check the value, use the `get boot volume` command and see the value of `vpus-per-gb`.

To change the value of the `firmware` property, provide a value for the `launchOptions` property. The default value is BIOS. You can alternatively specify UEFI_64. Other properties in `launchOptions` cannot be changed.

```
"launchOptions": {
  "bootVolumeType": "PARAVIRTUALIZED",
  "firmware": "UEFI_64",
  "isConsistentVolumeNamingEnabled": false,
  "isPvEncryptionInTransitEnabled": false,
  "networkType": "PARAVIRTUALIZED",
  "remoteDataVolumeType": "PARAVIRTUALIZED"
}
```

3. Run the instance configuration create command.

   Syntax:

   ```
   oci compute-management instance-configuration create -c compartment_OCID \
   --display-name IC_name --instance-details file://custom_config_file.json
   ```

   The specified compartment is where this instance configuration will be created. This compartment could be different from the compartment specified in the instance details JSON file, which is where the instances will be created.

   The specified display name is the name of the instance configuration. If you do not provide a value for the `--display-name` option, the default name of the instance configuration is `instanceconfiguration`***`YYYYMMDDhhmmss`***, where ***`YYYYMMDDhhmmss`*** is the creation date and time. (See Step 2 for a description of the display name specified in the instance details JSON file.)

   The output of this command is the same as the output of the `instance-configuration get` command.

Updating an Instance Configuration

You can change the name of the instance configuration and change the tags. To change configuration such as the compartment, subnet, or image, create a new instance configuration.

Using the Compute Web UI

1. On the Dashboard, click the Compute/View Instances button.
2. In the Compute menu, click Instance Configurations.
3. If the instance configuration that you want to update is not listed, use the Compartment drop-down menu above the instance configurations list to select the correct compartment.
4. For the instance configuration that you want to update, click the Actions menu, and click the Edit option.
5. In the Update Instance Configuration dialog, make the changes.
6. Click the Update Instance Configuration button in the dialog.

Using the OCI CLI

1. Get the OCID of the instance configuration that you want to update: `oci compute-management instance-configuration list`

2. Run the instance configuration update command.

   Example:

```
$ oci compute-management instance-configuration update \
--instance-configuration-id ocid1.instanceConfiguration.unique_ID \
--defined-tags file://instcfgdeftags.json
```

Moving an Instance Configuration to a Different Compartment

You can move an instance configuration to a different compartment within the same tenancy. When you move an instance configuration to a different compartment, instances and instance pools created by using this instance configuration are not moved.

New instances and instance pools that are created using this instance configuration are created in the compartment specified in the instance configuration, not in the compartment to which the instance configuration has been moved.

To move an instance configuration, you must use the OCI CLI.

Using the OCI CLI

1. Get the following information:
   - The OCID of the destination compartment: `oci iam compartment list`
   - The OCID of the instance configuration: `oci compute-management instance-configuration list`
2. Run the instance configuration change compartment command.

   Syntax:

```
oci compute-management instance-configuration change-compartment \
--compartment-id destination_compartment_OCID \
--instance-configuration-id instance_configuration_OCID
```

Deleting an Instance Configuration

An instance configuration that is being used by any pool cannot be deleted.

Using the Compute Web UI

1. In the navigation menu, click Compute, and then click Instance Configurations.
2. If the instance configuration that you want to delete is not listed, use the Compartment drop-down menu above the instance configurations list to select the correct compartment.
3. Click the name of the instance configuration that you want to delete.
4. On the instance configuration details page, click the Delete button.
5. Click the Confirm button.

Using the OCI CLI

1. Get the OCID of the instance configuration that you want to delete: `oci compute-management instance-configuration list`
2. Run the instance configuration delete command.

   Example:

```
$ oci compute-management instance-configuration delete \
--instance-configuration-id ocid1.instanceConfiguration.unique_ID
Are you sure you want to delete this resource? [y/N]: y
```

Using an Instance Configuration to Launch an Instance

This section shows how to use the instance configuration that you created in Creating an Instance Configuration to launch a compute instance.

This method of launching a compute instance is an alternative to the method described in Creating an Instance.

The name of the instance will be one of the following:

- If the instance configuration specifies a value for the `displayName` property, the name of the instance will be `displayName`. If you use the same instance configuration with multiple `launch-compute-instance` commands, all instances will have the same name. Instance names are not required to be unique.
- If the instance configuration does not specify a value for the `displayName` property, the default name of the instance will be `instance`***`YYYYMMDDhhmmss`***, where ***`YYYYMMDDhhmmss`*** is the creation date and time.

Using the OCI CLI

1. Get the OCID of the instance configuration that you want to use to launch the instance: `oci compute-management instance-configuration list`
2. Run the instance configuration launch instance command.

   Example:

   ```
   $ oci compute-management instance-configuration launch-compute-instance \
   --instance-configuration-id ocid1.instanceConfiguration.unique_ID
   ```

   The output of this command is the same as the output of the `compute instance get` command with the addition of a work request OCID. Use the `work-requests work-request get` command to check the status of the instance launch.

   If the launch operation fails because of resource constraints, see the suggested remedies in Creating an Instance.

Connecting to a Compute Instance

The image that was used to create the instance might not be the most up-to-date version that is available. Best practice is to check for and install operating system updates whenever you log in to an instance, especially when you log in for the first time.

Prerequisites

You need the following information to connect to an instance:

- **The public IP address of the instance.**

  You can get the address from the Instance Details page in the Compute Web UI. Click Dashboard, and click the Compute/View Instances button. Click the name of your instance. On the instance details page, click the Networking tab. The Public IP Address is in the Instance Access section.

- **For UNIX instances: The full path to the private key portion of the SSH key pair that you used when you launched the instance.**

  For more information about key pairs, see Managing Key Pairs.

- **The initial user name for the instance.**

  The initial user name for an instance is determined by the image that was used to create the instance. Images fall into these categories:

  – **Images provided with Private Cloud Appliance:**

    If you used an image that is provided with the appliance such as Oracle Linux or Oracle Solaris to launch the instance, the user name is `opc`.

  – **Custom images:**

    The initial user depends on how the image was configured before it was imported as a custom image.

- **(In some circumstances) The initial user password.**

  The initial user password for an instance is determined by the image that was used to create the instance. Images fall into these categories:

  – **Images provided with Private Cloud Appliance:**

    Instances launched using an Oracle Linux or Oracle Solaris image that was provided by Oracle use SSH to authenticate a user, and there is no initial password required.

  – **Custom images:**

    The initial password depends on how the image was configured before it was imported as a custom image.

Managing Key Pairs

The method you use to log into an instance depends on how the image that was used to launch the instance was configured.

- **Images provided with Private Cloud Appliance** launch instances that use an SSH key pair instead of a password to authenticate a remote user. These images also include the `cloud-init` toolkit (required for SSH authentication) in launched instances.
- **Custom images** might be configured with the `cloud-init` toolkit and use SSH for authentication, or the image might be configured to use its own set of credentials to authenticate a user. For example, the image might require a password. If the image requires a password, you don't need to create an SSH key pair.

> **Note:**
>
> Only instances that were created with the `cloud-init` toolkit can use SSH key pairs.

A key pair consists of a private key and public key. You keep the private key on your computer and provide the public key when you create an instance. When you connect to the instance using SSH, you provide the path to the private key in the SSH command.

You can have as many key pairs as you want, or you can keep it simple and use one key pair for all or several of your instances.

To create your own key pairs, you can use a third-party tool such as OpenSSH on UNIX systems (including Linux, Oracle Solaris, BSD, and macOS) or PuTTY Key Generator on Microsoft Windows.

Required SSH Public Key Format

If you provide your own key pair, it must use the OpenSSH format.

A public key has the following format:

```
key_type public_key optional_comment
```

For example, an RSA public key looks like this:

```
ssh-rsa AAAAB3BzaC1yc2EAAAADAQABAAABAQD9BRwrUiLDki6P0+jZhwsjS2muM
...
yXDus/5DQ== rsa-key-20201202
```

For images provided with the appliance, these SSH key types are supported: RSA, DSA, DSS, ECDSA, and ED25519.

If you bring your own image, you're responsible for managing the SSH key types that are supported.

For RSA, DSS, and DSA keys, a minimum of 2048 bits is recommended. For ECDSA keys, a minimum of 256 bits is recommended.

Prerequisites

- If you're using a UNIX system, you probably already have the `ssh-keygen` utility installed. To determine whether it's installed, type `ssh-keygen` on the command line. If it's not installed, you can download OpenSSH for UNIX from http://www.openssh.com/portable.html and install it.
- If you're using a Microsoft Windows operating system, you will need PuTTY and the PuTTY Key Generator. Download PuTTY and PuTTYgen from https://www.putty.org and install them.

Creating an SSH Key Pair on the Command Line

1. Open a shell or terminal for entering the commands.
2. At the prompt, enter `ssh-keygen` and provide a name for the key when prompted. Optionally, include a passphrase.
3. Do one of the following:
   - **On UNIX systems:**

     Use this command to set the file permissions so that only you can read the private key file:

     ```
     chmod 400 private_key_file
     ```

     private_key_file is the full path and name of the file that contains the private key associated with the instance you want to access.
   - **On a Microsoft Windows system using OpenSSH:**
     a. In Windows Explorer, navigate to the private key file, right-click the file, and then click Properties.
     b. On the Security tab, click Advanced.

c. Ensure that the Owner is your user account.

d. Click Disable Inheritance, and then select Convert inherited permissions into explicit permissions on this object.

e. Select each permission entry that is not your user account and click Remove.

f. Ensure that the access permission for your user account is Full control.

g. Save your changes.

Creating an SSH Key Pair Using PuTTY Key Generator

Perform this procedure on your Microsoft Windows system.

1. Open `puttygen.exe`.

   For example, navigate to `C:\Program Files\PuTTY` and double-click `puttygen.exe`.

   The PuTTY Key Generator window opens.

2. Specify a key size of 2048 bits.

   In the Parameters area at the bottom of the window, enter 2048 in the field for the Number of bits in a generated key.

3. Click the Generate button.

4. Move your mouse around the blank area in the PuTTY window to generate random data in the key.

   As you move your mouse, you should see the green progress bar advance.

   When the progress bar is full, the key is generated. Generating the key can take several seconds to several minutes.

   When the key generation is complete, the public key appears in the window under Public key for pasting into OpenSSH authorized_keys file.

5. Leave the Key passphrase field blank.

6. Click the Conversions menu at the top of the window, and then click Export OpenSSH key.

   When prompted to save this key without a passphrase, click Yes.

7. When prompted to save the private key, select a location and name of your choice.

8. Select all of the generated key that appears under Public key for pasting into OpenSSH authorized_keys file, copy it using Ctrl+C, paste it into a text file, and then save the file in the same location as the private key. (**Do not use Save public key because it does not save the key in the OpenSSH format.**)

   You can name the key anything you want, but for consistency, use the same name as the private key and a file extension of `.pub`. For example, `mykey.pub`.

9. Do one of the following:

   - **On a UNIX system:**

     Use the following command to set the file permissions so that only you can read the private key file:

     ```
     chmod 400 private_key_file
     ```

     private_key_file is the full path and name of the file that contains the private key associated with the instance you want to access.

   - **On a Microsoft Windows system:**

a. Navigate to the private key file, right-click on the file, and then click Properties.
b. On the Security tab, click Advanced.
c. Ensure that the Owner is your user account.
d. Click Disable Inheritance, and then select Convert inherited permissions into explicit permissions on this object.
e. Select each permission entry that is not your user account and click Remove.
f. Ensure that the access permission for your user account is Full control.
g. Save your changes.

10. Note the names and location of your public and private key files. You will need the public key when launching an instance. You will need the private key to access the instance via SSH.

Connecting to a Linux or Oracle Solaris Instance

You can connect to a running instance by using a Secure Shell (SSH) or Remote Desktop connection. Most UNIX systems include an SSH client by default.

> **Note:**
>
> If you created an instance without an SSH key, you can stop the instance, attach the boot volume to a new instance, and configure SSH on the new instance.

Connecting from a UNIX System

1. Open a terminal window or shell.
2. Use this command to connect to the instance:

```
ssh -i private_key_file username@public-ip-address
```

- ***private_key_file*** is the full path and name of the file that contains the private key associated with the instance you want to access.
- ***username*** is the default user name for the instance. See Prerequisites.
- ***public-ip-address*** is your instance IP address that you can get from the Compute Web UI. See Get the Instance IP Address.

Connecting from Microsoft Windows Using OpenSSH

1. Open Windows PowerShell.
2. Use this command to connect to the instance:

```
ssh -i private_key_file username@public-ip-address
```

- ***private_key_file*** is the full path and name of the file that contains the private key associated with the instance you want to access.
- ***username*** is the default user name for the instance. See Prerequisites.
- ***public-ip-address*** is your instance IP address that you can get from the Compute Web UI. See Get the Instance IP Address.

Connecting from Microsoft Windows Using PuTTY

Use the following procedure if the instance uses a key pair that you created using PuTTY Key Generator as described in Creating an SSH Key Pair Using PuTTY Key Generator.

1. Open PuTTY.
2. In the Category pane (on the left), select Session and enter the following:
   - **Host Name (or IP address):** ***username*@*public-ip-address***
     - ***username*** is the default user name for the instance. For instances launched from images provided with Private Cloud Appliance, the default user name is `opc`.
     - ***public-ip-address*** is your instance IP address.
   - **Port:** 22
   - **Connection type:** SSH
3. In the Category pane, expand Window, and then select Translation.
4. In the Remote character set drop-down list, select UTF-8. The default locale setting on Linux instances is UTF-8, and this setting configures PuTTY to use the same locale.
5. In the Category pane, expand Connection, expand SSH, and then click Auth.
6. Click Browse, and then select your `.ppk` private key file.
7. Click Open to start the session.

   If this is your first time connecting to the instance, you might see a message that the server's host key is not cached in the registry. Click Yes to continue the connection.

> **Tip:**
>
> If the connection fails, you might need to update your PuTTY proxy configuration.

Next Actions

- Add storage. See Block Volume Storage, Object Storage, and File System Storage.
- Install software on the instance.
- Configure and enable additional users to connect to your instance.

The utilities you use to perform the administrative tasks vary depending on the type of OS in the instance. For additional administrative information, refer to the documentation for the OS. These documentation libraries provide helpful information:

- Oracle Operating Systems Documentation
- Oracle Virtualization Documentation

Connecting to a Microsoft Windows Instance

You can connect to a Microsoft Windows instance using a Remote Desktop connection. Most Microsoft Windows systems include a Remote Desktop client by default.

Enabling Remote Desktop Protocol Access

To enable Remote Desktop Protocol (RDP) access to the Microsoft Windows instance, you need to add a stateful ingress security rule for TCP traffic on destination port 3389 from source 0.0.0.0/0 and any source port. You can implement this security rule in either a network security group that the Microsoft Windows instance belongs to, or a security list that is used by the instance's subnet.

Using the Compute Web UI

1. Click Dashboard, and click the Networking/View Virtual Cloud Networks button.
2. Select the compartment where your VCN is located.
3. Click the name of the VCN for which you want to enable RDP access.
4. Perform one of the following actions:
   - **Add an ingress security rule to an NSG:**

     NSG security rules provide a virtual firewall for cloud resources in the VCN.

     a. On the VCN details page, under Resources, click Network Security Groups.

     b. Click the name of the network security group for which you want to add a rule.

     c. On the network security group details page, under Resources, click Security Rules, and click the Create Security Rules button.

     d. In the Create New Network Security Group Rules dialog, click Allow Rules for Ingress, and enter the following values for the rule:

        – **Stateless:** Leave the check box empty to indicate stateful.

        – **Ingress Type:** CIDR

        – **Ingress CIDR:** 0.0.0.0/0

        – **IP Protocol:** TCP

        – **Source Port Range:** Leave empty to indicate All.

        – **Destination Port Range:** 3389

        – **Description:** An optional description of the rule.

     e. Click the Create button in the dialog.
   - **Add an ingress rule to a VCN security list:**

     Security list rules provide a virtual firewall for instances that use this VCN.

     a. On the VCN details page, under Resources, click Security Lists.

     b. Click the name of the security list for which you want to add a rule.

     c. On the security list details page, under Resources, click Ingress Rules, and click the Create Ingress Security Rule button.

     d. In the Create Security List Rule dialog, enter the following values for the rule:

        – **Stateless:** Leave the check box empty to indicate stateful.

        – **Ingress CIDR:** 0.0.0.0/0

        – **IP Protocol:** TCP

        – **Source Port Range:** Leave empty to indicate All.

- **Destination Port Range:** 3389
- **Description:** An optional description of the rule.

e. Click the Create Security List Rule button in the dialog.

Connecting with an RDP Client

1. Open the Remote Desktop client.
2. In the Computer field, enter the public IP address of the instance. You can retrieve the public IP address from the Compute Web UI. See Get the Instance IP Address.
3. The User name depends on how the image was configured. If you don't know the user name, consult with your administrator.

> **Note:**
>
> Depending on the Remote Desktop client you are using, you might have to connect to the instance before you can enter this credential.

4. Click Connect to start the session.
5. Accept the certificate if you are prompted to do so.
6. If you are connecting to the instance for the first time, enter the initial password that was provided to you by your administrator when you launched the instance. You will be prompted to change the password as soon as you log in. Your new password must be at least 12 characters long and must comply with the Microsoft password policy.

   Otherwise, enter the password that you created. If you are using a custom image, you might need to know the password for the instance that the image was created from.
7. Press Enter.

Next Actions

- Add storage. See Block Volume Storage, and Object Storage File System Storage.
- Install software on the instance.
- Configure and enable additional users to connect to your instance.

The utilities you use to perform the administrative tasks vary depending on the type of OS in the instance. For additional administrative information, refer to the documentation for the OS.

Remotely Troubleshooting an Instance by Using a Console Connection

> **Important:**
>
> Instance console connections are for troubleshooting purposes only. To connect to a running instance for administration and general use, use a Secure Shell (SSH) or Remote Desktop connection as described in Connecting to a Linux or Oracle Solaris Instance and Connecting to a Microsoft Windows Instance.

The following are example situations when you need to remotely troubleshoot an instance:

- An imported or customized image does not complete a successful boot
- A previously working instance stops responding

The following is the process to remotely connect to an instance by using a console connection:

1. Create an instance console connection.
2. Set up a secure tunnel.
   - To connect to the instance VNC console, set up a secure tunnel to the VNC server on the instance.
   - To connect to the instance serial console, set up a secure tunnel to the serial console device on the instance.
3. Complete the connection between the local system and the instance.
   - To connect to the instance VNC console, open a VNC viewer such as RealVNC Viewer on the local system.
   - To connect to the instance serial console, open an SSH connection on the local system to the serial console device on the instance. Only one serial connection can be made at one time.

To disconnect from the instance console, close the SSH connection that you initiated on the local host.

Console Connection Prerequisites

Ensure that you have the following resources on the system that you plan to use to connect to the instance console.

- SSH key pair

  If you do not already have an SSH key pair, you can use the `ssh-keygen` utility on UNIX systems or PuTTY `puttygen.exe` on Windows systems. Specify a key size of 2048 bits (this value should be the default). Give the key a name. You do not need to provide a passphrase; using a passphrase makes it more difficult to automate connecting. See also Managing Key Pairs.

- Command-line shell and SSH client

  On Windows systems, use one of the following:

  – Windows PowerShell

    If you use PowerShell to connect to the VNC server on the instance, `plink.exe` is required. `plink.exe` is the command link connection tool included with PuTTY. You can install PuTTY or install `plink.exe` separately. For installation information, see https://www.putty.org.

  – Git for Windows

    Git for Windows includes OpenSSH.

  – Windows Subsystem for Linux (WSL)

    WSL includes OpenSSH.

- VNC viewer to connect to the VNC console

Complete the following configuration on the Private Cloud Appliance where the instance that you need to connect to remotely resides.

- Ensure that you belong to a group that has the following permissions. See Managing Policies.

```
Allow group group_name to manage instance-console-connection in tenancy
Allow group group_name to read instance in tenancy
```

- Create an instance console connection. You will need your SSH key pair. See Creating an Instance Console Connection.

Creating an Instance Console Connection

Before you can connect to an instance VNC console or serial console you need to create an instance console connection.

> **Note:**
>
> Instance console connections are limited to one client at a time. If the client attempts to connect but fails to connect within five minutes, the connection is closed and a different client can connect. During the five-minute timeout, any attempt to connect a different client fails.

The instance console connection resource provides the command that you need to create the secure tunnel. The command is a little different depending on whether your local system is UNIX or Windows and whether you want to connect to the VNC console or the serial console on the instance.

One component that all of these commands have in common is ***proxy_host***. The ***proxy_host*** is the IP address of the master management node, which must be running the VM Console Service. The proxy host must be accessible on the public network at ***proxy_host***:443.

Using the Compute Web UI

1. On the Dashboard click the Compute/View Instances button.
2. If the instance where you want to create a console connection is not listed, use the Compartment drop-down menu above the instances list to select the correct compartment.
3. Click the name of the instance where you want to create a console connection.
4. On the instance details page, scroll to the Resources section and click Console Connection.
5. If a console connection does not already exist, click the Create Console Connection button.
6. Provide the public key portion of your SSH key.

   In the Create Console Connection dialog, do one of the following to enter your public SSH key:

   - Select the key file(s).
     - Click inside the Drag and Drop box to open a file browser and select the file.
     - Drag the file from your file browser listing and drop the file on the Drag and Drop box.
   - Paste the public key(s). Copy your public SSH key text, and paste the text into the field.
7. Click the Create Console Connection button in the dialog.

When the console connection has been created and is available, the state changes to Active.

Using the OCI CLI

1. Get the following information:
   - The OCID of the instance where you want to create the console connection: `oci compute instance list`
   - Your SSH public key file.
2. Determine whether a console connection already exists for this instance.

```
$ oci compute instance-console-connection list -c ocid1.compartment.unique_ID \
--instance-id ocid1.instance.unique_ID
```

3. Run the create console connection command.

```
$ oci compute instance-console-connection create --instance-id
ocid1.instance.unique_ID \
--ssh-public-key-file public_SSH_key_path
{
  "data": {
    "compartment-id": "ocid1.compartment.unique_ID",
    "connection-string": "ssh -i private_SSH_key_path -t -p 443 user_name@proxy_host
tty@instance_OCID",
    "defined-tags": {},
    "fingerprint": "SHA256:unique_ID",
    "freeform-tags": {},
    "id": "ocid1.instanceconnectionconsole.unique_ID",
    "instance-id": "ocid1.instance.unique_ID",
    "lifecycle-state": "ACTIVE",
    "service-host-key-fingerprint": null,
    "vnc-connection-string": "ssh -i public_SSH_key_path -p 443 -L
local_vnc_port:localhost:remote_vnc_port user_name@proxy_host
vnc@ocid1.instance.unique_ID"
  },
  "etag": "afc7eb68-5f1a-40cc-8dc3-8a1cae237230"
}
```

   The value of `connection-string` is the SSH connection string for the instance serial console connection. The value of `vnc-connection-string` is the SSH connection string for the instance VNC console connection.

   See the beginning of this topic for a description of ***proxy_host***.

4. When you are finished using this instance console connection, use the following command to delete the connection.

```
$ oci compute instance-console-connection delete \
--instance-console-connection-id ocid1.instanceconnectionconsole.unique_ID
```

What's Next

Continue to Connecting to the Instance VNC Console or Connecting to the Instance Serial Console.

Connecting to the Instance VNC Console

After you create the instance console connection, set up a secure tunnel to the VNC server on the instance, and connect using a VNC client.

Set Up a Secure Tunnel to the VNC Server

Use one of the following procedures to set up a secure tunnel to the VNC server on the instance:

- Connecting to the VNC Console from a Linux or macOS System

  In addition to native Linux or macOS systems, use this procedure if you are using Git for Windows or Windows Subsystem for Linux.
- Connecting to the VNC Console from a Microsoft Windows System

  Use this procedure if you are using PowerShell and `.ppk` keys.

The VNC console connection uses SSH port forwarding to create a secure connection from your local system to the VNC server attached to your instance's console.

> **Caution:**
>
> Although SSH port forwarding is a secure way to use VNC over the internet, opening a port on a multiuser system makes that port available to all users on that system until a VNC client connects. For this reason, Oracle does not recommend using this method on a multiuser system unless you secure the port or you isolate the VNC client by running it in a virtual environment.

Open a VNC Client

After the secure tunnel is established, open a VNC client on your local system. For example, execute a command such as the following on the local host, or open a VNC client on the local host in some other way.

```
$ vncviewer localhost:local_vnc_port
```

Specify `localhost` as the host to connect to, and set the port to the ***`local_vnc_port`*** port that is listed in the VNC connection string from Creating an Instance Console Connection. The procedures in Connecting to the VNC Console from a Linux or macOS System and Connecting to the VNC Console from a Microsoft Windows System describe how to retrieve this connection string. The default value for ***`local_vnc_port`*** is port 5900.

> **Note:**
>
> Remote management for Remote Desktop on macOS uses port 5900. Because VNC console connections in Private Cloud Appliance also use port 5900, VNC console connections are not compatible with remote management. To use VNC console connections on macOS, disable remote management.

> **Note:**
>
> The macOS built-in VNC client, Screen Sharing.app does not work with VNC console connections in Private Cloud Appliance. Use a different VNC client, such as RealVNC Viewer.

> **Note:**
>
> When you connect, you might see a warning from the VNC client that the connection is not encrypted. Because you are connecting through SSH, the connection is secure, so this warning is not an issue.

Connecting to the VNC Console from a Linux or macOS System

This procedure sets up a secure tunnel to the VNC server on the instance using OpenSSH on a Linux or macOS system. macOS and most Linux and other UNIX operating systems include the SSH client OpenSSH by default.

In addition to native Linux or macOS systems, use this procedure if you are using Git for Windows or Windows Subsystem for Linux.

Use either the Compute Web UI procedure or the OCI CLI procedure to get the VNC connection string from the instance console connection. Then open the VNC client on your local system.

The VNC connection string has the following format:

```
ssh -i private_key_path -p 443 -L local_vnc_port:localhost:remote_vnc_port
user_name@proxy_host vnc@instance_OCID
```

See Creating an Instance Console Connection for a description of ***proxy_host***.

Using the Compute Web UI

1. On the Dashboard, click the Compute/View Instances button.
2. If the instance that you want to remotely connect to is not listed, use the Compartment drop-down menu above the instances list to select the correct compartment.
3. Click the name of the instance that you want to remotely connect to.
4. On the instance details page, scroll to the Resources section, and click Console Connection.
5. For the active console connection, click the Actions menu, and then click Copy VNC Connection for Linux/Mac.
6. Verify the value of the `-i` parameter in the copied connection string.

   The `-i` parameter in the connection string specifies the location of the path to the private key to be used for authentication.

   If necessary, replace the value of the `-i` parameter with the correct path to your private key file.
7. Paste the VNC connection string from the preceding step into a terminal window, and then press Enter to set up the secure tunnel.
8. Go to Open a VNC Client.

Using the OCI CLI

1. Get the OCID of the console connection for the instance that you want to remotely connect to:

   ```
   $ oci compute instance-console-connection list -c ocid1.compartment.unique_ID \
   --instance-id ocid1.instance.unique_ID
   ```

2. Get the VNC connection string for Linux/Mac.

   ```
   $ oci compute instance-console-connection get \
   --instance-console-connection-id ocid1.instanceconnectionconsole.unique_ID
   ```

   Copy the value of the `vnc-connection-string` property.

3. Verify the value of the `-i` parameter in the copied connection string.

   The `-i` parameter in the connection string specifies the location of the path to the private key to be used for authentication.

   If necessary, replace the value of the `-i` parameter with the correct path to your private key file.

4. Paste the VNC connection string into a terminal window, and then press Enter to set up the secure tunnel.

5. Go to Open a VNC Client.

Connecting to the VNC Console from a Microsoft Windows System

This procedure sets up a secure tunnel to the VNC server on the instance using PowerShell and `.ppk` keys on a Microsoft Windows system.

Use either the Compute Web UI procedure or the OCI CLI procedure to get the VNC connection string from the instance console connection. Then open the VNC client on your Windows system.

The VNC connection string has the following format. If you use Windows PowerShell, you must use `plink.exe`.

```
plink.exe -ssh -i ppk_private_key_path -P 443 -L
local_vnc_port:localhost:remote_vnc_port user_name@proxy_host vnc@instance_OCID
```

See Creating an Instance Console Connection for a description of ***proxy_host***.

Using the Compute Web UI

1. On the Dashboard, click the Compute/View Instances button.
2. If the instance that you want to remotely connect to is not listed, use the Compartment drop-down menu above the instances list to select the correct compartment.
3. Click the name of the instance that you want to remotely connect to.
4. On the instance details page, scroll to the Resources section, and click Console Connection.
5. For the active console connection, click the Actions menu, and then click Copy VNC Connection for Windows.
6. Verify the value of the `-i` parameter in the copied connection string.

   The `-i` parameter in the connection string specifies the location of the path to the `.ppk` private key to be used for authentication. The default value for this parameter references an environment variable that might not be configured on your Windows client, or it might not represent the location where the private key file is saved.

   Replace the value of the `-i` parameter with the actual path to your `.ppk` private key file.

7. Paste the modified VNC connection string into a Windows PowerShell terminal window, and then press Enter to set up the secure connection.
8. Go to Open a VNC Client.

Using the OCI CLI

1. Get the following information:
   - The OCID of the console connection for the instance that you want to remotely connect to:

     ```
     $ oci compute instance-console-connection list -c ocid1.compartment.unique_ID \
     --instance-id ocid1.instance.unique_ID
     ```

   - The path to your `.ppk` private key file.
2. Get the VNC connection string for Windows.

   Run the get `plink` connection string command:

   ```
   $ oci compute instance-console-connection get-plink-connection-string \
   --instance-console-connection-id ocid1.instanceconnectionconsole.unique_ID \
   --private-key-file private key file
   ```

   The value of the `--private-key-file` option is the path to the `.ppk` private key to be used for authentication. This value is inserted into the generated connection string as the value of the `-i` option.

   Copy this output `plink.exe` VNC connection string.
3. Paste the `plink.exe` VNC connection string into a Windows PowerShell terminal window, and then press Enter to set up the secure tunnel.
4. Go to Open a VNC Client.

Connecting to the Instance Serial Console

After you create the instance console connection, set up a secure tunnel to the instance serial console. Use an SSH client to connect to the serial console. You can use the same SSH key for the serial console that was used when you launched the instance, or you can use a different SSH key.

Only one serial connection can be made to any given compute instance at one time. Therefore, only one user can connect to the serial console of any given compute instance at one time. A different user at another time can use the same SSH key for the serial console or a different SSH key.

When you are finished with the serial console, terminate the SSH connection and delete the serial console connection.

Set Up a Secure Tunnel to the Serial Console

Use one of the following procedures to set up a secure tunnel to the serial console on the instance:

- Connecting to the Serial Console from a Linux or macOS System

  In addition to native Linux or macOS systems, use this procedure if you are using Git for Windows or Windows Subsystem for Linux.
- Connecting to the Serial Console from a Microsoft Windows System

  Use this procedure if you are using PowerShell and `.ppk` keys.

Validating Server Host Keys

After the secure tunnel is established, connect to the instance serial console by using an SSH client.

When you first connect to the serial console, you are prompted to validate the fingerprint of the server host key. The fingerprint of the server host key is the SHA256 hash of the server host's public SSH key. The server SSH handshake response is signed with the associated private key. Validating the server host key's fingerprint protects against potential attacks.

When you make a manual connection to the serial console, the fingerprint of the server host key is not automatically validated. To manually validate the fingerprint, compare the value of the fingerprint that appears in the terminal when you connect to the fingerprint value that is displayed in the Compute Web UI or in the output from the instance console connection `get` command.

- Compute Web UI. Go to the instance details page, scroll to the Resources section, and click Console Connection. The table displays the fingerprint of the server host key.
- OCI CLI. Check the value of the `fingerprint` property in the output from the following command:

```
$ oci compute instance-console-connection get \
--instance-console-connection-id ocid1.instanceconnectionconsole.unique_ID
```

Connecting to the Serial Console from a Linux or macOS System

This procedure sets up a secure SSH-based communication channel from the local Linux or macOS host to the remote serial console device associated with the instance. macOS and most Linux and other UNIX operating systems include the SSH client OpenSSH by default.

> **Note:**
>
> The minimum version required for OpenSSH to connect to the serial console from Linux and macOS is OpenSSH 7.2.

In addition to native Linux or macOS systems, use this procedure if you are using Git for Windows or Windows Subsystem for Linux.

Use either the Compute Web UI procedure or the OCI CLI procedure to get the serial connection string from the instance console connection.

The serial connection string has the following format:

```
ssh -i private_key_path -t -p 443 user_name@proxy_host tty@instance_OCID
```

See Creating an Instance Console Connection for a description of ***proxy_host***.

Using the Compute Web UI

1. On the Dashboard, click the Compute/View Instances button.
2. If the instance that you want to remotely connect to is not listed, use the Compartment drop-down menu above the instances list to select the correct compartment.
3. Click the name of the instance that you want to remotely connect to.

4. On the instance details page, scroll to the Resources section, and click Console Connection.
5. For the active console connection, click the Actions menu, and then click Copy Serial Console Connection for Linux/Mac.
6. Verify the value of the `-i` parameter in the copied connection string.

   The `-i` parameter in the connection string specifies the location of the path to the private key to be used for authentication.

   If necessary, replace the value of the `-i` parameter with the correct path to your private key file.
7. Paste the serial connection string into a terminal window, and then press Enter to set up the secure tunnel to the serial console device on the instance.
8. If prompted, validate and accept the fingerprint of the server host key. See Validating Server Host Keys.
9. Press Enter again to activate the console.
10. On the Private Cloud Appliance, Reset or Soft Reset the instance. See Stopping, Starting, and Resetting an Instance.

    If the instance is functional and the connection is active, the serial output appears in your console. If serial output does not appear in the console, the instance operating system is not booting.

Using the OCI CLI

1. Get the OCID of the console connection for the instance that you want to remotely connect to:

   ```
   $ oci compute instance-console-connection list -c ocid1.compartment.unique_ID \
   --instance-id ocid1.instance.unique_ID
   ```
2. Get the serial connection string for Linux/Mac.

   ```
   $ oci compute instance-console-connection get \
   --instance-console-connection-id ocid1.instanceconnectionconsole.unique_ID
   ```

   Copy the value of the `console-connection-string` property.
3. Verify the value of the `-i` parameter in the copied connection string.

   The `-i` parameter in the connection string specifies the location of the path to the private key to be used for authentication.

   If necessary, replace the value of the `-i` parameter with the correct path to your private key file.
4. Paste the serial connection string into a terminal window, and then press Enter to set up the secure tunnel to the serial console device on the instance.
5. If prompted, validate and accept the fingerprint of the server host key. See Validating Server Host Keys.
6. Press Enter again to activate the console.
7. On the Private Cloud Appliance, `RESET` or `SOFTRESET` the instance. See Stopping, Starting, and Resetting an Instance.

   If the instance is functional and the connection is active, the serial output appears in your console. If serial output does not appear in the console, the instance operating system is not booting.

Connecting to the Serial Console from a Microsoft Windows System

This procedure sets up a secure SSH-based communication channel from the local Microsoft Windows host to the remote serial console device associated with the instance. Use PowerShell and `.ppk` keys on the local Windows system. Use an SSH client such as OpenSSH to connect to the serial console.

> **Note:**
>
> The minimum version required for OpenSSH to connect to the serial console from Windows is PuTTY (0.75).

Use either the Compute Web UI procedure or the OCI CLI procedure to get the serial connection string from the instance console connection.

The serial connection string has the following format. If you use Windows PowerShell, you must use `plink.exe`.

```
plink.exe -ssh -i ppk_private_key_path -t -P 443 user_name@proxy_host tty@instance_OCID
```

See Creating an Instance Console Connection for a description of ***proxy_host***.

Using the Compute Web UI

1. On the Dashboard, click the Compute/View Instances button.
2. If the instance that you want to remotely connect to is not listed, use the Compartment drop-down menu above the instances list to select the correct compartment.
3. Click the name of the instance that you want to remotely connect to.
4. On the instance details page, scroll to Resources, and click Console Connection.
5. For the active console connection, click the Actions menu, and then click Copy Serial Console Connection for Windows.
6. Verify the value of the `-i` parameter in the copied connection string.

   The `-i` parameter in the connection string specifies the location of the path to the `.ppk` private key to be used for authentication. The default value for this parameter references an environment variable that might not be configured on your Windows client, or it might not represent the location where the private key file is saved.

   Replace the value of the `-i` parameter with the actual path to your `.ppk` private key file.
7. Paste the modified serial connection string into a Windows PowerShell terminal window, and then press Enter to set up the secure tunnel to the serial console device on the instance.
8. If prompted, validate and accept the fingerprint of the server host key. See Validating Server Host Keys.
9. Press Enter again to activate the console.
10. On the Private Cloud Appliance, Reset or Soft Reset the instance. See Stopping, Starting, and Resetting an Instance.

If the instance is functional and the connection is active, the serial output appears in your console. If serial output does not appear in the console, the instance operating system is not booting.

Using the OCI CLI

1. Get the following information:
   - The OCID of the console connection for the instance that you want to remotely connect to:

     ```
     $ oci compute instance-console-connection list -c ocid1.compartment.unique_ID \
     --instance-id ocid1.instance.unique_ID
     ```

   - The path to your `.ppk` private key file.
2. Get the serial connection string for Windows.

   Run the get `plink` connection string command:

   ```
   $ oci compute instance-console-connection get-plink-connection-string \
   --instance-console-connection-id ocid1.instanceconnectionconsole.unique_ID \
   --private-key-file private key file
   ```

   The value of the `--private-key-file` option is the path to the `.ppk` private key to be used for authentication. This value is inserted into the generated connection string as the value of the `-i` option.

   Copy this output `plink.exe` serial connection string.
3. Paste the `plink.exe` serial connection string into a Windows PowerShell terminal window, and then press Enter to set up the secure tunnel to the serial console device on the instance.
4. If prompted, validate and accept the fingerprint of the server host key. See Validating Server Host Keys.
5. Press Enter again to activate the console.
6. On the Private Cloud Appliance, `RESET` or `SOFTRESET` the instance. See Stopping, Starting, and Resetting an Instance.

   If the instance is functional and the connection is active, the serial output appears in your console. If serial output does not appear in the console, the instance operating system is not booting.

Backing Up and Restoring an Instance

Oracle Private Cloud Appliance supports backing up and restoring instances. The instance backup is created in an Object Storage bucket. From there, you copy it to another server in your data center for safekeeping. When needed, you can import the backup into any Private Cloud Appliance 3.x Object Storage bucket, and use it to create instances.

The backup and restore process involves using a series of Compute Enclave API commands as described in the following sections.

For conceptual information about backing up and restoring instances, refer to "Instance Backup and Restore" in the Compute Instance Concepts chapter in the *Oracle Private Cloud Appliance Concepts Guide*.

Task Map - Backing Up an Instance

No.	Task	Links
1.	Ensure that you have an Object Storage bucket in the same tenancy where the instance is located.	• Listing Buckets • Creating a Bucket
2.	Create the instance backup.	Creating an Instance Backup
3.	Transfer the backup object from Object Storage to another system in your data center.	Transferring an Instance Backup to Another System

Task Map – Restoring an Instance From a Backup

The following tasks assume that you are restoring an instance from a backup that is on another system in your data center. If the backup is already on the Private Cloud Appliance where you plan to restore the instance, start with task number 3.

No.	Task	Links
1.	Ensure that you have an Object Storage bucket in the same tenancy where you plan to restore the instance.	• Listing Buckets • Creating a Bucket
2.	Upload the backup to the bucket.	Transferring an Instance Backup From Another System to Private Cloud Appliance
3.	Identify the backup OCID.	Listing Instance Backups
4.	Import the backup from the bucket into the appliance.	Importing an Instance Backup
5.	Finish restoring the instance by creating an instance using the imported instance backup.	Finishing the Instance Restore

Creating an Instance Backup

This section describes how to back up an instance to an Object Storage bucket.

The instance can be running or stopped. The boot volume and any block volumes must be attached.

> **Caution:**
>
> During the backup process, do not perform any volume attach or detach operations on the instance.

The duration of the backup varies based on the amount of data on the instance boot and block volumes. A small instance that only has a 50 GB boot volume takes only a few minutes to complete. If the instance volumes are as large as 32 TB, the backup can take up to 6 hours to complete.

Prerequisites

- You must have an Object Storage bucket in the tenancy where the instance is located. See Creating a Bucket.
- Quiesce instance activities such as running applications so that the backup is created at a known state.

Using the Compute Web UI

1. In the navigation menu, under Compute, click Instances.
2. If needed, select the compartment where your instance is located.
3. Click the name of the instance you plan to back up.
4. Click Controls and select Export.
5. In the dialog box, select these items:
   - If needed, change the compartment to the compartment where the bucket with the backup is located by clicking (Change).
   - Select the Bucket.
6. Click Create Export.
7. To see the progress, under Resources, click Work Requests.
8. To see the status of an instance backup, under Resources, click Instance Exports.

To transfer the backup to another server or to another appliance, see Transferring an Instance Backup to Another System.

Using the Compute API

`export` API – Creates an instance backup to an Object Storage bucket in the specified compartment.

API Endpoint

`https://<mgmt_node_VIP>:30003/20160918/instances/<instance_OCID>/actions/export`

where:

- ***<mgmt_node_VIP>*** is the management node VIP host name or IP address.
- ***<instance_OCID>*** is the instance ID.

Pass these key-value pairs:

```
{
    "bucketName": "<bucket_name>",
    "destinationType": "objectStorageTuple",
    "namespaceName": "<namespace_name>",
    "compartmentId": "<bucket_compartment_OCID>"
}
```

You can verify that the instance backup completed by using commands described in Listing Instance Backups.

Listing Instance Backups

The procedures in this section show you how to list backups for a given instance.

When you list instance backups, you are able to identify these components of the backup:

- Backup OCID
- Boot volume OCID
- Instance OCID
- Image OCID

Using the Compute Web UI

1. In the navigation menu, under Compute, click any of the following links.
   - Instance Exports: Displays the list of instance backups.
   - Instance Imports: Displays the list of imported instance backups that can be used to create new instances.

Using the Compute API

There are two API endpoints for viewing instance backups:

- **List All Instance Backups in a Bucket.**

  API endpoint

  `https://<mgmt_node_VIP>:30003/20160918/instances/instanceBackups?compartmentId=<bucket_compartment_OCID>`

  where:

  - ***<mgmt_node_VIP>*** is the management node VIP host name or IP address.
  - ***<bucket_compartment_OCID>*** is the compartment ID where the Object Storage bucket is located.

- **Get Instance Backup Details**

  API endpoint

  `https://<mgmt_node_VIP>:30003/20160918/instanceBackups/<instance_backup_id>`

  where:

  - ***<mgmt_node_VIP>*** is the management node VIP host name or IP address.
  - ***<instance_backup_id>*** is the backup ID, which you can get from the backup export output or from listing backups with the API.

Transferring an Instance Backup

The procedures in this section describe how to transfer an instance backup to another system and how to transfer an instance backup back to any Private Cloud Appliance 3.x.

Transferring an Instance Backup to Another System

You can use this procedure to transfer the instance backup to another system in your data center for safekeeping.

Instance backups are large files. Ensure that you have enough space on your system to store the backup. You can use the `oci os object list` command to display the size of the instance backup. See Viewing Objects in a Bucket.

Using the OCI CLI

1. Gather the information that you need to run the command.
   - Namespace name (see Obtaining the Object Storage Namespace)
   - Bucket name (`oci os bucket list`), see Listing Buckets
   - Object name (`oci os object list`), see Viewing Objects in a Bucket
2. Run the object get command.

   Syntax (entered on a single line):

   ```
   oci os object get
   --namespace-name object_storage_namespace
   --bucket-name bucket_name
   --name object_name
   --file file_location
   ```

   file_location is the destination of the file being downloaded.

   Example:

   ```
   oci os object get --namespace-name mytenant \
   --bucket-name my-backup-bucket --name ocid1.exportinstance.uniqueID \
   --file /home/myusername/backups/ocid1.exportinstance.uniqueID
   Downloading object  [########################------------]   68%  00:00:05
   ```

Transferring an Instance Backup From Another System to Private Cloud Appliance

Use this procedure to transfer an instance backup from another system in your data center to an Object Storage bucket in Private Cloud Appliance.

Using the OCI CLI

1. Gather the information that you need to run the command.
   - Namespace name. See Obtaining the Object Storage Namespace.
   - Bucket name (`oci os bucket list`). See Listing Buckets.
2. Upload the instance backup to an Object Storage bucket in the target appliance.

   Use the `oci os object put` command.

   Syntax (entered on a single line):

   ```
   oci os object put  \
   --namespace-name <namespace_name>  \
   --bucket-name <bucket_name>  \
   --file <instance_backup_pathname>
   ```

   The value of ***<instance_backup_pathname>*** is the path name of the object being uploaded, such as `C:\workspace\backups\ocid1.instance.uniqueID` or `/home/downloads/backups/ocid1.instance.uniqueID`.

   Example:

   ```
   oci os object put  \
   --namespace-name mytenant  \
   --bucket-name target-bucket  \
   --file ./ocid1.instance.uniqueID

   Upload ID: f000bf64-9a96-4008-b1cc-f6b2595b04b1
   ```

```
Split file into 35 parts for upload.
Uploading object  [####################################]  100%
{
  "etag": "ef7bdd67a72536e29da97f3414f4118e",
  "last-modified": "2022-07-06T17:50:00",
  "opc-multipart-md5": "htOEPyjWDFA4Bs2urJJPRQ==-35"
```

Restoring an Instance from an Instance Backup

You restore an instance by importing the instance backup from an Object Storage bucket. Next, create the instance using the boot volume from the backup as the image source. Then attach any block volumes that were included in the backup.

Importing an Instance Backup

Importing an instance backup copies the backup from an Object Storage backup to a location that is internal to the appliance.

For a given instance backup, you can only have one imported copy. If you need to import the same instance backup again, you must first delete the original instance backup. See Deleting an Instance Backup.

Prerequisites

- You must have an instance backup in an Object Storage bucket.

  If needed, transfer the backup to Private Cloud Appliance. See Transferring an Instance Backup From Another System to Private Cloud Appliance.

Using the Compute Web UI

1. Identify the OCID of the backup you plan to use.

   See Listing Instance Backups.
2. In the navigation menu, under Compute, click Instances.
3. Click Import.
4. In the dialog box, select these items:
   - If needed, change the compartment to the compartment where the bucket with the backup is located by clicking (Change).
   - Select the Bucket.
   - Select the Backup OCID.
5. Click Create Import.
6. When the import is finished, perform the steps in Finishing the Instance Restore.

Using the Compute API

`import` API – Imports the instance backup from an Object Storage bucket so that instances can be created from the backup.

API endpoint

`https://`***`<mgmt_node_VIP>`***`:30003/20160918/instanceBackups`***`<instance_backup_OCID>`***`/actions/import`

where:

- ***<mgmt_node_VIP>*** is the management node VIP host name or IP address.
- ***<instance_backup_OCID>*** is the instance backup ID.

Pass these key-value pairs:

```
{
    "bucketName": "<bucket_name>",
    "destinationType": "objectStorageTuple",
    "namespaceName": "<namespace_name>",
    "compartmentId": "<bucket_compartment_OCID>"
}
```

When the import is complete, perform the steps in Finishing the Instance Restore.

Finishing the Instance Restore

Perform these steps to create as many instances as you like from the instance backup.

1. Create an instance by following the instructions in Creating an Instance). While doing so, perform these actions:
   - For the source image, specify a backup boot volume instead of an image. Then select an imported instance.
   - If the instance requires access using SSH, ensure that you include an SSH public key.
2. (Optional) Attach any block volumes that were included in the instance backup.

   See Attaching a Volume.

Deleting an Instance Backup

You can use the `DELETE` API to delete exported and imported instance backups. The API sets the exported backup lifecycle state to TERMINATED and performs one of these actions based on the type of backup:

- **Imported instance backups:** Deletes the instance backup.
- **Exported instance backups:** Deletes the instance backup (source for importing) from object storage bucket

Terminated instance backups are eventually cleaned up by a background task.

Using the Compute API

`DELETE` API – Deletes instance backups

API endpoint

`https://<mgmt_node_VIP>:30003/20160918/instancesBackups<instance_backup_OCID>`

where:

- ***<mgmt_node_VIP>*** is the management node VIP host name or IP address.
- ***<instance_backup_OCID>*** is the instance backup ID.

8
Working with Instance Pools

Instance pools simplify the management of compute instances. An instance pool defines a set of compute instances that is managed as a group. Managing instances as a group enables you to efficiently provision instances and manage the state of instances.

In addition to provisioning or removing instances or stopping or starting instances by manually updating the instance pool, you can configure a pool to be scaled automatically according to a schedule. See Using Schedule-Based Autoscaling.

Creating an Instance Pool

An instance pool is a group of compute instances within the same region.

Performing operations such as reset or terminate on the pool object performs that operation on all instances that are members of the pool. Performing these operations on an individual instance that is a member of the pool does not affect any other member instances.

Creating an instance pool requires an instance configuration and a placement configuration. Instances that are added to the pool in a pool update can be created with different instance and placement configurations.

For instances in a pool, the value of the `displayName` property in the instance configuration is ignored. Instances in a pool are named `inst-`***`aaaaa-pool_name`***, where ***`aaaaa`*** is five random alphanumeric characters.

Placement Configuration

In addition to an instance configuration, pool creation requires a placement configuration. Values specified in a placement configuration override values specified in the instance configuration.

A placement configuration can specify fault domains, primary subnet, and secondary VNIC subnets.

Fault Domains

If you do not specify a fault domain in either the instance configuration or the placement configuration, the system automatically selects the best fault domains for the pool instances. If you specify only a single fault domain, all instances will be placed in only that fault domain. If you specify more than one fault domain, pool instances are placed in those fault domains evenly, providing better High Availability for the pool. If one fault domain cannot accommodate additional instances, instance creation stops. The system will not place more instances in one fault domain than in another fault domain.

If some instances cannot launch because of resource constraints, those instances remain in the Provisioning state and the pool remains in the Scaling state. Once `size` instances are launched, the pool can transition to the Running state. While the pool is in the Scaling state, pool instances that are in the Running state are available to use.

The following are examples of actions you can take if a pool instance fails to launch because of resource constraints:

- Update the pool and reduce the "Number of instances" or `size` value.
- Update the pool and change the Fault Domains specification in the Compute Web UI or in a new instance or placement configuration.
- Update the pool to specify a new instance configuration that creates instances that require fewer resources.
- Stop an instance that is not a member of a pool in the same fault domain where the pool instance is failing to launch because of resource constraints.
- Terminate an instance that is not a member of a pool in the same fault domain where the pool instance is failing to launch because of resource constraints.

Using the Compute Web UI

1. In the navigation menu, click Compute, and then click Instance Configurations.
2. If the instance configuration that you want to use to create this pool is not listed, use the Compartment drop-down menu above the instance configurations list to select the correct compartment.
3. Click the instance configuration that you want to use for the instances in this pool.
4. In the Resources box on the instance configuration details page, click Attached Instance Pools. Use the Compartment drop-down menu above the instance pools list to list pools in other compartments.

   Click the Create Instance Pool button.
5. In the Attach Instance Pool to ***`instance_configuration_name`*** dialog, enter the following information:
   - *Name*: Enter a name for the instance pool. The name does not need to be unique. This name is used in the names of the created instances. If you do not provide a name for the pool, the default name of the instance pool is `instancepool`***`YYYYMMDDhhmmss`***, where ***`YYYYMMDDhhmmss`*** is the creation date and time.
   - *Create in Compartment*: Select a compartment for this instance pool definition. Note that the instances in the pool will be created in the compartment that is specified in the instance configuration.
   - *Number of instances*: Specify the number of instances to create in this instance pool.
   - *Pool Placement*: Select the Fault Domains, VCN, and Subnet for instances in this instance pool. You can select a different compartment from which to choose the VCN and Subnet. See the descriptions of Placement Configuration and Fault Domains at the beginning of this section.
   - *Load Balancers*: Click the Attach Load Balancers box to specify load balancing for this pool. For information about load balancing, see Load Balancer as a Service. Provide the following information:
     - Select the load balancer to attach to this pool.
     - Select the backend set to which to add these pool instances.
     - Enter the port number on the instances to which the load balancer must direct traffic.
     - Select the VNIC to use when adding the instance to the backend set. The private IP address is used.

     To attach another load balancer, click the Add Load Balancer button. To attach a load balancer after the instance pool is created, see Managing Instance Pool Load Balancer Attachments.

- *Tagging*: (Optional) Add defined or free-form tags for this instance pool as described in Adding Tags at Resource Creation. Tags can also be applied later.

  These tags are applied to the pool definition, not to the member instances.

6. Click the Create Instance Pool button in the dialog.

   The details page of the new pool is displayed. The requested instances are listed in the Attached Instances table in the Resources section as they are created. The new instances are named `inst-aaaaa-pool_name`, where ***aaaaa*** is five random alphanumeric characters. If you change the name of the pool and then add new instances to the pool, the new instances will have the new name.

   Click Work Request(s) in the Resources box to check the status of the instance pool create.

Using the OCI CLI

1. Get the following information:

   - The OCID of the compartment where you want to create the instance pool definition: `oci iam compartment list`

     Note that the instances in the pool will be created in the compartment that is specified in the instance configuration.

   - The OCID of the instance configuration that you want to use: `oci compute-management instance-configuration list`
   - The size of the instance pool. This is the number of compute instances in the instance pool.
   - If you want load balancing for this pool, get the following information:
     - OCID of the load balancer to attach to this pool and name of the backend set to which to add these pool instances: `oci lb load-balancer list`
     - Port value to use when creating the backend set.
     - VNIC to associate with the load balancer. The value can be `PrimaryVnic` or the display name of one of the secondary VNICs on the instance configuration that is associated with the instance pool.

2. Construct an argument for the `--placement-configurations` option.

   See the descriptions of Placement Configuration and Fault Domains at the beginning of this section.

   Use the following command to show the content of the placement configurations argument:

   ```
   $ oci compute-management instance-pool create \
   --generate-param-json-input placement-configurations
   ```

3. If you want load balancing for this pool, construct an argument for the `--load-balancers` option.

   Use the following command to show the content of the load balancers argument:

   ```
   $ oci compute-management instance-pool create \
   --generate-param-json-input load-balancers
   ```

   To attach a load balancer after the instance pool is created, see Managing Instance Pool Load Balancer Attachments.

4. Run the instance pool create command.

   Syntax:

```
oci compute-management instance-pool create -c compartment_OCID \
--instance-configuration-id instance_configuration_OCID \
--placement-configurations file://placement_configuration.json \
--size number_of_instances
```

Example:

```
$ oci compute-management instance-pool create \
--compartment-id ocid1.compartment.unique_ID \
--display-name support-pool \
--instance-configuration-id ocid1.instanceConfiguration.unique_ID \
--placement-configurations file://./placement_configurations.json \
--load-balancers file://./load_balancers.json --size 10
```

The value of the `--display-name` option is the name of the pool. The pool name is not required to be unique. If you do not provide a value for the `--display-name` option, the default name of the instance pool is `instancepool`***YYYYMMDDhhmmss***, where ***YYYYMMDDhhmmss*** is the creation date and time.

The pool name is used in the names of the instances. Instances in a pool are named `inst-`***aaaaa-pool_name***, where ***aaaaa*** is five random alphanumeric characters. If you change the name of the pool and then add new instances to the pool, the new instances will have the new name.

The output of this command is the same as the output of the `instance-pool get` command. The list of instances in the pool is not shown.

To list the instances that belong to this pool, use the following command:

```
$ oci compute-management instance-pool list-instances -c compartment_OCID \
--instance-pool-id instance_pool_OCID
```

The output for each instance is abbreviated compared with the output from the `instance get` command.

The following command shows the same abbreviated output for only the specified instance:

```
$ oci compute-management instance-pool-instance get --instance-id
ocid1.instance.unique_ID \
--instance-pool-id ocid1.instancePool.unique_ID
```

Using Schedule-Based Autoscaling

Autoscaling instance pools enables you to effectively manage instance resource use.

An instance pool can have an autoscaling configuration and policies that scale the instance pool in the following ways according to a schedule:

- Scale out: add instances
- Scale in: remove instances
- Lifecycle or power action: stop, start, or reboot instances

When an instance pool scales out or scales in, instances are created or terminated as described in Updating an Instance Pool.

Policies define the schedule for autoscaling and the specific actions to take. An autoscaling configuration can have up to 50 schedule-based autoscaling policies, each with a different schedule and target pool size or lifecycle action. An instance pool can have only one autoscaling configuration.

If you manually change the pool size or lifecycle state as described in Updating an Instance Pool and Stopping and Starting Instances in an Instance Pool, autoscaling resets the pool size or lifecycle state to the value that is set in the policy the next time the scheduled autoscaling policy runs.

> **Note:**
>
> To use autoscaling, ensure that you have installed OCI CLI 3.15.1 or newer and Oracle Cloud Infrastructure Python SDK 2.80.0 or newer.

Multiple Schedule Management

When you create and enable an autoscaling configuration, the Autoscaling service evaluates the schedule rules in the policies in the configuration.

If multiple policies in the same configuration run at the same time, only one lifecycle state policy and one pool size policy will run. The lifecycle state policy runs first.

If multiple lifecycle state policies in the same configuration run at the same time, the policy with the highest priority action runs. The following list shows actions in priority order from highest to lowest priority:

1. Force Reboot
2. Reboot
3. Start
4. Force Stop

If multiple pool size policies in the same configuration run at the same time, the policy that specifies the largest pool size runs.

Creating an Autoscaling Configuration

An autoscaling configuration contains policies that schedule adding or removing instances in a specified pool, or stopping, starting, or rebooting all the instances in the pool.

Using the Compute Web UI

1. In the navigation menu, click Compute, and then click Autoscaling Configurations.
2. Click the Create Autoscaling Configuration button.
3. In the Create Autoscaling Configuration dialog, enter the following information:
   - *Name*: Enter a name for the autoscaling configuration.
   - *Create in Compartment*: Select the compartment where you want to create the autoscaling configuration.
   - *Instance Pool*: Select the instance pool that you want to scale with this autoscaling configuration.
   - *Autoscaling Policies*: For each policy, provide the following information:
     - *Action To Perform*: Select either Change Lifecycle State or Scale Pool Size.
     - *Policy Name*: Enter a name for the policy.

  - *Lifecycle Action*: If you selected Change Lifecycle State for Action To Perform, then select one of the following states to which to transition all instances of the pool when this policy is executed: Start, Stop, Soft Reset, Reset.
  - *Instance Pool Limit*: If you selected Scale Pool Size for Action To Perform, then enter a value for the pool size.
  - *Enable Schedule*: By default, the Schedule Enabled box is selected to enable the policy to execute at the next scheduled time. Uncheck the box to disable this policy.
  - *Policy Schedule*: Enter values for Minute, Hour, Day Of Month, Month, Day Of Week, and Year. Provide all schedule times in UTC. For more information, see Creating a Schedule-Based Autoscaling Policy.

  To add another policy, click the Add Policy button. You can also add policies after the autoscaling configuration is created, as described in Creating a Schedule-Based Autoscaling Policy.

  To delete a policy, click the trash can icon for that policy.

- *Tagging*: (Optional) Add defined or free-form tags for this instance as described in Adding Tags at Resource Creation. Tags can also be applied later.
- Click Submit.

  The details page for the new autoscaling configuration is displayed.

  On the details page, ignore the Cooldown Period value. Cooldown period does not apply to schedule-based autoscaling configurations.

  The new autoscaling configuration is enabled by default. To disable the configuration, see Updating an Autoscaling Configuration.

Using the OCI CLI

1. Get the following information:
   - The OCID of the compartment where you want to create this autoscaling configuration: `oci iam compartment list`
   - The OCID of the instance pool that will be managed by this autoscaling configuration: `oci compute-management instance-pool list`
2. Construct a file that contains all the input for the command.

   Use the following command to show the content and format of the command input:

   ```
   $ oci autoscaling configuration create \
   --generate-full-command-json-input > autoscalingCfgCreate.json
   ```

   The `resource` property is required and is the OCID of the instance pool that will be managed by this autoscaling configuration. The `type` of this resource must be `instancePool`.

   At least one policy is required to create an autoscaling configuration. To add policies after the autoscaling configuration is created, see Creating a Schedule-Based Autoscaling Policy.

   The optional display name is 1-255 characters, does not need to be unique, and can be updated. If you do not provide a value for `-displayName`, the default name of the autoscaling configuration is `autoscalingConfiguration`***`YYYYMMDDhhmmss`***, where `instanceconfiguration`***`YYYYMMDDhhmmss`*** is the creation date and time.

The autoscaling configuration is enabled by default. To disable the configuration, set `isEnabled` to `false`.

> **Note:**
>
> Do not specify values for `coolDownInSeconds` or `capacity` `min` or `max`. These properties do not apply to schedule-based autoscaling configurations.

The default values for `cool-down-in-seconds` and `capacity` `min` and `max` appear in the created autoscaling configuration but are not used for schedule-based autoscaling.

The following is an example autoscaling configuration create input file with one policy:

```
{
  "compartmentId": "ocid1.compartment.unique_ID",
  "displayName": "salesPoolCfg",
  "policies":
      {
        "displayName": "reboot policy",
        "executionSchedule":
          {
            "expression": "0 0 2 ? * 1#1 *",
            "timezone": "UTC",
            "type": "cron"
          },
        "policyType": "scheduled",
        "resourceAction": {
          "actionType": "power",
          "action": "SOFTRESET"
        }
      },
  "resource":
    {
      "id": "ocid1.instancePool.unique_ID",
      "type": "instancePool"
    }
}
```

3. Run the command to create the autoscaling configuration.

   Syntax:

   ```
   oci autoscaling configuration create --compartment-id compartment_OCID \
   --from-json file://input_file.json
   ```

   Example:

   ```
   $ oci autoscaling configuration create --c ocid1.compartment.unique_ID \
   --from-json file:///./salesPoolCfg.json
   {
     "data": {
       "compartment-id": "ocid1.compartment.unique_ID",
       "cool-down-in-seconds": 300,
       "defined-tags": {},
       "display-name": "salesPoolCfg",
       "freeform-tags": {},
       "id": "ocid1.autoScalingConfiguration.unique_ID",
       "is-enabled": true,
       "max-resource-count": null,
       "min-resource-count": null,
   ```

```
    "policies":
      {
        "capacity": null,
        "displayName": "reboot policy",
        "executionSchedule":
          {
            "expression": "0 0 2 ? * 1#1 *",
            "timezone": "UTC",
            "type": "cron"
          },
        },
        "id": "unique_ID",
        "is-enabled": true,
        "policy-type": "scheduled",
        "resourceAction": {
          "actionType": "power",
          "action": "SOFTRESET"
        },
        "time-created": "2023-01-25T21:28:56.131801+00:00"
      },
    "resource": {
      "id": "ocid1.instancePool.unique_ID",
      "type": "instancePool"
    },
    "time-created": "2023-01-25T21:28:56.140747+00:00"
  },
  "etag": "7c70532a-1d41-4861-a40f-bf840136a9c5"
}
```


Use the `work-requests work-request get` command to check the status of the autoscaling configuration creation.

Creating a Schedule-Based Autoscaling Policy

An autoscaling policy is part of an autoscaling configuration. Each policy of a schedule-based autoscaling configuration has a schedule and either a target pool size or a lifecycle action.

The procedures in this section describe how to create policies separate from creating the autoscaling configuration.

Designing Policies

This section provides some tips for designing and troubleshooting policies.

Create two separate policies to scale a pool in and out or to change the state of the pool between stopped and running.

- Scale example: One policy specifies a larger size for the pool at the beginning of a high demand period, and a second policy specifies a smaller pool size at the end of the high demand period.
- State example: One policy stops all instances in the pool at the beginning of a regular compute node maintenance period, and a second policy starts the pool at the end of the maintenance period.

Design the policy schedule as follows:

- Use `cron` expressions. Autoscaling uses a cron implementation similar to the Quartz cron implementation. All fields require a value. If fields conflict, such as day of month and day of week, use a specific value for one and a question mark for the other.
- Provide all schedule times in UTC.

- Use an online `cron` expression generator such as Cron Expression Generator & Explainer - Quartz to verify your schedule expressions.
- Ensure that policy schedules do not conflict. See the descriptions in Multiple Schedule Management of which policies run when schedules conflict.

Take the following steps if a policy fails to run, or appears to fail to run:

- Check that the autoscaling configuration and autoscaling policy are both enabled.
- Check the schedule expression. Is the policy set to run when you meant for it to run? Remember, all expression times must be provided in UTC.
- Was the policy set to start instances that were already running, or stop instances that were already stopped?

  In addition to a policy conflict, a power action might have been performed on the pool separate from any autoscaling policy. That separate power action could prevent the policy action from succeeding. The policy power action will not be retried.
- Was the policy set to scale out, but not enough resources were available?

  The scale policy sets the pool size, and the pool will continue to attempt to reach that size as resources become available.
- Is the operation specified by the policy still executing or waiting to execute?
  - Check whether the pool is in the Scaling, Starting, Stopping, or Rebooting state, which indicates that the policy operation is still running.
  - If a state change operation attempts to run while a state change operation is already running on the same pool, the second operation will fail to run.
  - A limited number of pools can be changing state concurrently. If too many other pools are already changing state, then your pool will need to wait to begin changing state. The time to change state is longer when more instances are involved since instances are started, stopped, or rebooted serially.
  - A limited number of pools can be changing size concurrently. If other pools are already changing size, your pool might need to wait to begin scaling. The time to scale is longer when more instances are involved since instances are terminated or created serially. Terminating and creating instances are both background operations and take some time to begin after the pool size has been updated by the policy.

Using the Compute Web UI

1. In the navigation menu, click Compute, and then click Autoscaling Configurations.
2. If the configuration to which you want to add a policy is not listed, use the Compartment drop-down menu above the autoscaling configurations list to select the correct compartment.
3. Click the name of the autoscaling configuration to which you want to add a policy.
4. On the autoscaling configuration details page, scroll to the Resources section, click Autoscaling Policies, and click the Create Scheduled Policy button.
5. In the Create policy dialog, enter the following information:
   - *Name*: Enter a name for the new autoscaling policy.
   - *Action to perform*: Select Scale pool size or Change lifecycle state of all instances.
     - *Scale pool size*: Enter the number of instances that the pool should scale to at the scheduled time.

  - *Change lifecycle state of all instances*: Select the state that all instances in the pool should transition to at the scheduled time.
  - *Enable Schedule*: By default, the Schedule Enabled box is selected to enable the policy to execute at the next scheduled time. Uncheck the box to disable this policy.
- *Execution schedule*: Define the schedule for implementing this autoscaling policy. See Designing Policies.

6. Click the Submit button.

Using the OCI CLI

1. Get the OCID of the autoscaling configuration where you want to add this autoscaling policy: `oci autoscaling configuration list`
2. Construct a file that contains the policy definitions.

   Use the following command to show the content and format of the file:

   ```
   $ oci autoscaling policy create \
   --generate-full-command-json-input > autoscalingPolicyCreate.json
   ```

   > **Note:**
   >
   > Do not specify values for `capacity` `min` or `max`. These properties do not apply to schedule-based autoscaling configurations.

   The default values for `capacity` `min` and `max` appear in the created autoscaling policy but are not used for schedule-based autoscaling.

   The display name is 1-255 characters, does not need to be unique, and can be updated.

   Use the references in Designing Policies for help with setting the policy execution schedule. The `timezone` must be `UTC` and the `type` must be `cron`.

   The policy is enabled by default.

   The policy type must be `scheduled`.

   The resource action type must be `power`, and the action must be one of `STOP`, `START`, `SOFTRESET`, `RESET`.

   The following is an example autoscaling policy create input file:

   ```
   {
     [
       {
         "capacity": {
           "initial": 10
         },
         "displayName": "size 10",
         "executionSchedule":
           {
             "expression": "0 0 10 ? 1 2#2 *",
             "timezone": "UTC",
             "type": "cron"
           },
         "isEnabled": true,
         "policyType": "scheduled"
       },
   ```

```
      "capacity": {
        "initial": 30
      },
      "displayName": "size 30",
      "executionSchedule":
        {
          "expression": "0 0 7 ? 11 5#1 *",
          "timezone": "UTC",
          "type": "cron"
        },
      "isEnabled": true,
      "policyType": "scheduled"
    },
    {
      "displayName": "stop policy",
      "executionSchedule":
        {
          "expression": "0 0 7 ? JAN,APR,JUL,OCT 4#3 *",
          "timezone": "UTC",
          "type": "cron"
        },
      "isEnabled": true,
      "policyType": "scheduled",
      "resourceAction": {
        "actionType": "power",
        "action": "STOP"
      }
    },
    {
      "displayName": "start policy",
      "executionSchedule":
        {
          "expression": "0 0 13 ? JAN,APR,JUL,OCT 4#3 *",
          "timezone": "UTC",
          "type": "cron"
        },
      "isEnabled": true,
      "policyType": "scheduled",
      "resourceAction": {
        "actionType": "power",
        "action": "START"
      }
    }
  ]
}
```

3. Run the command to create new policies for the specified autoscaling configuration.

   Syntax:

   ```
   oci autoscaling policy create \
   --auto-scaling-configuration-id autoscaling_configuration_OCID \
   --from-json file://policy_definitions.json --policy-type scheduled
   ```

   Example:

   ```
   $ oci autoscaling policy create \
   --auto-scaling-configuration-id ocid1.autoscalingConfiguration.unique_ID \
   --from-json file:///./salesPoolPolicies.json --policy-type scheduled
   ```

   Use the `work-requests work-request get` command to check the status of the autoscaling policy create command.

Updating an Autoscaling Configuration

You can change the display name of the autoscaling configuration, the tags, and whether the autoscaling configuration is enabled.

To add policies to an autoscaling configuration or update existing policies, see Creating a Schedule-Based Autoscaling Policy and Updating a Schedule-Based Autoscaling Policy.

Using the Compute Web UI

1. In the navigation menu, click Compute, and then click Autoscaling Configurations.
2. If the autoscaling configuration that you want to update is not listed, use the Compartment drop-down menu above the autoscaling configurations list to select the correct compartment.
3. For the autoscaling configuration that you want to update, click the Actions menu and click Edit.
4. In the Update Autoscaling Configuration dialog, make the changes. Ignore the Cooldown Period value.
5. Click the Submit button.

Using the OCI CLI

1. Get the OCID of the autoscaling configuration that you want to update: `oci autoscaling configuration list`
2. Run the update command.

   Example:

   ```
   $ oci autoscaling configuration update \
   --auto-scaling-configuration-id ocid1.autoscalingConfiguration.unique_ID \
   --is-enabled false
   ```

   Use the `work-requests work-request get` command to check the status of the update operation.

Updating a Schedule-Based Autoscaling Policy

These procedures describe how to update the specified autoscaling policy in the specified autoscaling configuration. You can update the policy display name, either the target pool size or the lifecycle action, and the execution schedule. You can enable or disable the policy.

Using the Compute Web UI, you can delete the policy or add a new policy. To add or delete a policy using the OCI CLI, see Creating a Schedule-Based Autoscaling Policy and Deleting an Autoscaling Policy.

Using the Compute Web UI

1. In the navigation menu, click Compute, and then click Autoscaling Configurations.
2. If the configuration for which you want to update a policy is not listed, use the Compartment drop-down menu above the autoscaling configurations list to select the correct compartment.
3. Click the name of the autoscaling configuration for which you want to update a policy.

4. On the autoscaling configuration details page, scroll to the Resources section and click Autoscaling Policies.
5. For the policy that you want to update, click the Actions menu and then click Edit.
6. In the Update Policy dialog, change the policy information.
7. Click the Submit button.

Using the OCI CLI

1. Get the following information:
   - The OCID of the autoscaling configuration from which you want to remove this autoscaling policy: `oci autoscaling configuration list`
   - The ID of the autoscaling policy within that autoscaling configuration: `oci autoscaling configuration get`
2. Construct a file that contains the updated policy definition.

   Use the following command to show the content and format of the file:

   ```
   $ oci autoscaling policy update \
   --generate-full-command-json-input > autoscalingPolicyUpdate.json
   ```

   See Creating a Schedule-Based Autoscaling Policy for information about policy properties.
3. Run the autoscaling policy update command.

   Syntax:

   ```
   oci autoscaling policy update \
   --auto-scaling-configuration-id autoscaling_configuration_OCID \
   --auto-scaling-policy-id autoscaling_policy_ID \
   --from-json file://policy_definition.json --policy-type scheduled
   ```

   Example:

   ```
   $ oci autoscaling policy update \
   --auto-scaling-configuration-id ocid1.autoscalingConfiguration.unique_ID \
   --auto-scaling-policy-id ID \
   --from-json file:///./stop_policy.json --policy-type scheduled
   ```

   Use the `work-requests work-request get` command to check the status of the policy update operation.

Deleting an Autoscaling Configuration

These procedures describe how to delete the specified autoscaling configuration.

Alternatively, you could disable the configuration as described in Updating an Autoscaling Configuration.

Using the Compute Web UI

1. In the navigation menu, click Compute, and then click Autoscaling Configurations.
2. If the autoscaling configuration that you want to delete is not listed, use the Compartment drop-down menu above the autoscaling configurations list to select the correct compartment.
3. For the autoscaling configuration that you want to delete, click the Actions menu and click Delete.

Using the OCI CLI

1. Get the OCID of the autoscaling configuration that you want to delete: `oci autoscaling configuration list`
2. Run the autoscaling configuration delete command.

   Example:

   ```
   $ oci autoscaling configuration delete \
   --auto-scaling-configuration-id ocid1.autoscalingConfiguration.unique_ID --force
   ```

   Use the `work-requests work-request get` command to check the status of the delete operation.

Deleting an Autoscaling Policy

These procedures describe how to delete the specified autoscaling policy in the specified autoscaling configuration.

Alternatively, you could disable the policy in the configuration as described in Updating a Schedule-Based Autoscaling Policy.

Using the Compute Web UI

1. In the navigation menu, click Compute, and then click Autoscaling Configurations.
2. If the autoscaling configuration from which you want to delete a policy is not listed, use the Compartment drop-down menu above the autoscaling configurations list to select the correct compartment.
3. Click the name of the autoscaling configuration for which you want to delete a policy.
4. On the autoscaling configuration details page, scroll to the Resources section and click Autoscaling Policies.
5. For the policy that you want to delete, click the Actions menu and then click Delete.

Using the OCI CLI

1. Get the following information:
   - The OCID of the autoscaling configuration from which you want to remove this autoscaling policy: `oci autoscaling configuration list`
   - The ID of the autoscaling policy within that autoscaling configuration: `oci autoscaling configuration get`
2. Run the autoscaling policy delete command.

   Syntax:

   ```
   $ oci autoscaling policy delete \
   --auto-scaling-configuration-id ocid1.autoscalingConfiguration.unique_ID \
   --auto-scaling-policy-id unique_ID --force
   ```

   Use the `work-requests work-request get` command to check the status of the policy delete operation.

Updating an Instance Pool

When you update an instance pool, you can change the name of the pool, the size of the pool, the instance configuration that is used to create new instances, the fault domains, VCN, and subnet.

To attach or detach an instance, see Attaching an Instance to an Instance Pool and Detaching an Instance from an Instance Pool.

To attach load balancers or detach load balancer attachments, see Managing Instance Pool Load Balancer Attachments.

Configuration changes do not affect existing instances; configuration changes only affect new instances. New instances will be provisioned using the new instance configuration and placement configuration.

If you increase the size of the pool, new instances are provisioned. The new instances are launched evenly across the fault domains specified by the instance configuration or the placement configuration.

If you decrease the size of the pool, instances are terminated evenly across the fault domains specified by the instance configuration or the placement configuration. In each fault domain, instances are terminated in creation date order, oldest first.

You cannot select which instances to terminate when you decrease the size of a pool. If you terminate an individual instance that is a member of a pool, as described in Terminating an Instance, a new instance is automatically provisioned to keep the pool at the specified pool size.

If you increase the size of the pool, and some new instances cannot be provisioned because of resource constraints, those instances remain in Provisioning state and the pool remains in Scaling state until all instances are provisioned. See the suggested remedies in Creating an Instance Pool.

Using the Compute Web UI

1. In the navigation menu, click Compute, and then click Instance Pools.
2. If the instance pool that you want to update is not listed, use the Compartment drop-down menu above the instance pools list to select the correct compartment.
3. For the instance pool that you want to update, click the Actions menu, and click the Edit option.
4. In the Update Instance Pool dialog, make the changes.
5. When you are finished editing, click the Update Instance Pool button in the dialog.

Using the OCI CLI

1. Get the OCID of the instance pool that you want to update: `oci compute-management instance-pool list`
2. Run the instance pool update command.

   Syntax:

   ```
   oci compute-management instance-pool update \
   --instance-pool-id instance_pool_OCID \
   options_with_values_to_update
   ```

Example:

```
$ oci compute-management instance-pool update \
--instance-pool-id ocid1.instancePool.unique_ID \
--instance-configuration-id new_instance_configuration_OCID --size 20
```

The output of this command is the same as the output of the `instance-pool get` command.

Attaching an Instance to an Instance Pool

When you attach an instance to an instance pool, the pool size increases.

> **Important:**
>
> If an autoscaling configuration is associated with the instance pool, then ensure that the autoscaling policy defines a target pool size that is large enough for the expanded pool. The next time the scheduled autoscaling policy runs, the target pool size is reset to the value that is set in the policy; if the policy size is smaller than the current size, instances will be deleted.

If load balancers are attached to the pool, then the instance is also added to the load balancers.

Ensure the following conditions before you attach an instance to an instance pool:

- Both the pool and the instance to be attached are running.
- The instance is not attached to another pool.
- The instance is in the same fault domain as the pool.
- The primary VNIC of the instance is in the same VCN and subnet as the pool.
- If secondary VNICs are defined, then the secondary VNIC of the instance is in the same VCN and subnet as the secondary VNICs used by other instances in the pool.

To attach an instance that is in a fault domain that is not included in the pool instance configuration, or is using a VCN and subnet that are not specified by the pool instance configuration, first update the instance configuration, then attach the instance. To update the instance configuration, create and attach a new instance configuration as shown in Updating an Instance Pool.

Using the Compute Web UI

1. In the navigation menu, click Compute, and then click Instance Pools.
2. If the instance pool that you want to update is not listed, use the Compartment drop-down menu above the instance pools list to select the correct compartment.
3. Click the name of the instance pool to which you want to attach an instance.
4. On the instance pool details page, scroll down to the Resources section and click Attached Instances.
5. Click the Attach Instance button on the Attached Instances table.
6. In the Input type field of the Attach Instance dialog, select either Instance name or Instance OCID.

- If you select Instance name, a list of instances is displayed. The list of instances is labeled "Potentially attachable instances" because instances must meet certain criteria to be eligible to be attached. For example, the instance must be in the same VCN and subnet that is specified by the instance pool configuration, and the instance must not already be attached to this pool or any other pool. See the list of criteria at the top of the Attach Instance dialog.

  If an instance that you think should be attachable is not shown in the list, try using Instance OCID.

- If you select Instance OCID, a text field labeled Instance OCID is displayed where you can paste the OCID of the instance that you want to attach.

7. Click the Attach button on the dialog.

   Even if the criteria listed in the dialog are met, the instance could fail to attach for some other reason. In the Resources box, click Work Requests and click the applicable work request in the list to troubleshoot any problems.

Using the OCI CLI

1. Get the information you need to run the command.
   - OCID of the instance pool that you want to update: `oci compute-management instance-pool list`
   - OCID of the instance that you want to attach: `oci compute instance list`
2. Run the instance pool attach instance command.

```
$ oci compute-management instance-pool-instance attach \
--instance-pool-id ocid1.instancePool.unique_ID \
--instance-id ocid1.instance.unique_ID
```

   The output of this command is the same as the output of the `instance get` command. If you run the `instance-pool get` command, you see that the `size` property is incremented.

Detaching an Instance from an Instance Pool

When you detach an instance from a pool, you can choose whether to delete the instance or to retain the instance separate from the pool.

Using the OCI CLI, you can choose whether to replace the detached instance by creating a new instance in the pool. If you don't replace the detached instance, then the pool size is decremented.

If load balancers are attached to the pool, then the instance is removed from the load balancers.

Using the Compute Web UI

1. In the navigation menu, click Compute, and then click Instance Pools.
2. If the instance pool that you want to update is not listed, use the Compartment drop-down menu above the instance pools list to select the correct compartment.
3. Click the name of the instance pool from which you want to detach an instance.
4. On the instance pool details page, scroll down to the Resources section and click Attached Instances to display the list of attached instances.
5. For the instance that you want to detach, click the Actions menu and click Detach to display the Confirm instance detachment dialog.

6. (Optional) To delete the instance and its boot volume, click the button under "Permanently terminate (delete) this instance and its attached boot volume."
7. Click the Confirm button to detach the instance.

   The pool size is decremented.

Using the OCI CLI

1. Get the information you need to run the command.
   - OCID of the instance pool that you want to update: `oci compute-management instance-pool list`
   - OCID of the instance that you want to detach: `oci compute-management instance-pool list-instances`
2. Run the instance pool detach instance command.

   Syntax:

   ```
   oci compute-management instance-pool-instance detach \
   --instance-pool-id instance_pool_OCID --instance-id instance_OCID \
   --is-auto-terminate [true|false] --is-decrement-size [true|false]
   ```

   Provide the following options if you do not want the default behavior:

   `--is-auto-terminate`

   If `true`, permanently terminate (delete) the instance and its attached boot volume when the instance is detached from the instance pool. The default value is `false`.

   `--is-decrement-size`

   If `true`, decrement the pool `size` when the instance is detached from the instance pool. This is the default.

   If `false`, provision a new, replacement instance using the pool's instance configuration after the existing instance is detached from the instance pool. The pool size remains the same as it was before you performed this detach operation.

   Example:

   In the following example, the specified instance is detached from the pool and terminated, and a new instance is provisioned for the pool.

   ```
   $ oci compute-management instance-pool-instance detach \
   --instance-pool-id ocid1.instancePool.unique_ID \
   --instance-id ocid1.instance.unique_ID \
   --is-auto-terminate true --is-decrement-size false
   ```

   The output of this command is the same as the output of the `instance-pool get` command.

Managing Instance Pool Load Balancer Attachments

These procedures describe how to attach a load balancer to an instance pool or detach a load balancer attachment from an instance pool.

Using the Compute Web UI

1. In the navigation menu, click Compute, and then click Instance Pools.

2. If the instance pool for which you want to manage load balancer attachments is not listed, use the Compartment drop-down menu above the instance pools list to select the correct compartment.
3. Click the name of the pool that you want to manage.
4. On the instance pool details page, scroll to the Resources section and click Load Balancers.
   - To attach a load balancer, click the Attach Load Balancer button.

     On the Attach Load Balancer dialog, specify the load balancer, backend set, port number, and VNIC as described in Creating an Instance Pool, and click the Attach Load Balancer button.
   - To remove a load balancer attachment, click the Actions menu for the load balancer attachment that you want to remove, and click Detach.

     The load balancer attachment remains visible in the load balancers list in the Detached state for at least 24 hours, up to 24.5 hours. No further action is needed to detach the load balancer attachment.

Using the OCI CLI

- To attach a load balancer to an instance pool:
  1. Get the following information:
     - OCID of the instance pool to which you want to attach a load balancer: `oci compute-management instance-pool list`
     - OCID of the load balancer and name of the backend set: `oci lb load-balancer list`
     - Port value to use when creating the backend set.
     - VNIC to associate with the load balancer. The value can be `PrimaryVnic` or the display name of one of the secondary VNICs on the instance configuration that is associated with the instance pool.
  2. Run the instance pool attach load balancer command.

     Example:

```
$ oci compute-management instance-pool attach-lb \
--instance-pool-id ocid1.instancePool.unique_ID \
--load-balancer-id ocid1.loadbalancer.unique_ID \
--backend-set-name BES1 --port 80 --vnic-selection PrimaryVnic
```

- To remove or detach a load balancer from an instance pool:
  1. Get the following information:
     - OCID of the instance pool
     - OCID of the load balancer
     - Backend set name
  2. Run the instance pool detach load balancer command.

     Example:

```
$ oci compute-management instance-pool detach-lb \
--instance-pool-id ocid1.instancePool.unique_ID \
--load-balancer-id ocid1.loadbalancer.unique_ID \
--backend-set-name BES1
```

When you `get` or `list` the instance pool, the load balancer attachment remains visible in state `DETACHED` for at least 24 hours, up to 24.5 hours. No further action is needed to detach the load balancer attachment.

Stopping and Starting Instances in an Instance Pool

Performing operations such as reset or stop on the pool object performs that operation on all instances that are members of the pool. Performing these operations on an individual instance that is a member of the pool does not affect any other member instances.

When instances are stopped, compute resources are released and the instances are disconnected and unassigned from their compute nodes. When instances are started, the Compute service restarts the instances in the same fault domain that they were in when they were stopped.

Instances continue to count toward the pool size while they are stopped, and configuration of stopped pool instances is preserved. Configuration changes such as fault domain changes do not apply to pool instances that are restarted or reset.

Using the Compute Web UI

1. In the navigation menu, click Compute, and then click Instance Pools.
2. If the instance pool that you want to manage is not listed, use the Compartment drop-down menu above the instance pools list to select the correct compartment.
3. For the instance pool that you want to manage, click the Actions menu, and click the Start, Stop, or Reboot option.

   By default, clicking Stop selects Soft Stop and clicking Reboot selects Soft Reboot. To stop or reboot all instances in a pool immediately, click the Force option on the confirmation dialog.

   All of the instances in the pool are stopped, started, or rebooted. See Stopping, Starting, and Resetting an Instance for how to prepare for an instance to stop. Stopping and starting an instance can take up to five minutes.
4. Click the Start Instance Pool, Stop Instance Pool, or Reboot Instance Pool button on the confirmation dialog.

   In the Resources section of the pool details page, click Work Request(s) to check the status of the instance pool stop, start, or reboot. Click Attached Instances to view the status of the instances.

Using the OCI CLI

1. Get the OCID of the instance pool that you want to manage: `oci compute-management instance-pool list`
2. Run the instance pool stop, start, or reset command.

   Syntax:

   ```
   oci compute-management instance-pool {start | stop | softstop | reset | softreset} \
   --instance-pool-id instance_pool_OCID
   ```

   For descriptions of these commands, enter:

   ```
   $ oci compute-management instance-pool -h
   ```

   Example:

```
$ oci compute-management instance-pool reset --instance-pool-id
ocid1.instancePool.unique_ID
```

Use the `work-requests work-request get` command to check the status of the instance pool management change.

Deleting an Instance Pool

When you delete an instance pool, the resources that were created by the pool are permanently deleted, including associated instances, attached boot volumes, and block volumes.

Using the Compute Web UI

1. In the navigation menu, click Compute, and then click Instance Pools.
2. If the instance pool that you want to delete is not listed, use the Compartment drop-down menu above the instance pools list to select the correct compartment.
3. For the instance pool that you want to delete, click the Actions menu, and click the Delete option.
4. On the confirmation dialog, click the Confirm button.

   All of the instances in the pool are terminated. Terminated instances are not attached and therefore are not listed in Attached Instances in the Resources box on the pool details page.

   Click Work Request(s) in the Resources box to check the status of the instance pool delete. Instances that had been attached to this pool remain visible in the instance list in state Terminated for at least 24 hours, up to 24.5 hours, as described in Terminating an Instance.

Using the OCI CLI

1. Get the OCID of the instance pool that you want to terminate: `oci compute-management instance-pool list`
2. Run the instance pool terminate command.

   Example:

```
$ oci compute-management instance-pool terminate \
--instance-pool-id ocid1.instancePool.unique_ID
Are you sure you want to delete this resource? [y/N]: y
{
  "etag": "34153f54-0cc9-4e6b-bc02-328166efbb4a",
  "opc-work-request-id": "ocid1.workrequest.unique_ID"
}
```

   Use the `work-requests work-request get` command to check the status of the instance pool terminate. Instances that had been attached to this pool remain visible in instance `list` or `get` in `lifecycle-state TERMINATED` for at least 24 hours, up to 24.5 hours, as described in Terminating an Instance.

9
Container Instances

Container Instances is a serverless compute service that enables you to quickly and easily run containers without managing any servers. Container Instances runs your containers on minimal compute instances that are optimized for container workloads.

Working with Container Instances

This section describes how to create a container instance, view information about container instances and their containers, update the name and tags of a container instance, and delete a container instance. You can stop, start, and restart a container instance and move a container instance to a new compartment. Creating a container instance creates a container in the instance.

Creating a Container Instance

A Private Cloud Appliance container instance includes a single container.

The following is the minimum information that you must provide to create a container instance:

- The compartment where you want to create the container instance
- The shape
- The VCN and subnet

Container instances leverage 15 GB of ephemeral storage that is shared between the container instance and the container. This ephemeral storage is created automatically when the container instance is created. You cannot manually create storage and attach it to the container instance.

Using the Compute Web UI

1. On the dashboard, select Containers / Container Instances.
2. Select the Create Container Instance button above the Container Instances list.
3. In the Create Container Instance dialog, enter the following information:
    - **Name**: Enter a name for the container instance. Instance names have the following characteristics:
        - Can be changed after the container instance is created.
        - Do not need to be unique.
        - Can contain only alphanumeric characters and the hyphen (-) character.
        - Can be a maximum of 63 characters.
    - General Information
        - **Create in Compartment**: Select the compartment where you want to create the container instance.
        - **Fault Domain**: (Optional) Select a fault domain. By default, the system automatically selects the best fault domain for the container instance when the

instance is created. If you specify a fault domain, and the requested fault domain cannot accommodate the container instance, instance create fails. The fault domain can be changed after the container instance is created.

- **Shape**: Select either the CI.Standard.x86.Generic shape or the CI.Standard.E5.Flex shape. For descriptions of these shapes, see Container Instances in the *Oracle Private Cloud Appliance Concepts Guide*.

  Use the sliders to specify the number of OCPUs you want, and the total amount of memory you want in gigabytes.

- **VNIC**
  - **Name**: Enter the name of the VNIC. By default, the container instance name is used.
  - **Networking**
    - **Private IP**: (Optional) Enter the private IP address to use.
    - **Hostname**: (Optional) Enter a hostname if you are using DNS within the cloud network. The hostname must be unique across all VNICs in the subnet. By default, the container instance name is used for the hostname.
    - **Assign Public IP**: The box is checked by default. Uncheck the box if you do not want a public IP address to be assigned to this container instance.
  - **Subnet**
    a. **VCN**: Select a VCN from the list. You might need to change the compartment to the compartment where the VCN is located.
    b. **Subnet**: Select a subnet from the list.
- **Container Image URL**: Enter the URL of the image to use to launch the container instance. Specify a container image from an external registry of your choice. The image must be signed by a well known Certificate Authority, and the image manifest must be an up-to-date version.
- **Tagging**: (Optional) Add defined or free-form tags for this instance as described in Adding Tags at Resource Creation. Tags can also be applied later.

4. Select the Create Container Instance button in the dialog.

   The instance details page is displayed. On the Container Instance Information tab, the Shape Configuration section shows the shape, the number of OCPUs, the network bandwidth, and the total memory. The Primary VNIC column shows the VCN and subnet, the primary private IP address and any assigned public IP address, and the MAC address.

   To check the status of the container instance create, scroll to the Resources section and select Work Request(s).

   Select Containers in the Resources section. The container is named `computecontainer-`***`unique_ID`***. See Working with Containers in Container Instances.

Using the OCI CLI

1. Get the following information:
   - The OCID of the compartment where you want to create the container instance: `oci iam compartment list`
   - The name of the shape to use. Use the following command to list the available shapes and their characteristics. Use the OCID of the compartment where you want to create the instance. For descriptions of these shapes, see Container Instances in the *Oracle Private Cloud Appliance Concepts Guide*.

```
$ oci container-instances container-instance list-shapes --compartment-id
compartment_OCID
```

You must also specify the shape configuration, as shown in the following example. You must provide a value for `ocpus`, and you can specify the `memoryInGBs`.

```
--shape-config '{"ocpus": 2, "memoryInGBs": 2}'
```

- The URL of the image to use to launch the container instance. Specify a container image from an external registry of your choice. The image must be signed by a well known Certificate Authority, and the image manifest must be an up-to-date version.

2. Construct an argument for the `--vnics` option.

   This is a list of the networks available to the container on this container instance. For each VNIC, provide the name of the VNIC, the OCID of the subnet that you want to use, the private IP address, and whether a public IP address should be assigned.

   Use the following command to show the syntax for this argument:

   ```
   $ oci container-instances container-instance create --generate-param-json-input vnics
   ```

3. Run the create container instance command.

   Syntax:

   Only the required parameters are shown.

   ```
   oci container-instances container-instance create --availability-domain AD-1 \
   --compartment-id compartment_OCID --containers '[{"imageUrl": "image_url"}]' \
   --shape shape --shape-config '{"ocpus": 2}' \
   --vnics file://network_cofig.json
   ```

   Example:

   If you set a display name for the container instance, see the Compute Web UI procedure for characteristics.

   For a complete list of required and optional parameters, use the following command:

   ```
   $ oci compute instance launch -h

   $ oci container-instances container-instance create --availability-domain AD-1 \
   --compartment-id ocid1.compartment.unique_ID --containers file://
   container_config.json \
   --shape CI.Standard.E5.Flex --shape-config '{"ocpus": 2,"memoryInGBs": 2}' \
   --vnics '[{"subnetId": "ocid1.subnet.unique_ID"}]'
   ```

   View the work request to monitor the status of the container instance create. The work request OCID is returned in the container instance create output.

   ```
   $ oci container-instances work-request get --work-request-id
   ocid1.workrequest.unique_ID
   ```

   If the status of the container instance create is Failed, list the errors reported in the work request:

   ```
   $ oci container-instances work-request list-errors --work-request-id
   ocid1.workrequest.unique_ID
   ```

Viewing Container Instances

This topic describes how to list all container instances in a compartment and how to view more information about a specified container instance.

Using the Compute Web UI

1. On the dashboard, select Containers / Container Instances.

   If the container instance that you want to view is not listed, use the Compartment drop-down menu above the container instances list to select the correct compartment.
2. To view more information about the container instance, select the name of the container instance in the list.

   The container instance details page is displayed.
3. On the container instance details page, scroll to the Resources section, and select Containers to view information about the container instance container.

Using the OCI CLI

1. List all container instances in a specified compartment.
   a. Get the OCID of the compartment where you want to list container instances: `oci iam compartment list`
   b. Run the list container instances command.

      ```
      $ oci container-instances container-instance list --compartment-id
      ocid1.compartment.unique_ID
      ```

      Add the `--display-name` option to list only container instances with the specified name. Recall that container instance names do not need to be unique.
2. Get more information about a specific container instance.
   a. Get the OCID of the container instance from the `container-instance list` command.
   b. Run the get container instance command.

      ```
      $ oci container-instances container-instance get --container-instance-id
      ocid1.container-instance.unique_ID
      ```

Updating a Container Instance

You can change the name and tags of a specified container instance.

Using the Compute Web UI

1. On the dashboard, select Containers / Container Instances.

   If the container instance that you want to update is not listed, use the Compartment drop-down menu above the container instances list to select the correct compartment.
2. In the container instances list, for the container instance that you want to update, select the Actions menu and select Edit.
3. When you are finished making your changes, select the Save Changes button.

Using the OCI CLI

1. Get the OCID of the container instance that you want to update: `oci container-instances container-instance list`
2. Run the update container instance command.

   Example:

```
$ oci container-instances container-instance update --container-instance-id
ocid1.container-instance.unique_ID \
--display-name new_name --defined-tags new_tags --force
```

Moving a Container Instance to a Different Compartment

This topic describes how to move a container instance to a new compartment within the same tenancy. To move a container instance, you must use the OCI CLI.

Using the OCI CLI

1. Get the OCID of the container instance that you want to move: `oci container-instances container-instance list`
2. Run the move container instance command.

   Example:

   ```
   $ oci container-instances container-instance change-compartment --container-instance-
   id ocid1.container-instance.unique_ID \
   --compartment-id ocid1.compartment.unique_ID
   ```

Stopping, Starting, and Restarting a Container Instance

This topic describes how to start, stop, and restart a container instance.

You can start a container instance that is in the Inactive state. You can stop a container instance that is in the Active state.

Using the Compute Web UI

1. On the dashboard, select Containers / Container Instances.

   If the container instance that you want to start, stop, or restart is not listed, use the Compartment drop-down menu above the container instances list to select the correct compartment.
2. For the container instance that you want to start, stop, or restart, select the Actions menu, and select the Start, Stop, or Restart option.

   Alternatively, in the container instance list, select the name of the container instance to display the details page for that container instance. Select the Controls menu, and select the Start, Stop, or Restart option.
3. Select the Start Container Instance, Stop Container Instance, or Restart Container Instance button on the confirmation dialog.

   In the Resources section of the container instance details page, select Work Request(s) to check the status of the container instance start, stop, or restart.

Using the OCI CLI

1. Get the OCID of the container instance that you want to stop, start, or restart: `oci container-instances container-instance list`
2. Start, stop, or restart the container instance.
   - Start the container instance if it is "Inactive."

     ```
     $ oci container-instances container-instance start --container-instance-id
     ocid1.container-instance.unique_ID
     ```

   - Stop the container instance if it is "Active."

```
$ oci container-instances container-instance stop --container-instance-id
ocid1.container-instance.unique_ID
```

- Restart the container instance.

```
$ oci container-instances container-instance restart --container-instance-id
ocid1.container-instance.unique_ID
```

Deleting a Container Instance

You can permanently delete container instances that you no longer need. Any attached VNICs are automatically detached when the container instance is deleted. Eventually, the container instance's public and private IP addresses are released and become available for other container instances. When the containers are deleted by the service, any data on the ephemeral storage is lost.

Using the Compute Web UI

1. On the dashboard, select Containers / Container Instances.

   If the container instance that you want to delete is not listed, use the Compartment drop-down menu above the container instances list to select the correct compartment.

2. For the container instance that you want to delete, click the Actions menu, and click the Delete option.

   Alternatively, in the container instance list, click the name of the container instance to display the details page for that container instance. Click the Controls menu, and click the Delete option.

3. On the confirm container instance delete dialog, select the Confirm button.

   Select Work Request(s) in the Resources box to check the status of the container instance delete. Deleted container instances temporarily remain in the list of container instances with the state Deleted.

Using the OCI CLI

1. Get the OCID of the container instance that you want to delete: `oci container-instances container-instance list`

2. Run the delete container instance command.

   ```
   $ oci container-instances container-instance delete --container-instance-id
   ocid1.container-instance.unique_ID
   ```

   View the work request to monitor the status of the container instance delete. The work request OCID is returned in the container instance delete output.

   ```
   $ oci container-instances work-request get --work-request-id
   ocid1.workrequest.unique_ID
   ```

Working with Containers in Container Instances

This section describes how to view information about containers in container instances, retrieve log data from a container instance container, and update the name and tags of a container instance container.

Viewing Container Instance Containers

This topic describes how to list all containers for a compartment or a container instance and how to view more information about a specified container instance container.

Using the Compute Web UI

1. On the dashboard, select Containers / Container Instances.

   If the container instance that you want to view is not listed, use the Compartment drop-down menu above the container instances list to select the correct compartment.
2. In the container instances list, select the name of the container instance for which you want to view the container.
3. On the container instance details page, scroll to the Resources section, and select Containers.
4. Select the name of the container in the list to show more information about the container, including log data.

Using the OCI CLI

1. List all containers in a specified compartment.
   a. Get the OCID of the compartment where you want to list containers: `oci iam compartment list`
   b. Run the list containers command.

   ```
   $ oci container-instances container list --compartment-id
   ocid1.compartment.unique_ID
   ```
2. List the container in a specified container instance.
   a. Get the OCID of the container instance for which you want to view the container: `oci container-instances container-instance list`
   b. Run the get container instance command to list the container in the container instance.

   ```
   $ oci container-instances container-instance get --container-instance-id
   ocid1.container-instance.unique_ID
   ```
3. Get more information about a specific container.
   a. Get the OCID of the container by using one of the two list containers commands.
   b. Run the get container command.

   ```
   $ oci container-instances container get --container-id ocid1.container.unique_ID
   ```
4. Retrieve log data from a specific container.
   a. Get the OCID of the container by using one of the two list containers commands.
   b. Retrieve the most recent 256 KB of logs from the specified container.

   ```
   $ oci container-instances container retrieve-logs --container-id
   ocid1.container.unique_ID
   ```

Updating a Container Instance Container

You can change the name and tags of a specified container instance container.

Using the Compute Web UI

1. On the dashboard, select Containers / Container Instances.
2. In the container instances list, select the name of the container instance for which you want to update the container.
3. On the container instance details page, scroll to the Resources section, and select Containers.
4. For the container in the list, select the Actions menu and select Edit.

Using the OCI CLI

1. Get the OCID of the container that you want to update by using one of the following list containers commands:

   ```
   $ oci container-instances container list --compartment-id compartment_OCID
   $ oci container-instances container-instance get --container-instance-id
   ocid1.container-instance.unique_ID
   ```

2. Run the update container command.

   Example:

   ```
   $ oci container-instances container update --container-id ocid1.container.unique_ID \
   --display-name new_name --defined-tags new_tags --force
   ```

10

Block Volume Storage

Block volumes provide high-performance network storage capacity that supports a broad range of I/O intensive workloads.

You can use block volumes to expand the storage capacity of your compute instances, to provide durable and persistent data storage that can be migrated across compute instances, and to host large databases.

The Block Volume service lets you dynamically provision and manage block storage volumes. You can create, attach, connect, and move volumes to meet your storage, performance, and application requirements.

After a block volume is created, you attach the volume to one or more instances. You can use the volume like a regular hard drive. You can also disconnect a volume and attach it to another instance without the loss of data.

There are two types of volumes:

- **Block volume:** A detachable block storage device that allows you to dynamically expand the storage capacity of an instance.
- **Boot volume:** A detachable boot volume device that contains the image used to boot a Compute instance.

When a volume is created, the volume is thin (sparse) provisioned: The volume consumes only the space that has been written to the volume. When the volume is attached to an instance, the volume is thick (non-sparse) provisioned: The volume reserves exactly enough space to completely fill the volume. This behavior avoids out-of-space errors. When the volume is detached, the volume is again thin provisioned if it is not still attached to another instance.

For more conceptual information, refer to the Block Volume Storage Overview section in the Oracle Private Cloud Appliance Concepts Guide.

This section provides instructions for managing block volumes.

Creating and Attaching Block Volumes

Task Flow

No.	Task	Links
1.	Create a block volume.	Creating a Block Volume
2.	Attach the block volume to one or more instances.	Attaching a Volume or Attaching a Volume to Multiple Instances
3.	Identify the added block volume and perform administrative tasks.	Find Your Volume in the Instance
4.	Configure the volume to automatically mount when the instance is rebooted.	Configuring Volumes to Automatically Mount (Linux Instances)

Creating a Block Volume

You can set values for the Synchronous Write Bias and Secondary Cache properties by using OraclePCA defined tags. If you use the OCI CLI or API, you can specify the OraclePCA tag namespace, tag key, and values for the parameters that you want to set. You do not need to first create the OraclePCA tag namespace and tag keys.

> **Note:**
>
> If you use the Compute Web UI to set these parameters, you must first create the OraclePCA tag namespace, tag keys, and value choices. See Creating OraclePCA Tags for instructions.

Using the Compute Web UI

1. On the dashboard, click the Block Storage/View Block Volumes button.
2. Click the Create Block Volume button.
3. Provide the following volume information:
   - **Name:** Provide a name or description for the volume. Avoid entering confidential information.
   - **Compartment:** Select the compartment in which to create the block volume.
   - **Size (in GBs):** The default size of 1024 GB is shown. To change the size, enter a value from 50 to 32768 (50 GB to 32 TB).
   - **High Performance Volume:** (Optional) By default, the volume uses balanced performance. To create a block volume that uses the high performance feature, click the Enable High Performance button. For more information, see "Block Volume Performance Options" in the Block Volume Storage Overview chapter in the *Oracle Private Cloud Appliance Concepts Guide*. Before you enable high performance, check with an appliance administrator to verify that a high performance pool is available.

     This selection cannot be changed after the volume is created.
   - **Backup Policy:** (Optional) Select a backup policy from the drop-down list. You might need to change the compartment.

     Oracle defined policies are listed, as well as any user defined policies. For information about Oracle defined policies (`bronze`, `silver`, and `gold`), see "Volume Backups and Clones" in the Block Volume Storage Overview chapter in the *Oracle Private Cloud Appliance Concepts Guide*.

     Backup policies can be assigned or changed after the volume is created. A volume can only have only one volume backup policy assigned at a time. For information about creating, editing, and assigning backup policies, see Managing Backup Policies. You can also back up this volume manually as described in Creating a Manual Boot or Block Volume Backup.
   - **Tagging:** (Optional) Add defined or free-form tags for this volume as described in Adding Tags at Resource Creation. Tags can also be applied later.

     See the OCI CLI procedure for descriptions of the Synchronous Write Bias (OraclePCA.logBias) and Secondary Cache (OraclePCA.secondaryCache) defined tags and the Volume Block Size (PCA_blocksize) free-form tag. Note that

PCA_blocksize must be set in Create Block Volume. PCA_blocksize cannot be set or updated after the volume is created.

4. Click Create Block Volume.

   The volume is ready to attach to an instance after its icon lists the volume in the Available state. See Attaching a Volume.

Using the OCI CLI

1. Get the OCID of the compartment where you want to create the block volume (`oci iam compartment list`)
2. Run the volume create command.

   This procedure does not show all available parameters for this command. For information about additional parameters, run the command with the `--help` option.

   Syntax:

   ```
   oci bv volume create --availability-domain AD-1 \
   --compartment-id compartment_OCID
   ```

   Example:

   This example specifies VPUs, log bias, secondary cache, and volume block size.

   VPUs per Gigabyte Option

   The value of the `--vpus-per-gb` option is the number of volume performance units (VPUs) that will be applied to this volume per GB. The default value for `vpus-per-gb` is 10, for balanced volume performance. For higher performance, you can specify 20 VPUs/GB. For more information, see "Block Volume Performance Options" in the Block Volume Storage Overview chapter in the *Oracle Private Cloud Appliance Concepts Guide*. Before you specify high performance, check with an appliance administrator to verify that a high performance pool is available.

   VPUs per GB cannot be changed after the volume is created.

   Private Cloud Appliance does not support volume performance auto-tuning.

   Synchronous Write Bias and Secondary Cache Properties

   Synchronous Write Bias and Secondary Cache properties can be set by using defined tags. Specify the `OraclePCA` tag namespace. To set a value for the Synchronous Write Bias, specify `logBias` for the tag key. To set a value for the Secondary Cache, specify `secondaryCache` for the tag key. See Adding Tags at Resource Creation for the syntax to specify a defined tag.

   The `logBias` property controls the use of the write cache flash devices for a share or LUN ("Logzilla"). The value of the `logBias` property must be either `LATENCY` or `THROUGHPUT`. If this value is not set, the value `LATENCY` is used.

   The `secondaryCache` property controls the use of the read cache flash devices for a share or LUN ("Readzilla"). The value of the `secondaryCache` property must one of `ALL`, `METADATA`, or `NONE`. If this value is not set, the value `ALL` is used.

   The values of the `logBias` and `secondaryCache` properties can be changed with the `update` command.

   Volume Block Size Property

   The volume block size can be set by using the `PCA_blocksize` free-form tag. The default block size is 8192 bytes. To specify a different block size, specify a value in bytes for the `PCA_blocksize` tag. Supported values are a power of 2 between 512 bytes and 1

megabyte, specified as a string and fully expanded. Note that Oracle recommends setting the value to at least 8192 bytes. See Adding Tags at Resource Creation for the syntax to specify a free-form tag.

The block size cannot be modified once the volume has been created.

```
$ oci bv volume create --availability-domain AD-1 \
--compartment-id ocid1.compartment.unique_ID\
--display-name myblockvolume --size-in-gbs 50 --vpus-per-gb 20 \
--defined-tags '{"OraclePCA":{"logBias":"THROUGHPUT","secondaryCache":"METADATA"}}'
--freeform-tags '{"PCA_blocksize": "65536"}'
{
  "data": {
    "auto-tuned-vpus-per-gb": null,
    "autotune-policies": null,
    "availability-domain": "AD-1",
    "block-volume-replicas": null,
    "compartment-id": "ocid1.compartment.unique_ID",
    "defined-tags": {
      "OraclePCA": {
        "logBias": "THROUGHPUT",
        "secondaryCache": "METADATA"
      }
    },
    "display-name": "myblockvolume",
    "freeform-tags": {
        "PCA_blocksize": "65536"
    },
    "id": "ocid1.volume.unique_ID",
    "is-auto-tune-enabled": null,
    "is-hydrated": null,
    "kms-key-id": null,
    "lifecycle-state": "PROVISIONING",
    "size-in-gbs": 50,
    "size-in-mbs": 51200,
    "source-details": null,
    "system-tags": null,
    "time-created": "2022-12-08T21:05:36.647925+00:00",
    "volume-group-id": null,
    "vpus-per-gb": 20
  },
  "etag": "08d0abc9-60c6-4fc7-b6fe-85d0af1c0308",
  "opc-work-request-id": "ocid1.workrequest.unique_ID"
}
```

A `vpus-per-gb` value of 10 indicates that this is a balanced performance volume. A `vpus-per-gb` value of 20 indicates that this is a high performance volume.

Any backup policy that you assigned is not shown in this output or in the volume list or get output. Instead, use the command shown at the end of Listing Block Volumes and Block Volume Details to show the OCID of any backup policy that is assigned to this volume.

When the volume is in the AVAILABLE state, you can attach the volume to an instance. See Attaching a Volume.

Attaching a Volume

You can attach a volume to an instance to expand the available storage on the instance. A volume can be attached to more than one instance at the same time. See Attaching a Volume to Multiple Instances.

You can also attach a boot volume that has been detached from its instance to a different instance as a data volume. This operation is convenient for troubleshooting a boot volume and for performing administrative operations while the boot volume is detached from its instance.

> **Important:**
>
> Only attach Linux volumes to Linux instances and Microsoft Windows volumes to Microsoft Windows instances.

> **Important:**
>
> If you are reattaching a volume that was detached, the volume might be associated with a different device name, and the instance operating system might not recognize the volume.

Using the Compute Web UI

1. In the navigation menu, click Compute, then click Instances.
2. Select the compartment where the instance resides.
3. In the Instances list, click the instance that you want to attach a volume to.
4. In the lower left panel, under Resources, select Attached Block Volumes.
5. In the Attached Block Volumes panel, click Attach Block Volume.
6. Select the compartment where the block volume resides.
7. Select a Block Volume.
8. Select one of the following access methods:
   - **Read/Write:** (Default) Configures the volume attachment with read/write capabilities. The volume cannot be shared with other instances. This option enables attachment to a single instance only.
   - **Read/Write - Shareable:** Configures the volume attachment as read/write, shareable with other instances. This option enables read/write attachment to multiple instances.
   - **Read Only - Shareable:** Configures the volume attachment as read-only, enabling attachment to multiple instances.
9. Click Attach to Instance.

Using the OCI CLI

This procedure shows the minimum required parameters for this command. For information about optional parameters, run the command with the `--help` option.

1. Gather the information that you need to run the command:
   - Instance OCID (`oci compute instance list`)
   - Volume OCID (`oci bv volume list`)
2. Run the volume attach command.

The `--type` option is required. The value of the `--type` option must be `paravirtualized`. In the attachment data, this type is shown as `attachment-type`.

The `--is-shareable` option is required to attach a shareable volume. The default value of this option is `false`.

```
--is-shareable true
```

Syntax:

```
oci compute volume-attachment attach --instance-id instance_OCID \
--volume-id volume_OCID --type paravirtualized
```

Example:

This example attaches a volume that is read-write and not shareable.

```
$ oci compute volume-attachment attach \
--instance-id ocid1.instance.uniqueID \
--volume-id ocid1.volume.uniqueID \
--type paravirtualized
{
  "data": {
    "attachment-type": "paravirtualized",
    "availability-domain": "AD-1",
    "compartment-id": "ocid1.compartment.uniqueID",
    "device": null,
    "display-name": "volumeattachment.uniqueID",
    "id": "ocid1.volumeattachment.uniqueID",
    "instance-id": "ocid1.instance.uniqueID",
    "is-pv-encryption-in-transit-enabled": null,
    "is-read-only": false,
    "is-shareable": false,
    "lifecycle-state": "ATTACHED",
    "time-created": "2021-06-01T17:24:13+00:00",
    "volume-id": "ocid1.volume.uniqueID"
  }
}
```

Attaching a Volume to Multiple Instances

The Block Volume service provides the capability to attach a block volume to multiple compute instances. With this feature, you can share block volumes across instances in read/write or read-only mode. Attaching block volumes as read/write and shareable enables you to deploy and manage cluster-aware solutions.

There are important limitations and considerations for attaching volumes to multiple instances. For more information, refer to the Block Volume Storage Overview chapter in the Oracle Private Cloud Appliance Concepts Guide.

> **Important:**
>
> If you are attaching a volume that was detached, the volume might be associated with a different device name and the instance operating system might not recognize the volume.

Configuring Multiple Instance Volume Attachments with Read/Write Access

The Block Volume service does not provide coordination for concurrent write operations to volumes attached to multiple instances. To prevent data corruption from uncontrolled read/write operations, you must install and configure a cluster aware system or solution such as Oracle Cluster File System version 2 (OCFS2) on top of the shared storage before you can use the volume.

Summary of the required steps:

1. Attach the block volume to an instance as Read/Write and Shareable using the UI, CLI, or API.

   See Attaching a Volume.
2. Set up your OCFS2/O2CB cluster nodes.
3. Create your OCFS2 file system and mount point.

Configuring Multiple Instance Volume Attachments with Read-Only Mode

Once you attach a block volume to an instance as read-only, it can only be attached to other instances as read-only. If you want to attach the block volume to an instance as read/write, you need to detach the block volume from all instances and then you can reattach the block volume to instances as read/write.

1. Attach the block volume to an instance as read-only using the UI, CLI, or API.

   See Attaching a Volume.
2. Attach the block volume to additional instances as read-only using the UI, CLI, or API.

   See Attaching a Volume.

Find Your Volume in the Instance

When a block volume is initially attached to an instance, the instance sees the volume as a new disk, for example as device `/dev/sdb`. This procedure describes how to list the disk devices in an instance so that you can find the volume.

For UNIX images, if you want to mount these volumes when an instance boots, you need to add the volume to the `/etc/fstab` file. See Configuring Volumes to Automatically Mount (Linux Instances).

Optionally, you can perform various administrative tasks to configure the storage to suit your storage requirements.

The utilities you use to perform the administrative tasks vary depending on the type of OS in the instance. For additional administrative information, refer to the documentation for the version of the OS that is on the instance. These documentation libraries provide access to helpful information:

- Oracle Operating Systems Documentation: https://docs.oracle.com/en/operating-systems/index.html
- Oracle Virtualization Documentation: https://docs.oracle.com/en/virtualization/index.html

Identifying the Boot Volume and the Attached Block Volume Devices in the Instance Using Linux Commands

1. Log on to your instance as described in Connecting to a Compute Instance.
2. List the disk devices.

> **Important:**
>
> On UNIX operating systems, the order in which volumes are attached is non-deterministic, so it can change with each reboot. If you refer to a volume using the device name, such as `/dev/sdb`, and you have more than one non-root volume, there is no guarantee that the volume you intend to mount for a specific device name will be the volume mounted. When configuring the OS to recognize the block volume (for example, adding the volume to the `/etc/fstab` file), use the volume's SCSI ID as described in this procedure.

```
sudo ls /dev/sd*
/dev/sda  /dev/sda1  /dev/sda2  /dev/sdb
```

In this example, two devices are listed, `/dev/sda` and `/dev/sdb`.

3. Use the `fdisk -l` command to view configuration information about the devices.

   In this example, `/dev/sda` is the boot volume and `/dev/sdb` is the attached block volume.

```
sudo fdisk -l

Disk /dev/sda: 53.7 GB, 53687091200 bytes, 104857600 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 8192 bytes / 8192 bytes
Disk label type: dos
Disk identifier: 0x000af694

   Device Boot Start End Blocks Id System
/dev/sda1 * 2048 2099199 1048576 83 Linux
/dev/sda2 2099200 61442047 29671424 8e Linux LVM

Disk /dev/mapper/ol-root: 27.2 GB, 27229421568 bytes, 53182464 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 8192 bytes / 8192 bytes


Disk /dev/mapper/ol-swap: 3145 MB, 3145728000 bytes, 6144000 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 8192 bytes / 8192 bytes


Disk /dev/sdb: 1099.5 GB, 1099511627776 bytes, 2147483648 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 8192 bytes / 8192 bytes
```

   This example output provides this information about `/dev/sda` and `/dev/sdb`:

   - The size of `/dev/sda` is 53.7 GB (boot volume).
   - `/dev/sda` has two partitions: `/dev/sda1` and `/dev/sda2`.
   - The size of `/dev/sdb` is 1099.5 GB (the attached block volume), and does not have any partitions.

4. Identify the devices that have file systems and are mounted in the OS.

```
sudo df -T
Filesystem          Type     1K-blocks    Used Available Use% Mounted on
devtmpfs            devtmpfs  16318164       0  16318164   0% /dev
tmpfs               tmpfs     16332596       0  16332596   0% /dev/shm
tmpfs               tmpfs     16332596    8744  16323852   1% /run
tmpfs               tmpfs     16332596       0  16332596   0% /sys/fs/cgroup
/dev/mapper/ol-root xfs       26578248 2907292  23670956  11% /
/dev/sda1           xfs        1038336  292512    745824  29% /boot
tmpfs               tmpfs      3266520       0   3266520   0% /run/user/0
```

In this example:

- `/dev/sda1` has an xfs file system and it is mounted on `/boot` (the boot volume).
- `/dev/sdb` is not listed because this block volume was just attached and hasn't had a file system created and is not mountable yet.

5. Find the SCSI ID for the newly attached volume.

```
sudo ls -l /dev/disk/by-id
total 0
lrwxrwxrwx. 1 root root 10 Dec  6 18:26 dm-name-ol-root -> ../../dm-0
lrwxrwxrwx. 1 root root 10 Dec  6 18:26 dm-name-ol-swap -> ../../dm-1
lrwxrwxrwx. 1 root root 10 Dec  6 18:26 dm-uuid-
LVM-83pr2aUrW2ZdCbWgsN4ZRFqvsXGGNZ8JO6il7j1YTWpywZeewYCiA6ywDmIeho1G -> ../../dm-0
lrwxrwxrwx. 1 root root 10 Dec  6 18:26 dm-uuid-
LVM-83pr2aUrW2ZdCbWgsN4ZRFqvsXGGNZ8JsaUihE3RWozk5u4p5nOwG9sFcj34AU3F -> ../../dm-1
lrwxrwxrwx. 1 root root 10 Dec  6 18:26 lvm-pv-uuid-Dh9ydC-Rj90-chhj-tkwq-ZI0Z-mfop-
Wtg5bh -> ../../sda2
lrwxrwxrwx. 1 root root  9 Dec  6 18:26 scsi-3600144f096933b92000061ae9bfc0025 -
> ../../sda
lrwxrwxrwx. 1 root root 10 Dec  6 18:26 scsi-3600144f096933b92000061ae9bfc0025-part1
-> ../../sda1
lrwxrwxrwx. 1 root root 10 Dec  6 18:26 scsi-3600144f096933b92000061ae9bfc0025-part2
-> ../../sda2
lrwxrwxrwx. 1 root root  9 Dec  8 15:17 scsi-3600144f096933b92000061b1129e0037 -
> ../../sdb
lrwxrwxrwx. 1 root root  9 Dec  6 18:26 wwn-0x600144f096933b92000061ae9bfc0025 -
> ../../sda
lrwxrwxrwx. 1 root root 10 Dec  6 18:26 wwn-0x600144f096933b92000061ae9bfc0025-part1
-> ../../sda1
lrwxrwxrwx. 1 root root 10 Dec  6 18:26 wwn-0x600144f096933b92000061ae9bfc0025-part2
-> ../../sda2
lrwxrwxrwx. 1 root root  9 Dec  8 15:17 wwn-0x600144f096933b92000061b1129e0037 -
> ../../sdb
```

In this example, the following line shows the SCSI ID assigned to `sdb`:

```
lrwxrwxrwx. 1 root root 9 Dec 8 15:17 scsi-3600144f096933b92000061b1129e0037 -
> ../../sdb
```

where `scsi-3600144f096933b92000061b1129e0037` is the SCSI ID.

The SCSI ID is a persistent device name for `/dev/sdb` and is used when performing administrative operations on the device, such as partitioning, creating a file system, and mounting.

For more information about mounting a block volume file system to an instance, see Configuring Volumes to Automatically Mount (Linux Instances).

6. Perform administrative tasks to configure the block volume to suit your storage requirements.

The specific tasks you perform depend on the type of OS that runs the instance and how you want the storage configured. Refer to your OS documentation for details.

Configuring Volumes to Automatically Mount (Linux Instances)

On Linux instances, if you want to automatically mount volumes during an instance boot, you need add the volumes to the `/etc/fstab` file.

Before You Begin

Get the SCSI ID for the block volume you plan to mount. See Find Your Volume in the Instance.

On Linux operating systems, specify the volume SCSI ID in the `/etc/fstab` file instead of the device name (for example, `/dev/sdb`). This is an example of a Volume SCSI ID:

```
/dev/disk/by-id/scsi-3600144f096933b92000061b1129e0037
```

Adding Volumes to the `/etc/fstab` File

1. Prepare the newly attached block volume for mounting.

   Use the disk administration utilities included with instance OS to perform tasks such as the following:

   - Partition the volume
   - Create file systems on the volume or partitions

   Consult the documentation for your instance OS for details.

   This is an example of creating an ext4 file system for a block volume attached to a Linux instance:

   ```
   mkfs.ext4 /dev/disk/by-id/scsi-3600144f096933b92000061b1129e0037
   mke2fs 1.42.9 (28-Dec-2013)
   /dev/disk/by-id/scsi-3600144f096933b92000061b1129e0037 is entire device, not just
   one partition!
   Proceed anyway? (y,n) y
   Filesystem label=
   OS type: Linux
   Block size=4096 (log=2)
   Fragment size=4096 (log=2)
   Stride=2 blocks, Stripe width=2 blocks
   67108864 inodes, 268435456 blocks
   13421772 blocks (5.00%) reserved for the super user
   First data block=0
   Maximum filesystem blocks=2415919104
   8192 block groups
   32768 blocks per group, 32768 fragments per group
   8192 inodes per group
   Superblock backups stored on blocks:
           32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
           4096000, 7962624, 11239424, 20480000, 23887872, 71663616, 78675968,
           102400000, 214990848

   Allocating group tables: done
   Writing inode tables: done
   Creating journal (32768 blocks): done
   Writing superblocks and filesystem accounting information: done
   ```

2. Create a mount point for each file system you plan to mount.

   ```
   mkdir /mnt/volume1
   ```

3. Add the volume to the `/etc/fstab` file.

   For this example, the following new line is added to the `/etc/fstab` file:

   ```
   /dev/disk/by-id/scsi-3600144f096933b92000061b1129e0037 /mnt/volume1 ext4
   _netdev,nofail 0 0
   ```

   Following are descriptions of these field values:

   - **Device:** Specified using the SCSI ID:

     `/dev/disk/by-id/scsi-3600144f096933b92000061b1129e003`

   - **Mount point:** The mount point created in the previous step: `/mnt/volume1`
   - **Type:** The type of file system: `ext4` in this example.
   - **Options:**
     - `_netdev` – Configures the mount process to initiate before the volumes are mounted.
     - `nofail` – If the device does not exist, no errors are reported. This is a good option to use when an instance is used to create a custom image. Future instances created with that image will not include the block volume and might fail to boot without this option.
   - **Dump:** The value `0` means do not use the obsolete `dump` utility.
   - **fsck:** The value `0` means do not run `fsck`.

4. Use the following command to mount the volumes that are in the `/etc/fstab` file:

   ```
   sudo mount -a
   ```

5. Verify that the file system is mounted:

   ```
   mount | grep /mnt
   /dev/sdb on /mnt/volume1 type ext4
   (rw,relatime,seclabel,stripe=2,data=ordered,_netdev)
   ```

Managing Block Volumes

You can manage these aspects of block volumes:

- List the block volumes in a compartment.
- List the details of a block volume.
- List block volume attachments.
- Edit the volume's configuration.
- Move a volume to another compartment.
- Clone a volume.
- Detach a volume.
- Delete a volume.

Listing Block Volumes and Block Volume Details

You can list all block volumes in a specific compartment, and detailed information about a single volume.

Using the Compute Web UI

1. In the navigation menu, under Block Storage, click Block Volumes.
2. Select the appropriate compartment.
3. To view block volume details, click the name of the block volume.

   The details are displayed.

Detail Item	Description
Block volume icon	Displays the status of the block volume.
Block volume name	The name of the block volume.
Block Volume Information and Tags	Tabs that you can click to display: • General Information • Tags that have been applied to this object
Created	The day and time that the volume was created.
Compartment	The compartment that the volume belongs to.
OCID	The volume's Oracle Cloud ID.
Backup Policy	The backup policy assigned to the volume.
Size	The size of the volume.
High Performance Enabled	Whether the volume is configured as a high performance volume, and the volume performance units (VPUs) per GB.

Using the OCI CLI

1. Get the OCID of the compartment where you want to list block volumes: (`oci iam compartment list`)
2. Run the volume list command.

```
$ oci bv volume list --compartment-id ocid1.compartment.unique_ID
{
  "data": [
    {
      "auto-tuned-vpus-per-gb": null,
      "availability-domain": "AD-1",
      "compartment-id": "ocid1.compartment.unique_ID",
      "defined-tags": {},
      "display-name": "volume2",
      "freeform-tags": {},
      "id": "ocid1.volume.unique_ID",
      "is-auto-tune-enabled": null,
      "is-hydrated": null,
      "kms-key-id": null,
      "lifecycle-state": "AVAILABLE",
      "size-in-gbs": 52,
      "size-in-mbs": 53248,
      "source-details": null,
      "system-tags": null,
      "time-created": "2021-06-01T17:33:24+00:00",
      "volume-group-id": null,
      "vpus-per-gb": 20
    },
    {
      "auto-tuned-vpus-per-gb": null,
      "availability-domain": "AD-1",
```

```
      "compartment-id": "ocid1.compartment.unique_ID",
      "defined-tags": {},
      "display-name": "volume20210106171509",
      "freeform-tags": {},
      "id": "ocid1.volume.unique_ID",
      "is-auto-tune-enabled": null,
      "is-hydrated": null,
      "kms-key-id": null,
      "lifecycle-state": "AVAILABLE",
      "size-in-gbs": 50,
      "size-in-mbs": 51200,
      "source-details": null,
      "system-tags": null,
      "time-created": "2021-06-01T17:15:09+00:00",
      "volume-group-id": null,
      "vpus-per-gb": 10
    },
}
```

The `is-hydrated` value is always `null` because this property does not apply to Private Cloud Appliance.

Private Cloud Appliance does not support volume performance auto-tuning.

To list only a specific block volume, use the `get` command:

```
$ oci bv volume get --volume-id ocid1.volume.unique_ID
```

Any backup policy that is assigned to a volume is not shown in the volume list or get output. Use the following command to show the OCID of the backup policy for a volume:

```
$ oci bv volume-backup-policy-assignment get-volume-backup-policy-asset-assignment \
--asset-id ocid1.volume.unique_ID
{
  "data": [
    {
      "asset-id": "ocid1.volume.unique_ID",
      "id": "ocid1.backuppolicyassignment.unique_ID",
      "policy-id": "ocid1.volumebackuppolicy.unique_ID",
      "time-created": "2023-06-07T02:03:53.466062+00:00"
    }
  ]
}
```

Listing Block Volume Attachments

Using the Compute Web UI

1. In the navigation menu, click Compute, then click Instances.
2. Select the compartment where the instance resides.
3. Click the instance name to display the details.
4. Scroll to the Resources section and select Attached Block Volumes.

   Block volumes that are attached to this instance are listed in the table.
5. To see details for a block volume, click the block volume name.

Using the OCI CLI

List block volume attachments for all instances in a compartment.

1. Get the OCID of the compartment where you want to list instance block volume attachments: (`oci iam compartment list`)
2. Run the volume attachment list command.

```
$ oci compute volume-attachment list \
--compartment-id oocid1.compartment.unique_ID
{
  "data": [
    {
      "attachment-type": "paravirtualized",
      "availability-domain": "AD-1",
      "compartment-id": "ocid1.compartment.unique_ID",
      "device": null,
      "display-name": "volumeattachment20210106172413",
      "id": "ocid1.volumeattachment.unique_ID",
      "instance-id": "ocid1.instance.unique_ID",
      "is-multipath": null,
      "is-pv-encryption-in-transit-enabled": null,
      "is-read-only": false,
      "is-shareable": false,
      "iscsi-login-state": null,
      "lifecycle-state": "ATTACHED",
      "time-created": "2021-06-01T17:24:13+00:00",
      "volume-id": "ocid1.volume.unique_ID"
    },
    {
      "attachment-type": "paravirtualized",
      "availability-domain": "AD-1",
      "compartment-id": "ocid1.compartment.unique_ID",
      "device": null,
      "display-name": "volumeattachment20210106175003",
      "id": "ocid1.volumeattachment.unique_ID",
      "instance-id": "ocid1.instance.unique_ID",
      "is-multipath": null,
      "is-pv-encryption-in-transit-enabled": null,
      "is-read-only": false,
      "is-shareable": false,
      "iscsi-login-state": null,
      "lifecycle-state": "ATTACHED",
      "time-created": "2021-06-01T17:50:03+00:00",
      "volume-id": "ocid1.volume.unique_ID"
    }
  ]
}
```

To list only block volume attachments for a specific instance, specify the instance OCID.

```
$ oci compute volume-attachment list --instance-id ocid1.instance.unique_ID
```

To list block volume attachments for a specific volume, specify the volume OCID in addition to either the compartment OCID or the instance OCID.

```
$ oci compute volume-attachment list \
--compartment-id oocid1.compartment.unique_ID \
--volume-id ocid1.volume.unique_ID
```

To list a specific block volume attachment, specify the volume attachment OCID.

```
$ oci compute volume-attachment get \
--volume-attachment-id ocid1.volumeattachment.unique_ID
```

Updating a Block Volume

You can change settings for a block volume while the volume is online, without any downtime. You can change the display name, increase the size, and change tags. For more information about resizing volumes, see Resizing Volumes.

To increase the volume size, see Resizing Volumes. To increase the size you must rescan the disk and extend the partition.

Using the Compute Web UI

1. In the navigation menu, under Block Storage, click Block Volumes.
2. Select the compartment where the block volume resides.
3. For the volume that you want to edit, click the Actions menu and then click Edit.
4. In the Edit Block Volume dialog, make the changes.
   - **Name:** The volume display name. The name does not have to be unique.
   - **Size (in GB):** You can increase the size in 1 GB increments up to 32768 (32 TB). You cannot decrease the size. Before you change the size, see Resizing Volumes to understand the effects of this change. To increase the size you must rescan the disk and extend the partition.
   - **Tagging:** Add, remove, or change tags. For details about tagging, see Working with Resource Tags.
5. Click Save Changes.

Using the OCI CLI

1. Get the OCID of the volume that you want to update: (`oci bv volume list`)
2. Run the volume update command.

   Syntax:

   ```
   oci bv volume update --volume-id volume_OCID options_with_values_to_update
   ```

   For descriptions of properties that you can change, enter the following command and scroll to Optional Parameters:

   ```
   $ oci bv volume update -h
   ```

   VPUs per GB cannot be changed after the volume is created. To update the `logBias` or `secondaryCache` values, see the instructions in Creating a Block Volume.

   Example:

   ```
   $ oci bv volume update --volume-id ocid1.volume.unique_ID \
   --display-name volumeA
   ```

Moving a Volume to a Different Compartment

You can move Block Volume resources such as block volumes, boot volumes, clones, volume backups, volume groups, and volume group backups from one compartment to another.

When you move a resource to a new compartment, associated resources might not be moved. For example, when you move a block volume, any backups of that volume are not moved.

> **Important:**
>
> Any access policies that exist on the new compartment apply immediately. Before you move resources to a different compartment, ensure that the resource users have sufficient access permissions on the compartment the resource is being moved to.

- You cannot move a block volume or boot volume from a security zone to a standard compartment.
- You cannot move a volume from a standard compartment to a compartment that is in a security zone if the volume violates any security zone policies.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Volume OCID (`oci bv volume list`)
   - Target Compartment OCID (`oci iam compartment list`)
2. Run the change compartment command.

   Example:

```
oci bv volume change-compartment \
--volume-id ocid1.volume.uniqueID \
--compartment-id ocid1.compartment.uniqueID
{
  "etag": "7e084c71-4729-4ddd-b131-d87bfc621e8c"
}
```

Cloning a Block Volume

Cloning a block volume enables you to make a copy of an existing block volume without performing the backup and restore operations.

A cloned volume is a point-in-time direct disk-to-disk deep copy of the source volume. All the data that is in the source volume is copied to the clone volume. Any subsequent changes to the data on the source volume are not copied to the clone.

By default, the clone is the same size as the source volume. You can specify a larger volume size when you create the clone.

The volume data is being copied in the background, and can take up to thirty minutes, depending on the size of the volume. You can attach and use the cloned volume as a regular volume when the state changes to Available.

For more information about cloning volumes, refer to "Volume Backups and Clones" in the Block Volume Storage Overview chapter in the *Oracle Private Cloud Appliance Concepts Guide*.

Using the Compute Web UI

1. In the navigation menu, under Block Storage, click Block Volumes.
2. Select the appropriate compartment.
3. For the volume you want to clone, click the Actions menu and then click Create Clone.
4. In the dialog, enter the following information:

- **Name:** A name or description for the volume. Avoid entering confidential information.
- **Compartment:** Select the compartment in which to clone the block volume.
- **Size (in GBs):** You can increase the size in 1 GB increments up to 32768 (32 TB). You cannot decrease the size.
- **High-Performance Volume:**(Optional) By default, the clone has the same performance setting as the source volume. Use this button to change the performance setting for this clone. For more information, see "Block Volume Performance Options" in the Block Volume Storage Overview chapter in the *Oracle Private Cloud Appliance Concepts Guide*.
- **Backup Policy:**(Optional) Select a backup policy from the drop-down list. You might need to change the compartment.

  Oracle defined policies are listed, as well as any user defined policies. For information about Oracle defined policies (`bronze`, `silver`, and `gold`), see "Volume Backups and Clones" in the Block Volume Storage Overview chapter in the *Oracle Private Cloud Appliance Concepts Guide*.

  Backup policies can be assigned or changed after the volume is cloned, or you can back up this volume manually. A volume can only have only one volume backup policy assigned at a time. For information about creating, editing, and assigning backup policies, see Managing Backup Policies. You can also back up this volume manually as described in Creating a Manual Boot or Block Volume Backup.
- **Tagging:**(Optional) Add defined or free-form tags for this instance as described in Adding Tags at Resource Creation. Tags can also be applied later.

5. Click Create Clone.

Using the OCI CLI

To clone a block volume, create a new block volume, specifying the volume that you want to clone as the source volume.

1. Gather the information that you need to run the command:
   - Compartment OCID that contains the source volume (`oci iam compartment list`)
   - Volume OCID of the volume to clone (`oci bv volume list`)
2. Run the volume create command, specifying the volume to clone.

   Syntax:

   ```
   oci bv volume create --availability-domain AD-1 \
   --compartment-id compartment_OCID \
   --source-volume-id OCID_of_volume_to_clone
   ```

   Example:

   ```
   $ oci bv volume create --availability-domain AD-1 -c ocid1.compartment.unique_ID \
   --source-volume-id ocid1.volume.unique_ID --display-name "MyVolumeClone"
   {
     "data": {
       "auto-tuned-vpus-per-gb": null,
       "autotune-policies": null,
       "availability-domain": "AD-1",
       "block-volume-replicas": null,
       "compartment-id": "ocid1.compartment.unique_ID",
       "defined-tags": {},
       "display-name": "MyVolumeClone",
       "freeform-tags": {},
   ```

```
    "id": "ocid1.volume.unique_ID",
    "is-auto-tune-enabled": null,
    "is-hydrated": null,
    "kms-key-id": null,
    "lifecycle-state": "PROVISIONING",
    "size-in-gbs": 51,
    "size-in-mbs": 52224,
    "source-details": {
      "id": "ocid1.volume.unique_ID",
      "type": "volume"
    },
    "system-tags": null,
    "time-created": "2023-06-07T17:34:34.234713+00:00",
    "volume-group-id": null,
    "vpus-per-gb": 20
  },
  "etag": "b9c13787-91a5-41ca-aa49-12e3ff80e97a",
  "opc-work-request-id": "ocid1.workrequest.unique_ID"
}
```


Detaching a Block Volume

When an instance no longer needs access to a volume, you can detach the volume from the instance without affecting the volume's data.

> **Caution:**
>
> If you later reattach the detached volume, the volume might be associated with a different device name and the instance operating system might not recognize the volume.

Using the Compute Web UI

1. Perform administrative tasks to remove any dependencies that any instances have for the block volume.

   For example, ensure that no applications are accessing the volume. Unmount the volume and remove it from the `/etc/fstab` file, and so on.
2. In the navigation menu, click Compute, then click Instances.
3. Select the compartment where the instance resides.
4. In the Instances list, click the instance that has the volume attached.
5. In the lower left corner, under Resources, select Attached Block Volumes.
6. Click the Actions icon (three dots) next to the volume you want to detach, and then click Detach.
7. Confirm when prompted.

Using the OCI CLI

1. Get the volume attachment OCID (`oci compute volume-attachment list`)
2. Run the detach command.

   Syntax:

```
oci compute volume-attachment detach --volume-attachment-id volume_attachment_OCID
```

Example:

```
oci compute volume-attachment detach --volume-attachment-id
ocid1.volumeattachment.uniqueID
Are you sure you want to delete this resource? [y/N]: y
```

To avoid the confirmation prompt, use the `--force` option.

Deleting a Block Volume

You can delete a volume that is no longer needed.

> **Caution:**
>
> You cannot undo this operation. Any data on a volume is permanently deleted when the volume is deleted.

> **Caution:**
>
> All policy-based (scheduled) backups expire. A manual backup expires if a scheduled backup of the same volume is created after the manual backup was created. To keep a volume backup indefinitely, cancel all future scheduled backups and create a manual backup before you delete the source volume. See Backing Up Block Volumes.

The result of deleting a block volume differs depending on the following conditions:

- **The volume has no backups or clones:** The volume is immediately deleted and the volume capacity is returned to the system for reuse. The volume is marked TERMINATED and eventually is no longer listed.
- **The volume has a backup or a clone:** The volume is marked TERMINATED, but the volume is not deleted and the capacity is not returned to the system until all of the backups and clones of the volume are deleted.
- **The volume is part of a DR configuration and replicating:** The volume is marked TERMINATED, but the volume is not deleted and the capacity is not returned to the system until DR completes the replication.

Using the Compute Web UI

1. Perform administrative tasks to remove any dependencies that any instances have for the block volume.

   For example, ensure that no applications are accessing the volume. Unmount the volume and remove it from the `/etc/fstab` file, and so on.
2. In the navigation menu, under Block Storage, click Block Volumes.
3. Select the compartment that contains the block volume that you plan to delete.
4. For the volume you plan to delete, click the Actions menu and then click Terminate.
5. Confirm the termination when prompted.

Using the OCI CLI

1. Get the volume OCID (`oci bv volume list`)
2. Run the volume delete command.

   Example:

```
$ oci bv volume delete --volume-id ocid1.volume.uniqueID
Are you sure you want to delete this resource? [y/N]: y
{
  "etag": "bd576de7-3193-4171-9792-uniqueID"
}
```

   To avoid the confirmation prompt, use the `--force` option.

Managing Boot Volumes

When you launch an instance, a new boot volume for the instance is created in the same compartment. That boot volume is associated with that instance until you terminate the instance. When you terminate the instance, you can preserve and reuse the boot volume and its data.

Boot volumes are encrypted by default.

For more conceptual information, refer to the Block Volume Storage Overview chapter in the Oracle Private Cloud Appliance Concepts Guide.

This section provides instructions for managing boot volumes.

Listing Boot Volumes

Using the Compute Web UI

1. In the navigation menu, under Block Storage, click Boot Volumes.
2. Select the appropriate compartment.

   A list of boot volumes is displayed.
3. To view details about a boot volume, click the boot volume name.

Using the OCI CLI

1. Get the OCID of the compartment where you want the list of boot volumes (`oci iam compartment list`)
2. Run the list boot volumes command.

```
$ oci bv boot-volume list --availability-domain AD-1 \
--compartment-id ocid1.compartment.unique_ID
{
  "data": [
    {
      "auto-tuned-vpus-per-gb": null,
      "autotune-policies": null,
      "availability-domain": "AD-1",
      "boot-volume-replicas": null,
      "compartment-id": "ocid1.compartment.unique_ID",
      "defined-tags": {},
      "display-name": "instance20231705214217(Boot Volume)",
```

```
    "freeform-tags": {},
    "id": "ocid1.bootvolume.unique_ID",
    "image-id": "ocid1.image.unique_ID",
    "is-auto-tune-enabled": null,
    "is-hydrated": null,
    "kms-key-id": null,
    "lifecycle-state": "AVAILABLE",
    "size-in-gbs": 50,
    "size-in-mbs": 51200,
    "source-details": null,
    "system-tags": null,
    "time-created": "2023-05-17T21:42:17+00:00",
    "volume-group-id": null,
    "vpus-per-gb": 0
  }
```


Listing Boot Volume Attachments

Using the Compute Web UI

1. In the navigation menu, click Compute, then click Instances.
2. Select the appropriate compartment.
3. Click the name of the instance for which you want to view the boot volume attachment.
4. Under Resources, click Attached Boot Volumes.

   The boot volume attachments are displayed.
5. To view the details about an attachment, click the boot volume attachment name.

Using the OCI CLI

1. Get the OCID of the compartment where you want the list of boot volume attachments (`oci iam compartment list`)
2. Run the list boot volume attachments command.


```
$ oci compute boot-volume-attachment list --availability-domain AD-1 \
--compartment-id ocid1.compartment.unique_ID
{
  "data": [
     {
      "availability-domain": "AD-1",
      "boot-volume-id": "ocid1.bootvolume.unique_ID",
      "compartment-id": "ocid1.compartment.unique_ID",
      "display-name": "bootvolumeattachment20211705214514",
      "encryption-in-transit-type": null,
      "id": "ocid1.bootvolumeattachment.unique_ID",
      "instance-id": "ocid1.instance.unique_ID",
      "is-pv-encryption-in-transit-enabled": null,
      "lifecycle-state": "ATTACHED",
      "time-created": "2023-05-17T21:45:14+00:00"
    }
```


Detaching a Boot Volume

If you think a boot volume issue is causing a compute instance problem, you can stop the instance and detach the boot volume using the steps described in this topic. Then you can attach the boot volume to another instance as a data volume to troubleshoot it.

Note– The instance must be in the Stopped state to detach a boot volume (described in this procedure). You must reattach a boot volume before you can start the instance. To stop and start the instance, see Stopping, Starting, and Resetting an Instance.

When you stop the instance, any application that is running on the instance will be immediately stopped, possibly resulting in data corruption. To avoid stopping the instance while applications are running, manually shut down the instance by using the commands available in the instance OS. After the instance is shut down from the OS, then stop the instance from the appliance.

Using the Compute Web UI

1. Stop the instance:
   a. Shut down the instance from the instance OS.
   b. On the appliance, in the navigation menu, click Compute and click Instances.
   c. Select the appropriate compartment.
   d. Click the name of the instance.
   e. On the instance details page, click Controls, then select Stop.

      Wait for the status to change from Stopping to Stopped. Stopping an instance can take up to five minutes.
2. Scroll to the Resources section and click Boot Volumes.
3. Click the boot volume Actions menu and then select Detach.
4. Confirm when prompted.

Using the OCI CLI

1. Gather the following information:
   - Instance OCID (`oci compute instance list`)
   - Boot volume attachment OCID (`oci compute boot-volume-attachment list`)
2. Stop the instance.
   a. Shut down the instance from the instance OS.
   b. Run the instance stop command.

      ```
      $ oci compute instance action --instance-id ocid1.instance.unique_ID --action
      STOP
      ```

      Wait for the status to change from `STOPPING` to `STOPPED`.

      ```
      $ oci compute instance get --instance-id ocid1.instance.unique_ID
      ```

      Stopping an instance can take up to five minutes.
3. Run the detach boot volume command.

   ```
   $ oci compute boot-volume-attachment detach \
   --boot-volume-attachment-id ocid1.bootvolumeattachment.unique_ID
   Are you sure you want to delete this resource? [y/N]: y
   ```

Reattaching a Boot Volume

A boot volume is automatically attached to an instance when the instance is launched.

Sometimes you need to detach and reattach a boot volume as a data volume for troubleshooting purposes. To detach a boot volume, see Detaching a Boot Volume. To attach the boot volume to another instance as a data volume, see Attaching a Volume.

This procedure describes how to reattach a boot volume as a boot volume.

Using the Compute Web UI

1. In the navigation menu, click Compute, then click Instances.
2. Select the compartment where the instance resides.
3. Click the instance name.
4. On the instance details page, scroll to the Resources section and click Boot Volumes.
5. Click the boot volume Actions menu and then click Attach.

   Confirm when prompted.

   Wait for the state of the boot volume to be Attached. Then you can restart the instance as described in Stopping, Starting, and Resetting an Instance.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Boot volume OCID (`oci bv boot-volume list`)
   - Instance OCID (`oci compute instance list`)
2. Run the attach boot volume command.

```
$ oci compute boot-volume-attachment attach \
--boot-volume-id ocid1.bootvolume.unique_ID \
--instance-id ocid1.instance.unique_ID
{
  "data": {
    "availability-domain": "AD-1",
    "boot-volume-id": "ocid1.bootvolume.unique_ID",
    "compartment-id": "ocid1.compartment.unique_ID",
    "display-name": "bootvolumeattachment20232405205526",
    "id": "ocid1.bootvolumeattachment.unique_ID",
    "instance-id": "ocid1.instance.unique_ID",
    "is-pv-encryption-in-transit-enabled": null,
    "lifecycle-state": "ATTACHED",
    "time-created": "2023-05-24T20:55:26+00:00"
  }
}
```

   When the lifecycle state is `ATTACHED`, you can restart the instance as described in Stopping, Starting, and Resetting an Instance.

Cloning a Boot Volume

Cloning a boot volume enables you to make a copy of an existing boot volume without performing the backup and restore operations.

For more information about cloned boot volumes, refer to "Cloning a Boot Volume" in the Compute Instance Concepts chapter in the *Oracle Private Cloud Appliance Concepts Guide*.

Using the Compute Web UI

1. In the navigation menu, under Block Storage, click Boot Volumes.

2. Select the appropriate compartment.
3. For the volume you want to clone, click the Actions menu and then click Create Clone.
4. In the dialog, enter the following information:
   - **Name:** A name or description for the volume. Avoid entering confidential information.
   - **Compartment:** Select the compartment in which to clone the block volume.
   - **High-Performance Volume:** (Optional) By default, the clone has the same performance setting as the source volume. Use this button to change the performance setting for this clone. For more information, see "Block Volume Performance Options" in the Block Volume Storage Overview chapter in the *Oracle Private Cloud Appliance Concepts Guide*.
   - **Tagging:** (Optional) Add defined or free-form tags for this instance as described in Adding Tags at Resource Creation. Tags can also be applied later.
5. Click the Create Clone button in the dialog.

   The new boot volume is ready to use when it reaches the Available state. For example, you could click the Actions menu and then click Create Instance to use this new boot volume to create a new instance.

Using the OCI CLI

To clone a boot volume, create a new boot volume, specifying the volume that you want to clone as the source volume.

1. Get the volume OCID of the boot volume to clone (`oci bv boot-volume list`)
2. Run the boot volume create command, specifying the boot volume to clone.

   Syntax:

   ```
   oci bv boot-volume create --source-boot-volume-id OCID_of_volume_to_clone
   ```

   Example:

   To create the clone in a compartment different from the compartment of the source, specify the compartment.

   ```
   $ oci bv boot-volume create --availability-domain AD-1 \
   --source-boot-volume-id ocid1.bootvolume.unique_ID
   {
     "data": {
       "auto-tuned-vpus-per-gb": null,
       "autotune-policies": null,
       "availability-domain": "AD-1",
       "boot-volume-replicas": null,
       "compartment-id": "ocid1.compartment.unique_ID",
       "defined-tags": {},
       "display-name": "instance20231306014509(Boot Volume)",
       "freeform-tags": {},
       "id": "ocid1.bootvolume.unique_ID",
       "image-id": "ocid1.image.unique_ID",
       "is-auto-tune-enabled": null,
       "is-hydrated": null,
       "kms-key-id": null,
       "lifecycle-state": "PROVISIONING",
       "size-in-gbs": 50,
       "size-in-mbs": 51200,
       "source-details": {
         "id": "ocid1.bootvolume.unique_ID",
   ```

```
      "type": "bootVolume"
    },
    "system-tags": null,
    "time-created": "2023-06-13T01:45:09.053390+00:00",
    "volume-group-id": null,
    "vpus-per-gb": 0
  },
  "etag": "ffe90d4d-355d-4293-b562-35a4a09c0a9a",
  "opc-work-request-id": "ocid1.workrequest.unique_ID"
}
```

Deleting a Boot Volume

When you terminate an instance, you choose to delete or preserve the associated boot volume. You can also delete a boot volume if it has been detached from the associated instance. See Detaching a Boot Volume.

> **Caution:**
>
> You cannot undo this operation. Any data on a volume is permanently deleted when the volume is deleted. You will not be able to restart the associated instance.

Using the Compute Web UI

1. In the navigation menu, under Block Storage, click Boot Volumes.
2. Select the appropriate compartment.
3. For the boot volume that you want to delete, click the Actions menu and then click Terminate.

   Confirm when prompted.

Using the OCI CLI

1. Get the boot volume OCID (`oci bv boot-volume list`)
2. Run the delete boot volume command.

```
$ oci bv boot-volume delete \
--boot-volume-id ocid1.bootvolume.unique_ID
Are you sure you want to delete this resource? [y/N]: y
```

Resizing Volumes

The Block Volume service lets you expand the size of block volumes and boot volumes. Use one of the following options to increase the size of a volume:

- Expand an existing volume in place with online resizing.
- Restore from a volume backup to a larger volume.
- Clone an existing volume to a new, larger volume.
- Expand an existing volume in place with offline resizing.

You cannot decrease the size of a volume.

For more conceptual information, refer to "Custom Boot Volume Sizes" in the Compute Instance Concepts chapter and the Block Volume Storage Overview chapter in the Oracle Private Cloud Appliance Concepts Guide

Online Volume Resizing

With online resizing, you can expand the volume size without detaching the volume from an instance. Online resizing requires you to rescan the disk and extend the partition.

Online Block Volume Resizing

Using the Compute Web UI

1. In the navigation menu, under Block Storage, click Block Volumes.
2. Select the appropriate compartment.
3. For the volume you plan to resize, click the Actions icon (three dots), and select Edit.
4. Change the size:
   - **Size (in GBs):** You can increase the size in 1 GB increments up to 32768 (32 TB). You cannot decrease the size.
5. Click Save Changes.
6. Rescan the disk.

   For details, consult the OS documentation for the OS type and version running in the instance.
7. Extend the partition.

   For details, consult the OS documentation for the OS type and version running in the instance.

Using the OCI CLI

1. Get the volume OCID (`oci bv volume list`)
2. Run the update block volume command.

   Syntax:

   ```
   oci bv volume update --volume-id volume_OCID --size-in-gbs size_in_GBs
   ```

   The ***size_in_GBs*** value is the size of the block volume. You can increase the size in 1 GB increments up to 32768 (32 TB). You cannot decrease the size.

   Example:

   ```
   oci bv volume update \
   --volume-id ocid1.volume.unique_ID --size-in-gbs 72
   {
     "data": {
       "auto-tuned-vpus-per-gb": null,
       "availability-domain": "AD-1",
       "compartment-id": "ocid1.compartment.unique_ID",
       "defined-tags": {},
       "display-name": "clone-w-policy",
       "freeform-tags": {},
       "id": "ocid1.volume.unique_ID",
       "is-auto-tune-enabled": null,
   ```

```
    "is-hydrated": null,
    "kms-key-id": null,
    "lifecycle-state": "PROVISIONING",
    "size-in-gbs": 72,
    "size-in-mbs": 71424,
    "source-details": {
      "id": "ocid1.volume.unique_ID",
      "type": "volume"
    },
    "system-tags": null,
    "time-created": "2021-07-02T20:48:20+00:00",
    "volume-group-id": null,
    "vpus-per-gb": 0
  },
  "etag": "58851b71-236d-4d99-8175-b27835d6b34f"
}
```


3. Rescan the disk.

   For details, consult the OS documentation for the OS type and version running in the instance.

4. Extend the partition.

   For details, consult the OS documentation for the OS type and version running in the instance.

Online Boot Volume Resizing

Using theCompute Web UI

1. In the navigation menu, under Block Storage, click Boot Volumes.
2. Select the appropriate compartment.
3. For the volume you plan to resize, click the Actions icon (three dots), and select Edit.
4. Change the size:
   - **Size (in GBs):** You can increase the size in 1 GB increments up to 16384 (16 TB). You cannot decrease the size.
5. Click Save Changes.
6. Rescan the disk.

   For details, consult the OS documentation for the OS type and version running in the instance.

7. Extend the partition.

   For details, consult the OS documentation for the OS type and version running in the instance.

Using the OCI CLI

1. Get the boot volume OCID (`oci bv boot-volume list`)
2. Run the update boot volume command.

   Syntax:

   ```
   oci bv boot-volume update --boot-volume-id volume_OCID --size-in-gbs size_in_GBs
   ```

The ***size_in_GBs*** value is the size of the boot volume. You can increase the size in 1 GB increments up to 16384 (16 TB). You cannot decrease the size.

If you specify a size that is smaller than the volume's current size, your request will be ignored.

Example:

```
oci bv boot-volume update \
--boot-volume-id ocid1.bootvolume.unique_ID --size-in-gbs 1024
{
  "data": {
    "auto-tuned-vpus-per-gb": null,
    "availability-domain": "AD-1",
    "compartment-id": "ocid1.tenancy.unique_ID",
    "defined-tags": {},
    "display-name": "MyInstance(Boot Volume)",
    "freeform-tags": {},
    "id": "ocid1.bootvolume.unique_ID",
    "image-id": "ocid1.image.unique_ID",
    "is-auto-tune-enabled": null,
    "is-hydrated": null,
    "kms-key-id": null,
    "lifecycle-state": "PROVISIONING",
    "size-in-gbs": 1024,
    "source-details": null,
    "system-tags": null,
    "time-created": "2021-08-10T20:14:03.053300+00:00",
    "volume-group-id": null,
    "vpus-per-gb": 0
  },
  "etag": "bd0677e3-c542-45f3-bf04-c473b184c795"
}
```

3. Rescan the disk.

   For details, consult the OS documentation for the OS type and version running in the instance.

4. Extend the partition.

   For details, consult the OS documentation for the OS type and version running in the instance.

Offline Volume Resizing

With offline resizing, you detach the volume from an instance before you expand the volume size. When the volume is resized and reattached, you need to extend the partition, but you do not need to rescan the disk.

Considerations When Resizing an Offline Volume

Whenever you detach and reattach volumes, there are complexities and risks for both UNIX and Microsoft Windows instances. Keep the following points in mind when resizing volumes:

- Before you resize a volume, create a full backup of the volume.
- When you reattach a volume to an instance after resizing, if you are not using consistent device paths, or if the instance does not support consistent device paths, device order and path might change. If you are using a tool such as Logical Volume Manager (LVM), you might need to fix the device mappings.

Offline Block Volume Resizing

Using the Compute Web UI

1. Detach the block volume.

   See Detaching a Block Volume.
2. In the navigation menu, click Block Storage, then click Block Volumes.
3. Select the appropriate compartment.
4. For the volume you plan to resize, click the Actions icon (three dots), and select Edit.
5. Change the size:
   - **Size (in GBs):** You can increase the size in 1 GB increments up to 32768 (32 TB). You cannot decrease the size.
6. Click Save Changes.
7. Reattach the volume.

   See Attaching a Volume
8. Extend the partition.

   For details, consult the OS documentation for the OS type and version running in the instance.

Using the OCI CLI

1. Detach the block volume.

   See Detaching a Block Volume.
2. Get the volume OCID (`oci bv volume list`)
3. Run the volume update command.

   Syntax:

   ```
   oci bv volume update --volume-id volume_OCID --size-in-gbs size_in_GBs
   ```

   The ***size_in_GBs*** value is the size of the block volume. You can increase the size in 1 GB increments up to 32768. You cannot decrease the size.

   Example:

   ```
   oci bv volume update   \
   --volume-id ocid1.volume.unique_ID --size-in-gbs 72
   {
     "data": {
       "auto-tuned-vpus-per-gb": null,
       "availability-domain": "AD-1",
       "compartment-id": "ocid1.compartment.unique_ID",
       "defined-tags": {},
       "display-name": "clone-w-policy",
       "freeform-tags": {},
       "id": "ocid1.volume.unique_ID",
       "is-auto-tune-enabled": null,
       "is-hydrated": null,
       "kms-key-id": null,
       "lifecycle-state": "PROVISIONING",
   ```

```
    "size-in-gbs": 72,
    "size-in-mbs": 71424,
    "source-details": {
      "id": "ocid1.volume.unique_ID",
      "type": "volume"
    },
    "system-tags": null,
    "time-created": "2021-07-02T20:48:20+00:00",
    "volume-group-id": null,
    "vpus-per-gb": 0
  },
  "etag": "58851b71-236d-4d99-8175-b27835d6b34f"
}
```

4. Reattach the volume.

   See Attaching a Volume.

5. Extend the partition.

   For details, consult the OS documentation for the OS type and version running in the instance.

Offline Boot Volume Resizing

Using the Compute Web UI

1. Stop the instance.

   See Stopping, Starting, and Resetting an Instance.

2. Detach the boot volume.

   See Detaching a Boot Volume.

3. In the navigation menu, under Block Storage, click Boot Volumes.
4. Select the appropriate compartment.
5. For the volume you plan to resize, click the Actions icon (three dots), and select Edit.
6. Change the size:
   - **Size (in GBs):** You can increase the size in 1 GB increments up to 16384 (16 TB). You cannot decrease the size.
7. Click Save Changes.
8. Attach the boot volume to a second instance as a data volume.

   See Attaching a Volume.

9. Extend the partition and grow the file system.

   For details, consult the OS documentation for the OS type and version running in the instance.

10. Detach the data volume.

    See Detaching a Block Volume.

11. Reattach the boot volume.

    See Reattaching a Boot Volume.

12. Restart the instance.

    See Stopping, Starting, and Resetting an Instance.

Using the OCI CLI

1. Stop the instance.

   See Stopping, Starting, and Resetting an Instance.

2. Detach the boot volume.

   See Detaching a Boot Volume.

3. Get the boot volume OCID (`oci bv boot-volume list`)

4. Run the boot volume update command.

   Syntax:

   ```
   oci bv boot-volume update --boot-volume-id volume_OCID --size-in-gbs size_in_GBs
   ```

   The ***size_in_GBs*** value is the size of the boot volume. You can increase the size in 1 GB increments up to 16384 (16 TB). You cannot decrease the size.

   If you specify a size that is smaller than the volume's current size, your request will be ignored.

   Example:

   ```
   oci bv boot-volume update \
   --boot-volume-id ocid1.bootvolume.unique_ID --size-in-gbs 1024
   {
     "data": {
       "auto-tuned-vpus-per-gb": null,
       "availability-domain": "AD-1",
       "compartment-id": "ocid1.tenancy.unique_ID",
       "defined-tags": {},
       "display-name": "MyInstance(Boot Volume)",
       "freeform-tags": {},
       "id": "ocid1.bootvolume.unique_ID",
       "image-id": "ocid1.image.unique_ID",
       "is-auto-tune-enabled": null,
       "is-hydrated": null,
       "kms-key-id": null,
       "lifecycle-state": "PROVISIONING",
       "size-in-gbs": 1024,
       "source-details": null,
       "system-tags": null,
       "time-created": "2021-08-10T20:14:03.053300+00:00",
       "volume-group-id": null,
       "vpus-per-gb": 0
     },
     "etag": "bd0677e3-c542-45f3-bf04-c473b184c795"
   }
   ```

5. Attach the boot volume to a second instance as a data volume.

   See Attaching a Volume.

6. Extend the partition and grow the file system.

   For details, consult the OS documentation for the OS type and version running in the instance.

7. Detach the data volume.

   See Detaching a Block Volume.

8. Reattach the boot volume.

See Reattaching a Boot Volume.

9. Restart the instance.

   See Stopping, Starting, and Resetting an Instance.

Managing Volume Groups

the Block Volume service enables you to organize multiple volumes into a volume group. A volume group can include both block and boot volumes.

You can use volume groups to create volume group backups and clones that are point-in-time and crash-consistent. This simplifies the process to create time-consistent backups of running enterprise applications that span multiple storage volumes across multiple instances. You can then restore an entire group of volumes from a volume group backup.

Similarly, you can also clone an entire volume group in a time-consistent and crash-consistent manner. A deep disk-to-disk and fully isolated clone of a volume group, with all the volumes associated in it, becomes available for use within a matter of seconds. This speeds up the process of creating new environments for development, quality assurance, user acceptance testing, and troubleshooting.

When working with volume groups and volume group backups, keep the following in mind:

- You can only add a volume to a volume group when the volume status is Available.
- A volume group can include up to 32 volumes, up to a maximum size of 128 TB. For example, if you wanted to add 32 volumes of equal size to a volume group, the maximum size for each volume would be 4 TB. You could add volumes that vary in size, however the overall combined size of all the block and boot volumes in the volume group cannot be more than 128 TB. Make sure you account for the size of any boot volumes in your volume group when considering volume group size limits.
- A volume can only be in one volume group.
- When you clone a volume group, a new group with new volumes is created. For example, if you clone a volume group containing three volumes, once this operation is complete, you will now have two separate volume groups and six different volumes with nothing shared between the volume groups.
- When you update a volume group using the CLI, SDKs, or REST APIs, specify all the volumes to include in the volume group. The list of volumes to include replaces the existing list. If you do not include a volume OCID in the updated list, that volume will be removed from the volume group.
- When you delete a volume group, the individual volumes in the group are not deleted.
- When you delete a volume that is part of a volume group, you must first remove the volume from the volume group and then delete the volume.
- When you delete a volume group backup, all the volume backups in the volume group backup are deleted.

Viewing the Volumes in a Volume Group

Using the Compute Web UI

1. In the navigation menu, under Block Storage, click Volume Groups.
2. Select the appropriate compartment.
3. In the Volume Groups list, click the volume group you want to view.

4. To view the block volumes for the volume group, in Resources, click Volumes.

Using the OCI CLI

1. Get the OCID of the compartment where you want to list volume groups: (`oci iam compartment list`)
2. Run the volume group list command.

   Example:

```
oci bv volume-group list --compartment-id ocid1.compartment.uniqueID
{
  "data": [
    {
      "availability-domain": "AD-1",
      "compartment-id": "ocid1.compartment.uniqueID",
      "defined-tags": {},
      "display-name": "myVolumeGroup",
      "freeform-tags": {},
      "id": "ocid1.volumeGroup.uniqueID",
      "is-hydrated": null,
      "lifecycle-state": "AVAILABLE",
      "size-in-gbs": 150,
      "size-in-mbs": 153600,
      "source-details": {
        "type": "volumeIds",
        "volume-ids": [
          "ocid1.volume.uniqueID-1",
          "ocid1.volume.uniqueID-2",
          "ocid1.volume.uniqueID-3"
        ]
      },
      "time-created": "2023-05-26T20:47:06+00:00",
      "volume-ids": [
          "ocid1.volume.uniqueID-1",
          "ocid1.volume.uniqueID-2",
          "ocid1.volume.uniqueID-3"
      ]
    },
    {
      "availability-domain": "AD-1",
      "compartment-id": "ocid1.compartment.uniqueID",
      "defined-tags": {},
      "display-name": "anotherVolumeGroup",
      "freeform-tags": {},
      "id": "ocid1.volumeGroup.uniqueID",
      "is-hydrated": null,
      "lifecycle-state": "AVAILABLE",
      "size-in-gbs": 100,
      "size-in-mbs": 102400,
      "source-details": {
        "type": "volumeIds",
        "volume-ids": [
          "ocid1.volume.uniqueID-4",
          "ocid1.volume.uniqueID-5"
        ]
      },
      "time-created": "2021-05-25T19:08:55+00:00",
      "volume-ids": [
          "ocid1.volume.uniqueID-4",
          "ocid1.volume.uniqueID-5"
      ]
```

```
    }
  ]
}
```

Creating a Volume Group

Using the Compute Web UI

1. In the navigation menu, under Block Storage, click Volume Groups.
2. Select the compartment where you want the volume group created.
3. Click Create Volume Group.
4. In the dialog, provide the following information:
   - **Name**: A user-friendly name or description for the volume group.
   - **Compartment**: The compartment for the volume group.
   - **Volumes**: Select a volume to add to the group from the volume drop-down list. You might need to select a different compartment above the volume drop-down list. Click **+ Add Volume** to add another volume.
   - **Backup Policy:** (Optional) Select a backup policy from the drop-down list. You might need to change the compartment.
5. **Tagging:**(Optional) Add defined or free-form tags for this instance as described in Adding Tags at Resource Creation. Tags can also be applied later.
6. Click Create Volume Group.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Compartment OCID (`oci iam compartment list`)
   - The OCID of each volume that you want to add to the volume group (`oci bv volume list`)
2. Construct an argument for the `--source-details` option.

   The `--source-details` option provides a list of the volumes to include in the group. The following example file named `group_volumes.json` shows the format of the list. Specify this content as the argument for the `--source-details` option, either as a string or as a `file://` argument.

   ```
   {
     "type": "volumeIds",
     "volumeIds": [
       "ocid1.volume.unique_ID_1",
       "ocid1.volume.unique_ID_2",
       "ocid1.bootvolume.unique_ID_3"
     ]
   }
   ```
3. Run the volume group create command.

   Syntax:

   ```
   oci bv volume-group create --availability-domain AD-1 \
   --compartment-id compartment_OCID
   --source-details JSON_list_of_volumes_to_include
   ```

Example:

```
oci bv volume-group create --availability-domain AD-1 \
--compartment-id ocid1.compartment.unique_ID \
--source-details file:///group_volumes.json
{
  "data": {
    "availability-domain": "AD-1",
    "compartment-id": "ocid1.compartment.unique_ID",
    "defined-tags": {},
    "display-name": "volumegroup20232605212205",
    "freeform-tags": {},
    "id": "ocid1.volumeGroup.unique_ID",
    "is-hydrated": null,
    "lifecycle-state": "PROVISIONING",
    "size-in-gbs": 100,
    "size-in-mbs": 102400,
    "source-details": {
      "type": "volumeIds",
      "volume-ids": [
        "ocid1.volume.unique_ID_1",
        "ocid1.volume.unique_ID_2",
        "ocid1.bootvolume.unique_ID_3"
      ]
    },
    "time-created": "2023-05-26T21:22:05+00:00",
    "volume-group-replicas": null,
    "volume-ids": [
      "ocid1.volume.unique_ID_1",
      "ocid1.volume.unique_ID_2",
      "ocid1.bootvolume.unique_ID_3"
    ]
  },
  "etag": "c7053513-6819-49ad-8785-dd3e2a45272a"
}
```

Adding Volumes to a Group

> **Note:**
>
> You cannot add a volume with an existing backup policy assignment to a volume group with a backup policy assignment. You must first remove the backup policy assignment from the volume before you can add it to the volume group.

Using the Compute Web UI

1. In the navigation menu, under Block Storage, click Volume Groups.
2. Select the compartment that contains the volume group.
3. In the Volume Groups list, click the volume group you want to add the volume to.
4. On the details page for the volume group, scroll to the Resources section, click Volumes, and click the Add Volumes button.
5. Select a volume to add to the group from the volume drop-down list. You might need to select a different compartment above the volume drop-down list. Click **+ Add Volume** to add another volume.

6. Click Update Volume Group.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Volume group OCID (`oci bv volume-group list`)
   - OCID of each volume that you want to add to the volume group (`oci bv volume list`)

     For details about the JSON format, run this command:

     ```
     oci bv volume-group update --generate-param-json-input volume-ids
     ```
2. Construct an argument for the `--volume-ids` option.

   The `--volume-ids` option provides a list of the volumes to add to the group.

   The `--volume-ids` argument must contain the full list of volumes to be included in the group. If a volume that is currently in the group is omitted from the `--volume-ids` argument, that volume will be removed from the group.

   The following example file named `group_volumes-2.json` shows the format of the list. Specify this content either as a string or as a `file://` argument.

   ```
   [
     "ocid1.volume.unique_ID_1",
     "ocid1.volume.unique_ID_2",
     "ocid1.bootvolume.unique_ID_3",
     "ocid1.volume.unique_ID_4"
   ]
   ```
3. Run the volume group update command.

   Syntax:

   ```
   oci bv volume-group update --volume-group-id volume_group_OCID \
   --volume-ids full_list_of_volumes_to_include
   ```

   Example:

   In this example, the first two volume IDs are already in the volume group, and the third volume ID is added to the group.

   ```
   oci bv volume-group update --volume-group-id ocid1.volumeGroup.unique_ID \
   --volume-ids file:///group_volumes-2.json
   {
     "data": {
       "availability-domain": "AD-1",
       "compartment-id": "ocid1.compartment.unique_ID",
       "defined-tags": {},
       "display-name": "volumegroup20232605212205",
       "freeform-tags": {},
       "id": "ocid1.volumeGroup.unique_ID",
       "is-hydrated": null,
       "lifecycle-state": "PROVISIONING",
       "size-in-gbs": 100,
       "size-in-mbs": 102400,
       "source-details": {
         "type": "volumeIds",
         "volume-ids": [
           "ocid1.volume.unique_ID_1",
           "ocid1.volume.unique_ID_2",
           "ocid1.bootvolume.unique_ID_3",
           "ocid1.volume.unique_ID_4"
         ]
   ```

```
    },
    "time-created": "2023-05-26T21:22:05+00:00",
    "volume-group-replicas": null,
    "volume-ids": [
      "ocid1.volume.unique_ID_1",
      "ocid1.volume.unique_ID_2",
      "ocid1.bootvolume.unique_ID_3",
      "ocid1.volume.unique_ID_4"
    ]
  },
  "etag": "eeb63423-1afa-43f1-9e3a-8d8e8d803ebc"
}
```


Removing Volumes from a Group

When you remove the last volume in a volume group, the volume group is deleted.

Using the Compute Web UI

1. In the navigation menu, under Block Storage, click Volume Groups.
2. Select the appropriate compartment.
3. In the Volume Groups list, click the volume group that contains the volume you plan to remove.
4. On the details page for the volume group, scroll to the Resources section and click Volumes. For the volume that you want to remove, click the Actions menu, and then click Remove.
5. Confirm the removal.

Using the OCI CLI

To remove a volume from a volume group, use the `oci bv volume-group update` command, and remove the volume from the list of volumes in the argument for the `--volume-ids` option as described in Adding Volumes to a Group.

Cloning a Volume Group

Cloning a volume group enables you to make a copy of a volume group without performing the backup and restore operations.

A cloned volume group is a point-in-time direct disk-to-disk deep copy of the source volume group, so all the data that is in the source volume group is copied to the clone volume group.

Any subsequent changes to the data on the source volume group are not copied to the clone.

For additional details about clones and how they differ from backups, refer to the "Volume Backups and Clones" section in the Block Volume Storage Overview chapter in the Oracle Private Cloud Appliance Concepts Guide.

Using the Compute Web UI

1. In the navigation menu, under Block Storage, click Volume Groups.
2. Select the appropriate compartment.
3. Click the name of the Volume Group you plan to clone.
4. On the details page for the volume group, scroll to the Resources section and click Volume Group Clones.

5. Click the Create Volume Group Clone button.
6. In the dialog, enter the following information:
   - **Volume Group Clone Name:** Enter a descriptive name for the clone.
   - **Create in Compartment:** Select the compartment where the clone will be created.
7. Click the Create Volume Group Clone button in the dialog.

Using the OCI CLI

To clone a volume group, create a new volume group, specifying the same set of member volumes as the volume group that you want to clone.

In the following example, `group_volumes.json` is the same list of volumes as are in the volume group that you are cloning.

```
$ oci bv volume-group create --availability-domain AD-1 \
--compartment-id ocid1.compartment.unique_ID \
--source-details file:///group_volumes.json
```

Deleting a Volume Group

When you delete a volume group, the individual volumes in the group are not deleted. Only the volume group is deleted.

Using the Compute Web UI

1. In the navigation menu, under Block Storage, click Volume Groups.
2. Select the appropriate compartment.
3. In the Volume Groups list, click the volume group you want to delete.
4. On the details page for the volume group, click Terminate.
5. Confirm the termination.

Using the OCI CLI

1. Get the OCID of the volume group (`oci bv volume-group list`)
2. Run the volume group delete command.

   Example:

   ```
   oci bv volume-group delete --volume-group-id ocid1.volumegroup.unique_ID
   ```

Backing Up Block Volumes

The backups feature for the Block Volume service lets you make a point-in-time snapshot of the data on a block volume. You can make a backup of a volume when the volume is attached to an instance or while it is detached. These backups can then be restored to new volumes any time.

There are two ways to initiate a backup:

- **Manual Backups:**These are full backups that you can launch immediately. See Creating a Manual Boot or Block Volume Backup.
- **Policy-Based Backups:**These are automated scheduled backups defined by the backup policy assigned to the volume. See Managing Backup Policies.

For more information about block volume backups, such as the differences between backups and clones, refer to "Volume Backups and Clones" in the Block Volume Storage Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

Viewing Volume Backups

Backups that are created by backup policies (also called automatic or scheduled backups) can take up to five minutes to show in the backups list either in the Compute Web UI or in OCI CLI list output.

Using the Compute Web UI

1. In the navigation menu, under Block Storage, click Block Volumes, Boot Volumes, or Volume Groups.

   Both automatic and manual backups are listed.
2. If you don't see your backup listed, ensure you are viewing the correct compartment, which is displayed above the list.
3. To view the details of a backup, click the backup name.

Using the OCI CLI

1. Get the compartment OCID (`oci iam compartment list`)
2. Run the list backups command.

   A large number of volume backups, for example hundreds of backups, can take a long time to list. Like most resource list commands, the backup list commands support pagination of results. Use the `--limit` option to specify the number of backups to show for each invocation of the list command. To show the next page of results, use the value of the `opc-next-page` property with the `--page` option. Re-specify the `--limit` option for each new page command; otherwise, all remaining resources will be listed. When no `opc-next-page` property is shown, all results have been listed.

   By default, the list is shown in ascending order by display name, and the display name is case sensitive. You can change the list order to try to move specific resources closer to the top of the list. You can sort by (`--sort-by`) `displayname` or `timecreated` and set the sort order (`--sort-order`) to either `asc` (ascending) or `desc` (descending). The default sort order for `timecreated` is descending.

   Syntax:

   ```
   oci bv backup list --compartment-id compartment_OCID
   oci bv boot-volume-backup list --compartment-id compartment_OCID
   oci bv volume-group-backup list --compartment-id compartment_OCID
   ```

   Both `SCHEDULED` and `MANUAL` backups are listed.

   Example:

   This example lists block volume backups.

   ```
   $ oci bv backup list --compartment-id ocid1.unique_ID --limit 100 \
   --sort-by timecreated
   {
     "data": [
       {
         "compartment-id": "ocid1.compartment.unique_ID",
         "defined-tags": {},
         "display-name": ".auto-20230607T160000UTC",
   ```

```
      "expiration-time": null,
      "freeform-tags": {},
      "id": "ocid1.volumebackup.unique_ID",
      "kms-key-id": null,
      "lifecycle-state": "AVAILABLE",
      "size-in-gbs": 0,
      "size-in-mbs": null,
      "source-type": "SCHEDULED",
      "source-volume-backup-id": null,
      "system-tags": null,
      "time-created": "2023-06-07T16:00:00.000001+00:00",
      "time-request-received": "2023-06-07T16:03:30.000001+00:00",
      "type": "FULL",
      "unique-size-in-gbs": 0,
      "unique-size-in-mbs": null,
      "volume-id": "ocid1.volume.unique_ID"
    },
    ...
  ],
  "opc-next-page": "NC0y"
}
```


To show the next page of results, use the following command:

```
$ oci bv backup list --compartment-id ocid1.unique_ID --limit 100 \
--sort-by timecreated --page "NC0y"
```

Creating a Manual Boot or Block Volume Backup

This procedure describes how to manually back up either a block volume or a boot volume. To create scheduled backups, see Managing Backup Policies.

Using the Compute Web UI

1. In the navigation menu, under Block Storage, click either Block Volumes or Boot Volumes.
2. If the volume that you want to back up is not listed, use the Compartment drop-down menu above the volume list to select the correct compartment.
3. For the volume you plan to back up, click the Actions menu, and click Create Manual Backup.
4. In the dialog, enter this information:
   - **Name:** Enter a descriptive name for the backup.
   - **Compartment:** For a boot volume backup, select the compartment where you want this backup created.
   - **Tagging:**(Optional) Add defined or free-form tags for this backup as described in Adding Tags at Resource Creation. Tags can also be applied later.
5. For a block volume backup, click Create Manual Backup. For a boot volume backup, click Create.

Using the OCI CLI

- **Creating a Block Volume Backup**
  1. Get the OCID of the block volume that you want to back up (`oci bv volume list`)
     - Volume OCID (`oci bv volume list`)
  2. Run the create block volume backup command.

```
$ oci bv backup create --volume-id ocid1.volume.unique_ID
{
  "data": {
    "compartment-id": "ocid1.compartment.unique_ID",
    "defined-tags": {},
    "display-name": "volumebackup20230806080408",
    "expiration-time": null,
    "freeform-tags": {},
    "id": "ocid1.volumeBackup.unique_ID",
    "kms-key-id": null,
    "lifecycle-state": "CREATING",
    "size-in-gbs": null,
    "size-in-mbs": null,
    "source-type": "MANUAL",
    "source-volume-backup-id": null,
    "system-tags": null,
    "time-created": "2023-06-08T08:04:08.430090+00:00",
    "time-request-received": "2023-06-08T08:04:08.000001+00:00",
    "type": "FULL",
    "unique-size-in-gbs": null,
    "unique-size-in-mbs": null,
    "volume-id": "ocid1.volume.unique_ID"
  },
  "etag": "616112e8-728c-43d6-b0d1-c6cfcc1a46e6"
}
```

- **Creating a Boot Volume Backup**

  1. Get the OCID of the boot volume that you want to back up (`oci bv boot-volume list`)
  2. Run the create boot volume backup command.

```
$ oci bv boot-volume-backup create
--boot-volume-id ocid1.bootvolume.unique_ID
{
  "data": {
    "boot-volume-id": "ocid1.bootvolume.unique_ID",
    "compartment-id": "ocid1.compartment.unique_ID",
    "defined-tags": {},
    "display-name": "bootvolumebackup20230806082217",
    "expiration-time": null,
    "freeform-tags": {},
    "id": "ocid1.bootvolumebackup.unique_ID",
    "image-id": "ocid1.image.unique_ID",
    "kms-key-id": null,
    "lifecycle-state": "CREATING",
    "size-in-gbs": null,
    "source-boot-volume-backup-id": null,
    "source-type": "MANUAL",
    "system-tags": null,
    "time-created": "2023-06-08T08:22:17.011857+00:00",
    "time-request-received": "2023-06-08T08:22:16.000001+00:00",
    "type": "FULL",
    "unique-size-in-gbs": null
  },
  "etag": "123a12b3-daa8-4557-8c83-uniqueID",
  "opc-work-request-id": "ocid1.workrequest.unique_ID"
}
```

Creating a Manual Backup of a Volume Group

This procedure describes how to manually back up a volume group. To create scheduled backups, see Managing Backup Policies.

Using the Compute Web UI

1. In the navigation menu, click Block Storage, then click Volume Groups.
2. Select the appropriate compartment.
3. For the volume group you want to back up, click the Actions menu and then click Create Volume Group Backup.
4. In the dialog, enter a descriptive name for the backup.
5. Click Create Volume Group Backup.

Using the OCI CLI

1. Get the OCID of the volume group (`oci bv volume-group list`)
2. Run the create volume group backup command.

```
$ oci bv volume-group-backup create \
--volume-group-id ocid1.volumegroup.unique_ID
{
  "data": {
    "compartment-id": "cid1.compartment.unique_ID",
    "defined-tags": {},
    "display-name": "volumegroupbackup20230806085452",
    "expiration-time": null,
    "freeform-tags": {},
    "id": "ocid1.volumegroupbackup.unique_ID",
    "lifecycle-state": "CREATING",
    "size-in-gbs": null,
    "size-in-mbs": null,
    "source-type": "MANUAL",
    "source-volume-group-backup-id": null,
    "time-created": "2023-06-08T08:54:52.286092+00:00",
    "time-request-received": "2023-06-08T08:54:51.000001+00:00",
    "type": "FULL",
    "unique-size-in-gbs": null,
    "unique-size-in-mbs": null,
    "volume-backup-ids": [],
    "volume-group-id": "ocid1.volumegroup.unique_ID"
  },
  "etag": "04761386-6ec5-4cfa-b88e-a085ad833eac",
  "opc-work-request-id": "ocid1.workrequest.unique_ID"
}
```

Restoring a Backup to a New Volume

Using the Compute Web UI

1. In the navigation menu, click Block Storage, then click Block Volume Backups.
2. If you don't see your backup listed, ensure you are viewing the correct compartment which is displayed at the top of the page.

3. For the block volume backup that you want to restore, click the Actions menu and click Restore Block Volume.
4. In the Create Block Volume dialog, provide the following information:
   - **Name:** A name or description for the volume. Avoid entering confidential information.
   - **Compartment:**Select the compartment in which to restore the block volume.
   - **Size (in GBs):** To change the size, enter a value from 50 to 32768 (50 GB to 32 TB) in 1 GB increments. You cannot enter a smaller value than the value that is shown.
   - **High Performance Volume:**(Optional) By default, the volume uses balanced performance. To create a block volume that uses the high performance feature, click the Enable High Performance button. For more information, see "Block Volume Performance Options" in the Block Volume Storage Overview chapter in the *Oracle Private Cloud Appliance Concepts Guide*.

     This selection cannot be changed after the volume is created.
   - **Backup Policy:**(Optional) Select a backup policy from the drop-down list. You might need to change the compartment.

     For more information about backup policies, see Managing Backup Policies.
   - **Tagging:**(Optional) Add defined or free-form tags for this volume as described in Adding Tags at Resource Creation. Tags can also be applied later.
5. Click Create Block Volume.

   The volume is ready to attach after its icon no longer lists it as PROVISIONING in the volume list.

Using the OCI CLI

1. Get the OCID of the volume backup (`oci bv backup list`)
2. Run the restore volume backup command.

   Syntax:

   ```
   oci bv volume create --availability-domain AD-1 \
   --volume-backup-id source_volume_backup_OCID
   ```

   The volume backup is the source of the data that will be restored to the newly created volume.

   Example:

   ```
   $ oci bv volume create --availability-domain AD-1 \
   --volume-backup-id ocid1.volumebackup.unique_ID
   {
     "data": {
       "auto-tuned-vpus-per-gb": null,
       "autotune-policies": null,
       "availability-domain": "AD-1",
       "compartment-id": "ocid1.compartment.unique_ID",
       "defined-tags": {},
       "display-name": "volume20230806093839",
       "freeform-tags": {},
       "id": "ocid1.volume.unique_ID",
       "is-auto-tune-enabled": null,
       "is-hydrated": null,
       "kms-key-id": null,
       "lifecycle-state": "PROVISIONING",
       "size-in-gbs": 50,
   ```

```
    "size-in-mbs": 51200,
    "source-details": {
      "id": "ocid1.volumebackup.unique_ID",
      "type": "volumeBackup"
    },
    "system-tags": null,
    "time-created": "2023-06-08T09:38:39.036730+00:00",
    "volume-group-id": null,
    "vpus-per-gb": 10
  },
  "etag": "13864f86-cd1c-49f7-b414-4c4800103b0c",
  "opc-work-request-id": "ocid1.workrequest.unique_ID"
}
```

Restoring a Volume Group from a Volume Group Backup

Using the Compute Web UI

1. In the navigation menu, under Block Storage, click Volume Group Backups.
2. Select the appropriate compartment.
3. For the volume group backup that you want to restore, click the Actions menu and click Restore Volume Group.
4. In the Create Volume Group dialog, provide the following information:
   - **Name**: Enter a descriptive name for the group.
   - **Compartment**: Select the compartment for the volume group.
   - **Backup Policy:** (Optional) Select a backup policy from the drop-down list. You might need to change the compartment.

     For more information about backup policies, see Managing Backup Policies.
   - **Tagging:**(Optional) Add defined or free-form tags for this volume as described in Adding Tags at Resource Creation. Tags can also be applied later.
5. Click Create Volume Group.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Compartment OCID (`oci iam compartment list`)
   - The OCID of the volume group backup that you want to restore to a new volume group (`oci bv volume-group list`)
2. Run the create volume group command.

   To restore a volume group backup to a new volume group, use the `oci bv volume-group create` command, specifying the volume group backup as the source.

   Syntax:

   ```
   oci bv volume-group create \
   --availability-domain AD-1 \
   --compartment-id compartment_OCID \
   --source-details volume_group_backup
   ```

   Example:

   ```
   $ oci bv volume-group create --availability-domain AD-1 \
   --compartment-id ocid1.compartment.unique_ID \
   ```

```
--source-details '{"type": "volumeGroupBackupId", "volumeGroupBackupId":
"ocid1.volumegroupbackup.unique_ID"}'
{
  "data": {
    "availability-domain": "AD-1",
    "compartment-id": "ocid1.compartment.unique_ID",
    "defined-tags": {},
    "display-name": "volumegroup20230806102749",
    "freeform-tags": {},
    "id": "ocid1.volumegroup.unique_ID",
    "is-hydrated": null,
    "lifecycle-state": "PROVISIONING",
    "size-in-gbs": 0,
    "size-in-mbs": 0,
    "source-details": {
      "type": "volumeGroupBackupId",
      "volume-group-backup-id": "ocid1.volumegroupbackup.unique_ID"
    },
    "time-created": "2023-06-08T10:27:49.025508+00:00",
    "volume-group-replicas": null,
    "volume-ids": []
  },
  "etag": "c7053513-6819-49ad-8785-dd3e2a45272a",
  "opc-work-request-id": "ocid1.workrequest.unique_ID"
}
```

Managing Backup Policies

The Block Volume service enables you to perform volume backups and volume group backups automatically according to a schedule that is defined in a backup policy. The backup policy also specifies how long to retain the backup.

You can use one of the backup policies that is defined by Oracle and available in every compartment, or you can create your own user defined backup policy as described in Creating a Backup Policy. For descriptions of the Oracle defined backup policies, see "Volume Backups and Clones" in the Block Volume Storage Overview chapter in the *Oracle Private Cloud Appliance Concepts Guide*. You cannot modify policies that are provided by Oracle. You cannot use an Oracle defined backup policy to back up a volume group.

All backups are full backups.

A particular resource cannot be assigned more than one backup policy. A particular policy can have more than one backup schedule.

Schedule notes:

These notes apply to both user defined and Oracle defined backup policies.

- Start time: A backup might not start at its scheduled start time. A backup can be delayed for hours if the system is very busy.
- Conflicts: The Block Volume service will not run more than one scheduled backup of a particular resource in one day. If more than one backup is scheduled to run on the same day (for example, daily, weekly, and monthly backups are all scheduled to run this Sunday), the Block Volume service will run the backup that has the longest schedule period.

> **Caution:**
>
> All policy-based (scheduled) backups expire. A manual backup expires if a scheduled backup of the same volume is created after the manual backup was created. To keep a volume backup indefinitely, cancel all future scheduled backups and create a manual backup as described in Backing Up Block Volumes.

Creating a Backup Policy

You can use an Oracle defined backup policy, or you can follow the procedures described in this section to create your own backup policy.

For details about Oracle defined backup policies, see Viewing Backup Policies or see "Volume Backups and Clones" in the Block Volume Storage Overview chapter in the *Oracle Private Cloud Appliance Concepts Guide*.

Using the Compute Web UI

When you create a backup policy by using the Compute Web UI, creating the backup policy schedule is a separate step. A policy cannot be assigned to a resource until the policy schedule is defined.

1. In the navigation menu, under Block Storage, click Backup Policies.
2. Click Create Backup Policy.
3. Enter the following information:
   - **Name:** Enter a descriptive name for the policy.
   - **Create in Compartment:** Select the compartment for the policy.
   - **Tagging:**(Optional) Add defined or free-form tags for this backup policy as described in Adding Tags at Resource Creation. Tags can also be applied later.
4. Click Create Backup Policy.

   A policy cannot be assigned to a resource until at least one schedule is defined.
5. Click the name of the new backup policy to go to the details page for the policy.
6. On the policy details page, scroll to the Resources section and click the Add Schedule button.
7. In the Add Schedule dialog box, enter the following schedule parameters:
   - **Schedule Type:** Select Daily, Weekly, or Monthly, and then specify the time for the backup to run and the length of time to retain the backup.
     - *Hourly:* Select the Retention Time In Hours (1 - 24).
     - *Daily:* Select the Hour of the Day (0 - 23) and the Retention Time In Days (1 - 365).
     - *Weekly:* Select the Day of the Week and the Hour of the Day, and select the Retention Time In Weeks (1 - 52).
     - *Monthly:* Select the Day of the Month (1 - 31) and the Hour of the Day, and select the Retention Time In Months (1 - 144).
   - **Time Zone:** Select either UTC or your Regional Time.
8. Click the Add Schedule button in the dialog.

The policy can now be assigned to one or more resources. See Assigning a Backup Policy to a Volume or Volume Group.

A policy can have more than one schedule. To add another schedule to this policy, click the Add Schedule button above the list of schedules again. See the schedule notes in Managing Backup Policies.

Using the OCI CLI

1. Get the OCID of the compartment where the policy will reside (`oci iam compartment list`).
2. Construct an argument for the `--schedules` option.

   A policy cannot be assigned to any resource until you add at least one schedule.

   The following command shows the names of the schedule properties and the required format of the `--schedules` argument:

   ```
   oci bv volume-backup-policy create --generate-param-json-input schedules
   ```

   The following example file, named `backup_schedules.json`, defines two backup schedules for a policy: a daily backup and a monthly backup.

   ```
   [
     {
       "hourOfDay": 20,
       "offsetType": "STRUCTURED",
       "period": "ONE_DAY",
       "retentionSeconds": 172800,
       "timeZone": "REGIONAL_DATA_CENTER_TIME"
     },
     {
       "dayOfMonth": 1,
       "hourOfDay": 20,
       "offsetType": "STRUCTURED",
       "period": "ONE_MONTH",
       "retentionSeconds": 5356800,
       "timeZone": "REGIONAL_DATA_CENTER_TIME"
     }
   ]
   ```

   The schedule properties can have the following values:

   - `backupType` - Use `FULL`.
   - `dayOfMonth` - 1-31.
   - `dayOfWeek` - English day name, all uppercase.
   - `hourOfDay` - 0-23.
   - `month` - Do not use.
   - `offsetSeconds` - Minimum 0. Maximum must be specified in multiples of 60 seconds:
     - 3540 (59 minutes) for a `ONE_HOUR` period schedule
     - 86340 (23 hours, 59 minutes) for a `ONE_DAY` period schedule
     - 604740 (6 days, 23 hours, 59 minutes) for a `ONE_WEEK` period schedule
     - 194340 (30 days, 23 hours, 59 minutes) for a `ONE_MONTH` period schedule

The value of `offsetSeconds` is the number of seconds that the volume backup start time should be shifted from the default interval boundaries specified by the `period`. The volume backup start time is the frequency start time plus the offset.

- `offsetType` - `STRUCTURED`, `NUMERIC_SECONDS`.

  If the value is `STRUCTURED`, then `hourOfDay`, `dayOfWeek`, and `dayOfMonth` are used when applicable for the `period`, and `offsetSeconds` is ignored.

  If the value is `NUMERIC_SECONDS`, then `offsetSeconds` is used and `hourOfDay`, `dayOfWeek`, and `dayOfMonth` are ignored.

- `period` - The volume backup frequency: `ONE_HOUR`, `ONE_DAY`, `ONE_WEEK`, `ONE_MONTH`.
- `retentionSeconds` - Minimum 3600 (one hour). Maximum:
  - 86400 (one day) for a `ONE_HOUR` period schedule
  - 32140800 (one year) for `ONE_DAY` and `ONE_WEEK` period schedules
  - 385689600 (12 years) for a `ONE_MONTH` period schedule
- `timeZone` - `UTC` or `REGIONAL_DATA_CENTER_TIME`. The default is `UTC`.

See also the schedule notes in Managing Backup Policies.

3. Run the create backup policy command.

   Syntax:

   ```
   oci bv volume-backup-policy create --compartment-id <compartment_OCID> \

   --schedules JSON_backup_schedules
   ```

   Example:

   ```
   $ oci bv volume-backup-policy create --compartment-id ocid1.compartment.unique_ID \
   --display-name daily-and-monthly  \
   --schedules file:///backup_schedules.json
   {
     "data": {
       "compartment-id": "ocid1.compartment.unique_ID",
       "defined-tags": {},
       "destination-region": null,
       "display-name": "daily-and-monthly",
       "freeform-tags": {},
       "id": "ocid1.volumebackuppolicy.unique_ID",
       "schedules": [
         {
           "backup-type": null,
           "day-of-month": null,
           "day-of-week": null,
           "hour-of-day": 20,
           "month": null,
           "offset-seconds": null,
           "offset-type": "STRUCTURED",
           "period": "ONE_DAY",
           "retention-seconds": 172800,
           "time-zone": "REGIONAL_DATA_CENTER_TIME"
         },
         {
           "backup-type": null,
           "day-of-month": 1,
           "day-of-week": null,
           "hour-of-day": 20,
   ```

```
            "month": null,
            "offset-seconds": null,
            "offset-type": "STRUCTURED",
            "period": "ONE_MONTH",
            "retention-seconds": 5356800,
            "time-zone": "REGIONAL_DATA_CENTER_TIME"
          }
        ],
        "time-created": "2023-06-08T19:04:21.805470+00:00"
      },
      "etag": "9dbe57ca-9a04-4dc0-9261-90172c1b757d",
      "opc-work-request-id": "ocid1.workrequest.unique_ID"
    }
```


4. Assign the policy to a volume.

   See Assigning a Backup Policy to a Volume or Volume Group.

Assigning a Backup Policy to a Volume or Volume Group

You can assign a backup policy to a volume or volume group at resource creation or later:

- During volume or volume group creation. Select from the Backup Policy list in the Compute Web UI or specify the `--backup-policy-id` option in the OCI CLI.
- After the volume or volume group is created. Follow the procedures described in this section to add a backup policy if no backup policy is assigned, or to change the backup policy that is assigned to a volume or volume group.

A volume or volume group can have only one backup policy assigned. If a backup policy is already assigned to this volume or volume group when you assign a new policy, the existing assignment is replaced with the new assignment.

A volume group cannot be assigned a backup policy if any of the volumes in the group already is assigned a backup policy. To remove a backup policy assignment from a volume, see Removing a Backup Policy Assignment.

> **Note:**
>
> Oracle defined backup policies cannot be used for volume group backups. Only user defined backup policies can be assigned to a volume group.

Using the Compute Web UI

1. In the navigation menu, under Block Storage, click Block Volumes, Boot Volumes, or Volume Groups.
2. Select the appropriate compartment.
3. For the volume or volume group to which you want to assign a backup policy, click the Actions menu and then click Assign Backup Policy.
4. In the Assign Backup Policy dialog, select a backup policy from the drop-down list.

   You cannot assign an Oracle defined policy to a volume group.
5. Click the Assign Backup Policy button in the dialog.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Boot volume, block volume, or volume group OCID. For example: `oci bv volume list`)
   - Backup policy OCID (`oci bv volume-backup-policy list`)
2. Run the command to assign the backup policy to the volume or volume group.

   Syntax:

   ```
   oci bv volume-backup-policy-assignment create \
   --asset-id volume_OCID --policy-id backup_policy_OCID
   ```

   Example:

   This example assigns a backup policy to block volume. Use this same command with a boot volume or volume group asset. Do not try to assign an Oracle defined policy to a volume group.

   ```
   $ oci bv volume-backup-policy-assignment create \
   --asset-id ocid1.volume.unique_ID \
   --policy-id ocid1.volumebackuppolicy.unique_ID
   {
     "data": {
       "asset-id": "ocid1.volume.unique_ID",
       "id": "ocid1.backuppolicyassignment.unique_ID",
       "policy-id": "ocid1.volumebackuppolicy.unique_ID",
       "time-created": "2023-05-26T23:31:53+00:00"
     },
     "etag": "57ddd89d-14bc-4ecf-b164-8a6cc9dc9014",
     "opc-work-request-id": "ocid1.workrequest.unique_ID
   }
   ```

Removing a Backup Policy Assignment

Use this procedure to remove a backup policy assignment from a volume or volume group.

A volume group cannot be assigned a backup policy if any of the volumes in the group already is assigned a backup policy.

Using the OCI CLI

1. Use the appropriate `list` command to get the OCID of the volume or volume group from which you want to remove the backup policy assignment. For example: `oci bv volume list`.
2. Use the volume or volume group OCID from the preceding step as the argument of the `--asset-id` option in the following command to get the backup policy assignment OCID.

   ```
   $ oci bv volume-backup-policy-assignment get-volume-backup-policy-asset-assignment \
   --asset-id ocid1.volume.unique_ID
   {
     "data": [
       {
         "asset-id": "ocid1.volume.unique_ID",
         "id": "ocid1.backuppolicyassignment.unique_ID",
         "policy-id": "ocid1.volumebackuppolicy.unique_ID",
         "time-created": "2023-06-07T02:03:53.466062+00:00"
       }
   ```

```
  ]
}
```

3. Use the backup policy assignment OCID from the preceding step to delete this assignment so that this resource has no backup policy assigned.

```
$ oci bv volume-backup-policy-assignment delete \
--policy-assignment-id ocid1.backuppolicyassignment.AK00661530.scasg01.unique_ID \
--force
{
  "etag": "7a0ca7dd-50f7-4d60-9689-02e442ac4348",
  "opc-work-request-id": "ocid1.workrequest.unique_ID"
}
```

The following command shows that this asset has no backup policy assigned.

```
$ oci bv volume-backup-policy-assignment get-volume-backup-policy-asset-assignment \
--asset-id ocid1.volume.unique_ID
$
```

Viewing Backup Policies

Using the Compute Web UI

1. In the navigation menu, under Block Storage, click Backup Policies.

   At a minimum, the Oracle defined policies are listed. If the compartment has any user defined policies, then both Oracle defined and user defined policies are listed.

2. To view a particular user defined policy, you might need to select a different compartment from the drop-down list above the policies list.

3. To see details about a policy, click the policy name. Scroll to the Resources section to see the schedules.

Using the OCI CLI

1. To list user defined backup policies, get the OCID of the compartment where the policies reside (`oci iam compartment list`)

2. Run the list backup policies command.

   Syntax:

   If no compartment OCID is provided, then only the Oracle defined policies are listed.

   ```
   oci bv volume-backup-policy list
   ```

   To list user defined policies, provide the compartment OCID of the user defined policies.

   ```
   oci bv volume-backup-policy list --compartment-id compartment_OCID
   ```

   Example:

   ```
   $ oci bv volume-backup-policy list --compartment-id ocid1.compartment.oc1.unique_ID
   {
     "data": [
       {
         "compartment-id": "ocid1.compartment.oc1.unique_ID",
         "defined-tags": {},
         "destination-region": null,
         "display-name": "daily-backup",
         "freeform-tags": {},
         "id": "ocid1.volumebackuppolicy.unique_ID",
   ```

```
      "schedules": [
        {
          "backup-type": null,
          "day-of-month": null,
          "day-of-week": null,
          "hour-of-day": 20,
          "month": null,
          "offset-seconds": null,
          "offset-type": "STRUCTURED",
          "period": "ONE_DAY",
          "retention-seconds": 172800,
          "time-zone": "REGIONAL_DATA_CENTER_TIME"
        }
      ],
      "time-created": "2023-06-08T19:04:21.805470+00:00"
    }
    {
      "compartment-id": "ocid1.compartment.oc1.unique_ID",
      "defined-tags": {},
      "destination-region": null,
      "display-name": "monthly-backup",
      "freeform-tags": {},
      "id": "ocid1.volumebackuppolicy.unique_ID",
      "schedules": [
        {
          "backup-type": null,
          "day-of-month": 1,
          "day-of-week": null,
          "hour-of-day": 20,
          "month": null,
          "offset-seconds": null,
          "offset-type": "STRUCTURED",
          "period": "ONE_MONTH",
          "retention-seconds": 5443200,
          "time-zone": "REGIONAL_DATA_CENTER_TIME"
        }
      ],
      "time-created": "2023-05-31T22:35:13.267843+00:00
    }
  ],
}
```


To list a specific backup policy, use the `get` command with the OCID of the backup policy:

```
$ oci bv volume-backup-policy get \
--policy-id ocid1.volumebackuppolicy.unique_ID
```

If the `compartment-id` property value is `null`, then this is an Oracle defined policy.

Editing a Backup Policy Schedule

You cannot edit an Oracle defined backup policy. You can only edit user defined backup policies.

Using the Compute Web UI

1. In the navigation menu, under Block Storage, click Backup Policies.
2. Select the appropriate compartment.
3. Click the name of the backup policy that has the schedule you want to edit.

4. On the backup policy details page, scroll to the Resources section.
5. For the schedule that you want to edit in the Schedules list, click the Actions menu and then click Edit.
6. Make your changes. See Creating a Backup Policy for details.
7. Click Update Schedule.

Using the OCI CLI

1. Get the backup policy OCID (`oci bv volume-backup-policy list`)
2. Construct an argument for the `--schedules` option.

   See Creating a Backup Policy for details.

   The content that you provide in the `--schedules` argument replaces all current schedule information. Specify all schedule properties, not just the ones you want to change.
3. Run the edit backup policy command.

   Syntax:

   ```
   oci bv volume-backup-policy update --policy-id backup_policy_OCID \
   --schedules JSON_backup_schedule_information
   ```

   Example:

   For complete output, see Creating a Backup Policy.

   ```
   $ oci bv volume-backup-policy update \
   --policy-id ocid1.volumebackuppolicy.unique_ID \
   --schedules file:///new_backup_schedule.json
   WARNING: Updates to schedules and defined-tags and freeform-tags will replace any
   existing values.
   Are you sure you want to continue? [y/N]: y
   ```

Deleting a Backup Policy Schedule

You cannot change an Oracle defined backup policy. You can only change or delete the schedules of user defined backup policies.

Using the Compute Web UI

1. In the navigation menu, under Block Storage, click Backup Policies.
2. Select the appropriate compartment.
3. Click the name of the backup policy that has the schedule you want to delete.
4. On the backup policy details page, scroll to the Resources section.
5. For the schedule that you want to delete in the Schedules list, click the Actions menu and then click Delete.
6. Confirm the deletion.

Using the OCI CLI

1. Get the backup policy OCID (`oci bv volume-backup-policy list`)
2. Run the delete backup policy command.

   ```
   $ oci bv volume-backup-policy delete \
   --policy-id ocid1.volumebackuppolicy.unique_ID --force
   ```

Deleting a Backup Policy

You cannot delete an Oracle defined backup policy. You can only delete user defined backup policies.

Using the Compute Web UI

1. In the navigation menu, under Block Storage, click Backup Policies.
2. Select the appropriate compartment.
3. For the policy you want to delete, click the Actions menu and then click Delete.
4. Confirm the deletion.

Using the OCI CLI

1. Get the OCID of the backup policy that you want to delete (`oci bv volume-backup-policy list`)
2. Run the delete backup policy command.

```
$ oci bv volume-backup-policy delete \
--policy-id ocid1.volumeBackupPolicy.unique_ID --force
{
  "etag": "63a6b74f-e86e-423c-9948-123456789012"
}
```

11 File System Storage

The File Storage service provides scalable and secure shared network file systems.

The File Storage service encrypts all file system and snapshot data at rest.

You can mount a File Storage service file system on any compute instance in your Virtual Cloud Network (VCN).

For more conceptual information, refer to the File Storage Overview chapter in the Oracle Private Cloud Appliance Concepts Guide.

Creating a File System, Mount Target, and Export

This section describes how to perform all the tasks that are required to create a file system and make it available for instances.

Task Flow

No.	Description	Links to Procedures
1.	Ensure that a mount target exists for the subnet that the instance where you want to mount a file system will use and the backing store pool that the file system will use. **Note**The file system and mount target must be in the same compartment and the same backing store pool when you create an export.	Creating a Mount Target
2.	Create the file system.	Creating a File System
3.	Create a file system export in the mount target.	Creating an Export for a File System
4.	Enable Security Rules for File Storage.	Controlling Access to File Storage
5.	Change NFS export options to control access to the file system.	Setting NFS Export Options

After the file system is exported, on the NFS client, perform these tasks to mount the file system:

1. (If needed) Install NFS client software.
2. Create a mount point.
3. On the client, mount the file system to the mount point.
4. On the client, add whatever files, directories, and data that you want in the file system.

For more information about mounting file systems, see Mounting File Systems on UNIX-Based Instances.

Creating a Mount Target

A mount target is an NFS endpoint assigned to a subnet of your choice. The mount target provides the IP address or DNS name that is used in the mount command when connecting NFS clients to a file system.

For an instance to mount a file system, the instance's VCN must have a mount target.

You can create at most two mount targets per VCN: at most one mount target per pool type. Pool type refers to the backing store pool for the file system, which can be either the default pool of the attached ZFS Storage Appliance or a high performance pool. See the poolName property (the OraclePCA.poolName defined tag) in Creating a File System. Two mount targets in a VCN are counted as one with regard to resource limits (see Tenancy Resource Configuration Limits in the *Oracle Private Cloud Appliance Release Notes*).

You can reuse a mount target to make many file systems available on the network. To reuse the same mount target for multiple file systems, create an export in the mount target for each file system. The file system and mount target must be in the same compartment and the same backing store pool when you create an export.

> **Caution:**
>
> Do not use /30 or smaller subnets for mount target creation because they might not have sufficient available IP addresses.

> **Important:**
>
> When exporting file systems to overlapping CIDRs in a VCN, exports to the longest CIDR (smallest network) must be done first. For more information and an example, see My Oracle Support article PCA File system as a Service Exports (Doc ID 2823994.1).

Before you can create a mount target, ensure that these items are configured:

- At least one Virtual Cloud Network (VCN) and subnet is configured. See Managing VCNs and Subnets.
- (Required for cross appliance mounting) A Dynamic Routing Gateway (DRG) with a route rule in the VCN. See Connecting to the On-Premises Network through a Dynamic Routing Gateway.
- (Optional) Security rules for the file system mount target. Security rules can be created in the security list for the mount target subnet, or in a Network Security Group (NSG) that you add the mount target to. See Controlling Access to File Storage.

  You don't need security rules to create a mount target, but you do need security rules to eventually mount file systems that are associated with this mount target.

Using the Compute Web UI

1. In the navigation menu, under File Storage, click Mount Targets.

In the compartment drop-down menu above the mount targets list, select the compartment where you plan to create the file system.

If a mount target is listed, click the name of the mount target to open the details page and check the following parameters:

- The mount target must be on the same subnet as the instance where you want to mount the file system.
- Click the Tags tab. The mount target must be in the same backing store pool that is specified for the file system. If the value of the OraclePCA.poolName tag is PCA_POOL_HIGH, then the mount target is in the high performance pool. If the value of the OraclePCA.poolName tag is PCA_POOL, or if there is no OraclePCA.poolName tag, then the mount target is in the default pool of the attached ZFS Storage Appliance.

If the mount target meets your needs, skip this procedure and go to Creating a File System.

2. Click Create Mount Target.
3. Enter the mount target information:
   - **Name:** It doesn't have to be unique. An Oracle Cloud Identifier (OCID) uniquely identifies the mount target. Avoid entering confidential information.

     > **Note:**
     >
     > The mount target name is different than the DNS hostname.

   - **Create in Compartment:** Specify the compartment.
   - **VCN:** Select the VCN for the new mount target.
   - **Subnet:** Select a subnet to attach the mount target to.
   - **IP Address:**(Optional) You can specify an unused IP address in the subnet you selected for the mount target. If left blank, an IP address is automatically assigned.
   - **Host Name:**(Optional) You can specify a hostname you want to assign to the mount target.

     > **Note:**
     >
     > The File Storage service constructs a fully qualified domain name (FQDN) by combining the hostname with the FQDN of the mount target subnet.
     >
     > For example, `myhostname.subnet123.dnslabel.examplevcn.com`.

   - **Enable Network Security Groups:** Select this option to add this mount target to an existing NSG.

> **Important:**
>
> Rules for the NSG that you select must be configured to allow traffic to the mount target's VNIC using specific protocols and ports. For more information, see Controlling Access to File Storage Configuring VCN Security Rules for File Storage.

- **Tagging:**(Optional) Add defined or free-form tags for this mount target as described in Adding Tags at Resource Creation. Tags can also be applied later.

  By default, the mount target is for the default pool of the attached ZFS Storage Appliance. To create a mount target for a high performance pool, select the OraclePCA tag namespace, the poolName tag key, and the value PCA_POOL_HIGH. For more information, see "Block Volume Performance Options" in the Block Volume Storage Overview chapter in the *Oracle Private Cloud Appliance Concepts Guide*. Before you specify PCA_POOL_HIGH, check with an appliance administrator to verify that a high performance pool is available. The poolName property can be set only when the mount target is created. You cannot set or change this property value after the mount target is created.

4. Click Create Mount Target.

   Next, create a file system. See Creating a File System.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - OCID of the compartment where you plan to create the file system (`oci iam compartment list`)
   - OCID of the subnet of the instance where you want to mount a file system (`oci network subnet list`)
2. Run the create mount target command.

   Syntax:

   ```
   oci fs mount-target create \
   --availability-domain AD-1 \
   --compartment-id compartment_OCID \
   --subnet-id subnet_OCID
   ```

   By default, the mount target is for the default pool of the attached ZFS Storage Appliance. To create a mount target for a high performance pool, specify the `OraclePCA.poolName` tag with a value of `PCA_POOL_HIGH` as shown in the following example. For more information, see "Block Volume Performance Options" in the Block Volume Storage Overview chapter in the *Oracle Private Cloud Appliance Concepts Guide*. Before you specify `PCA_POOL_HIGH`, check with an appliance administrator to verify that a high performance pool is available. The `poolName` property can be set only when the mount target is created. You cannot set or change this property value with the `update` command.

   Example:

   ```
   oci fs mount-target create --availability-domain AD-1 \
   --compartment-id ocid1.compartment.uniqueID --subnet-id ocid1.subnet.uniqueID \
   --defined-tags '{"OraclePCA":{"poolName":"PCA_POOL_HIGH"}}' \
   --display-name HighPerfPoolMT
   {
     "data": {
   ```

```
    "availability-domain": "AD-1",
    "compartment-id": "ocid1.compartment.uniqueID",
    "defined-tags": {
      "Oracle-Tags": {
        "CreatedBy": "pca_user",
        "CreatedOn": "2024-07-03T14:56:29.92Z"
      },
      "OraclePCA":{
        "poolName":"PCA_POOL_HIGH"
      }
    },
    "display-name": "HighPerfPoolMT",
    "export-set-id": "ocid1.exportset.uniqueID",
    "freeform-tags": {},
    "id": "ocid1.mounttarget.uniqueID",
    "lifecycle-details": null,
    "lifecycle-state": "CREATING",
    "nsg-ids": [],
    "private-ip-ids": [],
    "subnet-id": "ocid1.subnet.uniqueID",
    "time-created": "2024-07-03T14:56:29.921587+00:00"
  },
  "etag": "2d278b37-a74a-4fec-b74a-fd9e9a1c72de"
```


3. Next, create a file system. See Creating a File System.

Creating a File System

You can set values for the file system quota, the database record size, and the pool to use for the backing store by using OraclePCA defined tags. If you use the OCI CLI or API, you can specify the OraclePCA tag namespace, tag key, and values for the parameters that you want to set. You do not need to first create the OraclePCA tag namespace and tag keys.

> **Note:**
>
> If you use the Compute Web UI to set these parameters, you must first create the OraclePCA tag namespace, tag keys, and value choices. See Creating OraclePCA Tags for instructions.

Using the Compute Web UI

1. On the Dashboard, click the File Storage/View File Systems button.

   Ensure that the correct compartment is selected in the compartment drop-down menu above the file systems list. The file system and mount target must be in the same compartment and the same backing store pool when you create an export.

2. Click the Create File System button.
3. In the Create File System dialog, enter the following information:
   - **Name:** It doesn't have to be unique. An Oracle Cloud Identifier (OCID) uniquely identifies the file system. Avoid entering confidential information.
   - **Create in Compartment:** Select the compartment where the file system is created.
   - **Tagging:** (Optional) Add defined or free-form tags for this file system as described in Adding Tags at Resource Creation. Tags can also be applied later.

See the OCI CLI procedure for descriptions of the file system quota (OraclePCA.quota), database record size (OraclePCA.databaseRecordSize), and backing store pool (OraclePCA.poolName) defined tags. Note that databaseRecordSize and poolName must be set in Create File System. The database record size and backing store pool cannot be set or updated after the file system is created. For more information about backing store pools, see "Block Volume Performance Options" in the Block Volume Storage Overview chapter in the *Oracle Private Cloud Appliance Concepts Guide*. Before you specify the high performance backing store pool, check with an appliance administrator to verify that a high performance pool is available.

4. Click Create File System.

   The file system is created.

   Next, create an export for the file system. See Creating an Export for a File System.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Compartment OCID (`oci iam compartment list`)
   - File System Name: The display name you want assigned to this file system
2. Decide whether you need to set certain optional properties.

   The following properties are set by using defined tags. For the syntax to specify a defined tag, see Adding Tags at Resource Creation.

   Specify the `OraclePCA` tag namespace to set values for the following properties:

   - File system quota. Specify `quota` for the tag key. The default value of `quota` is 0, which means no quota is set. A quota that you set includes the data in the file system and all snapshots created under the file system. You can specify a quota value in gigabytes from 0 to 8000000 (8 petabytes). Any fractional portion of the gigabyte value is rounded to the next larger megabyte. The file system quota can be reset with the file system `update` command.
   - Database record size. Specify `databaseRecordSize` for the tag key. The default database record size is 131072 bytes. You can specify one of the following values (in bytes) for the value of `databaseRecordSize`: 512, 1024, 2048, 4096, 8192, 16384, 32768, 65536, 131072, 262144, 524288, 1048576. The `databaseRecordSize` property can be set only when the file system is created. You cannot set or change this property value with the `update` command.
   - Backing store pool. Specify `poolName` for the tag key. By default, the backing store of a file system instance is the default pool of the attached ZFS Storage Appliance, specified as `PCA_POOL`. You can specify `PCA_POOL_HIGH` for the value of `poolName` to indicate that you want to use a high performance pool for the backing store. For more information about backing store pools, see "Block Volume Performance Options" in the Block Volume Storage Overview chapter in the *Oracle Private Cloud Appliance Concepts Guide*. Before you specify `PCA_POOL_HIGH`, check with an appliance administrator to verify that a high performance pool is available. The `poolName` property can be set only when the file system is created. You cannot set or change this property value with the `update` command.

   See the following step for an example of setting these values.
3. Run the create file system command.

   Syntax:

```
oci fs file-system create --availability-domain AD-1 \
--compartment-id compartment_OCID
```

Example:

```
oci fs file-system create --availability-domain AD-1 \
--compartment-id ocid1.compartment.unique_ID --display-name MyFileSystem

{
  "data": {
    "availability-domain": "AD-1",
    "compartment-id": "ocid1.compartment.unique_ID",
    "defined-tags": {
      "Oracle-Tags": {
        "CreatedBy": "pca_user",
        "CreatedOn": "2024-07-05T13:15:11.19Z"
      }
    },
    "display-name": "MyFileSystem",
    "freeform-tags": {},
    "id": "ocid1.filesystem.unique_ID",
    "is-clone-parent": false,
    "is-hydrated": true,
    "is-targetable": null,
    "kms-key-id": "",
    "lifecycle-details": "",
    "lifecycle-state": "CREATING",
    "metered-bytes": 0,
    "source-details": {
      "parent-file-system-id": "",
      "source-snapshot-id": ""
    },
    "time-created": "2024-07-05T13:15:11.234434+00:00"
  },
  "etag": "58dec47e-4732-4730-9e18-6b5db1ac30d6"
}
```

Example using defined tags to set additional properties:

To set a quota for the file system, change the default database record size, or specify a high performance pool for the file system backing store, use `OraclePCA` defined tags as shown in the following example.

```
oci fs file-system create --availability-domain AD-1 \
--compartment-id ocid1.compartment.unique_ID --display-name myfilesystem \
--defined-tags '{"OraclePCA":
{"quota":100000,"databaseRecordSize":8192,"poolName":"PCA_POOL_HIGH"}}'
```

Alternatively, you can specify these properties in a JSON file.

```
{
  "OraclePCA": {
    "quota": 100000,
    "databaseRecordSize": 8192,
    "poolName": "PCA_POOL_HIGH"
  }
}
```

Then specify the file as the argument of the `--defined-tags` option.

```
--defined-tags file://./fs_options.json
```

4. Next, create an export for the file system. See Creating an Export for a File System.

Creating an Export for a File System

Exports control how NFS clients access file systems when they connect to a mount target.

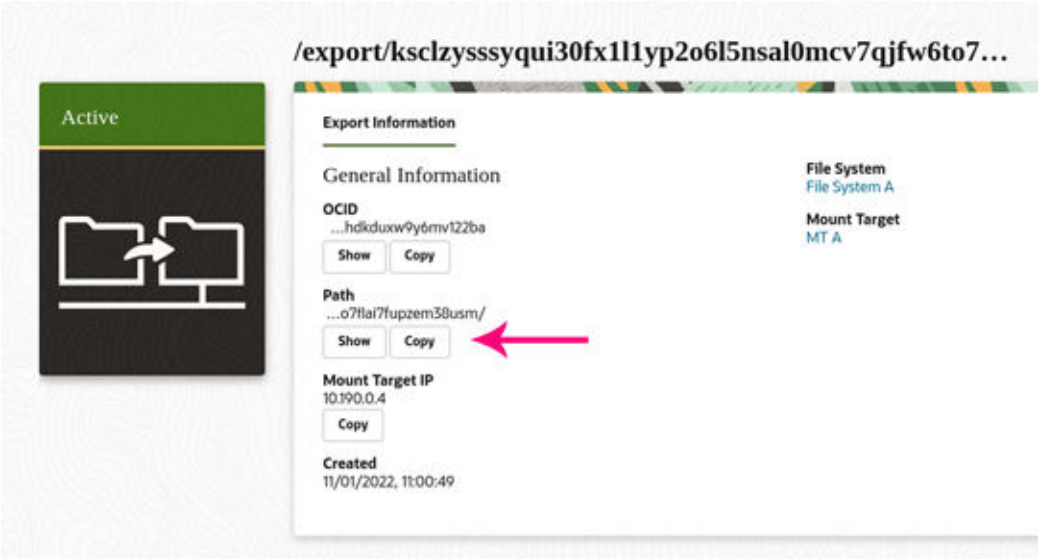
A file system must have at least one export in one mount target for instances to mount the file system.

> **Important:**
>
> When exporting file systems to overlapping CIDRs in a VCN, exports to the longest CIDR (smallest network) must be done first. For more information and an example, see My Oracle Support article PCA File system as a Service Exports (Doc ID 2823994.1).

Using the Compute Web UI

1. On the Dashboard, click the File Storage/View File Systems button.
2. If the file system that you want to export is not listed, use the Compartment drop-down menu above the file systems list to select the correct compartment.
3. Click the name of the file system that you plan to create an export for.
4. Scroll down to the Resources section, click Exports, and click Create Export.
5. Enter the following information:
   - Mount Target: Select a mount target from the list.
   - Source CIDR: Enter the longest CIDR (smallest network) in the CIDR range. Starting with the smallest CIDR range (largest network) will result in an error later in the process because CIDR ranges larger than existing ones will not be accepted. For example, 10.0.0.0/29 is a longer CIDR than 10.0.0.0/28, so add 10.0.0.0/29 first.
6. Click Create Export.

   The file system export is created and the export details page is displayed.
7. In the export details page, note the export path as shown in the following screen capture. The export path is used to mount the file system on an instance.



8. Scroll down to the Resources section and review the NFS Export Options.

   The NFS export options for that file system are set to the default values, which allow full access for all NFS client source connections. These defaults must be changed if you want to restrict access.

9. Consider your next action:

   - Mount the file system from an NFS client. See Mounting File Systems on UNIX-Based Instances.
   - Configure NFS options to secure the exported file system. See Setting NFS Export Options.

Using the OCI CLI

1. Gather the information that you need to run the command:

   - Export set OCID (`oci fs export-set list`)
   - File system OCID (`oci fs file-system list`)

2. Run the export create command.

> **Important:**
>
> The `path` option is required and its value must be `AUTOSELECT`. The value must be given in all uppercase letters. Providing any other value will cause the `export create` command to fail.

The export path is always automatically generated. See the `path` property in the command output for the export path.

Syntax:

```
oci fs export create --export-set-id export_set_OCID \
--file-system-id file_system_OCID --path AUTOSELECT
```

Example:

```
oci fs export create --export-set-id ocid1.exportset.uniqueID \
--file-system-id ocid1.filesystem.uniqueID --path AUTOSELECT
{
  "data": {
    "export-options": [
      {
        "access": "READ_WRITE",
        "anonymous-gid": 65534,
        "anonymous-uid": 65534,
        "identity-squash": "NONE",
        "require-privileged-source-port": false,
        "source": "0.0.0.0/0"
      }
    ],
    "export-set-id": "ocid1.exportset.uniqueID",
    "file-system-id": "ocid1.filesystem.uniqueID",
    "id": "ocid1.export.uniqueID",
    "lifecycle-state": "ACTIVE",
    "path": "/export/18lt6v4drhddiz2mn7vwmqt7mjiz3kfbw4reqaew33y50pdrj35p4ef5p04x",
    "time-created": "2023-06-06T04:34:28.829547+00:00"
  },
  "etag": "a0842b0b-b27b-4c98-a1ff-da85ae4bf150"
}
```

3. In the command output, note the value of `path`. The path value is used to mount the file system. To reprint this information, use the following command:

   ```
   oci fs export get --export-id ocid1.export.uniqueID
   ```

4. In the output, review the export options.

   In this example, the NFS export options for the file system are set to the default values, which allow full access for all NFS client source connections. These defaults must be changed if you want to restrict access

5. Next, control access to the file system.

   See Controlling Access to File Storage.

Mounting File Systems Across Private Cloud Appliances

You can create a file system on one Private Cloud Appliance, and mount the file system from an instance that is on another Private Cloud Appliance. To achieve this scenario, you must configure certain network parameters on each appliance.

Restriction

The appliance hosting the file system and the instance on the remote appliance that mounts the file system can't have overlapping VCN CIDR blocks.

On the Appliance Hosting the File System

1. Configure these network parameters:
   a. Create a Dynamic Routing Gateway (DRG).

      See Connecting to the On-Premises Network through a Dynamic Routing Gateway.

   b. For the VCN subnet that will be used by the mount target, attach the VCN to the DRG.

      See Attach VCNs to a Dynamic Routing Gateway.

   c. For the VCN subnet, add a route rule with the DRG as the target, and assign a destination CIDR that matches the remote appliance VCN CIDR.

For example, if the remote appliance instance that will mount the file system has a VCN with a 10.11.0.0/16 CIDR, the set the route rule destination CIDR to 10.11.0.0/16.

> **Important:**
>
> Don't specify 0.0.0.0/0 as the destination. Doing so causes serious internal network issues.

> **Note:**
>
> This route rule configuration is only required for mounting file systems across appliances. This configuration is not required for file system mounts within the same appliance.

See Working with Route Tables.

2. Create a mount target. See Creating a Mount Target.
3. Create a file system. See Creating a File System.
4. Create an export that exports to both the source and remote appliance VCN CIDR.

   The export CIDR must be big enough to cover both appliance VCN subnet CIDRs. For example, if the host VCN CIDR is 10.10.0.0/16, and the remote appliance instance VCN is 10.11.0.0/16, you can configure the export to use 10.10.0.0/15. This requirement only applies to the appliance hosting the file system.

   See Creating an Export for a File System.

On the Remote Appliance

1. Configure these network parameters:
   a. Create a Dynamic Routing Gateway (DRG).

      See Connecting to the On-Premises Network through a Dynamic Routing Gateway.

   b. For the VCN subnet, attach the VCN to the DRG.

      See Attach VCNs to a Dynamic Routing Gateway.

   c. For the VCN subnet, add a route rule with the DRG as the target, and assign a destination CIDR that matches the host appliance VCN CIDR.

      For example, if the host appliance mount target has a VCN 10.0.0.0/16 CIDR, set the route rule destination CIDR to 10.0.0.0/16.

> **Important:**
>
> Don't specify 0.0.0.0/0 as the destination. Doing so causes serious internal network issues.

> **Note:**
>
> This route rule configuration is only required for mounting file systems across appliances. It is not required for file system mounts within the same appliance.

See Working with Route Tables.

2. Log in to the instance and mount the file system.

   See:

   - Mounting File Systems on UNIX-Based Instances
   - Mounting File Systems On Microsoft Windows Instances

Controlling Access to File Storage

Before you can mount a file system, you must configure security rules to allow traffic to the mount target's VNIC using specific protocols and ports. Security rules enable traffic for the following protocols:

- Open Network Computing Remote Procedure Call (ONC RPC) rpcbind utility protocol
- Network File System (NFS) protocol
- Network File System (MOUNT) protocol

For more conceptual information, refer to the File Storage Overview chapter in the Oracle Private Cloud Appliance Concepts Guide.

Configuring VCN Security Rules for File Storage

You can add the required rules to a preexisting security list associated with a subnet, such as the default security list that is created along with the VCN.

For specific information about which security rules are required for the File Storage service, refer to *File Storage Network Ports* in the File Storage Overview chapter in the Oracle Private Cloud Appliance Concepts Guide.

For more information about managing VCNs and subnets, see Managing VCNs and Subnets.

Using the Compute Web UI

1. In the navigation menu, under Networking, click Virtual Cloud Networks.
2. Select the compartment where the VCN is located.
3. Click the name of the VCN.
4. Under Resources, click Security Lists.
5. Click the name of the security.
6. Under Resources, click Ingress Rules.
7. Click Create Ingress Security Rule, and enter the required information:
   - **Stateless check box:** Specify a stateful rule by leaving the check box unchecked.
   - **Ingress CIDR:** Enter the CIDR block for the subnet. For example, 10.0.0.0/24.

- **IP Protocol:** Choose the protocol. For example, TCP.
- **Description:** Enter a meaningful description for the rule.

8. Click Create Security List Rule.
9. Under Resources, click Egress Rules.
10. Click Create Egress Security Rule and enter the required information:
    - **Stateless check box:** Specify a stateful rule by leaving the check box unchecked.
    - **Egress Type:** To allow traffic from the subnet, select CIDR.
    - **Egress CIDR:** Enter the CIDR block for the subnet. For example, 10.0.0.0/24.
    - **IP Protocol:** Choose the protocol. For example, TCP.
    - **Description:** Enter a meaningful description for the rule.
11. Click Create Security List Rule.

Adding File Storage to a Network Security Group

Task Flow

No.	Description	Links to Procedures
1.	Create an NSG with the required security rules. (Alternatively, you can add them to a previously existing NSG.)	Controlling Traffic with Network Security Groups
2.	Add the mount target (or more specifically, the mount target's VNIC) to the NSG. You can do this task when you create the mount target, or you can update the mount target and add it to one or more NSGs that contain the required security rules.	Adding a Mount Target to a Network Security Group
3.	If you're setting up a mount target and instance in different subnets, add the instance (or more specifically, the instance's primary VNIC) to the NSG that contains the required security rules. You can do this task when you create the instance, or you can directly update the instance's primary VNIC.	Updating a VNIC

Adding a Mount Target to a Network Security Group

You can add the mount target to one or more Network Security Groups (NSGs). File storage requires specific rules to be configured for NSGs that are associated with mount targets.

Using the Compute Web UI

1. Ensure that an NSG with ingress and egress rules has been configured.

   See Configuring VCN Rules and Options.
2. Ensure that a mount target is created.

   See Managing VCNs and Subnets.

3. In the navigation menu, under File Storage, click Mount Targets.
4. Click the mount target name to see the details page.
5. Click Edit.
6. Enable Network Security Groups.
7. Select the NSG from the list.
8. Click Save Changes.

Using the OCI CLI

1. Ensure that an NSG with ingress and egress rules has been configured.

   See Configuring VCN Rules and Options.
2. Ensure that a mount target is created.

   See Managing VCNs and Subnets.
3. Gather the information that you need to run the command:
   - Mount target OCID (`oci fs mount-target list`)
   - NSG OCIDs (`oci network nsg list`)
4. Run this command.

   Syntax (entered on a single line):

   ```
   oci fs mount-target update
   --mount-target-id <mount_target_OCID>
   --nsg-ids '["<nsg1_OCID>","i"]'
   ```

   Example:

   ```
   oci fs export update  \
   --mount-target-id ocid1.mounttarget.….….….uniqueID  \
   --nsg-ids '["ocid1.networksecuritygroup.….….
   ….uniqueID-01","ocid1.networksecuritygroup.….….….uniqueID-02"]'
   ```

Setting NFS Export Options

When you create a file system and export, the NFS export options for that file system are set to the defaults listed in this table. The default values allow full access for all NFS client source connections. These defaults must be changed if you want to restrict access:

> **Caution:**
>
> If a file system is mounted by any clients, creating, deleting, or editing the Source value can disrupt file system I/O operations.

Export Option in the UI	Export Option in the CLI	Default Value	Description
Source:	`source`	0.0.0.0/0	The IP address or CIDR block of a connecting NFS client.

Export Option in the UI	Export Option in the CLI	Default Value	Description
Ports:	`require-privileged-source-port`	Any	Always set to: • UI: Any • CLI: `false`
Access:	`access`	Read/Write	Specifies the source NFS client access. Can be set to one of these values: • `READ_WRITE` • `READ_ONLY`
Squash:	`identity-squash`	None	Determines whether the clients accessing the file system as root have their User ID (UID) and Group ID (GID) remapped to the squash UID/GID. Possible values: • Root – Only the root user is remapped. • None – No users are remapped.
Squash UID/ GID:	`anonymous-uid` and `anonymous-gid`	65534	This setting is used along with the Squash option. When remapping a root user, you can use this setting to change the default anonymousUid and anonymousGid to any user ID of your choice.

Note – If you change the RW/RO permissions of an export option for an SMB share, the changes are only enforced for newly network-mapped drives of that share. Any previously mapped drives of the same share retain the original permissions. To have the changed permissions enforced on previously mapped drives on SMB clients, disconnect the shares and map them again.

For more information about configuring the options to suit various access scenarios, refer to the section titled *NFS Access Control and Export Options* in the File Storage Overview chapter in the Oracle Private Cloud Appliance Concepts Guide.

Using the Compute Web UI

1. In the navigation menu, under File Storage, click File Systems.
2. Select the appropriate compartment.
3. Click the file system name.
4. Under Resources, select Exports.
5. Click the export's export path.

   The NFS Export Options are displayed.
6. Click Edit Options.
7. In the NFS Export Options dialog, configure the NFS options.
8. Click Update Options.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Export ID (`oci fs export list --all --compartment-id` ***<compartment_OCID>***)

- Export options, listed in json format, in a json file or as a string on the command line.

2. Run this command.

> **Note:**
>
> This procedure shows the minimum required parameters for this command. For information about optional parameters, run the command with the `--help` option.

Syntax (entered on a single line):

```
oci fs export update
--export-id <export_id>
--export-options <file://json_file or json_string>
```

Note – The `require-privileged-source-port` option can only be set to `false`.

This example sets the export options for file system A to allow read/write access only to Client A, who is assigned to CIDR block 10.0.0.0/24. Client B and Client C are not included in this CIDR block, and cannot access the file system:

```
oci fs export update  \
--export-id File_system_A_export_ID  \
--export-options  \
'[{"source":"10.0.0.0/24","require-privileged-source-
port":"false","access":"READ_WRITE","identity-squash":"NONE","anonymous-
uid":"65534","anonymous-gid":"65534"}]'

WARNING: Updates to export-options will replace any existing values. Are you sure
you want to continue? [y/N]: y
{
  "data": {
    "export-options": [
      {
        "access": "READ_WRITE",
        "anonymous-gid": 65534,
        "anonymous-uid": 65534,
        "identity-squash": "NONE",
        "require-privileged-source-port": false,
        "source": "10.0.0.0/24"
      }
    ],
    "export-set-id": "ocid1.exportset.….….….uniqueID",
    "file-system-id": "ocid1.filesystem.….….….uniqueID",
    "id": "ocid1.export.oc1.pca.….….….uniqueID",
    "lifecycle-state": "ACTIVE",
    "path": "/export/85aiiadc1w81s8id63knxdq22nt95pe63sgs9c45yp3qovhut14cq9r6eqhn",
    "time-created": "2021-09-27T20:20:34.231009+00:00"
  },
  "etag": "bc660e11-644a-4043-9ad7-622d9581da9b"
}
```

Mounting File Systems on UNIX-Based Instances

Instance users of UNIX based operating systems, such as Linux and Oracle Solaris, can use OS commands to mount and access file systems.

Mount targets serve as network access points for file systems. After your mount target is assigned an IP address, you can use it together with the export path to mount the file system.

On the instance from which you want to mount the file system, you need to install an NFS client package and create a mount point. When you mount the file system, the mount point effectively represents the root directory of the File Storage file system, allowing you to write files to the file system from the instance.

Prerequisites

- The file system must be created and have at least one export in a mount target. See Creating a File System, Mount Target, and Export.
- The mount target must have correctly configured security rules or be assigned to an NSG. See Configuring VCN Security Rules for File Storage.

> **Note:**
>
> **Only for NFSv4 Mounts in Oracle Linux instances** – If you find that the file system owner is assigned as `nobody` instead of the actual user who mounts the file system, and if you have not set identity squash, you might need to edit the `/etc/idmapd.conf` file. In the file, set the DOMAIN entry to either `localdomain` or to the Active Directory domain name, if applicable. After the change, run `service rpcidmapd restart` to restart the `rpcidmapd` service.
>
> Defining settings in the `/etc/idmapd.conf` file is specific to Oracle Linux, and there are other ways to configure the domain depending on the OS in use. Consult your operating system documentation.

For more conceptual information, refer to the File Storage Overview chapter in the Oracle Private Cloud Appliance Concepts Guide.

Obtaining the Mount Target IP Address

To mount a file system, you need to know the private IP address of the mount target that has the export for the file system.

Using the Compute Web UI

1. In the navigation menu, under File Storage, click Mount Target.
2. Click the Mount Target name to see the details page.

   The IP address is displayed.

Using the OCI CLI

1. Gather the information that you need to run the commands:
   - Mount target OCID (`oci fs mount-target list`)
2. Run this command to get the mount target IP ID.

   Syntax:

   ```
   oci fs mount-target get
   --mount-target-id mount_target_OCID
   ```

Example:

```
oci fs mount-target get
--mount-target-id ocid1.mounttarget.uniqueID
{
  "data": {
    "availability-domain": "AD-1",
    "compartment-id": "ocid1.tenancy.uniqueID
    "defined-tags": {
      "Finance": {
        "CostCenter": "admin"
      }
    },
    "display-name": "mount-target01",
    "export-set-id": "ocid1.exportset.uniqueID",
    "freeform-tags": {},
    "id": "ocid1.mounttarget.uniqueID",
    "lifecycle-details": null,
    "lifecycle-state": "ACTIVE",
    "nsg-ids": [],
    "private-ip-ids": [
      "ocid1.privateip.uniqueID"
    ],
    "subnet-id": "ocid1.subnet.uniqueID",
    "time-created": "2021-09-01T18:45:25.251048+00:00"
  },
  "etag": "c2f84c0b-d0b5-422c-9761-9e43d7fc4214"
}
```

3. Run this command to get the mount target IP address.

   Syntax (entered on a single line):

   ```
   oci network private-ip get --private-ip-id mount_target_IP_OCID
   ```

   Example:

   ```
   oci network private-ip get  \
   --private-ip-id ocid1.….….….….uniqueID{
     "data": {
       "availability-domain": "AD-1",
       "compartment-id": "ocid1.tenancy..….….….uniqueID",
       "defined-tags": {},
       "display-name": "privateip20210901184525",
       "freeform-tags": {},
       "hostname-label": null,
       "id": "ocid1.privateip.….….….uniqueID",
       "ip-address": "10.200.0.3",
       "is-primary": false,
       "subnet-id": "ocid1.subnet.….….….uniqueID",
       "time-created": "2021-09-01T18:45:25.406808+00:00",
       "vlan-id": null,
       "vnic-id": "ocid1.vnic.….….….uniqueID"
     },
     "etag": "c98377e4-ae89-46cf-9c61-52aea68a3476"
   }
   ```

Mounting a File System on Linux, Red Hat, or CentOS

1. Log into the instance where you want to mount the file system.

   See Connecting to a Compute Instance.

   Example:

```
ssh user@192.0.2.0
```

2. Install the NFS client using this command:

   ```
   sudo yum install nfs-utils
   ```

3. Create a directory that will be used as the mount point.

   Replace ***<yourmountpoint>*** with a directory name of your choice. Example: `/mnt/mountpoint-A`

   ```
   sudo mkdir -p <yourmountpoint>
   ```

4. Mount the file system.

   > **Caution:**
   >
   > Omitting the `-o nosuid` option can allow unprivileged users to escalate their permissions to 'root'. The `nosuid` option disables set-user-identifier or set-group-identifier bits within the mounted system, which are rarely used.

   Example:

   ```
   sudo mount -t nfs -o nfsvers=<version>,nosuid <10.x.x.x>:<fs-export-path>
                              <yourmountpoint>
   ```

   - Replace ***<version>*** with one of the following, based on the NFS protocol version you want to use:
     - 3,noacl
     - 4.0
     - 4.1
   - Replace ***<10.x.x.x>*** with the mount target's private IP address. See Obtaining the Mount Target IP Address.
   - Replace ***<fs-export-path>*** with the export path that was generated when the export was created. See Creating an Export for a File System.
   - Replace ***<yourmountpoint>***with the full path to the local mount point.

5. View the mounted file system.

   ```
   df -h
   ```

6. Write a file to the file system.

   Replace ***<yourmountpoint>*** with the path to the local mount point and ***<filename>***with your file name.

   ```
   sudo touch /mnt/<yourmountpoint>/<filename>
   ```

7. Verify that you can access the file system and view the file.

   Replace yourmountpoint with the path to the local mount point.

   ```
   cd <yourmountpoint>
   ls
   ```

8. Add the file system mount information to the appropriate mount file for your OS.

So far, the file system is manually mounted to the client. If the client is rebooted, the file system won't automatically mount unless you add it to the mount file (for example the `/etc/fstab` or `/etc/vfstab` file).

Mounting a File System on Ubuntu or Debian

Operating Systems and versions of operating systems differ in the way software is added. Consult the documentation for our specific operating system for details.

1. On the NFS client, open a command window, and install the NFS client using this command:

   ```
   sudo apt-get install nfs-common
   ```

2. Create a directory that will be used as the mount point.

   Replace ***<yourmountpoint>*** with a directory name of your choice. Example: `/mnt/mountpoint-A`

   ```
   sudo mkdir -p <yourmountpoint>
   ```

3. Mount the file system.

   > **Caution:**
   >
   > Omitting the `-o nosuid` option might allow unprivileged users to escalate their permissions to 'root'. The `nosuid` option disables set-user-identifier or set-group-identifier bits within the mounted system, which are rarely used.

   Example:

   ```
   sudo mount -t nfs -o nfsvers=<version>,nosuid <10.x.x.x>:<fs-export-path>
                                <yourmountpoint>
   ```

   - Replace ***<version>*** with one of the following, based on the NFS protocol version you want to use:
     - 3,noacl
     - 4.0
     - 4.1
   - Replace ***<10.x.x.x>*** with the mount target's private IP address. See Obtaining the Mount Target IP Address.
   - Replace ***<fs-export-path>*** with the export path that was generated when the export was created.

     See Creating an Export for a File System.
   - Replace ***<yourmountpoint>***with the full path to the local mount point.

4. View the file system.

   ```
   df -h
   ```

5. Write a file to the file system.

   Replace ***<yourmountpoint>*** with the path to the local mount point and ***<filename>***with your file name.

```
sudo touch /mnt/<yourmountpoint>/<filename>
```

6. Verify that you can access the file system and view the file.

   Replace yourmountpoint with the path to the local mount point.

   ```
   cd <yourmountpoint>
   ls
   ```

7. Add the file system mount information to the appropriate mount file for your OS.

   So far, the file system is manually mounted to the client. If the client is rebooted, the file system won't automatically mount unless you add it to the mount file (for example the `/etc/fstab` or `/etc/vfstab` file).

Configuring a File System to Automatically Mount (Linux Instances)

On Linux instances, if you want to automatically mount exported file systems during an instance boot, you need to add the mount information in the `/etc/fstab` file.

1. Log into the instance where you want the file system mounted.

   See Connecting to a Compute Instance.

2. Create a mount point, if one has not been created.

   Example:

   ```
   mkdir /mnt/fs01
   ```

3. Open the `/etc/fstab` file in an editor and add a line for the nfs file systems you want automatically mounted.

   This is an example of an `/etc/fstab` file entry.

   ```
   192.0.2.0:/export/3ywflz8hhqfde81miewqwjfd049zju69502t9ouo6shzidr4dndaz1hd6qfi /mnt/
   fs01 nfs nfsvers=4.1,nosuid,nofail 0 0
   ```

   The `/etc/fstab` file space-separated fields are specified with these entries:

   - **Field 1:** Device to mount. For network file systems, specify: ***<mount target IP>*** : ***<export_path>***

     See Obtaining the Mount Target IP Address and Creating an Export for a File System.
   - **Field 2:** Full path of the mount point on the instance.
   - **Field 3:** File system type. In this case, specify `nfs`.
   - **Field 4:** NFS mount options separated with commas, such as:

     ```
     nfsvers=<version>,nosuid,nofail
     ```

     – `nfsvers=` where ***<version>*** is one of the following:

       * `3,noacl`
       * `4.0`
       * `4.1`

     – `nosuid` – prevents unprivileged users from escalating their permissions to root.

     – `nofail` – Ensures that an unavailable file system does not cause the instance reboot process to fail.

     In this case, use the same options as described in Mounting a File System on Linux, Red Hat, or CentOS. Each option is separated by a comma (no spaces).

- **Field 5:** Obsolete option for dump backups. Specify `0` (zero) for no dump backup.
- **Field 6:** File system check (fsck) order. Specify `0` (zero) for no check.

4. Use this command to mount the volumes that are in the `/etc/fstab` file:

   ```
   sudo mount -a
   ```

   If you get any error messages, fix the cause before proceeding.

5. Verify that the file systems are mounted:

   ```
   mount | grep nfs
   ```

6. To verify that the file system will automatically mount, reboot the instance.

   ```
   sudo reboot
   ```

7. After the reboot, log into the instance and check to see if the nfs file system is mounted.

   ```
   mount | grep nfs
   ```

Mounting File Systems On Microsoft Windows Instances

You can make file systems available to Microsoft Windows instances by mapping a network drive to the mount target IP address and export path provided by the File Storage service. You can accomplish this task using NFS or SMB protocols.

Using the SMB protocol requires that the Microsoft Windows instances and Oracle Private Cloud Appliance belong to the same Active Directory domain.

For more information about configuring Active Directory in the Service Enclave, refer to *Configuring the Active Directory Domain for File Storage* in the Hardware Administration chapter of the Oracle Private Cloud Appliance Administrator Guide.

For more conceptual information, refer to the File Storage Overview chapter in the Oracle Private Cloud Appliance Concepts Guide.

Mounting a File System On a Microsoft Windows Instance Using NFS

Prerequisites

- The file system must be created and have at least one export in a mount target. See Creating a File System, Mount Target, and Export.
- The mount target must have correctly configured security rules or be assigned to an NSG. See Configuring VCN Security Rules for File Storage.
- You must know the mount target's IP address. See Obtaining the Mount Target IP Address.
- You must be able to log into the Microsoft Windows OS on the instance with superuser or administrator privileges.

Before You Begin

The following tasks are included in this procedure, and you might want to be aware of them before you begin.

- **Installation of the Microsoft Windows NFS Client** – This service must be installed on the instance from which you want to mount the file system. Installing the client often requires a restart of the instance.
- **The `AnonymousGid` and `AnonymousUid` identity values must be configured to allow write access.** – Access to NFS file systems requires UNIX user and group identities, which are

not the same as Microsoft Windows user and group identities. By default, file systems write permissions are only granted to the root user. To enable user access to NFS shared resources, the Microsoft Windows client for NFS accesses file systems anonymously, using `AnonymousGid` and `AnonymousUid`.

> **Caution:**
>
> Updating the AnonymousGid and AnonymousUid values require registry changes to your instance.

Choose one the following methods:

- Using the Microsoft Windows Command Prompt
- Using Microsoft Windows File Explorer

Using the Microsoft Windows Command Prompt

1. Log into your Microsoft Windows instance.

   See Connecting to a Compute Instance.
2. Open Microsoft Windows PowerShell and run as Administrator:
   a. Go to Start and open Microsoft Windows PowerShell.
   b. In Microsoft Windows PowerShell, type the following to run as Administrator:

      ```
      Start-Process powershell -Verb runAs
      ```

   c. In the User Account Control window, click Yes. A new Administrator: PowerShell window opens. You can close the standard PowerShell window to avoid confusing them.
3. In Administrator: PowerShell, get the NFS client and update the registry by typing the following:

   ```
   Install-WindowsFeature -Name NFS-Client
   Set-ItemProperty HKLM:\SOFTWARE\Microsoft\ClientForNFS\CurrentVersion\Default -Name
   AnonymousUid -Value 0
   Set-ItemProperty HKLM:\SOFTWARE\Microsoft\ClientForNFS\CurrentVersion\Default -Name
   AnonymousGid -Value 0
   Stop-Service -Name NfsClnt
   Restart-Service -Name NfsRdr
   Start-Service -Name NfsClnt
   ```
4. Open a standard Command Prompt Window.

   > **Important:**
   >
   > NFS file systems mounted as Administrator are not available to standard users.
5. From the Command Prompt window, mount the file system.

   See the cautions and notes below the example.

   In the following example, replace:

   - `10.x.x.x` with the mount point IP address (see Obtaining the Mount Target IP Address)

- `fs-export-path` with the file system export path (see Creating an Export for a File System)
- `X` with the drive letter of any available drive you want to map the file system to.

Example:

```
mount 10.x.x.x:/fs-export-path X:
```

6. Verify that you can access and write to the file system.
   a. Access the file system.

      In the example, replace `X` with the drive letter you used to mount the file system.

      ```
      X:
      ```

   b. Write a file.

      ```
      echo > myfile.txt
      ```

   c. Verify that you can view the file.

      ```
      dir
      ```

Using Microsoft Windows File Explorer

1. Log into your Microsoft Windows instance.

   See Connecting to a Compute Instance.
2. Open Microsoft Windows PowerShell and run as Administrator:
   a. Go to Start and open Microsoft Windows PowerShell.
   b. In Microsoft Windows PowerShell, type the following to run as Administrator:

      ```
      Start-Process powershell -Verb runAs
      ```

   c. In the User Account Control window, click Yes. A new Administrator: PowerShell window opens. You can close the standard PowerShell window to avoid confusing them.
3. In Administrator: PowerShell, get the NFS client by typing the following:

   ```
   Install-WindowsFeature -Name NFS-Client
   ```
4. If necessary, restart your system.
5. Open the registry editor (regedit) to map the AnonymousGid and AnonymousUid to the root user.

   > **Caution:**
   >
   > User identity mapping requires changes to your system registry.

   a. Click Windows Search.
   b. Enter `regedit` in the Search field and press Enter.
   c. Click Yes to allow changes to your device.
   d. Click `HKEY_LOCAL_MACHINE`. Then, browse to: `Software\Microsoft\ClientForNFS\CurrentVersion\Default`.
6. Add a new DWORD32 registry entry for `AnonymousGid`:

a. Click Edit, and select New DWORD (32 bit) Value.

b. In the Name field, enter `AnonymousGid`. Leave the value at `0`.

7. Repeat the previous step to add a second DWORD32 registry entry named `AnonymousUid` with a value of `0`.

8. Open Microsoft Windows Command Line (CMD) and run as Administrator:

   a. Go to Start and scroll down to Apps.

   b. In the Windows System section, press Ctrl+Shift and click Command Prompt.

9. In the Microsoft Windows Command Line (CMD) window, restart the NFS Client by typing the following:

```
nfsadmin client stop

nfsadmin client start
```

10. Open File Explorer and select This PC. In the Computer tab, select Map network drive.

11. Select the Drive letter that you want to assign to the file system.

12. In the Folder field, enter the following line, replacing:

    - `10.x.x.x` with the mount point IP address (see Obtaining the Mount Target IP Address)
    - `fs-export-path` with the file system export path (see Creating an Export for a File System)

    Line:

```
\\10.x.x.x\fs-export-path
```

13. Click the Finish button when complete.

Mounting a File System on a Window Instance Using SMB

General Prerequisites

- The file system must be created and have at least one export in a mount target. See Creating a File System, Mount Target, and Export.
- The mount target must have correctly configured security rules or be assigned to an NSG. See Configuring VCN Security Rules for File Storage.
- You must know the mount target's IP address. See Obtaining the Mount Target IP Address.
- You must be able to log into the Microsoft Windows OS on the instance with superuser or administrator privileges.

Specific Prerequisites for SMB Support

SMB support for the File Storage service requires that both Oracle Private Cloud Appliance and the client Microsoft Windows instances belong to the same Active Directory (AD) domain.

This procedure assumes that the AD service is already configured in your data center infrastructure.

To add a Microsoft Windows instance to your AD service, perform the necessary administrative tasks according to the documentation for your version of Microsoft Windows OS.

To add the appliance to your AD service, an administrator with privileges to the OOracle Private Cloud Appliance Service Enclave must add the AD domain name to the appliance's

Active Directory Domain configuration. For information on how to perform this task, refer to Hardware Administration in the Oracle Private Cloud Appliance Administrator Guide.

Relaxing File System Permissions Before Network Mapping with SMB

By default, write permissions to a file system are limited to the UNIX superuser and group identity. To provide write permission to AD domain users, the permissions need to be relaxed.

1. Mount the network drive using NFS protocol.

   See Mounting a File System On a Microsoft Windows Instance Using NFS.
2. Relax the file system permissions:
   a. Open File Explorer, select the mapped drive and right-click on it, then select Properties.
   b. Select the NFS Attributes tab.
   c. Change File permissions by checking all RWX check boxes to relax the permissions for Owner, Group, and Other.
   d. Click OK.
3. Disconnect the NFS-mounted drive.

   Now that the file system permissions are relaxed, you can mount the file system using the SMB protocol.

Mounting a File System Using SMB

1. Log into your Microsoft Windows instance.

   See Connecting to a Compute Instance.
2. Open File Explorer and select This PC.
3. In the Computer tab, select Map network drive.
4. In the Folder field, enter the following line and replace these items:
   - `10.x.x.x` with the mount target IP address.
   - `fs-export-path-ID` with the file system export path (see Creating an Export for a File System)

     Note – Do not include `\export` in the `fs-export-path-ID` string when mounting using SMB.

   ```
   \\10.x.x.x\fs-export-path-ID
   ```

   Example:

   ```
   \\192.0.2.0\39u21btystm8x1axizezb9a3lfnpzjho98evi3ij450i96vj0a8jpf36au26
   ```
5. select the 'Drive' letter of any available drive you want to map the file system to.
6. If needed, select the Connect using different credentials check box.
7. Click Finish.
8. When prompted, provide the user name and password of the AD domain user used for mapping the network drive.
9. Click OK.
10. In a Command Prompt window (cmd), verify that the drive is properly mapped using this command:

```
C:\>net use
New connections will be remembered.
Status       Local     Remote                    Network
-------------------------------------------------------------------------------
OK           Z:        \
\10.0.0.2\uvj1iw6ytyecqijcbdgpy7ec15mgsv044i7609giqx7ukfn6t2pwgfqot0ma
                                                 Microsoft Windows Network
The command completed successfully.
C:\>
```

Managing Mount Targets and Exports

A mount target is an NFS endpoint assigned to a VCN subnet of your choice and provides network access for file systems. The mount target provides the IP address or DNS name that is used together with a unique export path to mount the file system.

For an instance to mount a file system, the instance's VCN must have a mount target.

You can reuse the same mount target to make as many file systems available on the network as you want. To reuse the same mount target for multiple file systems, create an export in the mount target for each file system.

> **Important:**
>
> When more than one file system is exported to the same mount target, you must export to the mount target with the smallest network (largest CIDR number) first. For detailed information and instructions, refer to My Oracle Support PCA File system as a Service Exports (Doc ID 2823994.1)

For instructions to create a mount target, see Creating a Mount Target.

For more conceptual information, refer to the File Storage Overview chapter in the *Oracle Private Cloud Appliance Concepts Guide*.

This section provides instructions for administering mount targets.

Listing Mount Targets and Viewing Details

Using the Compute Web UI

1. In the navigation menu, under File Storage, click Mount Targets.
2. Select the compartment where the mount target resides.

   The mount targets are displayed.
3. To see the mount target details, click the mount target name.

Using the OCI CLI

- **Listing Mount Targets**
  1. Get the OCID of the compartment where you want to list mount targets (`oci iam compartment list`)
  2. Run this command.

Syntax (entered on a single line):

```
oci fs mount-target list
--availability-domain AD-1 \
--compartment-id <compartment_id>
```

Example:

```
oci fs mount-target list --availability-domain AD-1 \
--compartment-id ocid1.compartment.uniqueID

{
  "data": [
    {
      "availability-domain": "AD-1",
      "compartment-id": "ocid1.compartment.uniqueID",
      "defined-tags": {},
      "display-name": "MyMountTarget",
      "export-set-id": "ocid1.exportset.uniqueID",
      "freeform-tags": {},
      "id": "ocid1.mounttarget.uniqueID",
      "lifecycle-state": "ACTIVE",
      "nsg-ids": null,
      "private-ip-ids": [
        "ocid1.privateip.uniqueID"
      ],
      "subnet-id": "ocid1.subnet.uniqueID",
      "time-created": "2021-07-16T22:56:57+00:00"
    },
    {
      "availability-domain": "AD-1",
      "compartment-id": "ocid1.compartment.uniqueID",
      "defined-tags": {},
      "display-name": "AnotherMountTarget",
      "export-set-id": "ocid1.exportset.uniqueID",
      "freeform-tags": {},
      "id": "ocid1.mounttarget.uniqueID",
      "lifecycle-state": "ACTIVE",
      "nsg-ids": [],
      "private-ip-ids": [
        "ocid1.privateip.uniqueID"
      ],
      "ocid1.privateip.uniqueID"
      "subnet-id": "ocid1.subnet.uniqueID",
      "time-created": "2021-06-16T22:56:57+00:00"
    }
  ]
}
```

- **Getting Mount Target Details**

  1. Gather the information that you need to run the command:
     - Mount target ID (`oci fs mount-target list`)
  2. Run this command.

     Syntax (entered on a single line):

     ```
     oci fs mount-target get
     --mount-target-id <mount_target_OCID>
     ```

     Example:

```
oci fs mount-target get  \
--mount-target-id ocid1.mounttarget.….….….uniqueID
{
  "data": {
      "availability-domain": "AD-1",
      "compartment-id": "ocid1.compartment.uniqueID",
      "defined-tags": {},
      "display-name": "MyMountTarget",
      "export-set-id": "ocid1.exportset.uniqueID",
      "freeform-tags": {},
      "id": "ocid1.mounttarget.uniqueID",
      "lifecycle-state": "ACTIVE",
      "nsg-ids": null,
      "private-ip-ids": [
        "ocid1.privateip.uniqueID"
      ],
      "subnet-id": "ocid1.subnet.uniqueID",
      "time-created": "2021-07-16T22:56:57+00:00"
    }
}
```

Updating a Mount Target

When you update a mount target, you can set or change the Network Security Groups and display name. You cannot change the backing store pool type assignment (default or high performance).

Using the Compute Web UI

1. In the navigation menu, under File Storage, click Mount Targets.
2. Select the compartment where the mount target resides.
3. Click the Action menu (three dots) for the mount target, and select Edit.
4. Make changes.
5. Click Save.

Using the OCI CLI

1. Get the OCID of the mount target (`oci fs mount-target list`)
2. Run this command.

   Syntax:

   ```
   oci fs mount-target update --mount-target-id mount_target_OCID \
   parameters to change
   ```

   Example:

   ```
   oci fs mount-target update
   --mount-target-id ocid1.mounttarget.uniqueID \
   --display-name "MyMountTarget"

   {
     "data": {
       "availability-domain": "AD-1",
       "compartment-id": "ocid1.compartment.uniqueID",
       "defined-tags": {},
       "display-name": "MyMountTarget",
       "export-set-id": "ocid1.exportset.uniqueID",
       "freeform-tags": {},
   ```

```
    "id": "ocid1.mounttarget.uniqueID",
    "lifecycle-details": null,
    "lifecycle-state": "ACTIVE",
    "nsg-ids": null,
    "private-ip-ids": [
      "ocid1.privateip.uniqueID"
    ],
    "subnet-id": "ocid1.subnet.uniqueID",
    "time-created": "2021-06-17T19:01:37+00:00"
  },
  "etag": "b7efb0d7-d5fb-45d8-8bdd-a4a2f3f0371d"
}
```

Listing Exports

Using the Compute Web UI

1. In the navigation menu, under File Storage, click Mount Targets.
2. Select the compartment where the mount target resides.
3. Click the mount target name.

   The exports are display at the bottom of the page.
4. To see the export details, click the export name.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Compartment OCID (`oci iam compartment list`)
2. Run this command.

   Syntax (entered on a single line):

   ```
   oci fs export list
   --compartment-id <compartment_id>
   ```

   Example:

   ```
   oci fs export list  \
   --compartment-id ocid1.….….….….uniqueID
   {
     "data": [
       {
         "export-set-id": "ocid1.exportset.….….….uniqueID",
         "file-system-id": "ocid1.filesystem.….….….uniqueID",
         "id": "ocid1.export.….….….uniqueID-1",
         "lifecycle-state": "ACTIVE",
         "path": "/export/8g0afgj16nuwx77a4wublc3ekkdaekef1bct2zt8qcbukfsconxmkp9su0ys",
         "time-created": "2021-06-17T21:15:44+00:00"
       },
       {
         "export-set-id": "ocid1.exportset.….….….uniqueID",
         "file-system-id": ".….….….uniqueID",
         "id": "ocid1.export.….….….uniqueID-2",
         "lifecycle-state": "ACTIVE",
         "path": "/export/8g0afgj16nuwx77a4wublc3ekkdaekef1bct2zt8qcbukfsconxmkp9su0ys",
         "time-created": "2021-06-17T21:20:55+00:00"
       }
     ]
   }
   ```

Listing Export Sets

Using the OCI CLI

1. Get the compartment where you want to list export sets (`oci iam compartment list`)
2. Run this command.

   Syntax (entered on a single line):

   ```
   oci fs export-set list
   --availability-domain AD-1 \
   --compartment-id compartment_OCID
   ```

   Example:

   ```
   oci fs export-set list --availability-domain AD-1 \
   --compartment-id ocid1.compartment.uniqueID

   {
     "data": [
       {
         "availability-domain": "AD-1",
         "compartment-id": "ocid1.compartment.uniqueID",
         "display-name": "MyMountTarget2 - export set",
         "id": "ocid1.exportset.uniqueID",
         "lifecycle-state": "ACTIVE",
         "time-created": "2021-06-17T19:01:37+00:00",
         "vcn-id": "ocid1.vcn.uniqueID"
       }
     ]
   }
   ```

Deleting an Export

Deleting an export deletes the file system path that clients use to mount the file system. Deleting an export does not delete any file systems.

> **Caution:**
>
> When you delete an export, you can no longer mount the file system using the file path specified in the deleted export. Any clients that use the export path to mount a file system will not be able to access the file system.

Using the Compute Web UI

1. In the navigation menu, under File Storage, click File Systems.
2. Select the appropriate compartment.
3. Click the name of a file system that uses the export you plan to delete.
4. Click the Action menu (three dots) for the export and select Delete.
5. Confirm the deletion.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - export OCID (`oci fs file-system list`)
2. Run this command.

   Syntax (entered on a single line):

   ```
   oci fs export delete
   --export-id <export_OCID>
   ```

   Example:

   ```
   oci fs export delete --export-id ocid1.export.….….….uniqueID
   Are you sure you want to delete this resource? [y/N]: y
   ```

Moving a Mount Target to a Different Compartment

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Mount target OCID (`oci fs mount-target list`)
   - Destination Compartment OCID (`oci iam compartment list`)
2. Run this command.

   Syntax (entered on a single line):

   ```
   oci fs mount-target change-compartment
   --mount-target-id <mount_target_OCID>
   --compartment-id <destination_compartment_OCID>
   ```

   Example:

   ```
   oci fs mount-target change-compartment  \
   --mount-target-id ocid1.….….….uniqueID  \
   --compartment-id ocid1.compartment.….….….uniqueID
   {
     "etag": "864d51bd-ed69-44bc-8c54-2a65d55fe07b"
   }
   ```

Deleting a Mount Target

> **Caution:**
>
> Deleting a mount target deletes all the exports that are associated with the mount target.

Using the Compute Web UI

1. In the navigation menu, under File Storage, click Mount Targets.
2. Select the compartment where the mount target resides.
3. Click the Action menu (three dots) for the mount target you plan to delete.

4. Select Delete.
5. Confirm the deletion.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Mount target OCID (`oci fs mount-target list`)
2. Run this command.

   Syntax (entered on a single line):

   ```
   oci fs mount-target delete
   --mount-target-id <mount_target_OCID>
   ```

   Example:

   ```
   oci fs mount-target delete  \
   --mount-target-id ocid1.mounttarget.….….….uniqueID
   Are you sure you want to delete this resource? [y/N]: y
   ```

Managing File Systems

A file system in the File Storage service represents a network file system that is mounted by one or more clients. File systems are associated with a single compartment. File systems must have at least one export in one mount target for any client to mount and use the file system. Data is added to a file system from the client.

This section describes how to manage file systems after they are created. For instructions to create a file system, see Creating a File System, Mount Target, and Export.

Listing and Viewing the Details of a File System

Using the Compute Web UI

1. In the navigation menu, under File Storage, click File Systems.
2. Select the appropriate compartment.

   The file systems for the compartment are listed.
3. To see file system details, click the name of the file system.

Using the OCI CLI

- **Listing File Systems**
  1. Get the OCID of the compartment where you want to list file systems (`oci iam compartment list`)
  2. Run this command.

     Syntax (entered on a single line):

     ```
     oci fs file-system list --availability-domain AD-1 \
     --compartment-id compartment_OCID
     ```

     Example:

     ```
     oci fs file-system list --availability-domain AD-1 \
     --compartment-id ocid1.compartment.uniqueID
     {
     ```

```
  "data": [
    {
      "availability-domain": "AD-1",
      "compartment-id": "ocid1.compartment.uniqueID",
      "defined-tags": {},
      "display-name": "MyFileSystem",
      "freeform-tags": {},
      "id": "ocid1.filesystem.uniqueID",
      "kms-key-id": null,
      "lifecycle-state": "ACTIVE",
      "metered-bytes": 180224,
      "time-created": "2021-06-16T19:48:18+00:00"
    },
    {
      "availability-domain": "AD-1",
      "compartment-id": "ocid1.compartment.uniqueID",
      "defined-tags": {},
      "display-name": "pluto",
      "freeform-tags": {},
      "id": "ocid1.filesystem.uniqueID",
      "kms-key-id": null,
      "lifecycle-state": "ACTIVE",
      "metered-bytes": 147456,
      "time-created": "2021-06-17T23:16:43+00:00"
    }
  ]
}
```

- **Getting the File System Details**

  1. Gather the information that you need to run the command:
     - File System OCID (`oci fs file-system list`)
  2. Run this command.

> **Note:**
>
> This procedure shows the minimum required parameters for this command. For information about optional parameters, run the command with the `--help` option.

Syntax (entered on a single line):

```
oci fs file-system get --file-system-id file_system_OCID
```

Example:

```
oci fs file-system get  \
--file-system-id ocid1.filesystem.uniqueID
{
  "data": {
    "availability-domain": "AD-1",
    "compartment-id": "ocid1.compartment.uniqueID",
    "defined-tags": {},
    "display-name": "MyFileSystem",
    "freeform-tags": {},
    "id": "ocid1.filesystem.uniqueID",
    "kms-key-id": null,
    "lifecycle-state": "ACTIVE",
```

```
    "metered-bytes": 180224,
    "time-created": "2021-06-16T19:48:18+00:00"
  },
  "etag": "58dec47e-4732-4730-9e18-6b5db1ac30d6"
}
```

Updating a File System

You can change the file system name and quota. You cannot change the database record size or the backing store pool type assignment (default or high performance).

To reduce the quota for the file system, first check the current file system usage. The quota cannot be set smaller than the current usage. Current usage includes the data in the file system and all snapshots created under the file system. To check the current usage, check the value of Metered Bytes on the file system details page in the Compute Web UI or `metered-bytes` in the file system `get` output in the OCI CLI.

> **Note:**
>
> The metered bytes value can take up to 15 minutes to refresh on a system with active I/O.

Fifteen minutes after setting a lower quota, compare the quota and metered bytes values for the file system. Check the value of quota in the OraclePCA defined tag on the Tags tab on the file system details page in theCompute Web UI or in `defined-tags` in the file system `get` output in the OCI CLI. If the quota is less than metered bytes, then the quota will not be enforced and you should set a higher quota.

Using the Compute Web UI

1. On the Dashboard, click the File Storage/View File Systems button.
2. If the file system that you want to update is not listed, use the Compartment drop-down menu above the instances list to select the correct compartment.
3. For the file system that you want to update, click the Actions menu, and click the Edit option.
4. Enter a new name in the name field, or enter a new quota in the OraclePCA:quota defined tag as described in Creating a File System. See the note at the beginning of this topic about reducing the quota.
5. Click Save Changes.

Using the OCI CLI

1. Get the OCID of the file system that you want to update: `oci fs file-system list`
2. Specify a new name for the file system, or set a new quota in the OraclePCA:quota defined tag as described in Creating a File System. See the note at the beginning of this topic about reducing the quota.
3. Run the update file system command.

   Example:

```
$ oci fs file-system update
--file-system-id ocid1.filesystem.unique_ID \
--defined-tags '{"OraclePCA":{"quota":500000}}'
```

Moving a File System to a Different Compartment

Using the OCI CLI

1. Gather the information that you need to run the command:
   - File System OCID (`oci fs file-system list`)
   - Destination compartment OCID (`oci iam compartment list`)
2. Run this command.

   Syntax (entered on a single line):

   ```
   oci fs file-system change-compartment
   --file-system-id <file-system_OCID>
   --compartment-id <destination_compartment_OCID>
   ```

   Example:

   ```
   oci fs file-system change-compartment  \
   --file-system-id ocid1.filesystem.….….….uniqueID  \
   --compartment-id ocid1.compartment.….….….destination-uniqueID
   {
     "etag": "0acc73ca-839d-451e-b079-4013889c233a"
   }
   ```

Deleting a File System

A file system that has an export cannot be deleted. To delete the export, see Deleting an Export.

You cannot delete file systems that have dependencies. For example, if you have created a snapshot of this file system and then created a new file system from the snapshot, you cannot delete the source file system. For details, see File Storage Overview in the Oracle Private Cloud Appliance Concepts Guide.

Using the Compute Web UI

1. In the navigation menu, under File Storage, click File Systems.
2. Select the appropriate compartment.
3. Click the Action menu (three dots) for the file system and select Delete.
4. Confirm the deletion.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - File System OCID (`oci fs file-system list`)
2. Run this command.

   Syntax (entered on a single line):

   ```
   oci fs file-system delete
   --file-system-id <file-system_OCID>
   ```

Example:

```
oci fs file-system delete  \
--file-system-id ocid1.filesystem.….….….uniqueID
Are you sure you want to delete this resource? [y/N]: y
```

Managing Snapshots

The File Storage service supports snapshots for data protection of your file system.

Snapshots are a consistent, point-in-time view of your file systems. Snapshots are copy-on-write, and scoped to the entire file system. The File Storage service encrypts all file system and snapshot data at rest. You can take as many snapshots as you need.

For more conceptual information, refer to Snapshots in the File Storage Overview chapter of the Oracle Private Cloud Appliance Concepts Guide.

This section provides instructions for managing file system snapshots.

Listing and Getting Snapshot Details

Using the Compute Web UI

1. In the navigation menu, under File Storage, click File Systems.
2. Select the appropriate compartment.
3. Click the file system name.
4. In the Resources panel, click Snapshots.

   The file system snapshots are listed.
5. To get the details for a specific snapshot, click the snapshot name.

Using the OCI CLI

- Listing Snapshots
  1. Gather the information that you need to run the command:
     - File system OCID (`oci fs file-system list`)
  2. Run this command.

     Syntax (entered on a single line):

     ```
     oci fs snapshot list
     --file-system-id <file-system_OCID>
     ```

     Example:

     ```
     oci fs snapshot list  \
     --file-system-id ocid1.filesystem.….….….uniqueID
     {
       "data": [
         {
           "defined-tags": {},
           "file-system-id": "ocid1.filesystem.….….….uniqueID
     ",
           "freeform-tags": {},
           "id": "ocid1.snapshot.….….….uniqueID-1",
           "lifecycle-state": "ACTIVE",
           "name": "MySnapshot",
     ```

```
      "time-created": "2021-06-21T17:12:37+00:00"
    }
  ]
    {
      "defined-tags": {},
      "file-system-id": "ocid1.filesystem.….….….uniqueID",
      "freeform-tags": {},
      "id": "ocid1.snapshot.….….….uniqueI-2",
      "lifecycle-state": "ACTIVE",
      "name": "MySnapshot2",
      "time-created": "2021-06-21T17:31:18+00:00"
    }
  ]
}
```

- Getting a Specific Snapshot

  1. Gather the information that you need to run the command:
     - Snapshot OCID (`oci fs snapshot list`)
  2. Run this command.

     Syntax (entered on a single line):

     ```
     oci fs snapshot get  \
     --snapshot-id <snapshot_OCID>
     ```

     Example:

     ```
      oci fs snapshot get --snapshot-id ocid1.snapshot.….….….uniqueID
     {
       "data": {
         "defined-tags": {},
         "file-system-id": "ocid1.filesystem.….….….uniqueID",
         "freeform-tags": {},
         "id": "ocid1.snapshot.….….….uniqueID",
         "lifecycle-state": "ACTIVE",
         "name": "MySnapshot",
         "time-created": "2021-06-21T17:12:37+00:00"
       },
       "etag": "f38aa070-0f3e-407f-a0b4-9bc841ff3fa4"
     }
     ```

Creating a Snapshot

You can create a snapshot of a file system. A snapshot is a point-in-time view of the file system. The snapshot is accessible at `.zfs/snapshot/`***`name`***.

Using the Compute Web UI

1. In the navigation menu, under File Storage, click File Systems.
2. Select the appropriate compartment.
3. Click the file system name.
4. In the Resources panel, click Snapshots.
5. Click Create Snapshot.
6. Enter a name for the snapshot.

   The name is limited to 64 characters and it must be unique among all other snapshots for this file system. The name can't be changed. Avoid entering confidential information.

7. Click Create Snapshot.

   The snapshot is accessible under the root directory of the file system at `.zfs/snapshot/`***`name`***.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - File system OCID (`oci fs file-system list`)
   - Snapshot name of your choice. The name is limited to 64 characters and it must be unique among all other snapshots for this file system. The name can't be changed. Avoid entering confidential information.
2. Run this command.

> **Note:**
>
> This procedure shows the minimum required parameters for this command. For information about optional parameters, run the command with the `--help` option.

Syntax (entered on a single line):

```
oci fs snapshot create
--file-system-id <file-system_OCID>
--name <snapshot_name>
```

Example:

```
oci fs snapshot create  \
--file-system-id ocid1.filesystem.….….….uniqueID  \
--name "MySnapshot"
{
  "data": {
    "defined-tags": {},
    "file-system-id": "ocid1.filesystem.….….….uniqueID",
    "freeform-tags": {},
    "id": "ocid1.snapshot.….….….uniqueID",
    "lifecycle-state": "CREATING",
    "name": "MySnapshot",
    "time-created": null
  },
  "etag": "f38aa070-0f3e-407f-a0b4-9bc841ff3fa4"
}
```

Accessing a Snapshot on the Mounted File System

When a file system snapshot is created, the snapshot is placed in the file system. If the file system is mounted in a client system, you can access the snapshot on the client system.

The snapshot is accessible in this directory path: ***`<mount-point>`***`/.zfs/snapshot/`***`<snapshot-name>`***.

Using a UNIX OS

1. Log into the instance OS that has the mounted the file system from which the snapshot was made.

2. List the snapshots.

   Syntax:

   ```
   ls -la <mount-point>/.zfs/snapshot/
   ```

   Example:

   ```
   ls -la  /mnt/MyMountPoint/.zfs/snapshot
   total 17
   dr-xr-xr-x. 4 root root 4 Sep  8 15:54 .
   dr-xr-xr-x. 4 root root 4 Sep  1 17:27 ..
   drwxr-xr-x. 4 root root 7 Sep  8 15:53 file-system-FS-snapshot-02
   drwxr-xr-x. 4 root root 6 Sep  1 18:12 file-system-FS-snapshot-01
   ```

3. Change to the directory of a snapshot.

   Example:

   ```
   cd /mnt/MyMountPoint/.zfs/snapshot/file-system-FS-snapshot-02
   ```

4. List the contents of the snapshot.

   ```
   ls -la
   total 3027
   drwxr-xr-x. 4 root root       7 Sep  8 15:53 .
   dr-xr-xr-x. 4 root root       4 Sep  8 15:54 ..
   -rwxr-xr-x. 1 root root     429 Sep  8 15:53 example1
   drwxr-x---. 2 root sys        3 Sep  1 17:28 .$EXTEND
   drwxr-xr-x. 2 root root       2 Sep  1 18:10 ABC-directory
   -rw-r--r--. 1 root root       0 Sep  1 18:10 xyz-file
   -rw-r--r--. 1 root root 3073219 Sep  1 18:12 zap.zip
   ```

Restoring a Snapshot (UNIX-Based Instances)

You can restore individual snapshot files or an entire snapshot using the `cp` command.

> **Note:**
>
> Optionally, you can use `rsync`, `tar`, or another tool that supports NFS to copy your data to another remote location.

Using the Instance OS

1. Log into the instance OS that has the mounted the file system from which the snapshot was made.
2. List the snapshots.

   Syntax:

   ```
   ls -la <mount-point>/.zfs/snapshot/
   ```

   Example:

   ```
   ls -la  /mnt/MyMountPoint/.zfs/snapshot
   total 17
   dr-xr-xr-x. 4 root root 4 Sep  8 15:54 .
   dr-xr-xr-x. 4 root root 4 Sep  1 17:27 ..
   drwxr-xr-x. 4 root root 7 Sep  8 15:53 file-system-FS-snapshot-02
   drwxr-xr-x. 4 root root 6 Sep  1 18:12 file-system-FS-snapshot-01
   ```

3. Use the `cp` command to copy individual snapshot files, or the entire snapshot to a location of your choice.

   Use the `-r` option when restoring a snapshot that contains subdirectories.

   Example:

   ```
   cp -r /mnt/MyMountPoint/.zfs/snapshot/<snapshot_name>/* <destination_directory>
   ```

Deleting a Snapshot

There are dependencies between file systems, snapshots, and clones. The appliance will not allow you to delete any resources for which there is a dependency. For details, see File Storage Overview in the Oracle Private Cloud Appliance Concepts Guide.

Using the Compute Web UI

1. In the navigation menu, under File Storage, click File Systems.
2. Select the appropriate compartment.
3. Click the name of the file system where the snapshot resides.
4. In the Resources panel, click Snapshots.
5. Click the Actions icon (three dots), and then click Delete.
6. Confirm the deletion.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Snapshot OCID (`oci fs snapshot list`)
2. Run this command.

   Syntax (entered on a single line):

   ```
   oci fs snapshot delete
   --snapshot-id <snapshot_OCID>
   ```

   Example:

   ```
   oci fs snapshot delete  \
   --snapshot-id ocid1.snapshot.….….….uniqueID
   Are you sure you want to delete this resource? [y/N]: y
   ```

Managing Clones

A clone is a new file system that is created based on a snapshot of an existing file system. Snapshots preserve the state of the data of a file system at a particular point in time. If you take snapshots of a file system at regular intervals, you can create clones of the file system as it existed at multiple points in its lifetime.

Cloned file systems are managed in the same way that any other file system is managed. See Managing File Systems.

Creating a File System Clone

Prerequisite

A snapshot of the file system must exist. See Creating a Snapshot.

Using the Compute Web UI

1. On the Dashboard, click the File Storage/View File Systems button.

   If necessary, select a different compartment from the menu above the list of file systems.
2. Click the name of the file system that you want to clone.
3. In the Resources box on the file system details page, click Snapshots.
4. For the snapshot that you want to clone, click the Actions menu and click Clone.
5. In the Clone Snapshot from File System dialog, provide the following information:
   - **Name**: Enter a name for the clone.
   - **Create in Compartment**: Select the compartment where you want to create the clone.
   - **Tagging**: (Optional) Add defined or free-form tags for this instance as described in Adding Tags at Resource Creation. Tags can also be applied later.
6. Click the Create File System button.

   The clone is created and the details page of the new file system clone is displayed.

   In the Clones section of the file system clone details page, Clone Parent is false and Clone Root is false. The name of the Parent File System is shown and is clickable, and the name of the Source Snapshot is shown and is clickable.

   If you click the name of the parent file system, the Clones section of the details page of that parent file system shows Clone Parent is true, Descendants is true, and Parent File System and Source Snapshot have no values.

Using the OCI CLI

1. Gather the information that you need to run the command:
   - Compartment OCID (`oci iam compartment list`)
   - Display Name: The display name you want assigned to this file system clone
   - Source snapshot OCID (`oci fs snapshot list`)
2. Run the create clone command.

   Syntax:

   ```
   oci fs file-system create --availability-domain AD-1 \
   --compartment-id compartment_OCID --display-name fs_clone_display_name \
   --source-snapshot-id fs_snapshot_OCID
   ```

   Example:

   ```
   oci fs file-system create --availability-domain AD-1 \
   --compartment-id ocid1.compartment.unique_ID \
   --display-name fs-1-clone-1 \
   --source-snapshot-id ocid1.snapshot.unique_ID
   ```

Deleting a File System Clone

Cloned file systems are managed in the same way that any other file system is managed; you delete a clone the same way you delete a file system. However, dependencies exist between file systems, snapshots, and clones. The appliance will not allow you to delete any resources that have a dependency. For details, see File Storage Overview in the Oracle Private Cloud Appliance Concepts Guide.

To delete a file system clone, see Deleting a File System.

12 Object Storage

The Object Storage service is a storage platform that offers reliable and cost-efficient data durability.

The Object Storage service stores unstructured data of any content type, including analytic data and rich content, like images and videos.

The data is stored as an object in a bucket. Buckets are associated with a compartment within a tenancy.

An Object Storage namespace serves as the top-level container for all buckets and objects. At account creation time, each tenant is assigned one unique system-generated and immutable Object Storage namespace name. The namespace spans all compartments.

With Object Storage, you can safely and securely store or retrieve data directly from the internet or from within the cloud appliance.

For more conceptual information, refer to the Object Storage Overview section in the Oracle Private Cloud Appliance Concepts Guide.

This chapter provides instructions for managing Object Storage.

Obtaining the Object Storage Namespace

An Object Storage namespace serves as the top-level container for all buckets and objects. Each tenant is assigned one unique system-generated and immutable Object Storage namespace name. The namespace spans all compartments. The namespace name is a required argument for many Object Storage CLI commands.

Using the Compute Web UI

1. Click your user name (upper right corner), and select Tenancy.

   The namespace string is listed under Object Storage Settings.

Using the OCI CLI

1. Run the following command and add the `iaas` endpoint to the command. For example:

```
oci os ns get --endpoint https://iaas.<mypca>.example.com
{
  "data": "<myobjstor_namespace_name>"
}
```

Managing Object Storage Buckets

A bucket is a container for storing objects in a compartment within an Object Storage namespace.

A bucket is associated with a single compartment. The compartment has policies that indicate what actions you can perform on a bucket and all objects in the bucket.

A bucket cannot contain other buckets.

For more conceptual information, refer to the Object Storage Overview section in the Oracle Private Cloud Appliance Concepts Guide.

Listing Buckets

Using the Compute Web UI

1. In the navigation menu, under Object Storage, click Object Storage.

   A list of the buckets in the compartment you're viewing is displayed.

2. If you don't see the bucket you're looking for, ensure that you're viewing the correct compartment (select from the list at the top of the page).

   The page shows only the resources in that compartment.

Using the OCI CLI

1. Gather the information you need to run the command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Compartment OCID (`oci iam compartment list`)
2. Run this command.

   Syntax (entered on a single line):

   ```
   oci os bucket list
   --namespace-name <object_storage_namespace>
   --compartment-id <compartment_OCID>
   ```

   Example:

   ```
   oci os bucket list  \
   --namespace-name examplenamespace  \
   --compartment-id ocid.compartment.….….….uniqueID

   {
     "data": [
       {
         "compartment-id": "ocid.compartment.….….….uniqueID",
         "created-by": "ocid1.user.….….….uniqueID",
         "defined-tags": null,
         "etag": "cdb5bc11561e476cb0d8aa5b8f8668f6",
         "freeform-tags": null,
         "name": "MyBucket",
         "namespace": "export/examplenamespace",
         "time-created": "2021-05-04T18:56:39+00:00"
       },
       {
         "compartment-id": "ocid.compartment.….….….uniqueID",
         "created-by": "ocid1.user.….….….uniqueID",
         "defined-tags": null,
         "etag": "aa7642fec45729ce7cb8b321d3ee1463",
         "freeform-tags": null,
         "name": "JoesBucket",
         "namespace": "export/examplenamespace",
         "time-created": "2021-05-04T20:26:33+00:00"
   ```

```
    }
  ]
}
```

Viewing Bucket Details

Use this task to view bucket details.

Using the Compute Web UI

1. In the navigation menu, under Object Storage, click Object Storage.

   A list of the buckets in the compartment you are viewing is displayed.
2. From the list at the top of the page, select the compartment where the bucket resides.
3. Click the bucket name to display the details.
4. Click View or Copy.

Using the OCI CLI

1. Gather the information you need to run the command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Bucket name (`oci os bucket list`), see Listing Buckets
2. Run this command.

   Syntax (entered on a single line):

   ```
   oci os bucket get
   --namespace-name <object_storage_namespace>
   --bucket-name <bucket_name>
   ```

   The OCID is identified as `id` in the output.

   Example:

   ```
   oci os bucket get  \
   --namespace-name examplenamespace  \
   --bucket-name MyBucket

   {
     "data": {
       "approximate-count": null,
       "approximate-size": null,
       "compartment-id": "ocid.compartment.….….….uniqueID",
       "created-by": "ocid1.user.….….….uniqueID",
       "defined-tags": null,
       "etag": "cdb5bc11561e476cb0d8aa5b8f8668f6",
       "freeform-tags": null,
       "id": ocid.bucket.….….….uniqueID,
       "is-read-only": null,
       "kms-key-id": null,
       "metadata": null,
       "name": "MyBucket",
       "namespace": "export/examplenamespace",
       "object-events-enabled": null,
       "object-lifecycle-policy-etag": null,
       "public-access-type": "NoPublicAccess",
       "replication-enabled": null,
       "storage-tier": "Standard",
       "time-created": "2021-05-04T18:56:39+00:00",
   ```

```
    "versioning": null
  },
  "etag": "cdb5bc11561e476cb0d8aa5b8f8668f6"
}
```

Creating a Bucket

Use this procedure to create an Object Storage bucket.

When you create a bucket, the bucket does not provide public access. To make the bucket publicly available, see Using Pre-Authenticated Requests.

Using the Compute Web UI

1. In the navigation menu, click Object Storage, then click Object Storage.
2. Click Create Bucket.
3. Enter the following details:
   - **Name:** Enter a name for the bucket.

     Specify a name that is unique within your tenancy Object Storage namespace.
   - **Create in Compartment:** Select the compartment in which to create this bucket.
   - **Enable Object Versioning:** Optionally, you can enable object versioning.

     For more information, refer to Managing Object Versioning.
   - **Tagging:** Optionally, add one or more tags to this resource.

     If you are not sure whether to apply tags, skip this option (you can apply tags later).

     For more information about tagging resources, see Working with Resource Tags.
4. Click Create Bucket.

   The bucket is created immediately and you can start uploading objects. See Uploading an Object.

Using the OCI CLI

1. Gather the information you need to run the next command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Compartment OCID (`oci iam compartment list`)
   - Bucket name: The name you want for this bucket.
2. Run this command.

   Syntax (entered on a single line):

   ```
   oci os bucket create
   --namespace-name <object_storage_namespace>
   --compartment-id <compartment_OCID>
   --name <bucket_name>
   ```

   The bucket is created immediately and you can start uploading objects. See Uploading an Object.

   Example:

   ```
   oci os bucket create \
   --namespace-name examplenamespace  \
   --compartment-id ocid.compartment.….….….uniqueID    \
   ```

```
--name MyBucket

{
  "data": {
    "approximate-count": null,
    "approximate-size": null,
    "compartment-id": "ocid1.compartment.….….….uniqueID",
    "created-by": "ocid1.user.….….….uniqueID",
    "defined-tags": null,
    "etag": "b78d4193ab3eb2270b1373aa52b443a1",
    "freeform-tags": null,
    "id": null,
    "is-read-only": null,
    "kms-key-id": null,
    "metadata": null,
    "name": "MyBucket",
    "namespace": "export/examplenamespace",
    "object-events-enabled": null,
    "object-lifecycle-policy-etag": null,
    "public-access-type": "NoPublicAccess",
    "replication-enabled": null,
    "storage-tier": "Standard",
    "time-created": "2021-06-11T20:11:02+00:00",
    "versioning": null
  },
  "etag": "b78d4193ab3eb2270b1373aa52b443a1"
}
```

Moving a Bucket to a Different Compartment

You can move a bucket from one compartment to another as long as both the source and target compartments are in the same tenancy. This capability includes moving a bucket from one compartment level down to a sublevel within the source compartment.

Using the OCI CLI

1. Gather the information you need for the next command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Compartment OCID of the compartment you are moving the bucket to (`oci iam compartment list`)
   - Bucket name (`oci os bucket list`), see Listing Buckets
2. Run this command to move the bucket.

   Syntax (entered on a single line):

   ```
   oci os bucket update
   --namespace-name <object_storage_namespace>
   --compartment-id <target_compartment_id>
   --bucket-name <bucket_name>
   ```

   Example:

   ```
   oci os bucket update \
   --namespace-name examplenamespace  \
   --compartment-id ocid1.compartment.….….….target-compartmentID   \
   --bucket-name MyBucket
   {
     "data": {
       "approximate-count": null,
   ```

```
    "approximate-size": null,
    "compartment-id": "ocid1.compartment.….….….target-compartmentID",
    "created-by": "ocid1.user.….….….uniqueID",
    "defined-tags": null,
    "etag": "5d72fb7ac4385e24f42ac830bc6490ca",
    "freeform-tags": null,
    "id": null,
    "is-read-only": null,
    "kms-key-id": null,
    "metadata": null,
    "name": "MyBucket",
    "namespace": "export/examplenamespace",
    "object-events-enabled": null,
    "object-lifecycle-policy-etag": null,
    "public-access-type": "NoPublicAccess",
    "replication-enabled": null,
    "storage-tier": "Standard",
    "time-created": "2021-06-02T20:44:57+00:00",
    "versioning": null
  },
  "etag": "5d72fb7ac4385e24f42ac830bc6490ca"
}
```

3. Run this command to verify that the bucket moved to the correct compartment:

   Syntax (entered on a single line):

   ```
   oci os bucket list
   --namespace-name <object_storage_namespace>
   --compartment-id <target_compartment_OICD>
   ```

   Example:

   ```
   oci os bucket list  \
   --namespace-name examplenamespace  \
   --compartment-id ocid1.compartment.….….….target-compartmentID
   {
     "data": [
       {
         "compartment-id": "ocid1.compartment.….….….target-compartmentID",
         "created-by": "ocid1.user.….….….uniqueID",
         "defined-tags": null,
         "etag": "5d72fb7ac4385e24f42ac830bc6490ca",
         "freeform-tags": null,
         "name": "MyBucket",
         "namespace": "export/examplenamespace",
         "time-created": "2021-06-02T20:44:57+00:00"
       }
   ```

Deleting a Bucket

> **Caution:**
>
> You cannot recover a deleted bucket.

You can permanently delete an empty bucket. You cannot delete a bucket that contains any of the following:

- Any objects

- Previous versions of an object
- A multipart upload in progress
- A pre-authenticated request

> **Tip:**
>
> When you delete an object in a version-enabled bucket, a previous version of that object is created. Select Show Deleted Objects to display the object versions that might prevent you from deleting the bucket. For more information, see Managing Object Versioning.

Using the OCI CLI

1. Gather the information you need to run the command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Bucket name (`oci os bucket list`), see Listing Buckets
2. Run this command.

   Syntax (entered on a single line):

   ```
   oci os bucket delete
   --namespace-name <object_storage_namespace>
   --bucket-name <bucket_name>
   ```

   Example:

   ```
   oci os bucket delete   \
   --namespace-name examplenamespace  \
   --bucket-name MyBucket
   Are you sure you want to delete this resource? [y/N]: y
   ```

Managing Storage Objects

In the Object Storage service, an object is a file or unstructured data you upload to a bucket within a compartment within an Object Storage namespace.

The object can be any type of data, for example, multimedia files, data backups, static web content, or logs. You can store objects that are up to 10 TiB. Objects are processed as a single entity. You can't edit or append data to an object, but you can replace the entire object.

Object Storage is not tied to any specific compute instance. You can access data from anywhere inside or outside of the Oracle Private Cloud Appliance, as long you have internet connectivity, access to the Object Storage endpoint, and authorization.

For more conceptual information, refer to the Object Storage Overview section in the Oracle Private Cloud Appliance Concepts Guide.

Viewing Objects in a Bucket

Using the Compute Web UI

1. In the navigation menu, under Object Storage, click Object Storage.

2. Choose the compartment that contains the bucket that contains your object.

   A list of buckets is displayed.

3. Click the bucket name that contains your object.
4. Click Objects under Resources.

Using the OCI CLI

- **Listing Objects in a bucket**

  1. Gather the information you need to run the command.
     - Namespace (see Obtaining the Object Storage Namespace)
     - Bucket name (`oci os bucket list`), see Listing Buckets
  2. Enter this command.

     Syntax (entered on a single line):

     ```
     oci os object list
     --namespace-name object_storage_namespace
     --bucket-name bucket_name
     ```

     Example:

     ```
     oci os object list \
     --namespace-name examplenamespace \
     --bucket-name MyBucket

     {
       "data": [
         {
           "etag": null,
           "md5": "Ucf+fZbCK/RN5gGsEl7G5w==",
           "name": "eventslogreference.htm",
           "size": 1363,
           "time-created": "2021-06-01T17:57:16+00:00",
           "time-modified": null
         }
       ],
       "prefixes": []
     }
     ```

- **Listing object details**

  1. Gather the information you need to run the command.
     - Namespace (see Obtaining the Object Storage Namespace)
     - Bucket name (`oci os bucket list`), see Listing Buckets
     - Object name (`oci os object list`), see previous example
  2. Run this command.

     Syntax (entered on a single line):

     ```
     oci os object head
     --namespace-name object_storage_namespace
     --bucket-name bucket_name
     --name object_name
     ```

     Example:

```
oci os object head \
--namespace-name examplenamespace \
--bucket-name MyBucket \
--name eventslogreference.htm

{
  "access-control-allow-credentials": "true",
  "access-control-allow-methods": "POST,PUT,GET,HEAD,DELETE",
  "access-control-allow-origin": "*",
  "access-control-expose-headers": "Content-Type,Etag,last-modified,Content-
MD5,Content-Length,opc-client-request-id,opc-request-id,Access-Control-Allow-
Origin,Access-Control-Allow-Methods,Access-Control-Allow-Credentials",
  "connection": "Keep-Alive",
  "content-length": "1363",
  "content-md5": "Ucf+fZbCK/RN5gGsEl7G5w==",
  "content-type": "application/octet-stream",
  "date": "Tue, 01 Jun 2021 18:05:32 GMT",
  "etag": "33ed1aff724eac56f00616552fc61f3e",
  "keep-alive": "timeout=5, max=100",
  "last-modified": "2021-06-01T17:57:16.000Z",
  "opc-client-request-id": "8965F8B5A9B84F00B51D4C965F029230",
  "opc-request-id": "txae7c2c9aa7094f16adee8-0060b676ec",
  "server": "Apache",
  "x-content-type-options": "nosniff"
}
```

Creating a Folder or Subfolder

Using the OCI CLI

1. Gather the information you need to run the command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Bucket name (`oci os bucket list`), see Listing Buckets
   - Object file location
   - Object name
2. Run this command.

   Syntax:

   ```
   oci os object put
   --namespace-name <object_storage_namespace>
   --bucket-name <bucket_name>
   --file <file_location>
   --name <object_name>
   ```

   Example:

   ```
   oci os object put   \
   --namespace-name examplenamespace  \
   --bucket-name Bucket1_objv-enabl   \
   --file /home/log_files/install.log   \
   --name /home/log_files/install.log

   oci os object put   \
   --namespace-name examplenamespace  \
   --bucket-name Bucket1_objv-enabl   \
   --file myfile   \
   --name /home/log_files/install.log
   ```

```
oci os object put   \
--namespace-name examplenamespace  \
--bucket-name Bucket1_objv-enabl   \
--file /home/log_files/install.log   \
--name /home/log_files/install.log

Uploading object [####################################] 100%
{
  "etag": "bae04836d4ea5d521c23cbee70566cf2",
  "last-modified": "2021-05-13T15:37:18.000Z",
  "opc-content-md5": "GWZbZ8CXPCjLcPxBs6cPCQ=="
}
```

Uploading an Object

Using the OCI CLI

An object can be uploaded as a single part or as multiple parts. Use the `--no-multipart` option to upload as a single part. For detailed information on multipart uploads, see Performing a Multipart Upload.

1. Gather the information you need to run the command.
   - Namespace. See Obtaining the Object Storage Namespace.
   - Bucket name (`oci os bucket list`). See Listing Buckets.
   - Object file location
2. Run the object put command.

   Syntax (entered on a single line):

   ```
   oci os object put
   --namespace-name <object_storage_namespace>
   --bucket-name <bucket_name>
   --file <file_location>
   ```

   The value of ***<file_location>*** is the full path name of the object being uploaded, such as `C:\workspace\Uploads\MyFile.txt` or `/home/user/Documents/Uploads/MyFile.txt`.

   If you specify the `--no-multipart` option, the file will upload as a single object with the same name as the source file.

   Example:

   ```
   oci os object put --namespace-name examplenamespace --bucket-name MyBucket \
   --file /home/user/Documents/Uploads/MyFile.txt --no-multipart

   Uploading object [####################################] 100%
   {
     "etag": "33ed1aff724eac56f00616552fc61f3e",
     "last-modified": "2021-06-01T17:57:16.000Z",
     "opc-content-md5": "Ucf+fZbCK/RN5gGsEl7G5w=="
   }
   ```

Performing a Multipart Upload

With multipart uploads, individual parts of an object can be uploaded in parallel to reduce the amount of time you spend uploading.

Multipart uploads accommodate objects that are too large for a single upload operation. Object parts must be no larger than 50 GiB.

You can pause between the uploads of individual parts, and resume the upload when your schedule and resources allow.

Using the OCI CLI

To upload an object, run `oci os object put` with the `--part-size` flag. The `--part-size` value represents the size of each part in mebibytes (MiBs). Object Storage waives the minimum part size restriction for the last uploaded part. The `--part-size` value must be an integer.

Optionally, you can use the `--parallel-upload-count` flag to set the maximum number of parallel uploads allowed.

1. Gather the information you need to run the command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Bucket name (`oci os bucket list`), see Listing Buckets
   - Object file location
2. Run the command.

   Syntax (entered on a single line):

   ```
   oci os object put
   --namespace-name <object_storage_namespace>
   --bucket-name <bucket_name>
   --file <file_location>
   --parallel-upload-count <maximum_number_parallel_uploads>
   --part-size <upload_part_size_in_MB>
   --force
   ```

   Example:

   ```
   oci os object put   \
   --namespace-name examplenamespace  \
   --file /boot/initramfs-0-rescue-e542c19f0fbf4e41a41428d933a7357f.img   \
   --parallel-upload-count 5   \
   --part-size 15   \
   --force

   Upload ID: a21bba2c-8922-4b9c-a98a-9ef3569c0138
   Split file into 6 parts for upload.
   Uploading object [####################################] 100%
   {
     "etag": "0964effc8dc4394fd317f03a025ae5d0",
     "last-modified": "2021-05-11T21:35:19",
     "opc-multipart-md5": "UIVRhiwSHY6o0E4pi/yfGg==-6"
   }
   ```

Listing the Parts of an Unfinished or Failed Multipart Upload

Using the OCI CLI

1. Gather the information you need to run the command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Bucket name (`oci os bucket list`), see Listing Buckets
2. Run this command.

   Syntax (entered on a single line):

```
oci os multipart list
--namespace-name <object_storage_namespace>
--bucket-name <bucket_name>
```

Example:

```
oci os multipart list
--namespace-name examplenamespace  \
--bucket-name MyBucket  \
{
  "data": [
    {
      "bucket": "MyBucket",
      "namespace": "examplenamespace",
      "object": "MyObject",
      "time-created": "2019-07-25T21:55:21.973000+00:00",
      "upload-id": "0b7abd48-9ff2-9d5f-2034-63a02fdd7afa"
    },
    {
      "bucket": "MyBucket",
      "namespace": "examplenamespace",
      "object": "MyObject",
      "time-created": "2019-07-25T21:53:09.246000+00:00",
      "upload-id": "1293ac9d-83f8-e055-a5a7-d1e13277b5c0"
    },
    {
      "bucket": "MyBucket",
      "namespace": "examplenamespace",
      "object": "MyObject",
      "time-created": "2019-07-25T21:46:34.981000+00:00",
      "upload-id": "33e7a875-9e94-c3bc-6577-2ee5d8226b53"
    }
...
```

Canceling a Multipart Upload

Using the OCI CLI

1. Gather the information you need to run the command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Bucket name (`oci os bucket list`), see Listing Buckets
   - Object name (`oci os object list`), see Viewing Objects in a Bucket
   - Upload ID (`oci os multipart list`), see Listing the Parts of an Unfinished or Failed Multipart Upload
2. Run this command.

   Syntax (entered on a single line):

   ```
   oci os multipart abort
   --namespace-name <object_storage_namespace>
   --bucket-name <bucket_name>
   --object-name <object_name>
   --upload-id <upload_ID>
   ```

   Example:

   ```
   oci os multipart abort  \
   --namespace-name examplenamespace  \
   --bucket-name MyBucket    \
   ```

```
--object-name MyObject    \
--upload-id 22d5f6d2-8e03-48ca-8593-0192d25770b8

"data": [
{
"etag": "dd434179cfbc22458a9739096ec43226",
"md5": "PBrT093rZrcSDwQsKh9azQ==",
"part-number": 13,
"size": 15728640
}
],
"opc-next-page": "00013"
}
WARNING: Are you sure you want to permanently remove this incomplete upload? [y/N]: y
```

Performing a Bulk Object Upload

Bulk operations at a specific level of the hierarchy do not affect objects in any level above.

Using the OCI CLI

1. Gather the information you need to run the command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Bucket name (`oci os bucket list`), see Listing Buckets
   - Source directory location – is the upload directory path, such as `C:\workspace\Upload\ or /home/user/Documents/Upload`. If your source directory has subdirectories, the subdirectory names are prepended to the names of the files stored in those subdirectories, delimited with a forward slash (`/`) character. For example, if a file named `maple.jpg` is stored in the subdirectory `trees`, when the file is uploaded, Object Storage assigns the name `trees/maple.jpg` to the object.
2. Run this command.

   Syntax (entered on a single line):

   ```
   oci os object bulk-upload
   --namespace-name <object_storage_namespace>
   --bucket-name <bucket_name>
   --src-dir <source_directory_location>
   ```

   Example:

   ```
   oci os object bulk-upload
   --namespace-name examplenamespace  \
   --bucket-name MyBucket    \
   --src-dir /home/log-dir/

   Uploaded Jan-logs [####################################] 100%
   Uploaded Feb-logs [####################################] 100%
   Uploaded Mar-logs [####################################] 100%
   Uploaded Apr-logs [####################################] 100%

   {
     "skipped-objects": [],
     "upload-failures": {},
     "uploaded-objects": {
       "Jan-logs": {
         "etag": "33ed1aff724eac56f00616552fc61f3e",
         "last-modified": "2021-06-01T20:42:50.000Z",
         "opc-content-md5": "Ucf+fZbCK/RN5gGsEl7G5w=="
   ```

```
    },
    "Feb-logs": {
      "etag": "e1875449257cc6ac6ab93cc9c7921c87",
      "last-modified": "2021-06-01T20:42:50.000Z",
      "opc-content-md5": "1B2M2Y8AsgTpgAmY7PhCfg=="
    },
    "Mar-logs": {
      "etag": "c784ac5216d889f55138ecfb428eee3c",
      "last-modified": "2021-06-01T20:42:51.000Z",
      "opc-content-md5": "1B2M2Y8AsgTpgAmY7PhCfg=="
    },
    "Apr-logs": {
      "etag": "3b4571c73bdb9e44bec0512a5e48fba7",
      "last-modified": "2021-06-01T20:42:51.000Z",
      "opc-content-md5": "1B2M2Y8AsgTpgAmY7PhCfg=="
    }
  }
}
```

Copying an Object to a Different Bucket

You can copy an object to a different bucket as long as the target bucket is located in the same Private Cloud Appliance.

Using the OCI CLI

1. Gather the information you need to run the command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Bucket name (`oci os bucket list`), see Listing Buckets
   - Name of the Source object (`oci os object list`), see Viewing Objects in a Bucket
   - Name of the destination bucket (`oci os bucket list`), see Listing Buckets
   - Name of the object in the new destination
2. Run this command.

   Syntax (entered on a single line):

   ```
   oci os object copy
   --namespace-name <object_storage_namespace>
   --bucket-name <source_bucket_name>
   --source-object-name <source_object>
   --destination-bucket <destination_bucket_name>
   --destination-object-name <destination_object_name>
   ```

   Example:

   ```
   oci os object copy
   --namespace-name examplenamespace
   --bucket-name MyBucket
   --source-object-name Compute_Logs.tar.gz
   --destination-bucket Bucket-log-backups
   --destination-object-name Compute_Logs.tar.gz.backup
   ```

   Verify that the copied object is in the bucket.

   ```
   oci os object list
   --namespace-name examplenamespace
   --bucket-name Bucket-log-backups
   ```

```
{
"data": [
{
"etag": null,
"md5": "XzYkstrjaprhbZyemalRbQ==",
"name": "Compute_Logs.tar.gz.backup",
"size": 132631,
"time-created": "2021-04-01T21:00:55+00:00",
"time-modified": null
}
],
"prefixes": []
}
```

Downloading an Object

Using the OCI CLI

1. Gather the information you need to run the command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Bucket name (`oci os bucket list`), see Listing Buckets
   - Object name (`oci os object list`), see Viewing Objects in a Bucket
   - Object file location
2. Run this command.

   Syntax (entered on a single line):

   ```
   oci os object get
   --namespace-name <object_storage_namespace>
   --bucket-name <bucket_name>
   --name <object_name>
   --file <file_location>
   ```

   <file_location> is the destination path for the file being downloaded, such as `C:\workspace\Downloads\MyFile.txt` or `/home/user/Documents/Downloads/MyFile.txt`.

   Example:

   ```
   oci os object get  \
   --namespace-name examplenamespace  \
   --bucket-name MyBucket  \
   --name photos  \
   --file /home/photos_backup

   Downloading object [#----------------------------------] 100%

   # ls -l
   total 8
   -rw-r--r-- 1 root root 1363 Jun 1 17:56 photo1
   -rw-r--r-- 1 root root 1363 Jun 1 21:40 photo1_backup
   -rw-r--r-- 1 root root 0 Jun 1 20:42 photo2
   -rw-r--r-- 1 root root 0 Jun 1 20:42 photo3
   -rw-r--r-- 1 root root 0 Jun 1 20:42 photo4
   ```

Performing a Multipart Download

Using the OCI CLI

1. Gather the information you need to run the command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Bucket name (`oci os bucket list`), see Listing Buckets
   - Object name (`oci os object list`), see Viewing Objects in a Bucket
   - Object file location
   - The byte-range for the download. Multipart object downloading is available using the byte-range request standard defined in RFC 7233, section 2.1
2. Run the command.

   Syntax (entered on a single line):

   ```
   oci os object get
   --namespace-name <object_storage_namespace>
   --bucket-name <bucket_name>
   --name <object_name>
   --file <file_location>
   --range bytes=<byte_range>
   ```

   Example:

   ```
   oci os object get    \
   --namespace-name examplenamespace  \
   --bucket-name MyBucket    \
   --name MyObject.mp4    \
   --file c:\workspace\Downloads\MyObject.mp4    \
   --range bytes=0-50

   cusobjstorenamespace --range bytes=0-50
   Downloading object [#----------------------------------] 3%

   # ls -l
   total 12
   -rw-r--r-- 1 root root 1363 Jun 1 17:56 abc.mp41
   -rw-r--r-- 1 root root 51 Jun 1 21:50 def.mp4
   -rw-r--r-- 1 root root 1363 Jun 1 21:40 ghi.mp4
   -rw-r--r-- 1 root root 0 Jun 1 20:42 jkl.mp4
   -rw-r--r-- 1 root root 0 Jun 1 20:42 mno.mp4
   -rw-r--r-- 1 root root 0 Jun 1 20:42 pqr.mp4
   ```

Performing a Bulk Download

Using the OCI CLI

1. Gather the information you need to run the command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Bucket name (`oci os bucket list`), see Listing Buckets
   - Download directory. ***<download_directory_location>*** is the destination path for the objects being downloaded, such as `C:\workspace\Downloads\` or `/home/user/`

`Documents/Downloads/`. If the directory does not exist, Object Storage creates the directory when the command runs.

2. Run the command.

   Syntax (entered on a single line):

   ```
   oci os object bulk-download
   --namespace-name <object_storage_namespace>
   --bucket-name <bucket_name>
   --download-dir <download_directory_location>
   ```

   Example:

   ```
   oci os object bulk-download    \
   --namespace-name examplenamespace  \
   --bucket-name MyBucket   \
   --download-dir c:\workspace\Downloads

   Downloaded MyFile.txt [####################################] 100%
   Downloaded logFile.log [####################################] 100%

   {
     "download-failures": {},
     "skipped-objects": []
   }
   ```

Deleting an Object

You can permanently delete an object from a bucket or folder. You cannot, however, recover a deleted object unless you have object versioning enabled. See Managing Object Versioning for details.

You cannot delete an object that has an active retention rule. See Defining Retention Rules for details.

Using the OCI CLI

1. Gather the information you need to run the command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Bucket name (`oci os bucket list`), see Listing Buckets
   - Object name (`oci os object list`), see Viewing Objects in a Bucket
2. Syntax (entered on a single line):

   ```
   oci os object delete
   --namespace-name <object_storage_namespace>
   --bucket-name <bucket_name>
   --object-name <object_name>
   ```

   Example:

   ```
   oci os object delete    \
   --namespace-name examplenamespace  \
   --bucket-name MyBucket    \
   --object-name MyFile.txt

   Are you sure you want to delete this resource? [y/N]: y
   ```

Performing a Bulk Delete of All Objects in a Bucket

Using the OCI CLI

1. Gather the information you need to run the command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Bucket name (`oci os bucket list`), see Listing Buckets
2. To see a list of the files impacted by a bulk delete command without actually deleting the files, use the `--dry-run` option.

   Syntax (entered on a single line):

   ```
   oci os object bulk-delete
   --namespace-name <object_storage_namespace>
   --bucket-name <bucket_name>
   --dry-run
   ```

   Example:

   ```
   oci os object bulk-delete    \
   --namespace-name examplenamespace  \
   --bucket-name MyBucket    \
   --dry-run
   {
     "delete-failures": {},
     "deleted-objects": [
       "MyFile.txt",
       "logFile.log"
     ]
   }
   ```
3. To perform the bulk deletion:

   Syntax (entered on a single line):

   ```
   oci os object bulk-delete
   --namespace-name <object_storage_namespace>
   --bucket-name <bucket_name>
   ```

   Example:

   ```
   oci os object bulk-delete \
   --namespace-name examplenamespace  \
   --bucket-name MyBucket

   WARNING: This command will delete 2 objects. Are you sure you wish to continue? [y/
   N]:y

   Deleted MyRenamedFile.txt [####################################] 100%
   Deleted logFile.log [####################################] 100%

   {
     "delete-failures": {},
     "deleted-objects": [
       "MyFile.txt",
       "logFile.log"
     ]
   }
   ```

Managing Object Versioning

Object versioning provides data protection against accidental or malicious object updates, overwrites, or deletions.

Object versioning is enabled at the bucket level. Versioning directs Object Storage to automatically create an object version each time a new object is uploaded, an existing object is overwritten, or when an object is deleted. You can enable object versioning at bucket creation time or later. A bucket that is versioning-enabled can have many versions of an object. There is always one latest version of the object and zero or more previous versions.

For more conceptual information, refer to the Object Storage Overview section in the Oracle Private Cloud Appliance Concepts Guide.

Enabling Versioning During Bucket Creation

Object versioning provides data protection against accidental or malicious object updates and deletions.

Using the OCI CLI

1. Gather the information you need to run the command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Compartment OCID (`oci iam compartment list -all`)
   - Bucket name: The name you want for this bucket.
2. Syntax (entered on a single line):

```
oci os bucket create
--namespace-name <object_storage_namespace>
--compartment-id <target_compartment_id>
--name <bucket_name>
--versioning enabled
```

Example:

```
oci os bucket create    \
--namespace-name examplenamespace  \
--compartment-id ocid.compartment.….….….exampleuniqueID    \
--name MyStandardBucket    \
--versioning enabled

{
  "data": {
    "approximate-count": null,
    "approximate-size": null,
    "compartment-id": "ocid1.compartment.….….….uniqueID",
    "created-by": "ocid1.user.….….….uniqueID",
    "defined-tags": null,
    "etag": "00b4edbb27012ae78a912428ad1e630c",
    "freeform-tags": null,
    "id": null,
    "is-read-only": null,
    "kms-key-id": null,
    "metadata": null,
    "name": "bucket-4-versioning",
    "namespace": "export/examplenamespace",
```

```
    "object-events-enabled": null,
    "object-lifecycle-policy-etag": null,
    "public-access-type": "NoPublicAccess",
    "replication-enabled": null,
    "storage-tier": "Standard",
    "time-created": "2021-06-10T18:39:12+00:00",
    "versioning": "Enabled"
  },
  "etag": "00b4edbb27012ae78a912428ad1e630c"
}
```

Enabling or Suspending Versioning (After Bucket Creation)

Object versioning provides data protection against accidental or malicious object updates and deletions.

Using the OCI CLI

1. Gather the information you need to run the command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Compartment OCID (`oci iam compartment list -all`)
   - Bucket name (`oci os bucket list`), see Listing Buckets
2. Run the command.

   Syntax (entered on a single line):

   ```
   oci os bucket update
   --namespace-name <object_storage_namespace>
   --compartment-id <target_compartment_id>
   --bucket-name <bucket_name>
   --versioning <enabled | suspended>
   ```

   For `--versioning`, choose one of the options: `enabled` or `suspended`.

   Example of enabling object versioning:

   ```
   oci os bucket update    \
   --namespace-name examplenamespace  \
   --compartment-id ocid.compartment.….….….uniqueID    \
   --bucket-name MyBucket    \
   --versioning Enabled
   {
     "data": {
       "approximate-count": null,
       "approximate-size": null,
       "compartment-id": "ocid1.compartment.….….….uniqueID",
       "created-by": "ocid1.user.….….….uniqueID",
       "defined-tags": null,
       "etag": "117f0608bdf83b9c7ea393db556a0ee4",
       "freeform-tags": null,
       "id": null,
       "is-read-only": null,
       "kms-key-id": null,
       "metadata": null,
       "name": "MyBucket",
       "namespace": "export/examplenamespace",
       "object-events-enabled": null,
       "object-lifecycle-policy-etag": null,
       "public-access-type": "ObjectRead",
       "replication-enabled": null,
   ```

```
    "storage-tier": "Standard",
    "time-created": "2021-06-02T17:06:18+00:00",
    "versioning": "Enabled"
  },
  "etag": "117f0608bdf83b9c7ea393db556a0ee4"
}
```

Viewing Object Versions and Details

Using the OCI CLI

1. Gather the information you need to run the command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Compartment OCID (`oci iam compartment list -all`)
   - Bucket name (`oci os bucket list`), see Listing Buckets
2. Run the command.

   Syntax (entered on a single line):

   ```
   oci os object list-object-versions
   --namespace-name <object_storage_namespace>
   --bucket-name <bucket_name>
   ```

   Example:

   ```
   oci os object list-object-versions    \
   --namespace-name examplenamespace  \
   --bucket-name MyBucket
   {
     "data": [
       {
         "etag": null,
         "is-delete-marker": false,
         "md5": "3DI5GbLmKiRxY/ozWxyXHQ==",
         "name": "bucket-data",
         "size": 103,
         "time-created": "2021-06-02T22:20:25+00:00",
         "time-modified": null,
         "version-id": null
       },
       {
         "etag": null,
         "is-delete-marker": false,
         "md5": "VIic5JncRWwDQj6CnsZ1Ww==",
         "name": "compute.log",
         "size": 4878456,
         "time-created": "2021-06-10T19:03:26+00:00",
         "time-modified": null,
         "version-id": "5f4ce7e8-656f-409a-b70a-ebfedddcfeda"
       }
     ],
     "prefixes": []
   }
   ```

Deleting the Previous Version of an Object

When versioning is enabled, deleting an object without targeting a specific version creates a delete marker and previous version of the object that can be recovered. However, deleting a previous version of an object is a permanent deletion.

Using the OCI CLI

1. Gather the information you need to run the command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Compartment OCID (`oci iam compartment list -all`)
   - Bucket name (`oci os bucket list`), see Listing Buckets
   - Object name (`oci os object list`), see Viewing Objects in a Bucket
2. Syntax:

> **Note:**
>
> If an object has a `version-id` of `null`, there is only one version of the object. To delete this object, omit the `--version-id` argument.

```
oci os object delete
--namespace-name <object_storage_namespace>
--bucket-name <bucket_name>
--version-id <bucket_version_id>
--object-name <object_name>
```

Example:

```
oci os object delete
--namespace-name examplenamespace  \
--bucket-name MyBucket  \
--version-id 7f1f537d-ec9c-4706-867a-b1dae355c263   \
--object-name compute.log
```

Recovering a Deleted Object Version

Recovering a deleted object version is as simple as deleting the delete marker that was created when you deleted the latest version of an object. The previous version of the object listed just below the delete marker is recovered and becomes the latest version of the object.

Using the OCI CLI

1. List the objects in the bucket. See Viewing Object Versions and Details. In the output, locate the object version that has `"is-delete-marker": true`.

   Use the version-id of that object with the delete command to delete the delete marker.

> **Note:**
>
> If an object has a `version-id` of `null`, there is only one version of the object. To delete this object marker, omit the `--version-id` argument.

2. Gather the information you need to run the command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Compartment OCID (`oci iam compartment list -all`)
   - Bucket name (`oci os bucket list`), see Listing Buckets
   - Object name (`oci os object list`), see Viewing Objects in a Bucket
   - Version ID (see previous step)
3. Syntax:

```
oci os object delete
--namespace-name <object_storage_namespace>
--bucket-name <bucket_name>
--object-name <object_name>
--version-id <bucket_version_id>
```

Example:

```
oci os object delete
--namespace-name examplenamespace  \
--bucket-name  MyBucket
--object-name application.log
--version-id  6ce3eb93-8850-4732-8949-cb6e67b722b0
Are you sure you want to delete this resource? [y/N]: y
```

Using Pre-Authenticated Requests

Pre-authenticated requests provide a way to let users access a bucket or an object without having their own credentials, as long as the request creator has permissions to access those objects.

For example, you can create a request that lets an operations support user upload backups to a bucket without owning API keys. Or, you can create a request that lets a business partner update shared data in a bucket without owning API keys.

When you create a pre-authenticated request, a unique URL is generated. Anyone you provide this URL to can access the Object Storage resources identified in the pre-authenticated request, using standard HTTP tools like curl and wget.

> **Important:**
>
> Assess the business requirement for and the security ramifications of pre-authenticated access to a bucket or objects.
>
> A pre-authenticated request URL gives anyone who has the URL access to the targets identified in the request. Carefully manage the distribution of the URL.
>
> For more conceptual information, refer to the Object Storage Overview section in the Oracle Private Cloud Appliance Concepts Guide.

Listing Pre-Authenticated Requests

Use this procedure to obtain information about pre-authenticated requests, such as obtaining the pre-authenticated requests id that you might need for other commands.

> **Note:**
>
> Listing pre-authenticated requests does not display the unique URL provided by the system when you created a pre-authenticated request. The URL is displayed only at the time of creation and cannot be retrieved later.

Using the OCI CLI

- **Listing All the Pre-Authenticated Requests in a Bucket**
  1. Gather the information you need to run the command.
     - Namespace (see Obtaining the Object Storage Namespace)
     - Bucket name (`oci os bucket list`), see Listing Buckets
  2. Run the command.

     Syntax (entered on a single line):

     ```
     oci os preauth-request list
     --namespace-name object_storage_namespace
     --bucket-name bucket_name
     ```

     Example:

     ```
     oci os preauth-request list \
     --namespace-name examplenamespace  \
     --bucket-name MyBucket
     {
       "data": [
         {
           "access-type": "ObjectRead",
           "id": "5299a6f9-55c7-4805-88ca-b270c9a9e94f",
           "name": "PAR_ObjRead",
           "object-name": "compute.log",
           "time-created": "2021-06-10T20:34:01+00:00",
           "time-expires": "2021-07-30T23:55:00+00:00"
         },
         {
           "access-type": "AnyObjectWrite",
     ```

```
      "id": "783cd56b-9df5-4518-aacf-f523deae5102",
      "name": "PAR-all-objectsRW",
      "object-name": null,
      "time-created": "2021-06-10T20:49:11+00:00",
      "time-expires": "2021-07-30T23:54:59+00:00"
    },
    {
      "access-type": "ObjectRead",
      "id": "2ea48624-16ed-4d81-95ca-b23ea750ed3d",
      "name": "PAR-OS-READ",
      "object-name": "backup.log",
      "time-created": "2021-06-10T21:16:47+00:00",
      "time-expires": "2021-07-30T23:55:00+00:00"
    }
  ]
}
```

- **Getting the Details for a Specific Pre-Authenticated Request**

  1. Gather the information you need to run the command.
     - Namespace (see Obtaining the Object Storage Namespace)
     - Bucket name (`oci os bucket list`), see Listing Buckets
     - Preauth ID (`oci os preauth-request list`), see Listing Pre-Authenticated Requests
  2. Run the command.

     Syntax (entered on a single line):

     ```
     oci os preauth-request get
     --namespace-name object_storage_namespace
     --bucket-name bucket_name
     --par-id preauth-id
     ```

     Example:

     ```
     oci os preauth-request get  \
     --namespace-name examplenamespace  \
     --bucket-name MyBucket  \
     --par-id 5299a6f9-55c7-4805-88ca-b270c9a9e94f
     {
       "data": {
         "access-type": "ObjectRead",
         "id": "5299a6f9-55c7-4805-88ca-b270c9a9e94f",
         "name": "PAR_ObjRead",
         "object-name": "compute.log",
         "time-created": "2021-06-10T20:34:01+00:00",
         "time-expires": "2021-07-30T23:55:00+00:00"
       }
     }
     ```

Creating a Pre-Authenticated Request for All Objects in a Bucket

Using the OCI CLI

1. Gather the information you need to run the command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Bucket name (`oci os bucket list`), see Listing Buckets
   - Name for this pre-authenticated request.

- Access type is one of these items:
  - `AnyObjectRead` permits reads on all objects in the bucket.
  - `AnyObjectWrite` permits writes to all objects in the bucket.
  - `AnyObjectReadWrite` permits reads and writes to all objects in the bucket.

> **Note:**
>
> Listing objects in a bucket is denied by default. If the `--access-type` is `AnyObjectRead` or `AnyObjectReadWrite`, you can specify the optional `--bucket-listing-action ListObjects` parameter when creating the pre-authenticated request that lets users list the objects in the bucket.

- Time expires is required and must be an RFC 3339 time stamp. For example: `2017-09-01T00:09:51.000+02:00`.

2. Run the command.

   Syntax (entered on a single line):

```
oci os preauth-request create
--namespace-name object_storage_namespace
--bucket-name bucket_name
--name preauthenticated_request_name
--access-type access_value
--time-expires timestamp
```

   This example creates a pre-authenticated request that allows reads and writes to all objects in the bucket:

```
 oci os preauth-request create  \
--namespace-name examplenamespace  \
--bucket-name MyBucket  \
--name PAR-all-objectsRW  \
--access-type AnyObjectWrite  \
--time-expires '2021-07-30 23:55'
{
  "data": {
    "access-type": "AnyObjectWrite",
    "access-uri": "/p/KOCRWzqBilJmIsaBbJNelKLWcOxwRLq/n/examplenamespace/b/
MyBucket/o/",
    "id": "783cd56b-9df5-4518-aacf-f523deae5102",
    "name": "PAR-all-objectsRW",
    "object-name": null,
    "time-created": "2021-06-10T20:49:11+00:00",
    "time-expires": "2021-07-30T23:54:59+00:00"
  }
}
```

3. **Important** – Copy the access-uri to durable storage.

   The unique `access-uri` provided by the system is the only way to construct a URL that a user can use to access the bucket or object specified as the request target.

   The `access-uri` is displayed only at the time of creation and cannot be retrieved later.

4. Construct a URL from the unique `access-uri`.

See Constructing the Pre-Authenticated Request URL.

Creating a Pre-Authenticated Request for a Specific Object

Using the OCI CLI

1. Gather the information you need to run the command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Bucket name (`oci os bucket list`), see Listing Buckets
   - Name for this pre-authenticated request.
   - Access type is one of the following values:
     - `AnyObjectRead` permits reads on all objects in the bucket.
     - `AnyObjectWrite` permits writes to all objects in the bucket.
     - `AnyObjectReadWrite` permits reads and writes to all objects in the bucket.

     > **Note:**
     >
     > Listing objects in a bucket is denied by default. If the `--access-type` is `AnyObjectRead` or `AnyObjectReadWrite`, you can specify the optional `--bucket-listing-action ListObjects` parameter when creating the pre-authenticated request that lets users list the objects in the bucket.

   - Time expires is required and must be an RFC 3339 time stamp. For example: `2017-09-01T00:09:51.000+02:00`.
   - Object name, or `null`
2. Syntax (entered on a single line):

```
oci os preauth-request create
--namespace-name object_storage_namespace
--bucket-name bucket_name
--name preauthenticated_request_name
--access-type access_value
--time-expires timestamp
-on object_name_or_null
```

Example:

```
oci os preauth-request create
--namespace-name examplenamespace  \
--bucket-name MyBucket
--name PAR-OS-READ
--access-type ObjectRead
--time-expires '2021-07-30 23:55'
-on compute.log

{
  "data": {
    "access-type": "ObjectRead",
    "access-uri": "/p/eWvgyLcDthhvVUNkVaejymgDTOILHli/n/examplenamespace/b/
MyBucket/o/compute.log",
    "id": "2ea48624-16ed-4d81-95ca-b23ea750ed3d",
```

```
    "name": "PAR-OS-READ",
    "object-name": "compute.log",
    "time-created": "2021-06-10T21:16:47+00:00",
    "time-expires": "2021-07-30T23:55:00+00:00"
  }
}
```

3. **Important** – Copy the access-uri to durable storage.

   The unique `access-uri` provided by the system is the only way to construct a URL that a user can use to access the bucket or object specified as the request target.

   The `access-uri` is displayed only at the time of creation and cannot be retrieved later.

4. Construct a URL from the unique `access-uri`.

   See Constructing the Pre-Authenticated Request URL.

Constructing the Pre-Authenticated Request URL

After you have a unique `access-uri`, you can construct the access URL that enables users to access pre-authenticated objects.

1. Construct the URL using this syntax.

   Syntax:

   `https://objectstorage.`***`pca_fqdn`***`/`***`access-uri`***

   where:

   - ***pca_fqdn*** is the fully qualified domain name of your appliance.
   - ***access-uri*** is the access URI that was obtained from one of these procedures:
     - Creating a Pre-Authenticated Request for All Objects in a Bucket
     - Creating a Pre-Authenticated Request for a Specific Object

   Example:

   ```
   https://objectstorage.mypca01.example.com/p/MrxLFkKlFkIlNDhvhcZnrjbUAlsoeah/n/
   mynamespace/b/my-bucket/o/my-object
   ```

Deleting a Pre-Authenticated Request

Using the OCI CLI

1. Gather the information you need to run the command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Bucket name (`oci os bucket list`), see Listing Buckets
   - Preauth ID (`oci os preauth-request list`), see Listing Pre-Authenticated Requests
2. Syntax (entered on a single line):

   ```
   oci os preauth-request delete
   --namespace-name object_storage_namespace
   --bucket-name bucket_name
   --par-id preauthenticated_request_id
   ```

   Example:

```
oci os preauth-request delete  \
--namespace-name examplenamespace  \
--bucket-name MyBucket  \
--par-id 2ea48624-16ed-4d81-95ca-b23ea750ed3d
Are you sure you want to delete this resource? [y/N]: y
```

Listing Objects for Pre-Authenticated Requests

Using the unique request URL, you can use a tool like curl to list, read, and write data using the pre-authenticated request.

Using curl

- Syntax (entered on a single line):

  ```
  $ curl -X GET unique-PAR-URL
  ```

  Example:

  ```
  $ curl -X GET    \
  https://objectstorage.us-example-1.example.com/p/CoO26YkSARiRevWlDWJD_QUvtFPUocn/n/
  examplenamespace/b/MyParBucket/o/

  {"objects":[{"name":"InfoWorld DeepDive - Tips  for Git and GitHub Users.pdf"},
  {"name":"OCI_User_Guide.pdf"},
  {"name":"OracleCorporateTerminologyUsageGuideRedwood.pdf"},{"name":"VPN.png"},
  {"name":"eventslogreference.htm"},
  {"name":"functionslogreference.htm"},{"name":"glob.txt"},
  {"name":"loadbalancerreference.htm"},{"name":"objectstoragelogreference.htm"},
  {"name":"servicechanges.html"},{"name":"servicediscovery.dita"},
  {"name":"serviceessentials.html"},{"name":"servicelogreference.htm"},
  {"name":"services.html"}]}
  ```

Uploading an Object Using a Pre-Authenticated Request

Using the unique request URL, you can use a tool like curl to read and write data using the pre-authenticated request.

Using curl

- Syntax (entered on a single line):

  ```
  $ curl -X PUT --data-binary '@local-filename' unique-PAR-URL
  ```

  Example:

  ```
  $ curl -X PUT    \
  --data-binary '@using-dita-guide.pdf'    \
  https://objectstorage.us-example-1.example.com/p/lnaqMuXWef_lhTxCiS9ngCw/n/
  examplenamespace/b/MyParBucket/o/using-dita-guide.pdf
  ```

Downloading an Object Using a Pre-Authenticated Request

Using the unique request URL, you can use a tool like curl to read and write data using the pre-authenticated request.

Using curl

- Syntax (entered on a single line):

  ```
  $ curl -X GET unique-PAR-URL
  ```

Example:

```
$ curl -X GET    \
https://objectstorage.example.com/p/tnjDhazP9o6s2KzLyFUxILQzSamEp/n/
examplenamespace/b/MyParBucket/o/OCI_User_Guide.pdf
'@data.1''@data.2''@data.3'
```

Defining Retention Rules

Retention rules provide immutable storage options for data written to Object Storage for data governance, regulatory compliance, and legal hold requirements. Retention rules can also protect your data from accidental or malicious writes or deletion. Retention rules can be locked to prevent rule modification and data deletion or modification even by administrators.

Retention rules are configured at the bucket level and are applied to all individual objects in the bucket.

For more conceptual information, refer to the Object Storage Overview in the *Oracle Private Cloud Appliance Concepts Guide*.

Viewing Retention Rules and Details

Using the OCI CLI

- **Listing the Retention Rules for a Bucket**

  1. Gather the information you need to run the command.
     - Namespace (see Obtaining the Object Storage Namespace)
     - Bucket name (`oci os bucket list`), see Listing Buckets
  2. Run the command.

     Syntax:

     ```
     oci os retention-rule list
     --namespace-name <object_storage_namespace>
     --bucket-name <bucket_name>
     ```

     Example:

     ```
     oci os retention-rule list  \
     --namespace-name examplenamespace  \
     --bucket-name MyBucket
     {
       "data": {
         "items": [
           {
             "display-name": "RegulatoryCompliance",
             "duration": {
               "time-amount": 5,
               "time-unit": "YEARS"
             },
             "etag": "72be3a47de931cd50ad9d93c077def64",
             "id": "72be3a47de931cd50ad9d93c077def64",
             "time-created": "2021-06-10T22:24:21+00:00",
             "time-modified": "2021-06-10T22:24:21+00:00",
             "time-rule-locked": "2021-06-30T17:00:00+00:00"
           },
           {
             "display-name": "TempHold",
     ```

```
        "duration": {
          "time-amount": 30,
          "time-unit": "DAYS"
        },
        "etag": "344a9c205187408699b51c7769dc1bb4",
        "id": "344a9c205187408699b51c7769dc1bb4",
        "time-created": "2021-06-10T22:17:50+00:00",
        "time-modified": "2021-06-10T22:17:50+00:00",
        "time-rule-locked": null
      },
      {
        "display-name": "LegalHold",
        "duration": null,
        "etag": "bd8d8efb964d1025f4305c86de630a4f",
        "id": "bd8d8efb964d1025f4305c86de630a4f",
        "time-created": "2021-06-10T22:13:37+00:00",
        "time-modified": "2021-06-10T22:13:37+00:00",
        "time-rule-locked": null
      }
    ]
  }
}
```

- **Getting Details for a Specific Retention Rule**

  1. Gather the information you need to run the command.
     - Namespace (see Obtaining the Object Storage Namespace)
     - Bucket name (`oci os bucket list`), see Listing Buckets
     - Retention rule ID (`oci os retention-rule list`), see Viewing Retention Rules and Details
  2. Run the command.

     Syntax:

     ```
     oci os retention-rule get
     --namespace-name <object_storage_namespace>
     --bucket-name <bucket_name>
     --retention-rule-id <retention_rule_identifier>
     ```

     Example:

     ```
     oci os retention-rule get  \
     --namespace-name examplenamespace  \
     --bucket-name MyBucket  \
     --retention-rule-id 72be3a47de931cd50ad9d93c077def64

     {
       "data": {
         "display-name": "RegulatoryCompliance",
         "duration": {
           "time-amount": 5,
           "time-unit": "YEARS"
         },
         "etag": "72be3a47de931cd50ad9d93c077def64",
         "id": "72be3a47de931cd50ad9d93c077def64",
         "time-created": "2021-06-10T22:24:21+00:00",
         "time-modified": "2021-06-10T22:24:21+00:00",
         "time-rule-locked": "2021-06-30T17:00:00+00:00"
       }
     }
     ```

Creating a Retention Rule

Using the OCI CLI

- **Creating an Indefinite Retention Rule**

  1. Gather the information you need to run the command.
     - Namespace (see Obtaining the Object Storage Namespace)
     - Bucket name (`oci os bucket list`), see Listing Buckets
     - Display name: The name you want to apply to this retention rule.
  2. Run this command.

     Syntax:

     ```
     oci os retention-rule create
     --namespace-name <object_storage_namespace>
     --bucket-name <bucket_name>
     --display-name <name_displayed_for_rule>
     ```

     Example:

     ```
     oci os retention-rule create   \
     --namespace-name examplenamespace  \
     --bucket-name MyBucket   \
     --display-name LegalHold
     {
       "data": {
         "display-name": "LegalHold",
         "duration": null,
         "etag": "bd8d8efb964d1025f4305c86de630a4f",
         "id": "bd8d8efb964d1025f4305c86de630a4f",
         "time-created": "2021-06-10T22:13:37+00:00",
         "time-modified": "2021-06-10T22:13:37+00:00",
         "time-rule-locked": null
       }
     }
     ```

- **Creating a time-bound, Unlocked Retention Rule**

  1. Gather the information you need to run the command.
     - Namespace (see Obtaining the Object Storage Namespace)
     - Bucket name (`oci os bucket list`), see Listing Buckets
     - Display name: The name you want to apply to this retention rule.
     - Time and unit(days|years). For example, 30 days or 5 years.
  2. Run this command.

     Syntax:

     ```
     oci os retention-rule create
     --namespace-name <object_storage_namespace>
     --bucket-name <bucket_name>
     --display-name <display_name>
     --time-amount <time_integer>
     --time-unit <days|years>
     ```

     Example:

```
 oci os retention-rule create  \
--namespace-name examplenamespace  \
--bucket-name MyBucket  \
--display-name TempHold  \
--time-amount 30  \
--time-unit days
{
  "data": {
    "display-name": "TempHold",
    "duration": {
      "time-amount": 30,
      "time-unit": "DAYS"
    },
    "etag": "344a9c205187408699b51c7769dc1bb4",
    "id": "344a9c205187408699b51c7769dc1bb4",
    "time-created": "2021-06-10T22:17:50+00:00",
    "time-modified": "2021-06-10T22:17:50+00:00",
    "time-rule-locked": null
  }
}
```

- **Creating a Time-Bound, Locked Retention Rule**

  1. Gather the information you need to run the command.
     - Namespace (see Obtaining the Object Storage Namespace)
     - Bucket name (`oci os bucket list`), see Listing Buckets
     - Display name: The name you want to apply to this retention rule.
     - Time and unit (days|years). For example, 30 days or 5 years.
     - Date and time to lock the rule.
  2. Run this command.

     Syntax:

     ```
     oci os retention-rule create
     --namespace-name <object_storage_namespace>
     --bucket-name <bucket_name>
     --display-name <display_name>
     --time-amount <time_integer>
     --time-unit <days|years>
     --time-rule-locked <date and time>
     ```

     Example:

     ```
     oci os retention-rule create  \
     --namespace-name examplenamespace  \
     --bucket-name MyBucket  \
     --display-name RegulatoryCompliance   \
     --time-amount 5   \
     --time-unit years   \
     --time-rule-locked "2021-06-30 17:00"
     {
       "data": {
         "display-name": "RegulatoryCompliance",
         "duration": {
           "time-amount": 5,
           "time-unit": "YEARS"
         },
         "etag": "72be3a47de931cd50ad9d93c077def64",
         "id": "72be3a47de931cd50ad9d93c077def64",
         "time-created": "2021-06-10T22:24:21+00:00",
     ```

```
    "time-modified": "2021-06-10T22:24:21+00:00",
    "time-rule-locked": "2021-06-30T17:00:00+00:00"
  }
}
```

Modifying a Retention Rule

Using the OCI CLI

- **Updating a Retention Rule**

  1. Gather the information you need to run the command.
     - Namespace (see Obtaining the Object Storage Namespace)
     - Bucket name (`oci os bucket list`), see Listing Buckets
     - Retention rule ID (`oci os retention-rule list`), see Viewing Retention Rules and Details
  2. Run this command.

     Syntax:

     ```
     oci os retention-rule update
     --namespace-name <object_storage_namespace>
     --bucket-name <bucket_name>
     --retention-rule-id <retention_rule_id>
     ```

     Followed by the retention rule item that you plan to change:

     ```
     --time-amount <time_integer>
     --time-unit <days|years>
     ```

     Example:

     ```
     oci os retention-rule update  \
     --namespace-name examplenamespace  \
     --bucket-name MyBucket  \
     --retention-rule-id 344a9c205187408699b51c7769dc1bb4  \
     --time-amount 60  \
     --time-unit days
     {
       "data": {
         "display-name": "TempHold",
         "duration": {
           "time-amount": 60,
           "time-unit": "DAYS"
         },
         "etag": "344a9c205187408699b51c7769dc1bb4",
         "id": "344a9c205187408699b51c7769dc1bb4",
         "time-created": "2021-06-10T22:17:50+00:00",
         "time-modified": "2021-06-10T22:45:16+00:00",
         "time-rule-locked": null
       }
     }
     ```

- **Removing a Retention Rule Lock During the Delay Period**

  1. Gather the information you need to run the command.
     - Namespace (see Obtaining the Object Storage Namespace)
     - Bucket name (`oci os bucket list`), see Listing Buckets

- Retention rule ID (`oci os retention-rule list`), see Viewing Retention Rules and Details

2. Run this command.

   Syntax:

```
oci os retention-rule update
--namespace-name <object_storage_namespace>
--bucket-name <bucket_name>
--retention-rule-id <retention_rule_id>
--time-rule-locked ""
```

   Example:

```
oci os retention-rule update
--namespace-name examplenamespace  \
--bucket-name MyBucket  \
--retention-rule-id b1a6c84c-57c4-416c-b006-f864b0904c9e
--time-rule-locked ""
{
  "data": {
    "display-name": "RegulatoryCompliance",
    "duration": {
      "time-amount": 6,
      "time-unit": "YEARS"
    },
    "etag": "5b4fa526-faec-47d4-9162-4acdf1813ee0",
    "id": "b1a6c84c-57c4-416c-b006-f864b0904c9e",
    "time-created": "2020-03-25T15:11:44.423000+00:00",
    "time-modified": "2020-03-25T22:02:43.745000+00:00",
    "time-rule-locked": null
  },
  "etag": "5b4fa526-faec-47d4-9162-4acdf1813ee0"
}
```

Deleting a Retention Rule

Using the OCI CLI

1. Gather the information you need to run the command.
   - Namespace (see Obtaining the Object Storage Namespace)
   - Bucket name (`oci os bucket list`), see Listing Buckets
   - Retention rule ID (`oci os retention-rule list`), see Viewing Retention Rules and Details
2. Syntax:

```
oci os retention-rule delete
--namespace-name <object_storage_namespace>
--bucket-name <bucket_name>
--retention-rule-id <retention_rule_identifier>
```

   Example:

```
oci os retention-rule delete  \
--namespace-name examplenamespace  \
--bucket-name MyBucket   \
--retention-rule-id 344a9c205187408699b51c7769dc1bb4
Are you sure you want to delete this resource? [y/N]: y
```