ORACLE

Cloud Infrastructure

Compute Cloud@Customer

March 27, 2026

Contents

Compute Cloud@Customer.....5
Overview.....5
Network Infrastructure.....8
Compute and Storage Configurations.....27
High Availability.....27
Using Your Own CA Trust Chain.....29
Oracle Operator Access Control.....29
GPU Expansion.....29
Compute Expansion Rack.....31
Emergency Information.....32
Supported Regions.....33
Guest OS Matrix.....33
Service Limits.....34
Quotas.....40
Where to Find Billing Information.....40
View as PDF.....40
Installation and Deployment.....40
Prepare Installation Site.....41
Site Checklists.....52
Initial System Installation Checklist.....58
Preparing Your Tenancy.....67
Managing Infrastructures.....71
Postinstallation Administration.....80
Infrastructure Administration.....81
Securing Compute Cloud@Customer.....81
Managing Upgrade Schedules.....94
Disaster Recovery.....100
Accessing User Interfaces.....102
Signing in to the Console.....102
Using the CLI.....103
Using the API.....107
Identity and Access Management (IAM).....108
Federated Identity Providers.....109
Identity Domains.....109
Policy Reference.....110
Tagging Resources (IAM in OCI).....115
Add User Names and Dates to Resources.....116
Adding Tags at Resource Creation (IAM in OCI).....118
Using Free-Form Tags for Extended Functionality.....120
Creating OraclePCA Tags (IAM in OCI).....120
Metrics.....123
Viewing Default Metric Charts.....124
Creating a Query with Metrics Explorer.....124
Example of Creating an Alarm.....125
Metrics Reference.....126
Virtual Cloud Networks.....127
Name Resolution.....128
Networking Scenarios.....129
Managing VCNs and Subnets.....140

Configuring VCN Rules and Options ... 145
Configuring VCN Gateways ... 168
Configuring VNICs ... 181
Managing Private IP Addresses ... 186
Managing Public IP Addresses ... 191
Configuring SR-IOV ... 197
Managing Public DNS Zones ... 198
Managing Traffic with Steering Policies ... 204
Load Balancers ... 209
Load Balancer as a Service ... 211
Network Load Balancers ... 262
Images ... 283
Listing Images and Viewing Details ... 283
Platform Images ... 284
Managing Custom Images ... 285
Marketplace Images ... 292
Bring Your Own Image (BYOI) ... 294
Compute Instances ... 299
Compute Shapes ... 301
Compute Instance Lifecycle ... 305
Extending Compute Resources ... 306
Tutorial: Launching Your First Instance ... 307
Working with Instances ... 316
Configuring Instances for Calling Services ... 330
Working with Instance Configurations ... 333
Connecting to a Compute Instance ... 341
Backing Up and Restoring an Instance ... 344
Instance Pools ... 349
Creating an Instance Pool ... 350
Using Schedule-Based Autoscaling ... 353
Updating an Instance Pool ... 362
Attaching an Instance to an Instance Pool ... 363
Detaching an Instance from an Instance Pool ... 364
Managing Instance Pool Load Balancer Attachments ... 365
Stopping and Starting Instances in an Instance Pool ... 367
Deleting an Instance Pool ... 368
Container Instances ... 368
Container Instance Shapes ... 369
Working with Container Instances ... 369
Working with Containers in Container Instances ... 379
Block Volume Storage ... 380
Creating and Attaching Block Volumes ... 383
Managing Block Volumes ... 392
Managing Boot Volumes ... 398
Resizing Volumes ... 403
Managing Volume Groups ... 407
Replicating Volumes ... 411
Backing Up Block Volumes ... 422
Managing Backup Policies ... 426
File Storage ... 433
VCN Security Rules for File Storage ... 436
File Storage Network Ports ... 437
Deleting File System Resources Overview ... 437
Export Options for File Storage ... 437
Creating a File System, Mount Target, and Export ... 442
Controlling Access to File Storage ... 450

Mounting File Systems on UNIX-based Instances....453
Mounting File Systems on Microsoft Windows Instances....457
Managing Mount Targets and Exports....461
Managing File Systems....465
Managing File System Snapshots....468
Managing File System Clones....471
Replicating File Systems....472
Object Storage....478
Obtaining the Object Storage Namespace....479
Managing Object Storage Buckets....479
Managing Storage Objects....482
Managing Object Versioning....492
Using Preauthenticated Requests....495
Defining Retention Rules....501
Kubernetes Engine (OKE)....505
OKE Workflow....506
OKE Best Practices....514
Creating OKE Network Resources....517
Creating and Managing OKE Clusters....589
Managing OKE Cluster Add-ons....600
Creating and Managing OKE Worker Node Pools....618
Exposing Containerized Applications....633
Adding Storage for Containerized Applications....634
Logs....646
Enabling Logging....646
Disabling Logging....647
Troubleshooting....647
Troubleshooting Versioning....647
Known Issues....648
Customers in certain OCI realms will not have access to usage/billing for resource consumption....648
Take caution when creating or deleting compartments and tags against Compute Cloud@Customer resources....649
System upgrades are not possible for specific OCI realms....649
Upgrade might impact Load Balancer as a Service functionality....649
GPU Drivers Not Included in Oracle Linux Platform Images....650
The `oci ccc get infrastructure` and `oci ccc infrastructure update` CLI commands return `null` for the `provisioning_pin` value....653
When using the ccc infrastructure list CLI command with the --compartment-id-in-subtree true option, no results are returned....653
Output from `oci iam user get` doesn't list user capabilities....655

Index....656

Compute Cloud@Customer

Use Compute Cloud@Customer to deploy Oracle Cloud Infrastructure (OCI) services on-premises so that you can meet data sovereignty and regulatory requirements, while using OCI's identity and governance services to manage access to them.

What's new

Get Started

- Overview
- Install and deploy the infrastructure
- Tutorial: Launching your first instance
- Video: Compute Cloud@Customer Overview

Set Up Resources

- Configure VCNs and subnets
- Create an instance
- Create and attach a block volume
- Create file storage
- Create an object storage bucket

Manage Resources

- Manage block volumes
- Manage file storage
- Mount file systems on UNIX-based instances
- Manage object storage
- Back up and restore an instance

Access Resources

- Access user interfaces
- Connect to a compute instance
- Find a volume in the instance OS

Manage the Infrastructure

- Manage infrastructures
- Schedule upgrades
- Secure Compute Cloud@Customer

Troubleshooting

- Known issues
- Troubleshooting Object Storage Versioning
- Troubleshooting Object Storage Retention Rules

Community

- Oracle Cloud Infrastructure blog
- Cloud infrastructure community forum
- Technical Papers

Support

- Ways to Get Help
- Creating a Support Request
- Sign in to My Oracle Support (MOS)

Developer Tools

- Compute Cloud@Customer API
- SDKs and the CLI
- Cloud Shell

Overview of Compute Cloud@Customer

Use Compute Cloud@Customer to deploy Oracle Cloud Infrastructure services on-premises, to meet data sovereignty and regulatory requirements, while using OCI's identity and governance services to manage access to them.

With a Compute Cloud@Customer subscription, you maintain absolute control over your data while leveraging the capabilities of Oracle Cloud Infrastructure managed by Oracle. The Compute Cloud@Customer rack is installed in your data center, connected to your Oracle Cloud Infrastructure tenancy, and fully managed by Oracle.

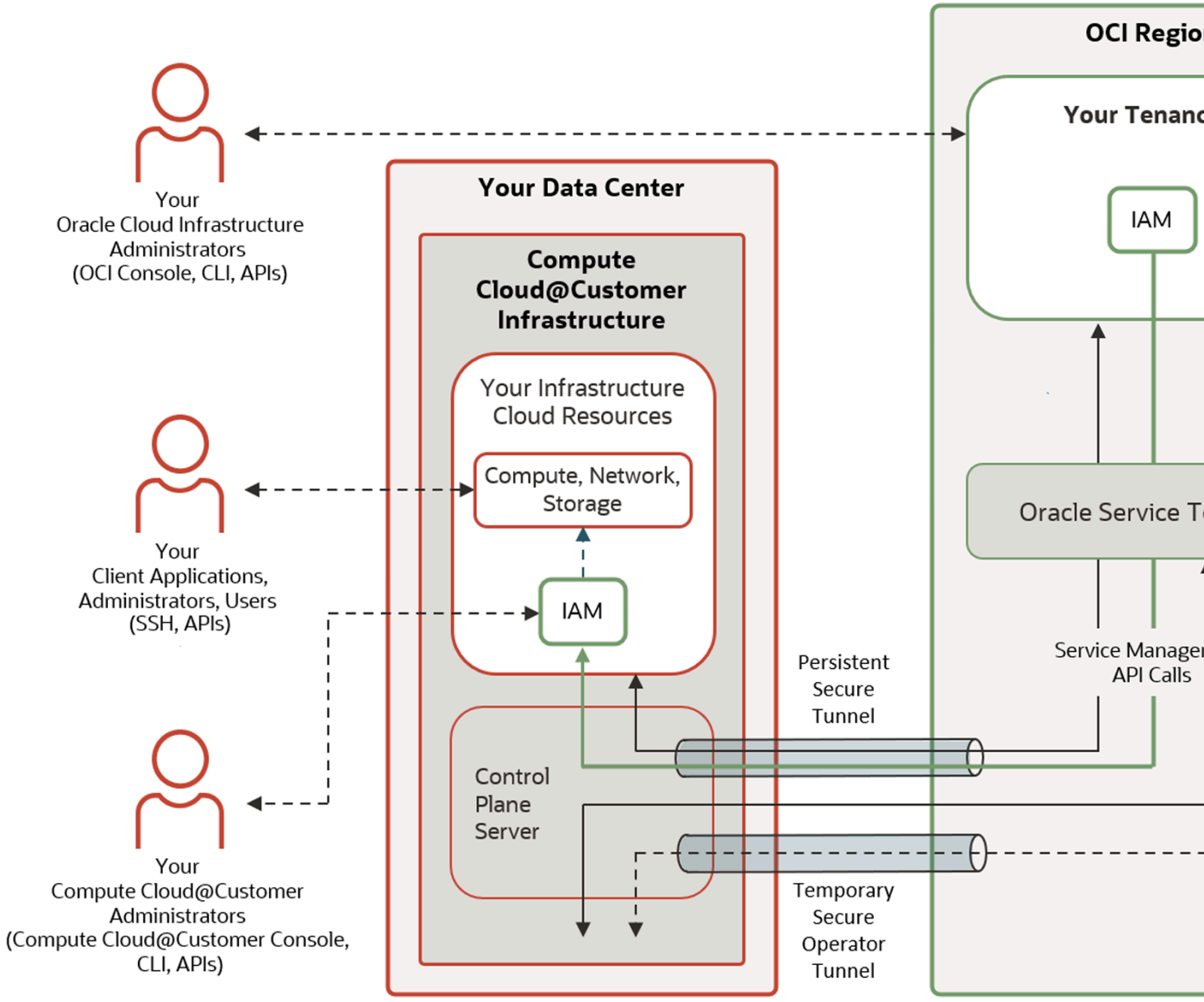
Compute Cloud@Customer is engineered to deliver a comprehensive suite of cloud infrastructure services within the secure environment of your on-premises data center. The system integrates all required hardware and software components, and has been tested, configured and tuned for the best performance by Oracle engineers. It's a flexible, general purpose IaaS (Infrastructure as a Service) solution in the sense that it supports a wide variety of workloads. Its pluggable platform provides an excellent foundation to layer PaaS (Platform as a Service) and SaaS (Software as a Service) solutions on top of the infrastructure.

The cloud operations team uses best-in-class operational processes to secure and monitor the underlying cloud infrastructure. The cloud operations team is responsible for ongoing maintenance of the IaaS platform (upgrades and patching), incident management, and monitoring with OCI-native services.

Oracle installs and initializes the Compute Cloud@Customer infrastructure in your data center. Oracle continues to monitor and maintain the infrastructure over the lifespan of the service.

OCI Connection

The following diagram shows how Compute Cloud@Customer infrastructure is securely connected to your tenancy in an Oracle Cloud Infrastructure region over an uninterruptible network. The connection is persistent. The system can tolerate short interruptions in network connectivity, but such interruptions are treated as faults and must be resolved as quickly as possible.



Disconnected State

In case connectivity with OCI is lost, certain functions are temporarily unavailable.

Preserved functionality:

- Access to all cloud resources on the system, including compute instances, application and network load balancers, OKE clusters, container instances
- Full lifecycle management (create, update, delete, stop, start, terminate) of cloud resources on the system
- Continued access through the Console, OCICLI, and infrastructure as code automation (such as Terraform), for using and managing the cloud resources on the system
- OCPU scaling
- Direct instance management through standard OS tools

Suspended functionality:

- All lifecycle management operations for IAM resources, including users, groups, policies, compartments, tagged namespaces
- Infrastructure management
- Service management through OCI interfaces: browser UI, OCI CLI, REST API/SDK, Terraform
- Oracle infrastructure monitoring, including security monitoring
- Oracle Cloud Automation access for scheduled tasks, for example patching or upgrade
- Oracle Cloud Ops access for remedial and security responses

Authentication

Compute Cloud@Customer uses the same federated identity provider that you use for Oracle Cloud Infrastructure to manage Console sign ins. See Federated Identity Providers (IAM in OCI) on page 109.

IAM Resources

Your Oracle Cloud Infrastructure IAM resources are regularly and securely cached in the Compute Cloud@Customer infrastructure. This enables you to manage IAM resources in one location. If you change IAM resources in the OCI tenancy, the changes are automatically applied to resources in Compute Cloud@Customer. See Identity and Access Management (IAM) on page 108.

User Interfaces

You use the Oracle Cloud Infrastructure Console, CLI, and APIs to manage the following items:

- Initial connection of the Compute Cloud@Customer infrastructure to your tenancy in Oracle Cloud Infrastructure.
- Schedules for when Oracle can upgrade Compute Cloud@Customer.
- Identity and Access Management (IAM), including users, groups, policies, compartments, and tags.
- Billing and payment information.
- Oracle's access to Compute Cloud@Customer using Oracle Operator Access Control.

To manage resources such as VCNs, instances, and storage, on Compute Cloud@Customer, use the following interfaces:

- **OCI API** (Compute Cloud@Customer supports a subset of operations)
- **OCI CLI** (Compute Cloud@Customer supports a subset of operations)
- **Compute Cloud@Customer Console** – a browser UI that offers a similar user experience to the Oracle Cloud Console.

For information about accessing your resources, see Accessing Compute Cloud@Customer User Interfaces on page 102.

Cloud Resources

When you sign in to Compute Cloud@Customer, you can create and manage the same types of resources that you can create in Oracle Cloud Infrastructure:

- **Instances:** You can select the most appropriate type of instance for your applications based on characteristics such as the number of CPUs, amount of memory, and network resources. You can deploy the instance with any of the provided platform images, or bring your own image. See Compute Instances on page 299 and Images for Compute Cloud@Customer Instances on page 283.
- **Virtual Cloud Networks (VCNs):** A virtual version of a traditional network, including subnets, route tables, and gateways, on which your instance runs. At least one cloud network must be set up before you create instances. For information about setting up cloud networks see Virtual Cloud Networks on page 127.
- **Block Volumes:** Lets you dynamically provision and manage block volumes that you can attach to one or more compute instances. See Block Volume Storage on page 380.
- **File Storage:** A durable, scalable, secure, enterprise-grade network file system that you can connect to from any compute instance in your virtual cloud network (VCN). See File Storage on page 433.
- **Object Storage:** A high-performance storage platform that lets you store unstructured data of any content type. This storage is regional and not tied to any specific compute instance. See Object Storage on page 478.

Upgrades

System upgrades are designed for minimum disruption and maximum availability. Health checks are performed before an upgrade to ensure that all components are in an acceptable state. The upgrade process is modular and only upgrades components – such as firmware, OSs, containerized services, or the system's main database – as needed. Oracle performs all upgrades, but you can determine the upgrade time frame.

Compute Cloud@Customer Network Infrastructure

For network connectivity, Compute Cloud@Customer relies on a physical layer that provides the necessary high-availability, bandwidth and speed. On top of this, a distributed network fabric composed of software-defined switches, routers, gateways, and tunnels enables secure and segregated data traffic – both internally between cloud resources, and externally to and from resources outside of Compute Cloud@Customer.

This section provides a network overview. For more details, see Customer Site Network Requirements on page 48.

Data Network

On Compute Cloud@Customer, data connectivity is built on redundant 100Gbit switches in two-layer design similar to a leaf-spine topology.

The leaf switches interconnect the rack hardware components, while the spine switches form the backbone of the network and provide a path for external traffic. Each leaf switch is connected to all the spine switches, which are also interconnected. The main benefits of this topology are extensibility and path optimization. A Compute Cloud@Customer rack contains two leaf and two spine switches.

For external connectivity, 5 ports are reserved on each spine switch. Four ports are available to establish the uplinks between the rack and the data center network; one port is reserved to segregate the administration network from the data traffic.

Administration Network

On Compute Cloud@Customer, the administration network physically separates configuration and management traffic from the operational activity on the data network by providing dedicated secured network paths for administration operations. This network includes all component management interfaces.

This network is only accessed by authorized Oracle operators.

Uplinks

The connections between the Compute Cloud@Customer and the customer data center are called *uplinks*. They're physical cable connections between the spine switches in the rack and one or, preferably, two next-level network devices in the data center.

Besides the physical aspect, there's also a logical aspect to the uplinks: how traffic is routed between the rack and the external network it's connected to.

Physical Connection

On each spine switch, ports 1 to 4 can be used for uplinks to the data center network. For speeds of 10 or 25 Gbps, the spine switch port must be split using an MPO-to-4xLC breakout cable. For speeds of 40 or 100 Gbps each switch port uses a single MPO-to-MPO cable connection. The correct connection speed must be specified during initial setup so that the switch ports are configured with the appropriate breakout mode and transfer speed.

The uplinks are configured during system initialization, based on information you provide as part of the installation checklist. Unused spine switch uplink ports, including unused breakout ports, are disabled for security reasons. The table shows the supported uplink configurations by port count and speed, and the resulting total bandwidth.

Uplink Speed	Number of Uplinks per Spine Switch	Total Bandwidth
10 Gbps	1, 2, 4, 8, or 16	20, 40, 80, 160, or 320 Gbps
25 Gbps	1, 2, 4, 8, or 16	50, 100, 200, 400, or 800 Gbps
40 Gbps	1, 2, or 4	80, 160, or 320 Gbps
100 Gbps	1, 2, or 4	200, 400, or 800 Gbps

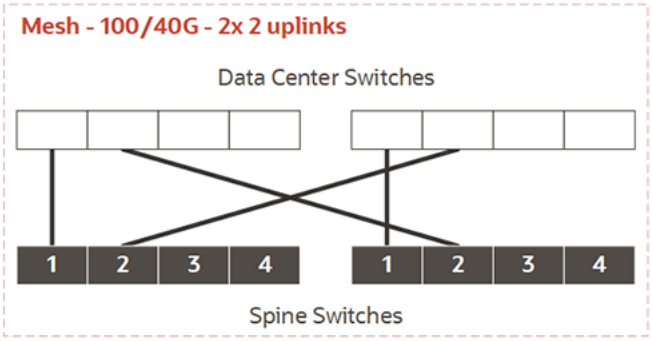
Regardless of the number of ports and port speeds configured, you also select a topology for the uplinks between the spine switches and the data center network. This information is critical for the network administrator to configure link aggregation (port channels) on the data center switches. The table shows the available options.

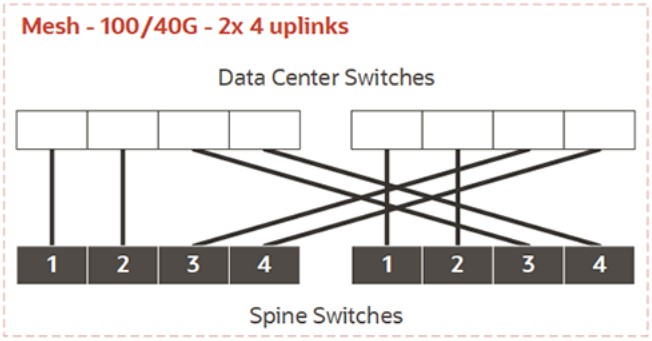
Topology	Description
Triangle	In a triangle topology, all cables from both spine switches are connected to a single data center switch.
Square	In a square topology, two data center switches are used. All outbound cables from a given spine switch are connected to the same data center switch.
Mesh	In a mesh topology, two data center switches are used as well. The difference with the square topology is that uplinks are created in a cross pattern. Outbound cables from each spine switch are connected in pairs: one cable to each data center switch.

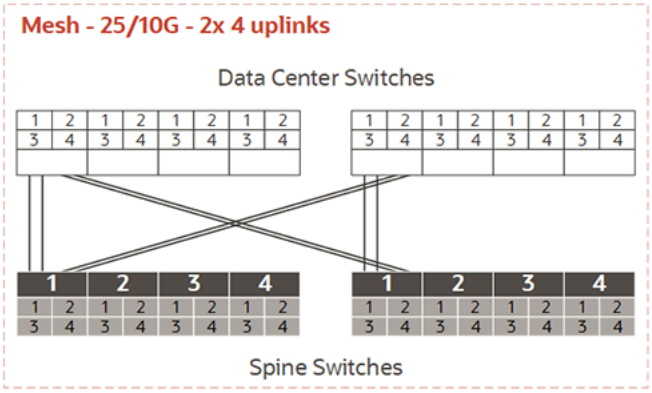
Topology

The physical topology for the uplinks from the rack to the data center network is dependent on bandwidth requirements and available data center switches and ports. Connecting to a single data center switch implies that you select a triangle topology. To increase redundancy you distribute the uplinks across a pair of data center switches, selecting either a square or mesh topology. Each topology allows you to start with a minimum bandwidth, which you can scale up with increasing need. The maximum bandwidth is 800 Gbps, assuming the data center switches, transceivers and cables allow it.

The following diagrams provide a simplified view of supported topologies, and can be used as initial guidance to integrate the rack into the data center network. Use the diagrams and the notes to determine the appropriate cabling and switch configuration for your installation. For more detailed uplink configuration examples, which have been tested by Oracle, see Reference Topologies on page 13.







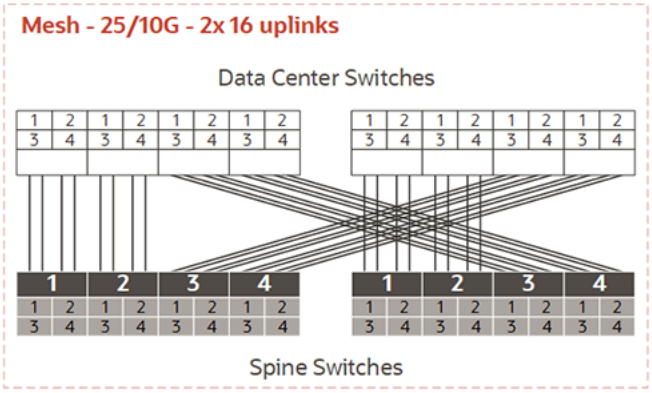


Diagram Notes

On the rack side there are two spine switches that must be connected to the data center network. Both spine switches must have identical port and cable configurations. In each example, the spine switches are shown at the bottom, with all uplink ports identified by their port number. The lines represent outgoing cable connection to the data center switches, which are shown at the top of each example without port numbers.

Cabling Pattern and Port Speed

There are six examples in total, organized in two rows by three columns.

- The top row shows cabling options based on full-port 100-Gbps or 40-Gbps connections. The bottom row shows cabling options using breakout ports at 25-Gbps or 10-Gbps speeds; the smaller boxes numbered 1-4 represent the breakout connections for each of the four main uplink ports per spine switch.
- The third column shows a triangle topology with full-port connections and breakout connections. The difference with column two is that all uplinks are connected to a single data center switch. The total bandwidth is the same, but the triangle topology lacks data center switch redundancy.
- There are no diagrams for the square topology. The square cabling configuration is similar to the mesh examples, but without the crossing patterns. Visually, all connectors in the diagrams would be parallel. In a square topology, all outgoing cables from one spine switch are connected port for port to the same data center switch. Unlike mesh, square implies that each spine switch is peered with only one data center switch.

Link Count

When connecting the uplinks, you are required to follow the spine switch port numbering. Remember that both spine switches are cabled identically, so each uplink or connection corresponds with a pair of cables.

- With one cable per spine port, using 100 or 40 Gbps transceivers, the first uplink pair uses spine switch ports numbered 1, the second uses port 2, and so on. In this configuration, the maximum number of uplinks is four per spine switch.
- When breakout cables are used, with 25 or 10-Gbps port speeds, the first uplink pair uses port 1/1. With two or four uplinks per spine switch there's still only one full port in use. When you increase the uplink count to 8 per spine switch, ports 1/1-2/4 are in use. At 16 uplinks per spine switch, all breakout connections of all four reserved ports will be in use.
- In a mesh topology, a particular cabling pattern must be followed: connect the first half of all uplinks to one data center switch, and the second half to the other data center switch. For example: if you have four uplinks then the first two go to the same switch; if you have eight uplinks (not shown in the diagrams) then the first four go to the same switch; if you have 16 uplinks then the first eight go to the same switch.

Mesh Topology Implications

- In a mesh topology, the spine switch configuration expects that the first half of all uplinks is connected to one data center switch, and the second half to the other data center switch. When you initially connect the rack to your data center network, it is straightforward to follow this pattern.
- However, if you increase the number of uplinks at a later time, the mesh cabling pattern has a significant implication for the existing uplinks. Compare the diagrams in the first two columns: when you double the uplink count, half of the existing connections must be moved to the other data center switch. For 100/40-Gbit uplinks, recabling is only required when you increase the link count from 2 to 4. Because of the larger number of cables, 25/10-Gbit uplinks require more re-cabling: when increasing uplink count from 2 to 4, from 4 to 8, and from 8 to 16.

Logical Connection

The logical connection between the rack and the data center is implemented entirely in layer 3. In the OSI model (Open Systems Interconnection model), layer 3 is known as the network layer, which uses the source and destination IP address fields in its header to route traffic between connected devices.

Compute Cloud@Customer supports two logical connection options: you must choose between static routing and dynamic routing. Both routing options are supported by all three physical topologies.

Connection Type	Description
Static Routing	When static routing is selected, all egress traffic goes through a single default gateway IP address configured on data center network devices. This gateway IP address must be in the same subnet as the rack uplink IP addresses, so it is reachable from the spine switches. The data center network devices can use SVIs (Switch Virtual Interfaces) with VLAN IDs in the range of 2-3899. All gateways configured within a virtual cloud network (VCN) will automatically have a route rule to direct all traffic intended for external destination to the IP address of the default gateway.

Connection Type	Description
Dynamic Routing	When dynamic routing is selected, BGP (Border Gateway Protocol) is used to establish a TCP connection between two Autonomous Systems: the rack network and the data center network. This configuration requires a registered or private ASN (Autonomous System Number) on each side of the connection. Compute Cloud@Customer BGP configuration uses ASN 136025 by default, this can be changed during initial configuration. For BGP routing, two routing devices in the data center must be connected to the two spine switches in the rack. Corresponding interfaces (port channels) between the spine switches and the data center network devices must be in the same subnet. It is considered good practice to use a dedicated /30 subnet for each point-to-point circuit, which is also known as a route hand-off network. This setup provides redundancy and multipathing. Dynamic routing is also supported in a triangle topology, where both spine switches are physically connected to the same data center network device. In this configuration, two BGP sessions are still established: one from each spine switch. However, this approach reduces the level of redundancy.

Supported Routing Designs

The following table shows which routing designs are supported depending on the physical topology in your data center and the logical connection you choose to implement.

Note that link aggregation across multiple devices (vPC or MLAG) is only supported with static routing. When dynamic routing is selected, link aggregation is restricted to ports of the same switch.

When the uplinks are cabled in a mesh topology, a minimum of 2 physical connections per spine switch applies. To establish BGP peering, 2 subnets are required. If the uplink count changes, the port channels are reconfigured but the dedicated subnets remain the same.

Logical Connection	Physical Topology	Routing Design		
		Single Subnet	Dual Subnet	vPC/MLAG
Static Routing	Square	Yes	Yes	Yes
	Mesh	Yes	Yes	Yes
	Triangle	Yes	Yes	Yes
Dynamic Routing	Square	Yes	--	--
	Mesh	--	Yes	--
	Triangle	Yes	--	--

Uplink Protocols

On Compute Cloud@Customer, the uplinks to the data center run various protocols to provide redundancy and reduce link failure detection and recovery times on these links. These protocols work with the triangle, square, or mesh topologies.

The suite of uplink protocols include:

- Bidirectional Forwarding Detection (BFD)
- Virtual Router Redundancy Protocol (VRRP)

- Hot Spare Router Protocol (HSRP)
- Equal Cost Multipath (ECMP)

Each is briefly described in the following sections of this topic.

BFD

In most router networks, connection failures are detected by loss of the "hello" packets sent by routing protocols. However, detection by this method often takes more than one second, routing a lot of packets on high-speed links to a destination that they can't reach, which burdens link buffers. Increasing the "hello" packet rate burdens the router CPU.

Bidirectional Forwarding Detection (BFD) is a built-in mechanism that alerts routers at the end of a failed link that there's a problem more quickly than any other mechanism, reducing the load on buffers and CPUs. BFD works even in situations where there are switches or hubs between the routers.

BFD requires no configuration and has no user-settable parameters.

VRRPv3

The Virtual Router Redundancy Protocol version 3 (VRRPv3) is a networking protocol that uses the concept of a virtual router to group physical routers together and make them appear as one to participating hosts. This increases the availability and reliability of routing paths through automatic default gateway selections on an IP subnetwork.

With VRRPv3, the primary/active and secondary/standby routers act as one virtual router. This virtual router becomes the default gateway for any host on the subnet participating in VRRPv3. One physical router in the group becomes the primary/active router for packet forwarding. However, if this router fails, another physical router in the group takes over the forwarding role, adding redundancy to the router configuration. The VRRPv3 "network" is limited to the local subnet and doesn't advertise routes beyond the local subnet.

HSRP

Cisco routers often use a redundancy protocol called the Hot Spare Router Protocol (HSRP) to improve router availability. Similar to the methods of VRRP, HSRP groups physical routers into a single virtual router. The failure of a physical default router results in another router using HSRP to take over the default forwarding of packets without stressing the host device.

ECMP

Equal Cost Multipath (ECMP) is a way to make better use of network bandwidth, especially in more complex router networks with many redundant links.

Normally, router networks with multiple router paths to another destination network choose one active route to a gateway router as the "best" path and use the other paths as a standby in case of failure. The decision about which path to a network gateway router to use is usually determined by its "cost" from the routing protocol perspective. In cases where the cost over several links to reach network gateways are equal, the router chooses one based on some criteria. This makes routing decisions easy but wastes network bandwidth as network links on paths not chosen sit idle.

ECMP is a way to send traffic on multiple path links with equal cost, making more efficient use of network bandwidth.

Reference Topologies

These detailed topologies are examples of uplink configurations that have been tested and are known to work as expected. Each example includes a diagram for illustration, and configuration guidelines.

When preparing for the installation of Compute Cloud@Customer, you decide how the system will be connected to the data center network: uplink count, cabling pattern, connection speed, routing design, and so on. Decisions about uplinks are based on the network infrastructure in your data center, and the bandwidth and availability requirements of your Compute Cloud@Customer environment. These reference topologies help you make informed decisions for your specific environment, and provide guidance to configure your switches for the topology you select.

> **Caution:**
>
> The configuration parameters provided in the reference topologies are intended as guidelines for network administrators. The exact configuration details of your specific environment must be aligned with your data center network design.

Mesh Topology with ECMP Static Routing

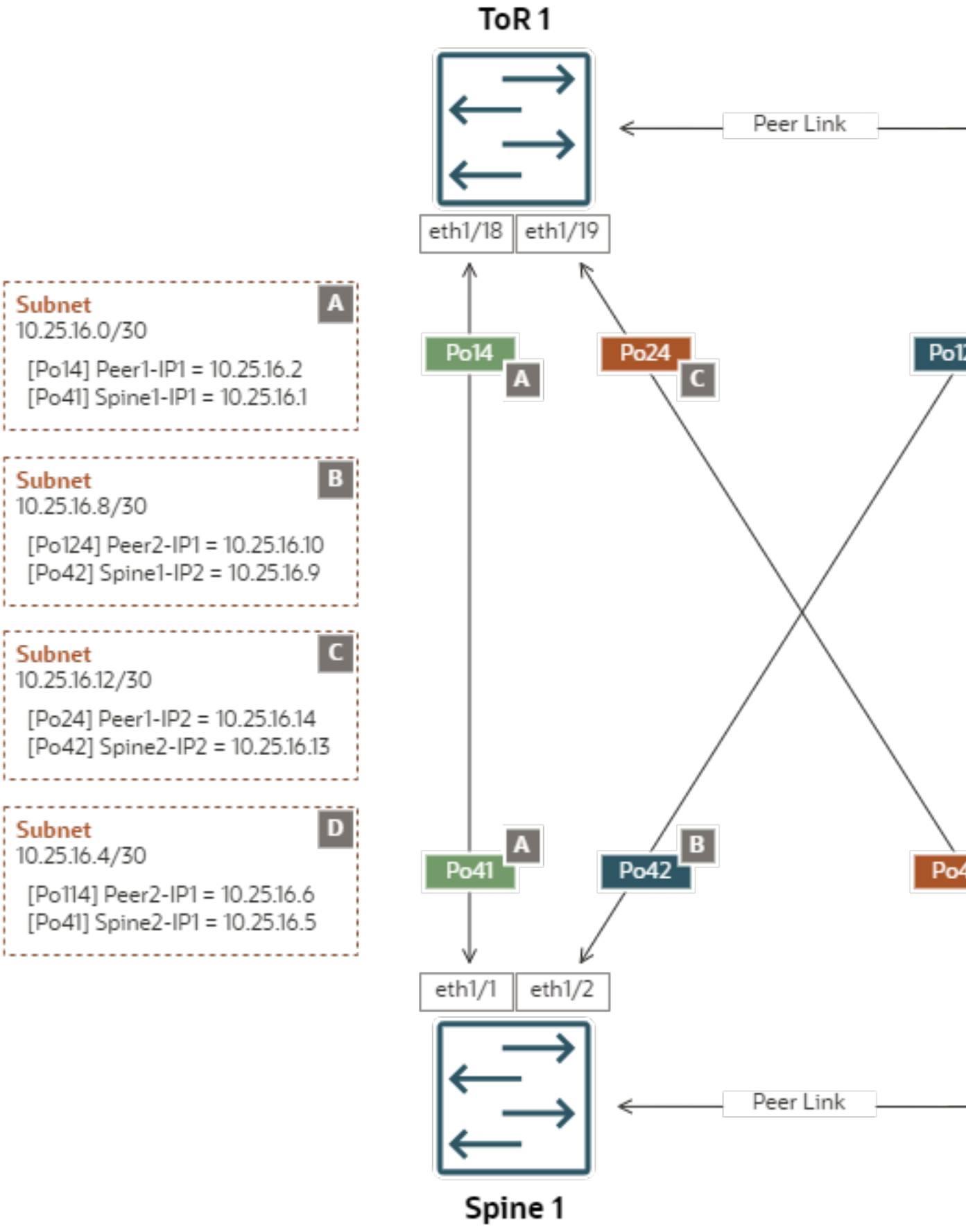
ECMP Mesh enables Layer 3 network deployment according to industry-proven best practices. This uplink topology is highly recommended.

Configuration Properties

- Mesh topology – each spine switch is connected to two independent data center switches
- Static routing – all egress traffic from an uplink goes through a single gateway IP configured on its peer network device in the data center
- ECMP – bandwidth optimization across multiple redundant links or paths
- Separate /30 subnets – each uplink connects one spine switch port channel to one data center switch port channel in a /30 subnet

Topology Highlights

- All uplinks are configured as LACP/active port channels with *rate=fast*
  - Port channel **Po41** represents the first set of links on both spine switches. They **connect straight** to the corresponding ToR switches.
  - Port channel **Po42** represents the second set of links on both spine switches. They **cross-connect** to the corresponding ToR switches.
- ToR switch ports connecting to the spine switches must be set up in access mode. Spanning tree protocol must be disabled.
- Requires 4 unique subnets: a /30 subnet size is recommended, but /31 is possible if the ToR switches support it.
  - Equal cost static routes to both ToR switches are set up automatically.
  - Egress traffic can hash to any of the 4 uplinks.
  - It is NOT possible to isolate specific VCN/VM egress traffic through one particular uplink.


ToR 1
Peer Link
eth1/18
eth1/19
Subnet
10.25.16.0/30
[Po14] Peer1-IP1 = 10.25.16.2
[Po41] Spine1-IP1 = 10.25.16.1
A
Subnet
10.25.16.8/30
[Po124] Peer2-IP1 = 10.25.16.10
[Po42] Spine1-IP2 = 10.25.16.9
B
Subnet
10.25.16.12/30
[Po24] Peer1-IP2 = 10.25.16.14
[Po42] Spine2-IP2 = 10.25.16.13
C
Subnet
10.25.16.4/30
[Po114] Peer2-IP1 = 10.25.16.6
[Po41] Spine2-IP1 = 10.25.16.5
D
Po14
A
Po24
C
Po41
A
Po42
B
eth1/1
eth1/2
Peer Link
Spine 1

Detailed Spine Switch Configuration Example

- Spine Switch 1

```
interface port-channel41
  description "customer uplink"
  no switchport
  mtu 9216
  speed 10000
  no negotiate auto
  ip access-group ingress-ports-acl in
  no ip redirects
  ip address 10.25.16.1/30
  ip nat outside

interface port-channel42
  description "customer uplink 2"
  no switchport
  mtu 9216
  speed 10000
  no negotiate auto
  ip access-group ingress-ports-acl in
  no ip redirects
  ip address 10.25.16.9/30
  ip nat outside

ip route 0.0.0.0/0 po41 10.25.16.2 20
ip route 0.0.0.0/0 po42 10.25.16.10 20
```

  Routes added:

```
0.0.0.0/0, ubest/mbest: 2/0
  *via 10.25.16.2, [20/0], 6d08h, static
  *via 10.25.16.10, [20/0], 6d08h, static
```

- Spine Switch 2

```
interface port-channel41
  description "customer uplink"
  no switchport
  mtu 9216
  speed 10000
  no negotiate auto
  ip access-group ingress-ports-acl in
  no ip redirects
  ip address 10.25.16.5/30
  ip nat outside

interface port-channel42
  description "customer uplink 2"
  no switchport
  mtu 9216
  speed 10000
  no negotiate auto
  ip access-group ingress-ports-acl in
  no ip redirects
  ip address 10.25.16.13/30
  ip nat outside

ip route 0.0.0.0/0 po41 10.25.16.6 20
ip route 0.0.0.0/0 po42 10.25.16.14 20
```

Routes added:

```
0.0.0.0/0, ubest/mbest: 2/0
  *via 10.25.16.6, [20/0], 6d07h, static
  *via 10.25.16.14, [20/0], 6d07h, static
```

Mesh Topology with VRRP Static Routing (Not Supported)

VRRP Mesh would have a negative impact on network performance. This uplink topology is NOT recommended.

When the uplinks are set up in a mesh topology, there are 4 distinct Layer 3 uplinks to the ToR switches. In a Virtual Router Redundancy Protocol switch configuration, egress traffic is typically routed through the uplink with the primary/active role. Thus, a VRRP mesh configuration would reduce the egress bandwidth to only 25 percent of the actual capacity. Therefore, this configuration is considered not supported.

Mesh Topology with Dynamic Routing

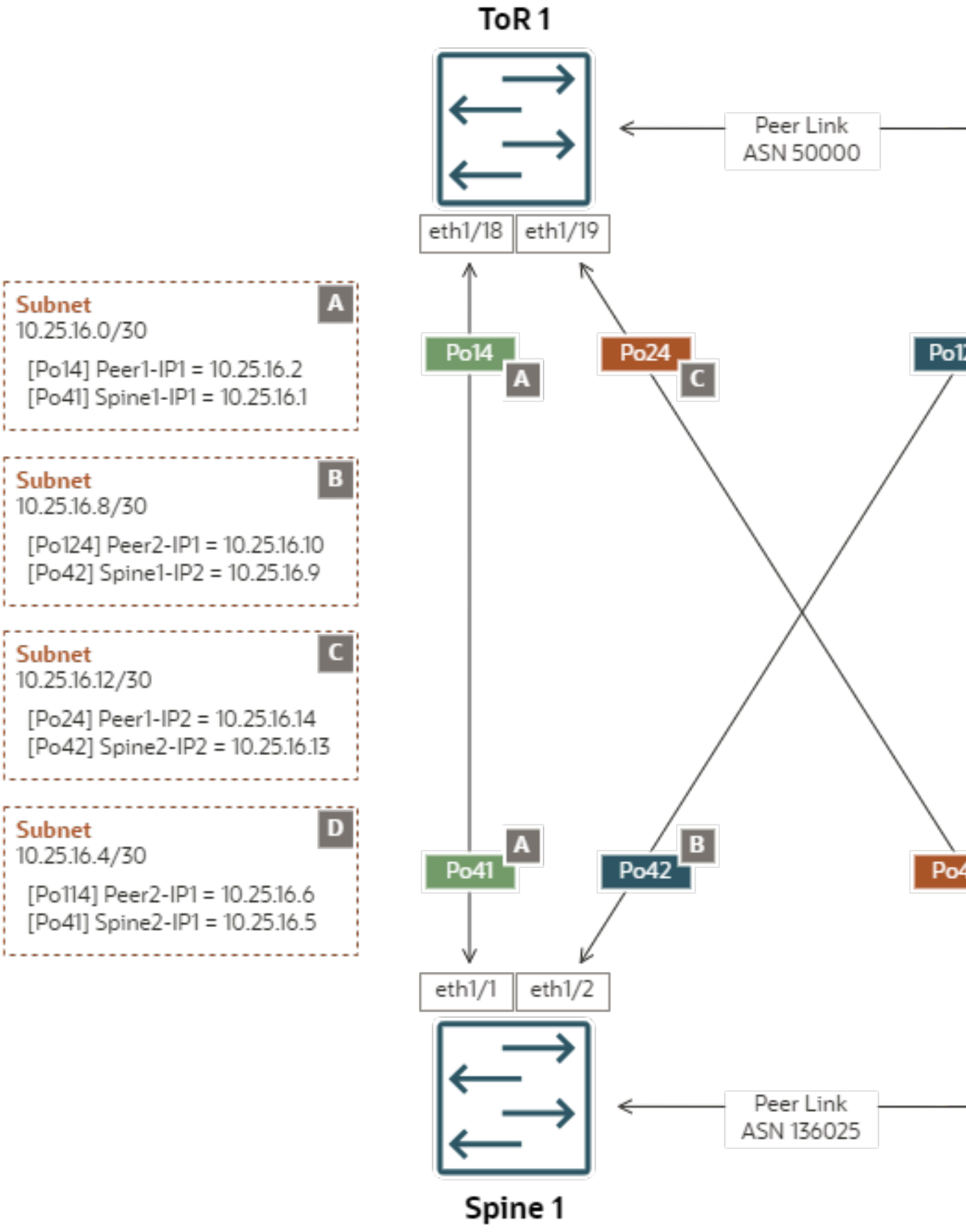
Dynamic Mesh enables Layer 3 network deployment according to industry-proven best practices. This uplink topology is highly recommended.

Configuration Properties

- Mesh topology – each spine switch is connected to two independent data center switches
- Dynamic routing – both peered Autonomous Systems, the appliance and the data center, exchange routing information using eBGP (external Border Gateway Protocol). The best routing path is dynamically adjusted based on network availability information advertized by each AS.
- Separate /30 subnets – each uplink connects one spine switch port channel to one data center switch port channel in a /30 subnet

Topology Highlights

- All uplinks are configured as LACP/active port channels with *rate=fast*
  - Port channel **Po41** represents the first set of links on both spine switches. They **connect straight** to the corresponding ToR switches.
  - Port channel **Po42** represents the second set of links on both spine switches. They **cross-connect** to the corresponding ToR switches.
- ToR switch ports connecting to the spine switches must be set up in access mode. Spanning tree protocol must be disabled.
- Requires 4 unique subnets: a /30 subnet size is recommended, but /31 is possible if the ToR switches support it.
  - Two eBGP peering sessions are established between each spine and both ToR switches.
  - Egress traffic can hash to any of the 4 uplinks.
  - It is NOT possible to isolate specific VCN/VM egress traffic through one particular uplink.


ToR 1
Peer Link
ASN 50000
eth1/18
eth1/19
Subnet
10.25.16.0/30
[Po14] Peer1-IP1 = 10.25.16.2
[Po41] Spine1-IP1 = 10.25.16.1
A
Subnet
10.25.16.8/30
[Po124] Peer2-IP1 = 10.25.16.10
[Po42] Spine1-IP2 = 10.25.16.9
B
Subnet
10.25.16.12/30
[Po24] Peer1-IP2 = 10.25.16.14
[Po42] Spine2-IP2 = 10.25.16.13
C
Subnet
10.25.16.4/30
[Po114] Peer2-IP1 = 10.25.16.6
[Po41] Spine2-IP1 = 10.25.16.5
D
Po14
Po24
Po41
Po42
eth1/1
eth1/2
Peer Link
ASN 136025
Spine 1

Spine Switch Configuration Details

- Spine Switch 1

```
interface port-channel41
  description "customer uplink"
  no switchport
  mtu 9216
  speed 10000
  no negotiate auto
  ip access-group ingress-ports-acl in
  no ip redirects
  ip address 10.25.16.1/30
  ip nat outside

interface port-channel42
  description "customer uplink 2"
  no switchport
  mtu 9216
  speed 10000
  no negotiate auto
  ip access-group ingress-ports-acl in
  no ip redirects
  ip address 10.25.16.9/30
  ip nat outside

router bgp 136025
  router-id 10.25.16.1
  neighbor 10.25.16.2
    bfd singlehop
    remote-as 50000
    address-family ipv4 unicast
  neighbor 10.25.16.10
    bfd singlehop
    remote-as 50000
    address-family ipv4 unicast

BGP Sessions:
ASN 136025
VRF default, local ASN 136025

Neighbor        ASN    Flaps LastUpDn|LastRead|LastWrit  St  Port(L/R)
 Notif(S/R)
10.25.16.2     50000  0     1w4d    |00:00:50|00:00:20  E   34408/179
   0/0
10.25.16.10    50000  0     1w4d    |00:00:43|00:00:20  E   57322/179
   0/0
```

- Spine Switch 2

```
interface port-channel41
  description "customer uplink"
  no switchport
  mtu 9216
  speed 10000
  no negotiate auto
  ip access-group ingress-ports-acl in
  no ip redirects
  ip address 10.25.16.5/30
  ip nat outside

interface port-channel42
  description "customer uplink 2"
  no switchport
```

```
  mtu 9216
  speed 10000
  no negotiate auto
  ip access-group ingress-ports-acl in
  no ip redirects
  ip address 10.25.16.13/30
  ip nat outside

router bgp 136025
  router-id 10.25.16.5
  neighbor 10.25.16.6
    bfd singlehop
    remote-as 50000
    address-family ipv4 unicast
  neighbor 10.25.16.14
    bfd singlehop
    remote-as 50000
    address-family ipv4 unicast

BGP Sessions:
ASN 136025
VRF default, local ASN 136025

Neighbor        ASN    Flaps LastUpDn|LastRead|LastWrit  St  Port(L/R)
 Notif(S/R)
10.25.16.6      50000  0     1w4d    |00:00:50|00:00:20  E   34408/179
   0/0
10.25.16.14     50000  0     1w4d    |00:00:43|00:00:20  E   57322/179
   0/0
```

Square Topology with ECMP Static Routing

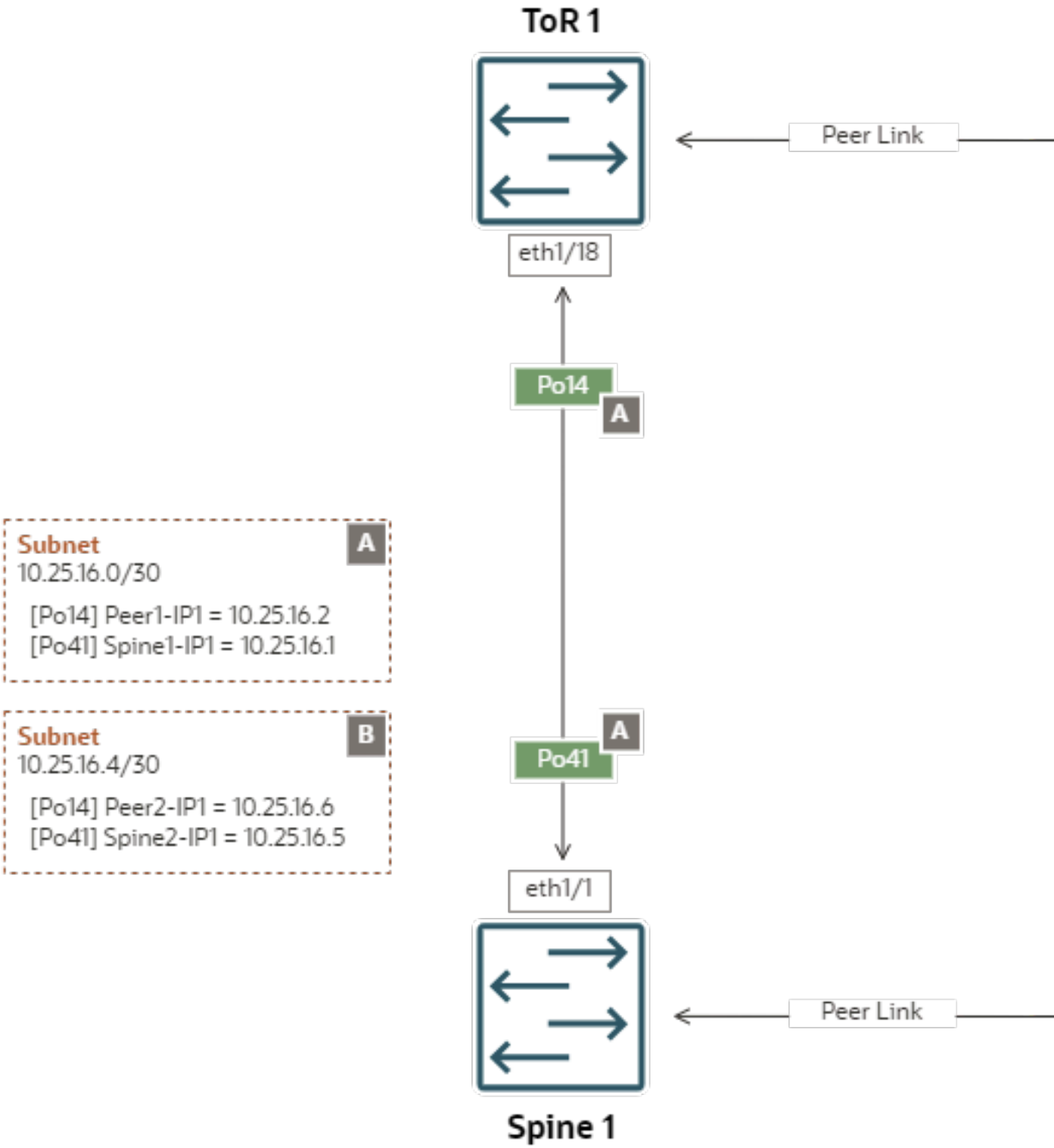
ECMP Square enables Layer 3 network deployment according to industry-proven best practices. This uplink topology is highly recommended.

Configuration Properties

- Square topology – each spine switch is connected to a different independent data center switch
- Static routing – all egress traffic from an uplink goes through a single gateway IP configured on its peer network device in the data center
- ECMP – bandwidth optimization across multiple redundant links or paths
- Separate /30 subnets – each uplink connects one spine switch port channel to one data center switch port channel in a /30 subnet

Topology Highlights

- All uplinks are configured as LACP/active port channels with *rate=fast*
- ToR switch ports connecting to the spine switches must be set up in access mode. Spanning tree protocol must be disabled. The ToR switches must NOT be configured with vPC.
- Requires 2 unique subnets: a /30 subnet size is recommended, but /31 is possible if the ToR switches support it.
  - Equal cost static routes to both ToR switches are set up automatically.
  - Egress traffic can hash to any of the 2 uplinks.
  - It is NOT possible to isolate specific VCN/VM egress traffic through one particular uplink.



Detailed Spine Switch Configuration Example

- Spine Switch 1

```
interface port-channel41
  description "customer uplink"
  no switchport
  mtu 9216
  speed 10000
  no negotiate auto
  ip access-group ingress-ports-acl in
  no ip redirects
  ip address 10.25.16.1/30
  ip nat outside
```

Routes added:

```
0.0.0.0/0, ubest/mbest: 2/0
  *via 10.25.16.2, [20/0], 6d08h, static
  *via 10.25.16.6, [100/0], 6d08h, static
```

- Spine Switch 2

```
interface port-channel41
  description "customer uplink"
  no switchport
  mtu 9216
  speed 10000
  no negotiate auto
  ip access-group ingress-ports-acl in
  no ip redirects
  ip address 10.25.16.5/30
  ip nat outside
```

Routes added:

```
0.0.0.0/0, ubest/mbest: 2/0
  *via 10.25.16.6, [20/0], 6d07h, static
  *via 10.25.16.2, [100/0], 6d07h, static
```

Triangle Topology with ECMP Static Routing

ECMP Triangle enables peering with a single data center router or switch.

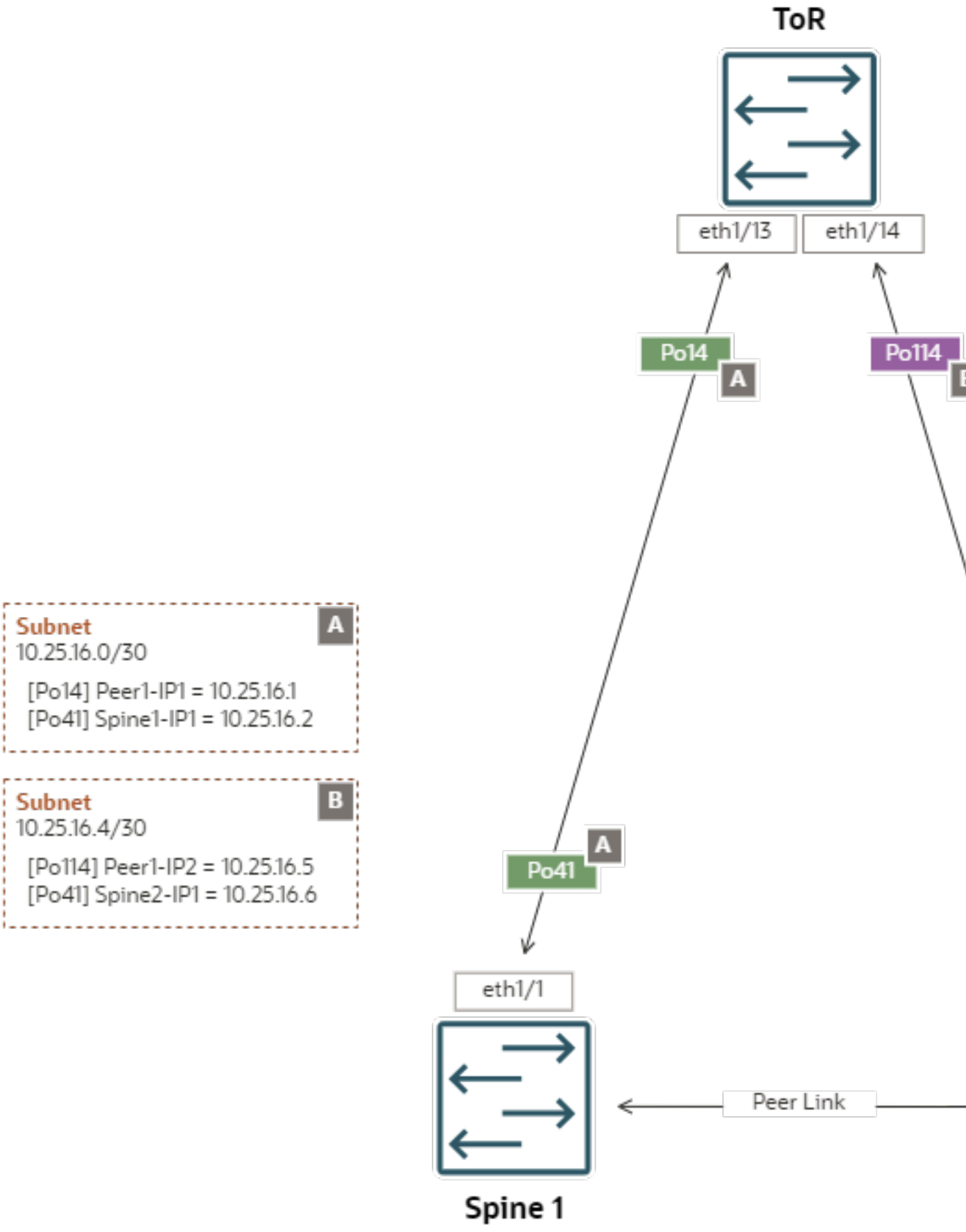
Configuration Properties

- Triangle topology – both spine switches are connected to a single data center switch
- Static routing – all egress traffic from an uplink goes through a single gateway IP configured on its peer network device in the data center
- ECMP – bandwidth optimization across multiple redundant links or paths
- Separate /30 subnets – each uplink connects one spine switch port channel to one data center switch port channel in a /30 subnet

Topology Highlights

- All uplinks are configured as LACP/active port channels with *rate=fast*
- Port channel **Po41** represents the set of links configured on both spine switches. Port channels **Po14** and **Po114** represent the corresponding sets of links configured on the ToR switch.
- ToR switch ports connecting to the spine switches must be set up in access mode. Spanning tree protocol must be disabled.
- Requires 2 unique subnets: a /30 subnet size is recommended, but /31 is possible if the ToR switch supports it.
  - Equal cost static routes to the ToR switch are set up automatically.
  - Egress traffic can hash to any of the 2 uplinks.

- It is NOT possible to isolate specific VCN/VM egress traffic through one particular uplink.


ToR
eth1/13
eth1/14
Po14
A
Po114
Subnet
10.25.16.0/30
A
[Po14] Peer1-IP1 = 10.25.16.1
[Po41] Spine1-IP1 = 10.25.16.2
Subnet
10.25.16.4/30
B
[Po114] Peer1-IP2 = 10.25.16.5
[Po41] Spine2-IP1 = 10.25.16.6
Po41
A
eth1/1
Peer Link
Spine 1

Detailed Spine Switch Configuration Example

- Spine Switch 1

```
interface port-channel41
  no shutdown
  mtu 9216
  ip address 10.25.16.2/30
  no ipv6 redirects
  ip proxy-arp
  ip nat outside

ip route 0.0.0.0/0 po41 10.25.16.1 20
ip route 0.0.0.0/0 po41 10.25.16.5 100
```

- Spine Switch 2

```
interface port-channel41
  no shutdown
  mtu 9216
  ip address 10.25.16.6/30
  no ipv6 redirects
  ip proxy-arp
  ip nat outside

ip route 0.0.0.0/0 po41 10.25.16.5 20
ip route 0.0.0.0/0 po41 10.25.16.1 100
```

Topology for Segregated Administration Traffic

With segregated administration traffic on Compute Cloud@Customer, you cable and configure the dedicated connections for the administration network using the same topologies as the data network. All the same principles apply.

The administration network topology does not need to match the data network: it can be based on another reference topology, using a different number of physical ports operating at a different link speed. The uplink configuration does require different subnets, IP addresses and port channels, but otherwise looks exactly like a data network.

On the spine switches, the administration network is hard-coded to use port channel 45 (Po45), whereas the data network uses Po41 and Po42. On the data center side of the uplinks you are allowed to use any valid port channel ID that fits the existing data center configuration.

Reserved Network Resources

On Compute Cloud@Customer, the network infrastructure and system components need many IP addresses and several VLANs for internal operation. It's critical to avoid conflicts with the addresses in use in the customer data center and the CIDR ranges configured in the virtual cloud networks (VCNs).

These IP address ranges are reserved for internal use by Compute Cloud@Customer:

Reserved IP Addresses	Description
CIDR blocks in Shared Address Space	The Shared Address Space, with IP range 100.64.0.0/10, was implemented to connect customer-premises equipment to the core routers of Internet service providers. To allocate IP addresses to the management interfaces and ILOMs of hardware components, two CIDR blocks are reserved for internal use: 100.96.0.0/23 and 100.96.2.0/23.

Reserved IP Addresses	Description
CIDR blocks in Class E address range	Under the classful network addressing architecture, Class E is the part of the 32-bit IPv4 address space ranging from 240.0.0.0 to 255.255.255.255. At the time, it was reserved for future use, so it can't be used on the public Internet. To accommodate the addressing requirements of all infrastructure networking over the physical 100Gbit connections, the entire 253.255.0.0/16 subnet is reserved. It is further subdivided into multiple CIDR blocks to group IP addresses by network function or type. The various CIDR blocks within the 253.255.0.0/16 range are used to allocate IP addresses for the Kubernetes containers running the microservices, the virtual switches, routers and gateways enabling the VCN data network, the hypervisors, the Compute Cloud@Customer chassis components, and so on.
Link Local CIDR block	A link-local address belongs to the 169.254.0.0/16 IP range, and is valid only for connectivity within a host's network segment, because the address isn't guaranteed to be unique outside that network segment. Packets with link-local source or destination addresses aren't forwarded by routers. The link-local CIDR block 169.254.239.0/24, and the IP address 169.254.169.254, are reserved for functions such as DNS requests, compute instance metadata transfer, and cloud service endpoints.

All VCN traffic – from one VCN to another, and between a VCN and external resources – flows across the 100-Gbit connections and is carried by VLAN 3900. Traffic related to server management is carried by VLAN 3901. All VLANs with higher IDs are also reserved for internal use, and VLAN 1 is the default for untagged traffic. The remaining VLAN range of 2 - 3899 is available for customer use.

Integrating External Systems with Flex Networks

Optionally, Compute Cloud@Customer can be integrated with external systems in the on-premises network using direct cable connections to reserved ports on the spine switches.

Compute Cloud@Customer can be integrated with external systems, such as Oracle Exadata, Oracle ZFS Storage Appliance, Oracle Database Appliance, and so on. In this configuration, external components are directly connected to reserved ports on the spine switches of the Compute Cloud@Customer. Four 100Gbit ports per spine switch are reserved and split into 4x25Gbit breakout ports, providing a maximum of 32 total cable connections. Each externally connected component is cabled directly to both spine switches, meaning up to 16 external connections of this type can be made. **This feature was formerly called Exadata Networks**.

> **Note:**
>
> The 25 Gbps breakout configuration applies to connectivity with Oracle Exadata nodes or Oracle Database Appliance nodes and does not indicate that Flex Networks are limited to 25 Gbps.

When the cable connections are in place, the infrastructure administrator configures a *Flex network*, which enables traffic between the connected system(s) and a set of compute instances. These prerequisites apply:

- The Flex network must not overlap with any subnets in the on-premises network.
- The VCNs containing compute instances that connect to the external nodes, must have a dynamic routing gateway (DRG) configured.
- The relevant subnet route tables must contain rules to allow traffic to and from the Flex network.

The Flex network configuration determines which external components are exposed and which subnets have access to those components. Access can be enabled or disabled per connected system and per compute subnet. In addition, the Flex network can be exposed through the appliance's external network, allowing other resources within the on-

premises network to connect to the external system through the spine switches of the appliance. The Flex network configuration is created and managed through the Service Enclave.

Compute and Storage Configurations

Compute Cloud@Customer offers different compute and storage configurations to suit various workloads.

The table that follows, lists the OCPU count and usable storage for balanced storage and performance storage options.

- Balanced or performance storage is expanded by adding disk trays to the ZFS Storage Appliance. The maximum total capacity for each type of storage is listed in the table. Balanced and performance trays can be combined up to a combined maximum of 20.
- Compute capacity can be further expanded in sets of 3 compute nodes in an expansion rack. Each 3-node expansion adds 552 OCPUs and 6.7TB memory to the specifications in the table.

Base Rack Configuration	OCPU Count	Usable Balanced Storage	Usable Performance Storage
Small (Oracle Compute Cloud@Customer Infrastructure - X11 - Base System with up to 552 OCPUs, 6.7 TB Memory, and 150 TB Usable Storage - Non-metered)	552	13 Trays: 2.425 PB	13 Trays: 780 TB
Medium (Oracle Compute Cloud@Customer Infrastructure - X11 - Base System with up to 1,104 OCPUs, 13.4 TB Memory, and 150 TB Usable Storage - Non-metered)	1,104	13 Trays: 2.425 PB	13 Trays: 780 TB
Large (Oracle Compute Cloud@Customer Infrastructure - X11 - Base System with up to 1,656 OCPUs, 20.1 TB Memory, and 150 TB Usable Storage - Non-metered)	1,656	20 Trays: 3.65 PB	20 Trays: 1,200 TB

High Availability

Compute Cloud@Customer is engineered to eliminate single points of failure, enabling the system and hosted workloads to remain operational in case of hardware or software faults, and during upgrades and maintenance operations.

Redundancy is built into the architecture at every level: hardware, controller software, master database, services, and so on. Features such as backup, automated service requests and optional disaster recovery further enhance the system's serviceability and continuity of service.

Hardware Redundancy

The minimum base rack configuration contains redundant networking, storage and server components to ensure that failure of any single element doesn't affect overall system availability.

Data connectivity throughout the system is built on redundant pairs of leaf and spine switches. Link aggregation is configured on all interfaces: switch ports, host NICs and uplinks. The leaf switches interconnect the rack components using cross-cabling to redundant network interfaces in each component. Each leaf switch also has a connection to each of the spine switches, which are also interconnected. The spine switches form the backbone of the network and enable traffic external to the rack. Their uplinks to the data center network consist of two cable pairs, which are cross-connected to two redundant ToR (top-of-rack) switches.

The management cluster, which runs the controller software and system-level services, consists of three fully active management nodes. Inbound requests pass through the virtual IP of the management node cluster, and are distributed across the three nodes by a load balancer. If one of the nodes stops responding and fences from the cluster, the load balancer continues to send traffic to the two remaining nodes until the failing node is healthy again and rejoins the cluster.

Storage for the system and for the cloud resources in the environment is provided by the internal ZFS Storage Appliance. Its two controllers form an active-active cluster, providing high availability and excellent throughput at the same time. The ZFS pools are built on disks in a mirrored configuration for the best data protection.

System Availability

The software and services layer are deployed on the three-node management cluster, and take advantage of the high availability that's inherent to the cluster design. The Kubernetes container orchestration environment also uses clustering for both its own controller nodes and the service pods it hosts. Many replicas of the microservices are running at any particular time. Nodes and pods are distributed across the management nodes, and Kubernetes ensures that failing pods are replaced with new instances to keep all services running in an active/active setup.

All services and components store data in a common, central MySQL database. The MySQL cluster database has instances deployed across the three management nodes. Availability, load balancing, data synchronization and clustering are all controlled by internal components of the MySQL cluster.

A significant part of the system-level infrastructure networking is software-defined. The configuration of virtual switches, routers and gateways isn't stored and managed by the switches, but is distributed across several components of the network architecture. The network controller is deployed as a highly available containerized service.

The upgrade framework leverages the hardware redundancy and the clustered designs to provide rolling upgrades for all components. During the upgrade of one component instance, the remaining instances ensure that there's no downtime. The upgrade is complete when all component instances have been upgraded and returned to normal operation.

Compute Instance Availability

For a compute instance, high availability refers to the automated recovery of an instance in case the underlying infrastructure fails. The state of the compute nodes, hypervisors, and compute instances is monitored continually. Each compute node is polled with a 5 minute interval. When compute instances go down, by default the system tries to recover it automatically.

By default, the system attempts to restart instances in their selected fault domain but will restart instances in other fault domains if insufficient resources are available in the selected fault domain. The selected fault domain is the fault domain that's specified in the instance configuration.

If a compute node goes down because of an unplanned reboot, when the compute node successfully returns to normal operation, instances are restarted. At the next polling interval, by default if instances are found that should be running but are in a different state, the start command is issued again. If any instances have stopped and remain in that state, the hypervisor tries to restart them up to 5 times. Instances that weren't running before the compute node became unavailable, remain shut down when the compute node is up and running again.

A compute node is considered failing when it has been disconnected from the data network or has been in a powered-off state for about 5 minutes. This 5-minute timeout corresponds with two unsuccessful polling tries, and is the threshold for placing the compute node in `FAIL` state and its agent in `EVACUATING` state. This condition is required before the reboot migration can start.

Reboot migration implies that all compute instances from the failing compute node are stopped and restarted on another compute node. When migration is complete, the failing compute node's agent indicates that instances have been evacuated. If the compute node eventually reboots successfully, it must go through a cleanup process that removes all stale instance configurations and associated virtual disks. After cleanup, the compute node can host compute instances again.

During the entire reboot migration, the instances remain in "moving" configuration state. After migration is completed, the instance configuration state is changed to "running". Instances that were stopped before the failure aren't migrated, because they're not associated with any compute node.

Continuity of Service

Compute Cloud@Customer offers several features that further enhance high availability. Health monitoring at all levels of the system is a key factor. Diagnostic and performance data is collected from all components, then centrally stored and processed, and made available to Oracle personnel.

To mitigate data loss and support the recovery of system and services configuration in case of failure, consistent and complete backups are made regularly.

Optionally, workloads deployed on Compute Cloud@Customer can be protected against downtime and data loss through the implementation of disaster recovery. To achieve this, two Compute Cloud@Customer infrastructures need to be set up at different sites, and configured to be each others replica. Resources under disaster recovery control are stored separately on the ZFS Storage Appliances in each system, and replicated between the two. When an incident occurs at one site, the environment is brought up on the replica system with minimal downtime. We recommend that disaster recovery is implemented for all critical production systems.

Using Your Own CA Trust Chain

On Compute Cloud@Customer, you can provide your own Certificate Authority (CA) certificates which enables you to use your CA trust chain. To use your own certificates, open a Support Request, and request to use your own certificates. See Creating a Support Request. To access support, sign in to the Oracle Cloud Console as described in Sign In to the OCI Console.

Oracle Operator Access Control

Compute Cloud@Customer supports the use of Oracle Operator Access Control to manage requests for temporary access to your organization's cloud resources from OCI authorized operators.

Authorized operators need to access resources to upgrade Compute Cloud@Customer, troubleshoot or help resolve an issue. Oracle Operator Access Control provides a secure workflow through which operators request access your organization's cloud environment.

Oracle Operator Access Control:

- Provides the operator temporary user credentials for a specific duration.
- Enables you to approve or deny the access requests.
- Specifies the access level for the representative.
- Creates logs of all actions, providing an audit trail.

For more information about Oracle Operator Access Control, see Oracle Operator Access Control.

For common tasks, see Managing Infrastructure Access with Operator Access Control

> **Attention:**
>
> When using the Oracle Cloud Console to manage Oracle Operator Access Control, specify Compute Cloud@Customer as the Resource Type.

GPU Expansion

To enable GPU-accelerated workloads in the local data center, a Compute Cloud@Customer installation can be expanded with server nodes that have GPUs installed.

GPU nodes are delivered in an expansion rack containing power distribution units (PDUs) and networking components to integrate the additional physical resources with the base rack. A GPU expansion rack contains at least 1 and a maximum of 6 factory-installed GPU nodes. More nodes can be installed after initial deployment. Up to two expansion racks can be connected to a base rack, for a maximum of 12 GPU nodes.

A GPU node is a 3 RU server with Intel Xeon Platinum 8480+ architecture, high-speed Ethernet connectivity, and four NVIDIA L40S GPUs with 48GB GDDR6 memory and 1466 peak FP8 TFLOPS. After these nodes have been fully provisioned, their use is seamless: when launching a new compute instance, users select a dedicated compute shape to allocate one or more GPUs to the instance. For descriptions of available shapes, see Compute Shapes on page 301.

For detailed component specifications, refer to the manufacturer website.

- Intel Xeon Platinum 8480+ Processor specifications
- NVIDIA L40S GPU data sheet

Oracle Compute Cloud@Customer with GPU expansion provides a scalable platform to build AI and graphics intensive applications at the edge. It's built to power the next generation of data center workloads, including:

- Generative AI inference: real time inferencing for multimodel generative AI pipelines (text, image, audio, video)
- LLM training and fine-tuning: accelerated performance for fine-tuning medium LLMs and training small LLMs with NVIDIA's transformer engine and FP8 support
- Graphics-intensive and VDI applications: 3D graphics and rendering workflows with NVIDIA’s RTX and ray tracing capabilities
- Digital twins using NVIDIA Omniverse: develop and operate complex 3D industrial digitization workflows
- Media streaming: increased encode/decode density and AV1 support for 4K video streaming
- HPC: scientific data analysis and simulation workloads with FP32 support

Installation Requirements

Site Preparation

If you have decided to expand your Compute Cloud@Customer environment with GPU nodes, carefully plan ahead for the installation of the additional hardware. The GPU expansion rack has the same external dimensions as the base rack, and contains the same type of hardware. Therefore, the base rack site requirements also apply for the expansion rack. They are described in detail in the installation section Preparing the Installation Site on page 41.

Rack Cabling

The cable connections between the base rack and the GPU expansion rack must not exceed 25 meters. Allocate a space for the expansion rack near the base rack, ensuring that the inter-rack cabling is within the specified maximum length when routed through the floor or ceiling. The required cable length must be specified with the order.

High-Performance Storage

The GPU compute shapes are optimized for high speed and low latency. They use high-performance storage exclusively, meaning the system's ZFS Storage Appliance must provide a high performance storage pool consisting of one or more *performance* disk trays. In case no performance tray is present in the existing installation, one is added to the GPU expansion order. If the base rack has no rack units available to add the performance tray, it will be installed in a storage expansion rack. The high performance storage pool must be configured before the GPU expansion rack is activated.

Installation Process

Physical Installation

All installation tasks are performed by Oracle. When the GPU expansion rack is in its allocated space, it must be connected to the base rack. The expansion rack leaf switches are cross-connected to the base rack spine switches to extend the data network into the expansion rack. Similarly, the expansion rack components are added to the internal management network through a cable connection between the management switches of both racks. The ports required for this setup have been reserved on all connected switches. The GPU nodes are internally connected to the expansion rack switches at the factory.

Rack Activation

When the physical connections are in place, the expansion rack is activated by running a script from one of the management nodes. The script powers on the switches and enables the required ports so the new hardware components can be detected and registered. When the script is finished, the data and management networks are operational across the interconnected racks. The system proceeds with the installation and

configuration of the operating system and additional software on the new nodes, preparing them for provisioning. When the GPU nodes have been provisioned, they are fully integrated and ready to use.

GPU nodes are added to the existing fault domains alongside other compute nodes, but server families operate separately from each other and migrations between them are not supported. The fault domains might become unbalanced because, unlike standard compute nodes, GPU nodes can be added one by one.

Compute Expansion Rack

Compute Cloud@Customer with compute expansion provides a scalable platform to increase the compute capacity of the system beyond the physical limitations of a base rack.

A compute expansion rack contains a minimum of 3 factory-installed compute nodes. More nodes can be installed after initial deployment, but must be added in sets of 3. Two expansion racks can be connected to a base rack, for a maximum combined total of 24 expansion compute nodes. An expansion rack contains power distribution units (PDUs) and networking components to integrate the additional physical resources with the base rack. This rack does not include storage hardware.

Installation Requirements

Site Preparation

If you have decided to expand your Compute Cloud@Customer environment with a compute expansion rack, carefully plan ahead for the installation of the additional hardware. The compute expansion rack has the same external dimensions as the base rack, and contains the same type of hardware. Therefore, the base rack site requirements also apply for the expansion rack. They are described in detail in the installation section Preparing the Installation Site on page 41.

Rack Cabling

The cable connections between the base rack and the compute expansion rack must not exceed 25 meters. Allocate a space for the expansion rack near the base rack, ensuring that the inter-rack cabling is within the specified maximum length when routed through the floor or ceiling. The required cable length must be specified with the order. The default length of cables included in the shipment is 10 meters.

Installation Process

Physical Installation

All installation tasks are performed by Oracle. When the compute expansion rack is in its allocated space, it must be connected to the base rack. The expansion rack leaf switches are cross-connected to the base rack spine switches to extend the data network into the expansion rack. Similarly, the expansion rack components are added to the internal management network through a cable connection between the management switches of both racks. The ports required for this setup have been reserved on all connected switches. The compute nodes are internally connected to the expansion rack switches at the factory.

Rack Activation

When the physical connections are in place, the expansion rack is activated by running a script from one of the management nodes. The script powers on the switches and enables the required ports so the new hardware components can be detected and registered. When the script is finished, the data and management networks are operational across the interconnected racks. The system proceeds with the installation and configuration of the operating system and additional software on the new nodes, preparing them for provisioning. When the compute nodes have been provisioned, they are fully integrated and ready to use. The system treats the compute nodes in the expansion rack the same way as all other compute nodes.

Emergency Information for Compute Cloud@Customer Installations

Review this Compute Cloud@Customer emergency information to minimize the risk of injury and damage in the event of an emergency.

Cautions and Warnings

The following cautions and warnings apply to Compute Cloud@Customer:

> **Warning:**
>
> Do not touch the parts of this product that use high-voltage power. Touching them might result in serious personal injury.

> **Warning:**
>
> Do not power off the system unless there is an emergency.

> **Caution:**
>
> Keep the front and rear cabinet doors closed. Failure to do so might cause system failure or result in damage to hardware components.

> **Caution:**
>
> Keep the top, front, and back of cabinets clear to allow proper airflow and prevent overheating of components.

> **Caution:**
>
> Use only the supplied hardware.

Emergency Power-off Considerations

If there is an emergency, then power to Compute Cloud@Customer should be halted immediately. The following emergencies might require powering off the system:

- Natural disasters such as an earthquake, flood, hurricane, tornado or cyclone
- Abnormal noise, smell or smoke coming from the system
- Threat to human safety

> **Note:**
>
> If you use the compute instance high availability feature, it's advised to disable it before a system shutdown and enable it again after restart, when the compute nodes are back in normal operation. This requires access to the Service CLI, which is available on Private Cloud Appliance, but restricted to the cloud operations team on Compute Cloud@Customer.
>
> ```
> PCA-ADMIN> disableVmHighAvailability
> PCA-ADMIN> enableVmHighAvailability
> ```

Emergency Power-off Procedure

To perform an emergency power-off procedure, turn off power at the circuit breaker or pull the emergency power-off switch in the computer room.

Emergency Power-off Switch

Emergency power-off (EPO) switches are required in a computer room when computer equipment contains batteries capable of supplying more than 750 volt-amperes for more than five minutes. Systems that have these batteries include internal EPO hardware for connection to a site EPO switch or relay. Use of the EPO switch will remove power from the Compute Cloud@Customer installation.

Supported Regions for Compute Cloud@Customer

The Compute Cloud@Customer service is supported in many regions across different realms.

All OCI regions in the commercial OC1 realm enable you to register a Compute Cloud@Customer installation. See About Regions and Availability Domains for an overview. The service is also available in Government Cloud realms:

- Oracle US Government Cloud
- Oracle US Defense Cloud
- Oracle EU Sovereign Cloud
- Oracle UK Sovereign Cloud

Each installation must correspond with an infrastructure resource in the home OCI tenancy. Ensure that the realm and regions associated with your tenancy align with the intended location of your Compute Cloud@Customer installation. The service is not available in Oracle Alloy and Dedicated Regions by default. Contact your Oracle representative to discuss your specific access and deployment requirements.

Guest Operating System Matrix

On Compute Cloud@Customer, you can use choose to create instances from various images. Each image provides a particular instance guest OS. This section lists the types of guest OSs you can use.

Images of Oracle Linux and Oracle Solaris are provided as part of the Compute Cloud@Customer software (platform images), and new image versions are added over time through upgrades.

You can bring your own images (BYOI). Several guest OSs are part of Oracle testing and are known to work in Compute Cloud@Customer instances. The following table provides an overview.

Also see Images for Compute Cloud@Customer Instances on page 283.

Guest Operating System	Oracle-Provided Image	Custom Image	Oracle-Tested
Oracle Linux 9.x	Y	Y	Y
Oracle Linux 8.x	Y	Y	Y
Oracle Linux 7.x	Y	Y	Y
Oracle Solaris 11.x	Y	Y	Y
Red Hat Enterprise Linux 9.x		Y	Y
Red Hat Enterprise Linux 8.x		Y	Y
Red Hat Enterprise Linux 7.x		Y	Y
CentOS Linux 8.x		Y	
CentOS Linux 7.x		Y	
SUSE Linux Enterprise Server 15 (latest)		Y	
SUSE Linux Enterprise Server 12 SP4		Y	
Ubuntu 20.04 and later		Y	
Ubuntu 18.04 and later		Y	

Guest Operating System	Oracle-Provided Image	Custom Image	Oracle-Tested
AlmaLinux OS 9.2		Y	Y
Kali Linux		Y	Y
Microsoft Windows Server 2025		Y	Y
Microsoft Windows Server 2022		Y	Y
Microsoft Windows Server 2019		Y	Y
Microsoft Windows Server 2016		Y	Y

Service Limits for Compute Cloud@Customer

Learn about the default limits on Oracle Cloud Infrastructure resources for the Compute Cloud@Customer service.

Resource	Scope	Oracle Universal Credits	Pay-as-You-Go or Promo
Compute Cloud@Customer Infrastructures	Tenancy	24	Not available
Compute Cloud@Customer Upgrade Schedules	Tenancy	48	Not available

Limits on Resources Provided by Compute Cloud@Customer

Review the limits on the resources provided to customers by a Compute Cloud@Customer infrastructure and request new limits.

Review the following sections:

- Resource Limits per Infrastructure on page 34
- Limits on Concurrent Operations on page 37
- Identity and Access Management (IAM) Limits on page 38

Resource Limits per Infrastructure

The numbers provided here apply to any Compute Cloud@Customer infrastructure, regardless of its hardware configuration.

For some services, you can request a limit increase for a particular resource. See Creating a Support Request. To access support, sign in to the Oracle Cloud Console as described in Sign In to the OCI Console.

Service	Resource Type	Limit
Networking Service	VCNs	80 with up to 16 SR-IOV VCNs
Networking Service	Subnets	40 per VCN 320 in total

Service	Resource Type	Limit
Networking Service	Dynamic routing gateways (DRG)	32 with up to 16 SR-IOV DRGs
Networking Service	DRG attachments	10 per DRG 80 in total
Networking Service	Internet gateways	1 per VCN
Networking Service	Local peering gateways	5 per VCN 150 in total
Networking Service	NAT gateways	1 per VCN
Networking Service	Service gateways	1 per VCN
Networking Service	Storage gateways	2 per VCN 80 in total, standard and high-performance combined
Networking Service	Reserved public IPs	The range specified on the Compute Cloud@Customer Network Spreadsheet
Networking Service	Ephemeral public IPs	2 per compute instance
Networking Service	DHCP options	30 per VCN 500 in total
Networking Service	Route tables	20 per VCN 500 in total
Networking Service	Route rules	50 per route table 10000 in total
Networking Service	Network security groups	100 per VCN 5 per VNIC 600 in total
Networking Service	VNICs in network security group	As many VNICs as are in the VCN A VNIC can belong to max. 5 network security groups
Networking Service	Security rules	50 per network security group 12000 in total

Service	Resource Type	Limit
Networking Service	Security lists	20 per VCN 5 per subnet 600 in total
Networking Service	Ingress rules	30 per security list 12000 in total
Networking Service	Egress rules	30 per security list 12000 in total
Networking Service	DNS zones	1000 (in addition to any internal zones)
Networking Service	DNS records	25000 per zone 8000000 in total
Networking Service	Flex networks	128 flex networks 32 DRGs 128 logical ports 32 networks per physical port
Compute Service	Custom images	100
Block Storage Service	Aggregated size of block volumes	100 TB (with default storage capacity)
Block Storage Service	Block volume backups	100000
File Storage Service	File systems	800
File Storage Service	Mount targets	PCA_POOL 80 PCA_POOL_HIGH 80
File Storage Service	File system size	3.3 PB
Object Storage Service	Buckets	10000
(Network) Load Balancing Service	Load balancers (Network LB and LBaaS combined)	20 in a single VCN up to 144 in total, depending on system compute capacity
(Network) Load Balancing Service	IP address	1 per load balancer
(Network) Load Balancing Service	Network security groups	5 per load balancer
(Network) Load Balancing Service	Listeners	16 per load balancer
(Network) Load Balancing Service	Backend sets	16 per load balancer

Service	Resource Type	Limit
(Network) Load Balancing Service	Backend servers	512 per load balancer and per backend set
Load Balancing Service	Request routing rules	20 per rule set 50 per load balancer
Kubernetes Engine (OKE)	Clusters	20 per tenancy
Kubernetes Engine (OKE)	Worker nodes	128 per cluster (across all pools)
Kubernetes Engine (OKE)	Nodes per node pool/group	128
Kubernetes Engine (OKE)	Pods	110 per node (Kubernetes default)

Limits on Concurrent Operations

Find out how many concurrent operations of a specific type Compute Cloud@Customer can manage at any particular time. These limits assume that no other operations of any kind are running at the same time. When a limit is exceeded, an error with code 409 or 429 is displayed.

Resource Type	Operation	Concurrency Limit
compute instance	back up or restore an instance	10
compute instance	launch/terminate instance	15
compute instance	reset/stop/start instance	15
compute instance	update fault domain (live migration)	10
compute image	create image from instance	10
compute image	import image	10
block volume	create/delete volume	10
block volume	attach/detach boot volume	15
block volume	attach/detach data volume	15
block volume	resize volume	15
file system	create/delete file system	10
mount target	create/delete mount target	10
VCN	create/delete VCN	10
VCN gateway	create/delete gateway (all types)	10
subnet	create/delete subnet	10
route table	create/delete route table	10

Resource Type	Operation	Concurrency Limit
security list	create/delete security list	10
network security group	create/delete network security group	10
VNIC	attach/detach VNIC	15
public IP	create/delete public IP	10
private IP	create/delete private IP	10
all networking resources	update network resource	10
Kubernetes cluster	create/update/delete cluster	10
Kubernetes node pool	create/update/delete node pool	5
Kubernetes node	create/update/delete node	15

Note:

In addition, there is a system limit on the number of concurrent user sessions:

- Console: 10 tenancy users

An authentication error is displayed when the limit is reached. An inactive user session times out after 1 hour.

Identity and Access Management (IAM) Limits

On Compute Cloud@Customer, IAM services are provided by Oracle Cloud Infrastructure (OCI).

IAM resources are managed on the OCI side and synchronized to Compute Cloud@Customer. This is why the OCI tenancy limits apply.

For information about limits in OCI, see Limits by Service. The following sections apply to IAM:

- Service Limits for IAM with Identity Domains
- Service Limits for IAM without Identity Domains

To check current status for your tenancy, follow the instructions in Viewing a Tenancy's Limits and Usage. Filter by service *Identity*.

Viewing Resource Limits

Compute Cloud@Customer has a set of service limits configured. A service limit is the quota or allowance set on a resource.

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 Console

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.
3. If you select a different service from the drop-down menu, also select the Refresh button to refresh the list.

Using the CLI

Use the oci limits service list command and required parameters to return a list of supported services.

```
oci limits service list --compartment-id compartment_OCID [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Examples:

- **List Limits Services**

  List services that support changing resource limits.

  Syntax:

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

- **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 isn't available, a message is shown that resource availability isn't supported for that particular service and limit.

Using the API

Use the ListServices operation to return a list of supported services.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Quotas for Compute Cloud@Customer

Tenancy administrators can set quota policies to enforce restrictions on the number of resources used.

For information about how Oracle Cloud Infrastructure handles quotas, see Compartment Quotas.

Family name: `ccatc`

Resource Name	Scope	Description
ccc-infrastructure-count	Tenancy	Number of Compute Cloud@Customer Infrastructures
ccc-upgrade-schedule-count	Tenancy	Number of Compute Cloud@Customer Upgrade Schedules

Example Quota Statements for Compute Cloud@Customer

```
Set ccatc quota ccc-infrastructure-count to <value> in
 compartment <compartment_name>
```

Where to Find Billing and Cost Information

Find information about billing and cost management for Compute Cloud@Customer.

Compute Cloud@Customer uses the standard Oracle Cloud Infrastructure billing structure. For more information about billing and cost management, see Billing and Cost Management.

Viewing a PDF of this Documentation

You can view and download a copy of the Compute Cloud@Customer documentation set provided here.

Use the following link to display the Compute Cloud@Customer in PDF format. You can save it to your local computer for convenient offline use.

Compute Cloud@Customer Documentation

> **Note:**
>
> You can view the PDF file offline. However, it contains links to other Oracle Cloud Infrastructure documentation, which require an internet connection to view.

Installing and Configuring Compute Cloud@Customer

Before you can use Compute Cloud@Customer, you must prepare your site for the installation of the rack, prepare your tenancy, and initialize the connection of the Compute Cloud@Customer infrastructure to Oracle Cloud Infrastructure.

The following table describes the steps you perform to install Compute Cloud@Customer. Oracle performs some steps. Steps 3 and 4 can be performed at the same time, or in reverse order, depending on when the hardware arrives at your site. You can also watch this Video: Introduction to Installing and Configuring Oracle Compute Cloud@Customer.

No.	Task	Links
1.	Prepare your data center for the Compute Cloud@Customer rack installation.	Preparing the Installation Site on page 41 and Site Checklists for Compute Cloud@Customer on page 52
2.	Oracle validates site requirements in your data center.	
3.	Prepare your OCI tenancy by setting up these items: • Federated identity provider • Users and groups • Compartments • Policies • Virtual cloud network (VCN) with a subnet	Preparing Your Tenancy on page 67
4.	The Compute Cloud@Customer rack is delivered to your site. Oracle installs the rack in your data center and works with you to connect it to your network. Oracle initializes Compute Cloud@Customer and configures it with site-specific details such as network parameters.	
5.	Create an infrastructure in the compartment for the physical infrastructure in the data center.	Creating a Compute Cloud@Customer Infrastructure in OCI on page 72
6.	Open a Support Request to engage with Oracle. You exchange information with Oracle to enable the service within OCI.	See Creating a Support Request. To access support, sign in to the Oracle Cloud Console as described in Sign In to the OCI Console.

You can now use Compute Cloud@Customer to provide compute, storage and other resources. See Signing in to the Compute Cloud@Customer Console on page 102.

Preparing the Installation Site

Before the arrival of Compute Cloud@Customer, prepare the installation site and ensure it meets the requirements outlined in this section.

To track progress as you prepare your site, use the Site Checklists for Compute Cloud@Customer on page 52.

Follow these steps prior to unpacking and situating your Compute Cloud@Customer:

1. Thoroughly clean and vacuum the area in preparation for the installation.
2. Note problems or peculiarities at the site that require special equipment.
3. Verify that the installation site flooring has a strength rating to withstand the combined weight of the Compute Cloud@Customer and any other installed equipment.

   For more information, see Flooring Requirements on page 44.
4. Install all necessary electrical equipment and ensure that sufficient power is provided.

   For more information, see Electrical Power Requirements on page 44.
5. Ensure that the installation site provides adequate air conditioning.

   For more information, see Environmental Requirements on page 47 (Ventilation and Cooling).
6. Operate the air conditioning system for 48 hours to bring the room temperature to the appropriate level.

Review Safety Guidelines

Review the following safety precautions to ensure that the site is safe and ready for delivery. Failure to observe these precautions can result in personal injury, equipment damage, or malfunction.

- Do not block ventilation openings.
- Do not install the system in a location that is exposed to direct sunlight or near a device that may become hot.
- Do not install the system in a location that is exposed to excessive dust, corrosive gases, or air with high salt concentrations.
- Do not install the system in a location that is exposed to frequent vibrations. Install the system on a flat, level surface.
- Use a power outlet that uses proper grounding. When using shared grounding, the grounding resistance must not be greater than 10 ohms. Ensure that your facility administrator or a qualified electrical engineer verifies the grounding method for the building, and performs the grounding work.
- Do not place cables under the equipment or stretch the cables too tightly.
- Do not disconnect power cords from the equipment while its power is on.
- Do not place anything on top of the system or perform any work directly above it.
- Do not let the room temperature rise sharply, especially in winter. Sudden temperature changes can cause condensation to form inside the system. Allow for a sufficient warm-up period prior to server operation.
- Do not install the system near a photocopy machine, air conditioner, welding machine, or any other equipment that generates loud, electronic noises.
- Avoid static electricity at the installation location. Static electricity transferred to the system can cause malfunctions. Static electricity is often generated on carpets.

Site Requirements

Ensure the appliance installation site meets the requirements described in the following sections:

Space Requirements on page 42

- Product dimensions and weight
- Area surrounding the rack
- Receiving area and access route
- Maintenance access

Flooring Requirements on page 44

Electrical Power Requirements on page 44

- Power distribution units
- Facility power
- Circuit breakers and grounding

Environmental Requirements on page 47

- Temperature and humidity
- Ventilation and cooling

Space Requirements

Ensure that your site meets the space requirements for each Oracle Compute Cloud@Customer rack.

Use the information in this section along with the Logistics Checklist on page 56, Data Center Room Checklist on page 52, and Access Route Checklist on page 54 to prepare your site.

Dimensions and Weight Specifications

Specification	Metric	Standard
Height	2000 mm	78.74 "

Specification	Metric	Standard
Width	600 mm	23.62 "
Depth (front door handle to rear door handle)	1197 mm	47.12 "
Depth (doors removed)	1112 mm	43.78 "
Weight (base rack, fully populated)	1000 kg	2204 lbs

Space Requirements

Minimum Ceiling Height

The minimum ceiling height for the cabinet is 2914 mm (114.72 inches), measured from the true floor or raised floor, whichever is higher. This includes an additional 914 mm (36 inches) of space required above the rack height for maintenance access.

Surrounding Space

The space surrounding the cabinet must not restrict the movement of cool air between the air conditioner and the front of the systems within the cabinet, or the movement of hot air coming out of the rear of the cabinet.

Receiving Specifications

Before the Compute Cloud@Customer rack arrives, ensure that the receiving area is large enough for the package. The following are the package dimensions and weights for a rack:

- Shipping height: 2159 mm (85 inches)
- Shipping width: 1118 mm (44 inches)
- Shipping depth: 1543 mm (60.75 inches)
- Shipping weight (base rack, fully populated): 1118 kg (2465 lbs)

If your loading dock meets the height and ramp requirements for a standard freight carrier truck, then you can use a pallet jack to unload the rack. If the loading dock does not meet the requirements, then you must provide a standard forklift or other means to unload the rack. You can also request that the rack be shipped in a truck with a lift gate.

When the rack arrives, leave the rack in its shipping packaging until it arrives at its installation site. Use a conditioned space to remove the packaging material to reduce particles before entering the data center. The entire access route to the installation site should be free of raised-pattern flooring that can cause vibration.

Allow enough space for unpacking the system from its shipping cartons. Ensure that there is enough clearance and clear pathways for moving the rack from the unpacking location to the installation location. The following table lists the access route requirements for the rack.

Access Route Item	With Shipping Pallet	Without Shipping Pallet
Minimum door height	2159 mm (85 inches)	2000 mm (78.74 inches)
Minimum door width	1118 mm (44 inches)	600 mm (23.62 inches)
Minimum elevator depth	1543 mm (60.75 inches)	1058.2 mm (41.66 inches)

Access Route Item	With Shipping Pallet	Without Shipping Pallet
Maximum incline	6 degrees	6 degrees
Maximum elevator, pallet jack, and floor loading capacity	1145 kg (2520 lbs)	1145 kg (2520 lbs)

Maintenance Access Requirements

The maintenance area must be large enough for the rack, and have the required access space. For example, the required space to remove the side panels is 675.64 mm (26.6 inches).

The following table lists the maintenance access requirements for the Compute Cloud@Customer rack.

Location	Maintenance Access Requirement
Rear maintenance	914 mm (36 inches)
Front maintenance	1232 mm (48.5 inches)
Top maintenance	914 mm (36 inches)

Flooring Requirements

Oracle allows the installation of the Compute Cloud@Customer rack on solid flooring and on raised flooring (recommended). Review important flooring requirements before the system arrives at your site.

Use the information in this section along with the Data Center Room Checklist on page 52 to prepare your site.

The site floor and the optional raised flooring must be able to support the total weight of the system as specified in Space Requirements on page 42. The following table lists the floor load requirements.

Description	Requirement
Maximum allowable weight of installed rack equipment	952.54 kg (2100 lbs)
Maximum allowable weight of installed power distribution units	52.16 kg (115 lbs)
Maximum dynamic load (maximum allowable weight of installed equipment including power distribution units)	1004.71 kg (2215 lbs)

Electrical Power Requirements

Follow these important electrical power guidelines for Compute Cloud@Customer.

Use the information in this section along with the Facility Power Checklist on page 55 to prepare your site.

The Compute Cloud@Customer can operate effectively over a wide range of voltages and frequencies. However, it must have a reliable power source. Damage might occur if the ranges are exceeded. Electrical disturbances such as the following might damage the system:

- Fluctuations caused by brownouts
- Wide and rapid variations in input voltage levels or in input power frequency
- Electrical storms
- Faults in the distribution system, such as defective wiring

To protect your system from such disturbances, you should have a dedicated power distribution system, power-conditioning equipment, as well as lightning arresters or power cables to protect from electrical storms.

Power Distribution Units (PDUs)

Each rack has two preinstalled power distribution units. The PDUs accept different power sources. You must specify the type of PDU that is correct for your data center. The following tables list PDU requirements.

Global PDU Specifications

Item	15kVA - type 6440A	15kVA - type 6441A	15kVA - type 6442A	22kVA - type 7100873	22kVA - type 7100874	24kVA - type 6444A	24kVA - type 6445A
Region	North and South America, Japan and Taiwan	EMEA and APAC (except Japan and Taiwan)	North and South America, Japan and Taiwan	North and South America, Japan and Taiwan	EMEA and APAC (except Japan and Taiwan)	North and South America, Japan and Taiwan	EMEA and APAC (except Japan and Taiwan)
Number of inputs	1	1	3	3	3	2	2
Source voltage	200-220V	220/380 - 240/415VAC	200-240V	200-240V	200-240V	200-220V	220/380 - 240/415VAC
Frequency	50/60 Hz	50/60 Hz	50/60 Hz	50/60 Hz	50/60 Hz	50/60 Hz	50/60 Hz
Max. line current	40A	21A	24A	36.8A	32A	34.6A	16A
Phase current	23A	21A	N/A	N/A	32A	20A	16A
Recommended circuit breaker	50A	25A	30A	50A	32A	50A	25A
Outlets	42 x C13 6 x C19	42 x C13 6 x C19	42 x C13 6 x C19	42 x C13 6 x C19	42 x C13 6 x C19	42 x C13 6 x C19	42 x C13 6 x C19
Outlet groups	6	6	6	6	6	6	6
Max. group current	20A	20A	20A	20A	20A	20A	18A
Output voltage	200-240V	220-240V	200-240V	200-240V	200-240V	220-240V	200-240V
Data center receptacle	3ph 3W + ground 60A IEC60309	3ph 4W + ground 32A IEC60309 5P 6h 400V	1ph 2W + ground 30A NEMA L6-30	1ph 2W + ground 50A Hubbell CS8265C	1ph 2W + ground 32A IEC60309 3P 6H 230V	3ph 3W + ground 60A IEC60309	3ph 4W + ground 32A IEC60309 5P 6h 400V

PDU Specifications for Korea

Item	15 kVA - type 7600785	22KVA - type 7600788	24KVA - type 7600787
Region	South Korea only	South Korea only	South Korea only
Nominal Power Rating	15KVA	22KVA	24KVA
Number of inputs	1	3	2
Source voltage	220/380 - 240/415VAC 3Ø	220-240V	220/380 - 240/415VAC 3Ø
Frequency	50/60 Hz	50/60 Hz	50/60 Hz
Max. line current	21A	32A	18A
Phase current	21A	32A	18A
Recommended circuit breaker	25A	32A	25A
Outlets	42 x C13 6 x C19	42 x C13 6 x C19	42 x C13 6 x C19
Outlet groups	6	6	6
Max. group current	20A	16A	20A
Output voltage	220-240V	220-240V	220-240V
Data center receptacle	3ph 4W + ground 32A IEC60309 5P 6h 400V	1ph 2W + ground 32A IEC60309 3P 6H 230V	3ph 4W + ground 32A IEC60309 5P 6h 400V

Facility Power Requirements

Electrical work and installations must comply with applicable local, state, or national electrical codes. Contact your facilities manager or qualified electrician to determine what type of power is supplied to the building.

To prevent catastrophic failures, design the input power sources to ensure adequate power is provided to the power distribution units (PDUs). When planning for power distribution requirements, balance the power load between available AC supply branch circuits. In the United States and Canada, ensure that the overall system AC input current load doesn't exceed 80 percent of the branch circuit AC current rating.

Provide a stable power source, such as an uninterruptable power supply (UPS) to reduce the possibility of component failures. If computer equipment is subjected to repeated power interruptions and fluctuations, then it's susceptible to a higher rate of component failure.

PDU power cords are 4 meters (13.12 feet) long, and 1 to 1.5 meters (3.3 to 4.9 feet) of the cord will be routed within the rack cabinet. The installation site AC power receptacle must be within 2 meters (6.6 feet) of the rack.

Circuit Breaker Requirements

Use dedicated AC breaker panels for all power circuits that supply power to the rack PDUs. Electrical work and installations must comply with applicable local, state, or national electrical codes. Rack PDUs require electrical circuits to be grounded to the Earth.

> **Note:**
>
> Circuit breakers are supplied by the customer. One circuit breaker is required for each power cord.

Grounding Guidelines

Compute Cloud@Customer racks are shipped with grounding-type power cords (three-wire). Always connect the cords to grounded power outlets. Because different grounding methods are used, depending on location, check the grounding type, and refer to documentation, such as IEC documents, for the correct grounding method. Ensure that the facility administrator or qualified electrical engineer verifies the grounding method for the building, and performs the grounding work.

Environmental Requirements

Ensure that the Compute Cloud@Customer installation site meets these requirements for best environmental operating conditions.

Use the information in this section along with the Data Center Environment Checklist on page 53 to prepare your site.

Temperature and Humidity Requirements

Airflow through the Compute Cloud@Customer rack is from front to back. For information, see Ventilation and Cooling Requirements on page 48.

Studies have shown that temperature increases of 10 degrees Celsius (18 degrees Fahrenheit) above 20 degrees Celsius (68 degrees Fahrenheit) reduce long-term electronics reliability by 50 percent. Excessive internal temperatures might result in full or partial shutdown of the rack.

Condition	Operating Requirement	Non-operating Requirement	Optimum
Temperature	5 ° to 32 ° Celsius (41 ° to 89.6 ° Fahrenheit)	-40 ° to 68 ° Celsius (-40 ° to 154 ° Fahrenheit)	For optimal rack cooling, data center temperatures from 21 ° to 23 ° Celsius (69.8 ° to 73.4 ° Fahrenheit)
Relative humidity	10 to 90 percent relative humidity, non-condensing	Up to 93 percent relative humidity	For optimal data center rack cooling, 45 to 50 percent non-condensing
Altitude	3,000 meters (9,840 feet) maximum	12,000 meters (39,370 feet)	Ambient temperature is reduced by 1 ° Celsius per 300 meters above 900 meters altitude above sea level

Set conditions to the optimal temperature and humidity ranges to minimize the chance of downtime because of component failure. Operating a rack for extended periods at or near the operating range limits, or installing it in an environment when it remains at or near non-operating range limits could significantly increase hardware component failure.

The ambient temperature range of 21 ° to 23 ° Celsius (69.8 ° to 73.4 ° Fahrenheit) is optimal for server reliability and operator comfort. Most computer equipment can operate in a wide temperature range, but near 22 ° Celsius (71.6 ° Fahrenheit) is preferable because it is easier to maintain safe humidity levels. Operating in this temperature range provides a safety buffer in the event that the air conditioning system goes down for a period of time.

The ambient relative humidity range of 45 to 50 percent is suitable for safe data processing operations. Most computer equipment can operate in a wide range (20 to 80 percent), but the range of 45 to 50 percent is recommended for the following reasons:

- Optimal range helps protect computer systems from corrosion problems associated with high humidity levels.

- Optimal range provides the greatest operating time buffer in the event of air conditioner control failure.
- This range helps to avoid failures or temporary malfunctions caused by intermittent interference from static discharges that may occur when relative humidity is too low.

> **Note:**
>
> Electrostatic discharge (ESD) is easily generated, and hard to dissipate in areas of low relative humidity, such as below 35 percent. ESD becomes critical when humidity drops below 30 percent. It is not difficult to maintain humidity in a data center because of the high-efficiency vapor barrier and low rate of air changes normally present.

Ventilation and Cooling Requirements

Always provide adequate space in front of and behind the rack to allow for proper ventilation. Do not obstruct the front or rear of the rack with equipment or objects that might prevent air from flowing through the rack. Rack-mountable servers and equipment typically draw cool air in through the front of the rack and let warm air out the rear of the rack. There is no airflow requirement for the left and right sides because of front-to-back cooling.

If the rack is not completely filled with components, then cover the empty sections will filler panels. Gaps between components can adversely affect airflow and cooling within the rack.

Relative humidity is the percentage of the total water vapor that can exist in the air without condensing, and is inversely proportional to air temperature. Humidity goes down when the temperature rises, and goes up when the temperature drops. For example, air with a relative humidity of 45 percent at a temperature of 24 ° Celsius (75.2 ° Fahrenheit) has a relative humidity of 65 percent at a temperature of 18 ° Celsius (64.4 ° Fahrenheit). As the temperature drops, the relative humidity rises to more than 65 percent, and water droplets are formed.

Air conditioning facilities usually don't precisely monitor or control temperature and humidity throughout an entire computer room. Generally, monitoring is done at individual points corresponding to multiple exhaust vents in the main unit, and other units in the room. Special consideration should be paid to humidity when using underfloor ventilation. When underfloor ventilation is used, monitoring is done at each point close to an exhaust vent. Distribution of the temperature and humidity across the entire room is uneven.

The rack has been designed to function while installed in a natural convection air flow. The following requirements must be followed to meet the environmental specification:

- Ensure that there is adequate airflow through the system.
- Ensure that the system has front-to-back cooling. The air intake is at the front of the system, and the air outlet is at the rear of the system.
- Allow a minimum clearance of 1219.2 mm (48 inches) at the front of the system, and 914 mm (36 inches) at the rear of the system for ventilation.

Customer Site Network Requirements

Review the topics in this section to prepare your network environment before the arrival of the Compute Cloud@Customer rack.

> **Note:**
>
> In addition to the network requirements listed here, an Oracle representative works with you to ensure that your data center network configuration is prepared to accommodate Compute Cloud@Customer before the rack arrives.

Use the network information in this section with the Initial System Installation Checklist on page 58 to prepare your network.

Oracle Compute Cloud@Customer network architecture relies on physical high speed Ethernet connectivity.

The networking infrastructure in Compute Cloud@Customer is integral and must not be altered. The networking doesn't integrate into any data center management or provisioning frameworks such as Cisco ACI, Network Director, or the like.

> **Caution:**
>
> No changes to the networking switches in the Compute Cloud@Customer rack are supported unless directed by an Oracle Support document (KM note) or Oracle personnel.

Network Overview

> **Note:**
>
> Additional network overview concepts are described in Compute Cloud@Customer Network Infrastructure on page 8.

The Compute Cloud@Customer infrastructure uses these networks.

- **Data Network:** The infrastructure's data connectivity is built on redundant 100-Gbit switches in two-layer design similar to a leaf-spine topology. An infrastructure contains two leaf and two spine switches. The leaf switches interconnect the rack hardware components, while the spine switches form the backbone of the network and provide a path for external traffic.
- **Uplinks:** Uplinks are the connections between the infrastructure and the customer data center. For external connectivity, 5 ports are reserved on each spine switch. Four ports are available to establish the uplinks between the infrastructure and the data center network; one port is used to segregate the administration network from the data traffic.
- **Administration Network:** This required network is configured as a physically separated network from the data network. This isolates configuration and management traffic from the operational activity on the data network by providing dedicated secured network paths for administration operations. This network includes all component management interfaces.

  Setting up a segregated administration network involves the following actions:

  - Extra Ethernet connections from the next-level data center network devices to port 5 on each of the spine switches in the rack.
  - Inside the administration network, the spine switches must each have one IP address and a virtual IP shared between the two.
  - A default gateway is required to route traffic, and NTP and DNS services must be enabled.
  - The management nodes must be assigned host names and IP addresses in the administration network, one each individually, and one shared between all three.

  A separate administration network can be used with both static and dynamic routing. The use of a VLAN is supported, but when combined with static routing the VLAN ID must be different from the one configured for the data network.
- **Reserved Network Resources:** The infrastructure requires many IP addresses and several VLANs for internal operation.

Network Configuration Requirements

On each spine switch, ports 1-4 can be used for uplinks to the data center network. For speeds of 10-Gbps or 25-Gbps, the spine switch port must be split using a 4-way splitter or breakout cable. For higher speeds of 40-Gbps or 100-Gbps each switch port uses a single direct cable connection.

The uplinks are configured during system initialization, based on information you provide. Unused spine switch uplink ports, including unused breakout ports, are disabled for security reasons.

It is critical that **both** spine switches have the same connections to the next-level data center switches. This configuration provides redundancy and load splitting at the level of the spine switches, the ports and the data center switches. This outbound cabling depends on the network topology you deploy. The cabling pattern is essential to the continuation of service during failover scenarios.

- Before installation, you must run network cables from your existing network infrastructure to the Compute Cloud@Customer rack installation site.

- Plan to connect at least 1 high-speed Ethernet port on each *spine* switch to your data center public Ethernet network.
- Configuring the required Administration network requires 2 more cable connections (one each from port 5 on the two spine switches) to a pair of next-level data center switches.
- Uplink connectivity is based on layer 3 of the OSI model.
- If you have security policies requiring the use of proxies for all internet connections,you can use an `HTTP` proxy. This is a passive/corporate proxy supported for the connection from Compute Cloud@Customer to Oracle Cloud Infrastructure. Customer `HTTPS`, challenge proxy, and traffic inspection aren't supported.
- A corporate HTTP proxy between the Compute Cloud@Customer infrastructure and Oracle Cloud Infrastructure isn't recommended with FastConnect because FastConnect is already a dedicated network. If a corporate proxy is required, then the proxy needs to have additional routing to ensure that Compute Cloud@Customer network traffic is sent over FastConnect.

Data Center DNS Configuration for Compute Cloud@Customer

Review the Compute Cloud@Customer DNS information to select the configuration that best suits your network environment.

To integrate the data of the infrastructure's dedicated DNS zone into the data center DNS configuration, two options are supported:

- Zone Delegation (Preferred) on page 50
- Manual Configuration on page 50

If you select manual configuration, it's good practice to register network host names and IP addresses for the management network, client network, and more public networks in the data center Domain Name System (DNS) before the initial configuration. In particular, all public addresses, VIP addresses, and infrastructure services endpoints must be registered in DNS before the installation.

All addresses registered in DNS must be configured for forward resolution; reverse resolution isn't supported in the services zone.

Zone Delegation (Preferred)

For zone delegation to work, the data center's recursive caches must be able to reach TCP/UDP port 53 on the virtual IP address shared by the Compute Cloud@Customer management nodes. It might be necessary to change your firewall configuration.

Configure the data center DNS server so that it operates as the parent zone of the Compute Cloud@Customer DNS zone. Thus, all DNS requests for the child zone are delegated to the Compute Cloud@Customer internal DNS server. In the data center DNS configuration, add a name server record for the child zone and an address record for the authoritative server of that zone.

In the example it is assumed that the data center DNS domain is `example.com`, that the Compute Cloud@Customer rack is named `myccc`, and that the management node cluster virtual IP address is 192.0.2.102. The Compute Cloud@Customer internal DNS server host name is `ns1`.

```
$ORIGIN example.com.
[...]
myccc       IN    NS    ns1.myccc.example.com.
ns1.myccc   IN    A     192.0.2.102
```

Manual Configuration

Manually add DNS records for all labels or host names required by Compute Cloud@Customer. Many of the DNS entries are required for maintenance performed by Oracle.

In the examples, the data center DNS domain is `example.com`, that the Compute Cloud@Customer rack is named `myccc`, and that the management node cluster virtual IP address is 192.0.2.102.

> **Note:**
>
> For object storage, you must point the DNS label to the Object Storage Public IP. This is the public IP address assigned for this purpose when setting up the data center public IP ranges during Initial Setup.

The examples assume that the data center DNS domain is `example.com`, that the infrastructure is named `myccc`, and that the management node cluster virtual IP address is 192.0.2.102 in the data network and 203.0.113.12 in the administration network.

Compute Cloud@Customer Service	Data Center DNS Record with Admin Network Enabled
Admin service ***admin***`.myccc.example.com`	`admin` `A 203.0.113.12`
Networking, Compute, Block Storage, Work Requests services ***iaas***`.myccc.example.com`	`iaas` `A 192.0.2.102`
Identity and Access Management service ***identity***`.myccc.example.com`	`identity` `A 192.0.2.102`
DNS service ***dns***`.myccc.example.com`	`dns` `A 192.0.2.102`
Object storage ***objectstorage***`.myccc.example.com` **Note**: Use the Object Storage Public IP from the Appliance Initial Setup.	`objectstorage` `A 198.51.100.33`
File storage ***filestorage***`.myccc.example.com`	`filestorage` `A 192.0.2.102`
Alert manager ***alertmanager***`.myccc.example.com`	`alertmanager` `A 203.0.113.12`
API ***api***`.myccc.example.com`	`api` `A 203.0.113.12`
OKE service ***containerengine***`.myccc.example.com`	`containerengine` `A 192.0.2.102`
Resource principal service ***rps***`.myccc.example.com`	`rps` `A 203.0.113.12`
Grafana ***grafana***`.myccc.example.com`	`grafana` `A 203.0.113.12`
Prometheus ***prometheus***`.myccc.example.com`	`prometheus` `A 203.0.113.12`

Compute Cloud@Customer Service	Data Center DNS Record with Admin Network Enabled
Prometheus-gw ***prometheus-gw***.myccc.example.com	prometheus-gw A 203.0.113.12
Service Web UI ***adminconsole***.myccc.example.com	adminconsole A 203.0.113.12
Compute Cloud@Customer Console ***console***.myccc.example.com	console A 192.0.2.102

Data Center Switch Configuration Notes

When configuring the data center switches to accept incoming Compute Cloud@Customer uplinks (default uplinks as well as any custom uplinks you define) take these notes into account.

- All uplinks, default and customer, are configured to use link aggregation (LACP). All switch ports included in an uplink configuration must belong to the same link aggregation group (LAG). The switch ports on the data center side of the uplinks must be configured accordingly.
- Compute Cloud@Customer supports layer 3 based uplink connectivity to the customer data center. Static routing and BGP4-based dynamic routing are supported in layer 3.
- Autonegotiation isn't available for uplink ports. Transfer speed must be specified on the customer switches' end.

For more information, see Uplinks on page 9 and Uplink Protocols on page 12.

Site Checklists for Compute Cloud@Customer

To determine your readiness for the installation of the Compute Cloud@Customer rack in your data center, work with your Oracle representative to review and complete the checklists before the rack arrives.

Preparation is crucial to ensure a fluent installation process. These site checklists are designed to guide you through each phase and remind you of all aspects you need to consider.

System Components Checklist

For specifications, see Space Requirements on page 42.

Item	Comment
How many racks do you plan to install?	
Is the rack installation a new system or an addition to an existing system?	
Are all machine racks adjacent to each other?	
If the connecting racks are not within the specified proximity, have you Oracle Support Services to provide and schedule the custom multirack cabling service with the installation?	

Data Center Room Checklist

For specifications, see Space Requirements on page 42 and Flooring Requirements on page 44.

Item	Comment
Has a vacant rack location been allocated for each rack?	
Does the floor layout meet the equipment maintenance access requirements?	
Will the rack be positioned so that the exhaust air of one rack doesn't enter the air inlet of another rack?	
Have cabinet stabilization measures been considered?	
If the data center has a raised floor: • Does the raised floor satisfy the weight requirements for the rack? • Is permission required to remove floor tiles for cabling and servicing below the floor? • Is the depth of the raised floor a minimum of 46 cm (18 inches)? • Are there cable routing channels or cutouts?	
Are you providing any additional hardware?	
Is the hardware you are providing fully compatible with the Compute Cloud@Customer system?	
Will the new hardware location require any nonstandard cable lengths?	
Is the floor to ceiling height a minimum of 3214 mm or 3.2 m (10.5 feet)?	

Data Center Environment Checklist

For specifications, see Environmental Requirements on page 47.

Item	Comment
Does the computer room air conditioning meet temperature and humidity requirements?	
Does the installation floor layout satisfy the ventilation requirements?	
Will the rack be positioned so that the exhaust air of one rack doesn't enter the air inlet of another rack?	
If the room cooling is from a raised floor: • Are the perforated floor tiles each rated at 400 CFM or greater? • Can additional perforated floor tiles be obtained if required for additional cooling?	
Does the data center air conditioning provide sufficient front-to-back airflow?	

Item	Comment
Is airflow adequate to prevent hot spots?	
Can the data center continuously satisfy the environmental requirements?	

Access Route Checklist

For specifications, see Space Requirements on page 42 and Flooring Requirements on page 44.

Item	Comment
Has the access route been checked for clearance of the rack, including the minimum width and height requirements for all doors on the route?	
Do all the doors and entry ways conform to the width and height requirements for transportation, including the width of the unpacked unit?	
Do all the doors meet the height requirement of minimum 218 cm (86 inches) for packaged delivery?	
Does the access route provide sufficient space for transport of the packed devices?	
Are there any stairs, ramps, or thresholds that are of concern? If yes, provide details.	
Are all access route incline angles within the permitted range (under 6 degrees)?	
Have you confirmed that the access route is free of any obstacles that would expose the device to shock?	
Can the complete access route support the weight of the rack?	
Is the access route onto the raised floor rated for dynamic loading of the rack?	
Are all the surfaces acceptable for rolling the new unpacked and packed equipment?	
If a pallet jack is to be used: • Can the pallet jack support the weight of the rack? • Are the pallet jack tines compatible with the shipping pallet?	
If there are stairs, is a loading elevator available for the equipment?	

Item	Comment
If an elevator is to be used: • Does the elevator car meet the height, width, and depth requirements for carrying the rack? • Do the elevator doors meet the height and width requirements for moving the rack? • Does the elevator meet the weight requirements for transporting the rack?	

Facility Power Checklist

For specifications, see Electrical Power Requirements on page 44.

Item	Comment
Have the operating voltage and electric current requirements of the device and peripherals been reviewed?	
What type of power supply will be used? • Single-phase or 3-phase. • Low-voltage or High-voltage.	
Are enough power outlets provided within 2 meters (6.5 feet) for each rack?	
Do the power outlets have appropriate socket receptacles for the planned Power Distribution Units (PDUs)?	
Will optional ground cables be attached to the rack?	
Are the electrical circuits suitable in terms of voltage and current-carrying capacities?	
Does the power frequency meet the equipment specifications?	
Are power outlets available for the new equipment at the designated location?	
Will system power be delivered from two separate grids?	
Is there a UPS to power the equipment?	
Are the minimum required power sources available to support the power load (kW or kVA) for the new hardware?	

Safety Checklist

Complete this checklist to ensure that the safety requirements are met. For information about safety, see Emergency Information for Compute Cloud@Customer Installations on page 32 and Environmental Requirements on page 47.

Item	Comment
Is there an emergency power shut off?	
Is there a fire protection system in the data center room?	
Is the computer room adequately equipped to extinguish a fire?	
Is antistatic flooring installed?	
Is the area below the raised floor free of obstacles and blockages?	

Logistics Checklist

Complete this checklist to ensure that the logistics requirements are met. For information about receiving and unpacking, see Space Requirements on page 42.

Item	Comment
Is contact information for the data center personnel available?	
Is there security or access control for the data center?	
Are there any security background checks or security clearances required for Oracle personnel to access the data center? If yes, then provide the process for Oracle to follow.	
How many days in advance must background checks be completed?	
Are there any additional security access issues?	
Is computer room access available for installation personnel?	
Are laptops allowed in the data center?	
Are cell phones allowed in the data center?	
Are cameras allowed in the data center?	
Does the building have a delivery dock?	
Is the site prepared for uncrating and packaging removal? Package removal should take place outside the data center room.	
Does the building have adequate receiving space?	
Is there a delivery / unpacking / staging area?	
Is the delivery / unpacking / staging area protected from the elements?	
Is inside delivery planned (direct to the final rack location in the data center room)?	

Item	Comment
Is the unpacking area air-conditioned to avoid thermal shock for various hardware components?	
Will sufficient moving personnel be available to transport the rack?	
Is union labor required for any part of the delivery or installation?	
Is uncrating of cabinet and packaging removal required?	
Are there any restrictions on delivery truck length, width, or height?	
Is there storage space (cabinet) for the ride along spares? If not, does the customer allow cardboard boxes and other packing material in the computer room, since the spares are packed in cardboard boxes?	
Is there a time constraint on dock access? If yes, provide time constraints.	
Is a tail or side lift required on the delivery carrier to unload the equipment at the delivery dock?	
Will any special equipment be required to place the rack in the data center room? For example: • Stair walkers • Lifters • Ramps • Steel plates • Floor covers	
Does the delivery carrier require any special equipment, such as non-floor-damaging rollers, transport dollies, pallet jacks, or fork lifts?	

Final Preinstallation Checklist

Complete the following checklist to ensure that the network specification requirements are met. For more information about network requirements and IP addresses, see Customer Site Network Requirements on page 48.

Network Specification Considerations	Yes, No, or N/A	Comment
Did you complete the networking worksheet, which is included in the “Initial System Installation Checklist”?		
Have you received the site-specific installation template from your Oracle technical representative?		
Did you review the installation template and consult with your Oracle technical representative regarding any site-specific changes, if any?		

Network Specification Considerations	Yes, No, or N/A	Comment
Did you verify that IP addresses in the installation template are currently not in use?		
Have you performed the required configuration within your network infrastructure to allow the Compute Cloud@Customer to use the IP addresses specified in the template?		
Have you registered IP addresses in the installation template with DNS?		
Did you run the required network cables from your network equipment to the location where the Compute Cloud@Customer will be installed?		
Did you label the network cables that will connect to the Compute Cloud@Customer?		

Initial System Installation Checklist

Complete this checklist to plan for the initial installation wizard, which walks you through the system initialization parameters and first boot process of your Private Cloud Appliance.

Work with your network and site administrators to plan the integration of the appliance into your data center. For initial configuration you can select either a dynamic network or static network configuration. This checklist contains information for both options, but you only need data for the implementation you select.

For more information about network requirements, see Customer Site Network Requirements on page 48.

> **Note:**
>
> **Checklist Legend**
>
> ***** -- required fields for all configurations.
>
> **†** -- required for static network configurations
>
> **‡** -- required for dynamic network configurations

General Configuration Parameters

Item	Your Configuration	Description and Example
Administrator user name and password		The rack doesn't ship with a default administrative user account. You create an administrator account during the initial installation. Passwords must contain at least 12 characters with at least one of each: uppercase character, lowercase character, digit, punctuation character, and no double quote ("").

Item	Your Configuration	Description and Example
Logical connection		Choose ***static*** or ***dynamic*** routing. Both routing options are supported by all three physical topologies (Mesh, Square, Triangle). Static routing implies that all egress traffic goes through a single default data center gateway IP address. It must be in the same subnet as the rack uplink IP addresses, so it is reachable from the spine switches. Dynamic routing implies that BGP (Border Gateway Protocol) is used to establish a TCP connection between two Autonomous Systems: the rack network and the data center network.
(If required) Proxies for internet connections		you can use a `HTTP` proxy (not HTTPS). This is a passive/ corporate proxy supported for the connection from Compute Cloud@Customer to Oracle Cloud Infrastructure.
OCI Region*****		Specify the Oracle Cloud Infrastructure region name that Compute Cloud@Customer will be connected to. See Public Cloud Regions.
OCI Availability Domain*****		Specify the Oracle Cloud Infrastructure availability domain that Compute Cloud@Customer will be connected to.

Compute Cloud@Customer Rack Details

Item	Your Configuration	Description and Example
System Name*****		Name for the system, and used as the Compute Cloud@Customer short name or Realm. This attribute has a maximum length of 24 characters. Acceptable characters are "a" to "z", "A" to "Z", "0" to "9", and "-". **Once set, this parameter can't be changed.**
Domain*****		Domain name for your system which is used as the base domain for the internal network, and by Compute Cloud@Customer public facing services. This attribute has a maximum length of 190 characters. Acceptable characters are "a" to "z", "A" to "Z", "0" to"9", "-" Example: `us.example.com` **Once set this parameter can't be changed.**
Rack Name		Provide a unique name for this Compute Cloud@Customer system.

Item	Your Configuration	Description and Example
Description		Provide an optional description for Compute Cloud@Customer. This is a free text field with a 256 character limit.

Static Routing Network Details

Item	Your Configuration	Description and Example
Uplink Gateway†		IP address for the uplink switch to the default gateway in your data center. Chose a valid IP address in the customer data center subnet that the rack uplink switches are attached to. Example: `10.68.48.86`
Spine switch virtual IP†		Virtual IP address which acts as the public VIP for the spine switches in your company network. Chose a valid IP address in the data center subnet that the rack uplink switches are attached to. Example: `10.68.49.103`
Uplink VLAN		VLAN used to connect to an uplink switch. Chose a VLAN value between 2 and 3899. VLANs 3900 to 4095 are reserved. Example: `322`
Uplink HSRP Group		Assign an HSRP group number to the rack. Acceptable values are 0 to 255, and the default value is 151. If there are multiple racks connected to the same data center infrastructure switches, ensure they use *different* HSRP groups.
Management node 1 IP addresses and host name*	10/25/40/100G*: 1G: host name:	Static IP addresses for management node 1 in your on-premises network. Provide a valid host name. An appended domain name is used if you don't provide a host name.
Management node 2 IP addresses and host name*	10/25/40/100G*: 1G: host name:	Static IP addresses for management node 2 in your on-premises network. Provide a valid host name. An appended domain name is used if you don't provide a host name.
Management node 3 IP addresses and host name*	10/25/40/100G*: 1G: host name:	Static IP addresses for management node 3 in your on-premises network. Provide a valid host name. An appended domain name is used if you don't provide a host name.

Item	Your Configuration	Description and Example
Management virtual IP address and host name*	10/25/40/100G*: 1G: host name*:	Virtual IP addresses of the management node cluster in your on-premises network. This is the IP used to DNAT to the internal management node VIP. Compute Cloud@Customer DNS will resolve Compute Cloud@Customer endpoints to this IP. Chose a valid IP address in the data center CIDR. Example: `10.68.49.170` Provide a valid host name. An appended domain name is used if you don't provide a host name.
Spine switch 1 IP address*		IP address for the spine switch 1 in your on-premises network. Chose a valid IP address in the data center subnet that the rack uplink switches are attached to. Example: `10.68.49.101`
Spine switch 2 IP address*		IP address for the spine switch 2 in your on-premises network. Chose a valid IP address in the data center subnet that the rack uplink switches are attached to. Example: `10.68.49.102`

Dynamic Routing Network Details

Item	Your Configuration	Description and Example
Peer 1 IP‡		Neighbor BGP IP - IP address of customer data center router-1 that the rack uplink switches are attached to. Example: `10.68.48.86`
Peer 1 ASN‡		BGP ASN of customer data center router-1. Example: `64512-65533 or 4200000000-4294967294` (Note: Oracle Spine switch default ASN is 136025.)

Item	Your Configuration	Description and Example
Peer 2 IP‡		Neighbor BGP IP - IP address of customer data center router-2 that the rack uplink switches are attached to. Example: `10.68.48.88`
Peer 2 ASN‡		BGP ASN of customer data center router-2 Example: `64512-65533 or 4200000000-4294967294` (Note: Oracle Spine switch default ASN is 136025.)
Uplink Gateway*		IP address for the uplink switch to the default gateway in your data center. Chose a valid IP address in customer data center subnet that the rack uplink switches are attached to. Example: `10.68.48.86`
Oracle ASN		The default is 136025.
BGP Topology		Options are triangle, square, or mesh. The default is mesh.
BGP Authentication (Optional) Admin BGP Authentication		Enable BGP authentication for your network, and admin network if used.
BGP Authentication Password (Required)		Enter the BGP authentication password for your network, and admin network if used.
Admin BGP Authentication Password (Required if admin network used)		Enter password in hex format. Password must be less than 64 characters (<64).
BGP KeepAlive Timer		The default is 60.
BGP Holddown Timer		The default is 180.
Enable MDA Authentication		The default is false.

Uplink Details

Item	Your Configuration	Description and Example
Uplink Port Speed*		All uplink ports must have the identical speed. The options are 10, 25, 40, 100. The default is 100.

Item	Your Configuration	Description and Example
Uplink Port Count*		The number of uplink ports per spine switch. Connectivity must be identical on both spine switches to provide redundancy and load-splitting. For 100G and 40G speeds, valid values are 1, 2 and 4. For 10G and 25G ports, valid values are 1, 2, 4, 8 and 16. The default for both is 4.
Uplink VLAN MTU*		MTU size determines the maximum packet size that can be transmitted over your uplink connection. The valid range is 576 to 9216. The default is 9216.
Uplink Netmask*		Netmask of the subnet rack is connected to in your data center. Example: `255.255.252.0`
Uplink Port FEC		Configure Forward Error Correction (FEC) for the uplink port. The default option is auto, with on and off as other valid options.

NTP and DNS Server Details

Item	Your Configuration	Description and Example
NTP servers*		At least one valid IP address for an NTP server in your data center. You can enter multiple IP addresses in a comma-separated list. Example: `10.147.24.1,10.211.17.1`
DNS server 1		IP address for primary DNS server. Example: `10.68.50.60`
DNS server 2		IP address for secondary DNS server. Example: `10.147.36.60`
DNS server 3		IP address for tertiary DNS server. Example: `206.233.27.1`

Administration Network Details

Item	Your Configuration	Description and Example
Admin Networking	enable	Enable a separate Administration network to transport administrative traffic.
Management virtual IP address and host name*	10/25/40/100G*: 1G: host name*:	Virtual IP address of the management node cluster in the administration network. This is the IP used to DNAT to the internal management node VIP. Compute Cloud@Customer DNS will resolve Compute Cloud@Customer endpoints to this IP. Chose a valid IP address in the administration network CIDR. Example: `10.168.141.170` Provide a valid host name. An appended domain name is used if you don't provide a host name.
Management node 1 IP addresses and host name	10/25/40/100G: 1G: host name:	Static IP addresses for management node 1 in the administration network. Provide a valid host name. An appended domain name is used if you don't provide a host name.
Management node 2 IP addresses and host name	10/25/40/100G: 1G: host name:	Static IP addresses for management node 2 in the administration network. Provide a valid host name. An appended domain name is used if you don't provide a host name.
Management node 3 IP addresses and host name	10/25/40/100G: 1G: host name:	Static IP addresses for management node 3 in the administration network. Provide a valid host name. An appended domain name is used if you don't provide a host name.
Admin DNS servers		IP addresses for 1-3 DNS servers providing name resolution in the administration network. Example: `10.168.20.31,10.168.20.32,10.147.36.60`
Admin Port Speed		Port speed options are 10, 25, 40, 100. The port speed of your administration port can be different from the data port speed.
Admin Port Count		For 100G and 40G speeds, valid value is 1. For 10G and 25G ports, valid values are 1 to 4.
Admin HSRP Group		Assign an HSRP group number to your Administration network. Acceptable values are 0 to 255, and the default value is 152. If there are multiple racks connected to the same data center infrastructure switches, ensure they use *different* HSRP groups.

Item	Your Configuration	Description and Example
Admin VLAN		VLAN used to connect to the Administration network (only access mode supported). The valid range for users to select is 2 to 3899. The default is 3915, in the Oracle-reserved range.
Admin VLAN MTU		MTU size determines the maximum packet size that can be transmitted over your administration connection. The valid range is 576 to 9216. The default is 9216.
Admin Port FEC		Configure Forward Error Correction (FEC) for the administration port. The default option is auto, with on and off as other valid options.
Admin Gateway IP		IP address of the default gateway in your Administration network. Example: `10.168.141.1`
Admin Netmask		Netmask of the subnet the Administration network is connected to.
Admin CIDR		CIDR range for which the default route is the Administration gateway IP.
Admin Spine 1 IP		Public IP address of spine switch 1. Needed for HSRP configuration or the spine virtual IP.
Admin Spine 2 IP		Public IP address of spine switch 2. Needed for HSRP configuration or the spine virtual IP.
Admin Spine VIP		Public virtual IP of the spine switches.

Compute Cloud@Customer Public IP Addresses

Item	Your Configuration	Description and Example
Public IP Addresses		A range of customer data center IP addresses can be used for Compute Cloud@Customer components that require public IP addresses. Note: In this context, public IP addresses refer to IP addresses that have access to the data center network from the Compute Cloud@Customer subnet. You must specify IP addresses or ranges that are routed to the Compute Cloud@Customer from the data center. Route tables must be correct and consistent. Enter a string containing a comma separated list of valid CIDRs. Example: `"10.68.49.249","10.68.50.32/32","10.68.51.4/31"` Partial CIDR deletion isn't supported.
Object Storage IP address		Valid IP address for an `objectstorage` endpoint that is outside the public IP range.

Important Endpoints

Compute Cloud@Customer infrastructures need to access the following endpoints from your network. Depending on your Firewall configuration, you might need to explicitly enable access to these endpoints.

Service Name	Your Value	Endpoint FQDN
Management plane*		**`ccc-mp.${region}.oci.${oraclecloud}`** Example: `https://ccc-mp.us-ashburn-1.oci.oraclecloud.com`
Websocket*		**`connect.ws.ccc.${region}.oci.${oraclecloud}`** Example: `https://connect.ws.ccc.us-ashburn-1.oci.oraclecloud.com`
Stunnel*		**`connect.se.ccc.${region}.oci.${oraclecloud}`** Example: `https://connect.se.ccc.us-ashburn-1.oci.oraclecloud.com`

Service Name	Your Value	Endpoint FQDN
OCI Object Storage*		**objectstorage.${region}.${oraclecloud}** Example: `https://objectstorage.us-ashburn-1.oraclecloud.com`
OCI Identity*		**identity.${region}.oci.${oraclecloud}** `Example: https://identity.us-ashburn-1.oci.oraclecloud.com`
OCI Authentication*		**auth.${region}.${oraclecloud}** `Example: https://auth.us-ashburn-1.oraclecloud.com`
Resource Principal Service*		**certificates.${region}.oci.${oraclecloud}** `Example: https://certificates.us-ashburn-1.oci.oraclecloud.com`

Preparing Your Tenancy

Before the Compute Cloud@Customer infrastructure is connected to Oracle Cloud Infrastructure, the tenancy administrator must set up compartments, create policies, and configure a virtual cloud network. This setup is used to connect the Compute Cloud@Customer infrastructure to Oracle Cloud Infrastructure.

You can prepare your tenancy before the Compute Cloud@Customer rack is delivered to your site.

If working in the Oracle Cloud Infrastructure environment is new to you, consider reviewing Learn Best Practices for Setting Up Your Tenancy.

Prepare your tenancy by completing these activities:

- Establish a Federated Identity Provider on page 68
- Create Users and Groups on page 68
- Create or Identify Compartments on page 68
- Add Required Policies on page 69
- Create a VCN and Subnet on page 70
- Subscribe to Compute Cloud@Customer Notifications on page 71

You can also watch this Video: Prepare your Oracle Cloud Infrastructure Tenancy for Compute Cloud@Customer.

> **Note:**
>
> The tasks you perform in this section are required to establish the connection between OCI and the Compute Cloud@Customer infrastructure. You need to perform similar administrative tasks, beyond what is described here, before you can create resources such as instances on Compute Cloud@Customer infrastructure. Most of the additional administrative tasks can be performed after you prepare your tenancy, or after the installation. See Postinstallation Administration on page 80.

Establish a Federated Identity Provider

Before Compute Cloud@Customer is installed, your tenancy must be set up to use a federated identity provider to manage authentication.

When Oracle installs Compute Cloud@Customer, Oracle configures the Compute Cloud@Customer infrastructure to use the same federated identity provider. This enables you to use the same credentials to access Oracle Cloud Infrastructure and Compute Cloud@Customer.

If your tenancy is already configured to use a federated identity provider, including Oracle's Identity Cloud Service, you're all set. Share your federated identity information with your Oracle representative. Otherwise, work with your Oracle representative to establish a federated identity provider.

You can use an external identity provider or Oracle Identity Cloud Service. The type of identity provider you can use depends on the type of tenancy you have (a tenancy with IAM identity domains or without IAM identity domains).

For more information, see these resources:

- Determining the Tenancy Type
- Tenancies with Identity Domains – Federating with Identity Providers
- Tenancies without Identity Domains – Federating with Identity Providers

> **Important:**
>
> If you change your identity provider configuration in Oracle Cloud Infrastructure, Oracle must make the same administrative changes on Compute Cloud@Customer. In this situation, open an Oracle Support Request to request help. See Creating a Support Request.

For information about securing IAM Federation, see IAM Federation.

Create Users and Groups

To prepare your Oracle Cloud Infrastructure tenancy, identify users and create groups for the people in your organization who administer the Compute Cloud@Customer infrastructure.

Perform this task before Compute Cloud@Customer is connected to Oracle Cloud Infrastructure.

For information about how to add users and groups, see Managing Oracle Identity Cloud Service Users and Groups in the Oracle Cloud Infrastructure Console.

1. Identify your tenancy administrator.
2. Create at least one group with users who can perform these administrative tasks:
   - Create, update, and delete Compute Cloud@Customer infrastructures.
   - Create, update, and delete Compute Cloud@Customer upgrade schedules.
   - Run the certificate-based registration process that establishes the secure connection of the infrastructure to your tenancy. We recommend that you create a specific group for this administrative task, and only grant permissions limited to performing this task.

The groups are included in policies you define later. See Add Required Policies on page 69.

Create or Identify Compartments

When Compute Cloud@Customer is connected to Oracle Cloud Infrastructure, one or more compartments are needed.

A compartment is a collection of related resources. Compartments are a fundamental component of Oracle Cloud Infrastructure for organizing and isolating your cloud resources. You use them to separate resources for the purposes of controlling access (using policies), and isolation (separating the resources for one project or business unit from another).

For Compute Cloud@Customer, at least one compartment is needed for the following items:

- Compute Cloud@Customer infrastructure connection to Oracle Cloud Infrastructure.

- The VCN you eventually created for the connection to Oracle Cloud Infrastructure.

Compute Cloud@Customer can be connected to your tenancy (root compartment), to an existing compartment, or to a new compartment. You can use multiple compartments. For example, you can use one compartment for the infrastructure connection, and another for the VCN.

1. Create or choose a compartment based on how you use compartments to control access to resources.

   If you plan to create a new compartment, sign in to OCI and use the Oracle Cloud Console, or use the CLI or API to create the compartment in your tenancy.

   > **Note:**
   >
   > Compartments used for Compute Cloud@Customer, including compartments for the installation, are created and managed in OCI. Compartments aren't managed in the Compute Cloud@Customer infrastructure. All compartments in the tenancy are automatically synchronized to the Compute Cloud@Customer infrastructure, every ten minutes or so.

   For an introduction to compartments, and for instructions for managing compartments, see Managing Compartments.

Add Required Policies

Certain IAM policies must be configured before Compute Cloud@Customer is connected to your tenancy.

1. Configure the following policies in your tenancy.

   For information about how to work with policies, see Managing Policies.

   If your tenancy supports Identity Domains, you can create policies that specify the dynamic group. To determine if your tenancy has Identity Domains or not, see Determining the Tenancy Type.

   > **Note:**
   >
   > Different policy statements can be constructed to achieve the same level of access to resources. The following list of policies provide examples. You can use the example, or create policy variations, as long as the policies allow access to the correct user or group for the particular resource.

   Policy 1 – Allows users to create, read, update, and delete Compute Cloud@Customer infrastructures and upgrade schedules.

   > **Important:**
   >
   > Specify an IAM group that only includes the users who require permissions to manage infrastructures and upgrade schedules. Administration of these resources is critical to the functionality of Compute Cloud@Customer, and must not be allowed for unauthorized users.

   Policy example for IAM with or without Identity Domains:

   ```
   allow group <group_name> to manage ccc-family in tenancy
   ```

   Policy 2 – Allows Compute Cloud@Customer to use your IAM data for identity and access management on Compute Cloud@Customer resources.

   Policy example for IAM with or without Identity Domains:

   ```
   allow any-user to {COMPARTMENT_INSPECT, USER_INSPECT, GROUP_INSPECT,
    DYNAMIC_GROUP_INSPECT, POLICY_READ, TAG_NAMESPACE_INSPECT,
    USER_READ, TAG_DEFAULT_INSPECT, TAG_NAMESPACE_READ, DOMAIN_READ,
   ```

```
 DOMAIN_INSPECT } in tenancy where all { request.principal.id='<ccc-
infrastructure_OCID>', request.principal.type='cccinfrastructure' }
```

Policy example for IAM with Identity Domains:

```
allow dynamic-group <dynamic-group> to {COMPARTMENT_INSPECT,
 USER_INSPECT, GROUP_INSPECT, DYNAMIC_GROUP_INSPECT, POLICY_READ,
 TAG_NAMESPACE_INSPECT, USER_READ, TAG_DEFAULT_INSPECT,
 TAG_NAMESPACE_READ, DOMAIN_READ, DOMAIN_INSPECT} in tenancy
```

Policy 3 – Allows the Compute Cloud@Customer infrastructure service to send you notifications about upgrades.

For more information about upgrade notifications and how to subscribe to receive them, see Subscribing to Upgrade Notifications on page 96.

The following examples show policies for IAM with or without Identity Domains:

```
allow any-user to manage ons-topics in tenancy where
 request.principal.type ='cccinfrastructurenotifier'
```

The policy can be modified to restrict access to the root compartment as shown in the following example:

```
allow any-user to manage ons-topics in tenancy where all
 {request.principal.type='cccinfrastructurenotifier',
 target.compartment.name = 'root_compartment' }
```

If you restrict access to a compartment, it must be to the root compartment (tenancy).

Policy 4 – Allows a user in the specified group to initiate the registration process that enables the infrastructure to communicate with your OCI tenancy.

> **Note:**
>
> Don't specify a regular admin group. Instead, create a group with a user whose sole purpose is to run the registration process.
>
> For more information, see Connecting a Compute Cloud@Customer Infrastructure to OCI on page 73.

The following policy examples are for IAM with or without Identity Domains.

This example sets the policy at the tenancy level:

```
allow group <group_name> to { CCC_CERTIFICATE_REGISTER } in tenancy
```

This example sets the policy at the compartment level. The compartment must be the compartment that's associated with the infrastructure:

```
allow group <group_name> to { CCC_CERTIFICATE_REGISTER } in compartment
 '<compartment_name>'
```

For information about Compute Cloud@Customer policies you can use to control access to Compute Cloud@Customer infrastructure and upgrade schedule operations, see Compute Cloud@Customer Policy Reference on page 110.

Create a VCN and Subnet

Before Compute Cloud@Customer is connected to your tenancy, create a VCN with a subnet in the tenancy.

Infrastructures require the following network resources in the tenancy:

1. One Virtual Cloud Network (VCN). See Creating a VCN. We recommend a small CIDR block, for example `192.168.100.0/29`.
2. For each infrastructure, create one subnet in the VCN. See Creating a subnet. For example, `192.168.100.0/30`.

What's Next?

Work with Oracle to install and connect Compute Cloud@Customer to your network. After Oracle initializes Compute Cloud@Customer, you then create a Compute Cloud@Customer infrastructure, and connect it to your OCI tenancy. See Creating a Compute Cloud@Customer Infrastructure in OCI on page 72.

For a complete list of installation tasks, see Installing and Configuring Compute Cloud@Customer on page 40.

Subscribe to Compute Cloud@Customer Notifications

Subscribing to Compute Cloud@Customer notifications enables you to receive notifications and view Cloud Console announcements about upgrades.

This feature uses the Oracle Cloud Infrastructure (OCI) Notification service to inform you of the following upgrade activities:

- When an upgrade is scheduled, you're notified 7 days before the upgrade starts.
- When an upgrade starts.
- When the upgrade is finished.

You can subscribe to receive notifications in Email. You can also view the notifications in the Oracle Cloud Console in your tenancy.

Subscribe to the Compute Cloud@Customer topics

To enable upgrade notifications, subscribe to `ComputeCloudCustomer` notifications. See the following sections in the OCI Notification Service documentation:

- Listing Subscriptions – Look for subscriptions starting with **ComputeCloudCustomer**.
- Creating an Email Subscription

See also Subscribing to Upgrade Notifications on page 96.

Managing Compute Cloud@Customer Infrastructures

Compute Cloud@Customer infrastructure are the resources in Oracle Cloud Infrastructure that communicate with the corresponding hardware and software in the customer's data center.

You must create an infrastructure before connecting the hardware and software.

See Creating a Compute Cloud@Customer Infrastructure in OCI on page 72 and Connecting a Compute Cloud@Customer Infrastructure to OCI on page 73.

You can change the name and upgrade schedule associated with an infrastructure later. See Editing a Compute Cloud@Customer Infrastructure in OCI on page 78.

If you need to change the compartment where infrastructures are stored, you can do this. See Moving a Compute Cloud@Customer Infrastructure Between Compartments on page 79.

Don't try to delete an infrastructure if it has been used to connect to the Compute Cloud@Customer infrastructure in the data center.

> **Important:**
>
> Infrastructures depend on continuous connection to Oracle Cloud Infrastructure. When Oracle Cloud Infrastructure isn't accessible, IAM configuration changes aren't synchronized on Compute Cloud@Customer and the required monitoring and maintenance tasks aren't performed.

Creating a Compute Cloud@Customer Infrastructure in OCI

Create a Compute Cloud@Customer infrastructure in Oracle Cloud Infrastructure (OCI) to communicate with the corresponding infrastructure in the data center.

Before creating an infrastructure, ensure that the following tenancy setup has been completed. See Preparing Your Tenancy on page 67 .

- A compartment is set up to host the infrastructure definitions.
- A VCN and subnet are set up as required by the infrastructure.
- IAM policies are set up to provide appropriate user and resource access to the infrastructure compartment.

Also ensure that an upgrade schedule for this infrastructure is defined. If you don't set up an upgrade schedule, Oracle can upgrade the infrastructure at any time. See Creating a Compute Cloud@Customer Upgrade Schedule on page 95.

Using the Console

To create an infrastructure, follow these steps:

1. In the Oracle Cloud Console, open the navigation menu, select **Hybrid & Multicloud**, and then select **Oracle Compute Cloud@Customer**.
2. Select the compartment that you want to create the infrastructure in.
3. Select **Create infrastructure**.
4. Define the following infrastructure parameters. Avoid entering confidential information.
   - **Display name**: The name shown when listing infrastructures in the Oracle Cloud Console. This name must match the System Name supplied to Oracle in the Initial System Installation Checklist on page 58.
   - **Description**: (Optional) Information about the infrastructure to distinguish it from others.
   - **Compartment**: The compartment that you want to create the infrastructure in.
   - **Networking**: The VCN and subnet for this infrastructure. This value can't be edited later.
   - **Upgrade schedule**: (Optional) The upgrade schedule associated with this infrastructure, which defines when it can be upgraded. You can create and attach upgrade schedules later. See Managing Upgrade Schedules on page 94.
5. (Optional) To add tags to the infrastructure, select **Show advanced options**.

   If you have permissions to create a resource, then you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags.
6. Select **Create**.

What's Next?

Connect the Compute Cloud@Customer infrastructure to your OCI tenancy. See Connecting a Compute Cloud@Customer Infrastructure to OCI on page 73.

Using the CLI

Use the ccc infrastructure create command and required parameters to create an infrastructure:

```
oci ccc infrastructure create --compartment-id <compartment ocid> --display-
name <Descriptive name> --subnet-id <subnet to connect infrastructure to>...
 [OPTIONS]
```

Avoid entering confidential information.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

What's Next?

Connect the Compute Cloud@Customer infrastructure to your OCI tenancy. See Connecting a Compute Cloud@Customer Infrastructure to OCI on page 73.

Using the API

Use the CreateCccInfrastructure operation to create a Compute Cloud@Customer infrastructure.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

What's Next?

Connect the Compute Cloud@Customer infrastructure to your OCI tenancy. See Connecting a Compute Cloud@Customer Infrastructure to OCI on page 73.

Connecting a Compute Cloud@Customer Infrastructure to OCI

The Compute Cloud@Customer infrastructure in the data center needs to be connected to Oracle Cloud Infrastructure (OCI) before it can be used. This task involves a registration process during which a secure connection is established.

As of June 2025, a certificate chain of trust is used to authenticate the registration process that establishes the connection. Prior to June 2025, the connection used PIN-based authentication as described in (Legacy Method) Connecting an Infrastructure. We recommend all Compute Cloud@Customer infrastructures transition to the certificate-based registration process.

The first time implementing this new registration process requires you to work with Oracle to establish certificate-based authentication with OCI. After that, the certificate is automatically renewed every 30 days, which is within the 60 day certificate lifespan.

Connecting an infrastructure with this new registration process applies to the following situations:

- **New installations:** Establishing the connection is part of the Oracle managed installation, and you don't need to request a connection.
- **Existing installations:** If the infrastructure was installed before <date>, the infrastructure needs to be transitioned to the certificate-based registration before the 90-day legacy connection expires. In this case, Oracle might contact you to plan a connection, or you can request a connection.
- **Connection loss:** If a prolonged connection outage occurs, and the connection is in the REJECTED state, request a connection.

Requesting a Connection

Prerequisites:

- A Compute Cloud@Customer infrastructure must be set up in OCI. See Creating a Compute Cloud@Customer Infrastructure in OCI on page 72.
- The Compute Cloud@Customer infrastructure must be installed in your data center.
- In your tenancy, have a group with a user whose sole purpose is to run the registration process. See Create Users and Groups on page 68.
- Create a policy for the registration group that allows the group the CCC_CERTIFICATE_REGISTER permission. Don't grant the group any other permissions. See Add Required Policies on page 69.
- The connection must be in the REJECT connection state for a minimum of 20 minutes before you reconnect an infrastructure. See Viewing the Connection State on page 76.

  Newly installed infrastructures are in the REJECT state by default. If you need to put the infrastructure into a REJECT state (for example, if you believe the security of the connection has been compromised), see Rejecting a Connection on page 78.

Open a Support Request

If you need to reconnect the infrastructure for any reason, open a Support Request. See Creating a Support Request. To access support, sign in to the Oracle Cloud Console as described in Sign In to the OCI Console.

(Legacy Method) Connecting a Compute Cloud@Customer Infrastructure to OCI

This section describes the original way that Compute Cloud@Customer infrastructures were connected to Oracle Cloud Infrastructure (OCI). This task involves a PIN-based registration process during which a secure connection is established.

> **Important:**
>
> All new infrastructure installations must use the certificate-based registration process as described in Connecting a Compute Cloud@Customer Infrastructure to OCI on page 73.
>
> Infrastructures that are configured to use this legacy registration process must transition to the certificated-based registration process before the current PIN-based registration is set to expire. Contact Oracle by opening a Support Request to schedule the transition. See Creating a Support Request. To access support, sign in to the Oracle Cloud Console as described in Sign In to the OCI Console.

> **Note:**
>
> A connected system moves to a REJECT state after 90 days of continuous operation to force a refresh of the connection. After the system is moved to REJECT, you must reconnect the infrastructure.
>
> When the infrastructure is in a REJECT state, your Compute Cloud@Customer workloads continue to run. However, the synchronization of the information between the infrastructure and OCI stops until the connection is reestablished.

Use the instructions in this section to connect the Compute Cloud@Customer infrastructure that's in your data center to your OCI tenancy.

Connecting an infrastructure applies to the following situations:

- After a new Compute Cloud@Customer infrastructure is installed in your data center.
- If a prolonged connection outage occurs, lasting more than 15 minutes.
- You're coming up on 90 days of uninterrupted connectivity and you want to proactively refresh the connection to avoid the forced connection refresh.
- You believe the security of the connection has been compromised.

Prerequisites:

- A Compute Cloud@Customer infrastructure must be set up in OCI. Follow the steps in Creating a Compute Cloud@Customer Infrastructure in OCI on page 72.
- The Compute Cloud@Customer infrastructure must be installed in your data center. Oracle does this using the information that you provided.
- The connection must be in the **REJECT** connection state for a minimum of 20 minutes before you reconnect an infrastructure. See Viewing the Connection State on page 76.

  Newly installed infrastructures are in the REJECT state by default. If you need to put the infrastructure into a REJECT state (for example, if you believe the security of the connection has been compromised), see Rejecting a Connection on page 78.

Using the Console

To connect an infrastructure to Oracle Cloud Infrastructure, follow these steps:

1. In the Oracle Cloud Console, open the navigation menu, select **Hybrid & Multicloud**, and then select **Oracle Compute Cloud@Customer**.
2. In the list of infrastructures, find the one that you want to use to connect to the infrastructure in your data center. If you don't see the infrastructure that you want, you might need to change compartments.

The connection state is **REJECT** at this stage.

3. Select the display name to open the infrastructure details page.
4. Copy the **Provisioning PIN**.
5. In a terminal window, use the `curl` command to provide the PIN and return the fingerprint:

```
curl -k -sS --header "Content-Type: application/json" --request POST --
data '{"pin":"<Provisioning_PIN>"}' https://rps.<system>.<domain>/rps_pin
```

The connection state is now **Request** and the bootstrap process starts.

The fingerprint is returned in SHA 512 format as the value of a JSON key/value pair. For example `{"fingerprint": "<provisioning_fingerprint>"}`

6. Make note of the infrastructure fingerprint from the JSON output from the command in the previous step.
7. Back on the infrastructure details page in the Oracle Cloud Console, view the provisioning fingerprint.
8. Compare the fingerprints.
   - If the fingerprints match, continue to the next step.
   - If they don't match, select the **Actions** drop-down menu, and select **Reject connection** to set the state back to **REJECT**. Then start again with a new provisioning PIN.
9. Select **Accept connection**.
10. In the **Accept connection** dialog box, select **Yes** to confirm that the fingerprints match and you want the connection to be made.

    The connection state is now **Ready** and changes to **Connected** when the secure link is fully established.

What's Next?

Create an upgrade schedule. See Creating a Compute Cloud@Customer Upgrade Schedule on page 95.

Using the CLI

1. Use the ccc infrastructure get command and required parameters to obtain the infrastructure details, including the **Provisioning PIN**:

```
oci ccc infrastructure get --infrastructure-id <OCID of infrastructure to
 connect to> ... [OPTIONS]
```

2. Copy the **Provisioning PIN**.
3. Connect to the infrastructure using `curl` to provide the PIN and return the fingerprint:

```
curl -k -sS --header "Content-Type: application/json" --request POST --
data '{"pin":"<Provisioning_PIN>"}' https://rps.<system>.<domain>/rps_pin
```

The fingerprint is returned in SHA 512 format as the value of a JSON key/value pair. For example `{"fingerprint": "<provisioning_fingerprint>"}`

4. Use the ccc infrastructure get command again to query the infrastructure in OCI. Confirm that the state is **REQUEST**, and obtain the **Provisioning Fingerprint**.
5. Compare the fingerprints.
   - If they match, use the ccc infrastructure update command to update the state to READY and accept the connection.

```
oci ccc infrastructure update <OCID for infrastructure> --connection-
state READY
```

- If the fingerprints don't match, use the ccc infrastructure update command to update the state to REJECT and start again.

```
oci ccc infrastructure update <OCID for infrastructure> --connection-
state REJECT
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

What's Next?

Create an upgrade schedule. See Creating a Compute Cloud@Customer Upgrade Schedule on page 95.

Using the API

1. Use the **GetCccInfrastructure** operation to obtain the infrastructure details, including the **Provisioning PIN**.
2. Copy the **Provisioning PIN**.
3. Connect to the infrastructure using `curl` to provide the PIN and return the fingerprint:

```
curl -k -sS --header "Content-Type: application/json" --request POST --
data '{"pin":"<Provisioning_PIN>"}' https://rps.<system>.<domain>/rps_pin
```

   The fingerprint is returned in SHA 512 format as the value of a JSON key/value pair. For example `{"fingerprint": "<provisioning_fingerprint>"}`
4. Use the **GetCccInfrastructure** operation again to query the infrastructure in OCI. Confirm that the state is **REQUEST**, and obtain the **Provisioning Fingerprint**.
5. Compare the fingerprints.
   - If they match, use the **UpdateInfrastructure** operation to update the state to READY and accept the connection.
   - If the fingerprints don't match, use the use the **UpdateInfrastructure** operation to update the state to REJECT and start again.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

What's Next?

Create an upgrade schedule. See Creating a Compute Cloud@Customer Upgrade Schedule on page 95.

Viewing the Connection State

You can view the connection state of a Compute Cloud@Customer infrastructure. The connection state indicates the condition of the connection of the Compute Cloud@Customer infrastructure in your data center to your OCI tenancy.

These are the possible states:

- `REJECT`
- `REQUEST`
- `READY`
- `CONNECTED`
- `DISCONNECTED`

You can only update the state from `REQUEST` to `READY` (see Connecting a Compute Cloud@Customer Infrastructure to OCI on page 73), or from any state back to `REJECT` (see Rejecting a Connection on page 78).

The system automatically handles the `REJECT` to `REQUEST`, `READY` to `CONNECTED`, or `CONNECTED` to `DISCONNECTED` transitions.

Using the Console

1. In the Oracle Cloud Console, open the navigation menu, select **Hybrid & Multicloud**, and then select **Oracle Compute Cloud@Customer**.

2. In the list of infrastructures, view the Connection state column. If you don't see the infrastructure that you want, you might need to change compartments.

Using the CLI

Use the oci ccc infrastructure list command and required parameters to view a list of Compute Cloud@Customer infrastructures and parameters such as the connection state.

```
oci ccc infrastructure list [OPTIONS]
```

Example of listing all infrastructures in a specified compartment:

```
oci ccc infrastructure list -c <compartment_OCID> --all
{
 "data": {
      "items": [
           {
                "compartment-id": "<compartment_OCID>",
                "connection-state": "REJECT",
                "defined-tags": {
                     "Oracle-Tags": {
                          "CreatedBy": "user",
                          "CreatedOn": "2023-07-28T12:53:26.934Z"
                     }
                },
                "display-name": "TestInfrastructure",
                "freeform-tags": {},
                "id": "<infrastructure_OCID>",
                "lifecycle-state": "ACTIVE",
                "short-name": "<infrastructure_SHORTNAME>",
                "subnet-id": "<subnet_OCID>",
                "system-tags": {},
                "time-created": "2023-07-28T12:53:27.141000+00:00"
           },
           {
                "compartment-id": "<compartment_OCID>",
                "connection-state": "CONNECTED",
                "defined-tags": {
                     "Oracle-Tags": {
                          "CreatedBy": "user",
                          "CreatedOn": "2023-07-18T08:39:53.201Z"
                     }
                },
                "display-name": "some text",
                "freeform-tags": {},
                "id": "<infrastructure_OCID>",
                "lifecycle-state": "ACTIVE",
                "short-name": "<infrastructure_SHORTNAME>",
                "subnet-id": "<subnet_OCID>",
                "system-tags": {},
                "time-created": "2023-07-18T08:39:53.433000+00:00"
           },
 }
}
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListCccInfrastructures operation to view a list of Compute Cloud@Customer infrastructures.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Rejecting a Connection

If the Compute Cloud@Customer infrastructure connection is broken, or you believe the security of the connection has been compromised, or you want to make a new connection, reject the connection. This puts the connection into a REJECT state.

For more information about connection states, see Viewing the Connection State on page 76.

> **Important:**
>
> The infrastructure connection state must remain in the REJECT state for a minimum of 20 minutes before you connect the infrastructure. See Connecting a Compute Cloud@Customer Infrastructure to OCI on page 73.

Using the Console

1. In the Oracle Cloud Console, open the navigation menu, select **Hybrid & Multicloud**, and then select **Oracle Compute Cloud@Customer**.
2. In the list of infrastructures, find the one that you want to disconnect from the infrastructure in your data center. If you don't see the infrastructure that you want, you might need to change compartments.
3. Reject the connection in one of the following ways:
   - If you're on the Infrastructures list page, select the Actions menu (•••), and select **Reject connection**.
   - If you're on the Infrastructure details page, select the **Actions** drop-down menu, and select **Reject connection**.

   The connection state changes to **REJECT**.
4. To reconnect the infrastructure, see Connecting a Compute Cloud@Customer Infrastructure to OCI on page 73.

Using the CLI

Use the ccc infrastructure update command and required parameters to update a Compute Cloud@Customer infrastructure, including the connection state.

```
oci ccc infrastructure update --infrastructure-id <infrastructure_OCID> --
connection-state REJECT [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateCccInfrastructure operation to update a Compute Cloud@Customer infrastructure, including the connection state.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Editing a Compute Cloud@Customer Infrastructure in OCI

Edit the display name, description, or tags of a Compute Cloud@Customer infrastructure, or associate a new upgrade schedule with the infrastructure.

Using the Console

To delete an infrastructure, follow these steps:

1. In the Oracle Cloud Console, open the navigation menu, select **Hybrid & Multicloud**, and then select **Oracle Compute Cloud@Customer**.
2. For the infrastructure that you want to edit, select the Actions menu (•••), then select **Edit**.
3. Update the infrastructure parameters. Avoid entering confidential information. For a description of the parameters, see Creating a Compute Cloud@Customer Infrastructure in OCI on page 72.
4. Select **Save**.

Using the CLI

Use the ccc infrastructure update command and required parameters to update an infrastructure:

```
oci ccc infrastructure update --infrastructure-id <infrastructure ocid> ...
 [OPTIONS]
```

Avoid entering confidential information.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateCccInfrastructure operation to update a Compute Cloud@Customer infrastructure.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Compute Cloud@Customer Infrastructure in OCI

You must delete only an infrastructure that is not connected to a Compute Cloud@Customer infrastructure in your data center. An infrastructure that is not connected is in the **Reject** state.

If you need to disconnect an infrastructure, see Rejecting a Connection on page 78.

Using the Console

To delete an infrastructure, follow these steps:

1. In the Oracle Cloud Console, open the navigation menu, select **Hybrid & Multicloud**, and then select **Oracle Compute Cloud@Customer**.
2. In the list of infrastructures, find the one that you want to delete. If you don't see the infrastructure that you want, you might need to change compartments.

   The connection state must be **Reject**.
3. From the Actions menu (•••) for the infrastructure, select **Copy OCID**. You need the OCID to confirm deletion of the infrastructure.
4. From the Actions menu (•••) for the infrastructure, select **Delete**.
5. Confirm deletion by pasting the OCID of the infrastructure.
6. Select **Delete**.

Using the CLI

Use the ccc infrastructure delete command and required parameters to delete an infrastructure:

```
oci ccc infrastructure delete --infrastructure-id <infrastructure ocid>
 [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteCccInfrastructure operation to delete a Compute Cloud@Customer infrastructure.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Moving a Compute Cloud@Customer Infrastructure Between Compartments

Move an existing Compute Cloud@Customer infrastructure to a different compartment within the same tenancy.

You must have the required permissions in both compartments. See Preparing Your Tenancy on page 67 for details.

Using the Console

To move an infrastructure, follow these steps:

1. In the Oracle Cloud Console, open the navigation menu, select **Hybrid & Multicloud**, and then select **Oracle Compute Cloud@Customer**.
2. Select the name of the infrastructure that you want to move to a different compartment. If you don't see the infrastructure that you want, you might need to change compartments.
3. From the **Actions** drop-down menu, select **Move resource**.
4. Select the compartment to move the infrastructure to, and then select **Move resource**.

Using the CLI

Use the ccc infrastructure change-compartment command and required parameters to move an infrastructure to a different compartment:

```
oci ccc infrastructure change-compartment --compartment-id <compartment ocid
 of the new compartment> --infrastructure-id <OCID of the infrastructure to
 be moved> ... [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ChangeCccInfrastructureCompartment operation to move a Compute Cloud@Customer infrastructure to a different compartment in the tenancy.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Postinstallation Administration

After the Compute Cloud@Customer infrastructure is installed and connected to Oracle Cloud Infrastructure (OCI), you need to perform additional administrative tasks before you can create resources such as virtual cloud networks, instances, and storage on the Compute Cloud@Customer infrastructure.

> **Note:**
>
> The tasks you performed in Preparing Your Tenancy on page 67 (creating a user, compartment, policies, and a VCN) are used only to establish the connection between OCI and the Compute Cloud@Customer infrastructure. These resources aren't intended to be used for organizing your cloud resources. You need to create similar IAM and network resources for organizing and isolating your cloud resources.

Start by securing Compute Cloud@Customer. Perform initial security tasks such as securing the hardware, your tenancy, and the network. See Securing Compute Cloud@Customer on page 81.

After security administration, postinstallation administration tasks you perform depend on the state of your tenancy.

- If your Identity Access Management (IAM) resources such as users, groups, compartments, and policies are already set up in your tenancy, the IAM configuration in your tenancy is automatically synchronized on the Compute Cloud@Customer infrastructure. Administrators can begin to create cloud resources on Compute Cloud@Customer.
- If your tenancy is new, or if you need to update IAM resources to accommodate Compute Cloud@Customer, you need to configure these resources in your tenancy before creating cloud resources on the Compute Cloud@Customer infrastructure:
  - **Users:** Create user accounts for people who manage Compute Cloud@Customer resources.
  - **Groups:** Create groups and add users to groups.
  - **Compartments:** Create compartments to organize and secure your Compute Cloud@Customer resources.
  - **Policies:** Create policies to restrict access to Compute Cloud@Customer resources.

- **Defined tags:** (Optional) Create defined tags to organize resources based on your business needs. Note that free-form tags are created when you create resources on the Compute Cloud@Customer infrastructure. See Tagging Resources (IAM in OCI) on page 115.

These Compute Cloud@Customer IAM resources are managed in your tenancy, not on the Compute Cloud@Customer infrastructure. The tenancy IAM configuration is synchronized to the Compute Cloud@Customer infrastructure every ten minutes or so.

The way you perform these tasks depends on the tenancy type – either with or without Identity Domains. For more information about tenancy types on Compute Cloud@Customer, see Identity and Access Management (IAM) on page 108.

If you're new to Compute Cloud@Customer, one way to learn about creating cloud resources is to create a sample instance. See Tutorial: Launching Your First Instance on page 307.

Compute Cloud@Customer Infrastructure Administration

Learn how to administer Compute Cloud@Customer.

The following sections describe the administrative functions and operations you can access on Compute Cloud@Customer:

- Securing Compute Cloud@Customer on page 81
- Managing Upgrade Schedules on page 94

Securing Compute Cloud@Customer

Compute Cloud@Customer provides effective and manageable security that enables you to run mission-critical workloads and to store data with confidence.

Compute Cloud@Customer is a fully-managed, rack-scale, OCI regional resource bringing Oracle's second-generation cloud services on-premises. The system is installed by Oracle, which provides a level of security independent of local practices. However, this also requires the system administrators to understand exactly what's provided as a security baseline. Then the administrators can adjust security practices and configurations to achieve the required level of security needed for their specific circumstances.

> **Note:**
>
> For comprehensive information about Oracle Cloud Infrastructure security, see Oracle Cloud Infrastructure: Security.

Main Security Areas

Compute Cloud@Customer security is managed in three areas:

- **Compute Cloud@Customer Infrastructure:** This is the physical rack hardware that's owned by Oracle, and installed on the customer's premises. Some security-related tasks are performed at this basic level when the system is installed.

  This infrastructure layer also includes software for controlling the infrastructure. Access to this layer is restricted to, and closely monitored by authorized Oracle personnel only. You can control when authorized Oracle personnel can access the infrastructure.
- **Compute Cloud@Customer infrastructure-based resources:** This is where your workloads are created, configured and hosted, and where cloud resources such as compute instances, networks, and storage are managed.

  You manage security in this area by configuring the resources (networks, instances and storage) in a secure way. For example, to secure your VCN, you can use network security groups (NSGs) and security lists to secure network access, and use other network security features. You can deploy instances that use user SSH keys for authentication. You can use storage features to secure block, file, and object storage.
- **Oracle Cloud Infrastructure Identity and Access Management (IAM) service:** This is where you configure compartments, and policies to control who has access to your infrastructure-based resources.

The IAM service handles authentication – identifies users through confidential information such as username and password, or shared keys. IAM also handles authorization – users can only access the resources with the level of access that has been given to them.

> **Note:**
>
> For Compute Cloud@Customer, IAM resources are managed in OCI within your tenancy, and synchronized to Compute Cloud@Customer approximately every ten minutes. IAM resources can't be managed on the Compute Cloud@Customer infrastructure.

For information about managing IAM, see IAM with Identity Domains.

When configured, the preceding security areas enable the following secure environments:

- **Survivability of Mission-Critical Workloads:** Compute Cloud@Customer prevents or minimizes the damage caused from accidental and malicious actions taken by internal users or external parties. This is accomplished by security testing of components, checking protocols for vulnerabilities, and verifying software continuity even during security breaches.
- **Defense in Depth to Secure the Operating Environment:** Compute Cloud@Customer employs multiple, independent, and mutually-reinforcing security controls to help organizations create a secure operating environment for their workloads and data. All levels of the system are protected by an array of security capabilities.
- **Least-Privilege Access for Services and Users:** Compute Cloud@Customer promotes the use of security policies that ensure that applications, services, and users have access to the capabilities that they need to perform their tasks. However, it's equally important to ensure that access to unnecessary capabilities, services, and interfaces are limited. Users and administrators are confined to their particular areas of concern.
- **Accountability of Events and Actions:** Compute Cloud@Customer offers detailed audit trails at each layer and controls to help account for resources. This helps an administrator detect and report incidents as they occur (such as a denial of service attack) or after they occurred if it wasn't preventable (through traceability through audit logs to resulting changes to resources).
- **Accounting:** Accounting lets administrators track inventories of hardware and cloud resources. From the Oracle Cloud Console, an administrator can retrieve the Compute Cloud@Customer rack serial number.

Security Responsibilities

Security in Compute Cloud@Customer is a shared responsibility between you and Oracle. We use best-in-class security technology and operational processes to secure our cloud services. However, for you to securely run your workloads in OCI and Compute Cloud@Customer, you must know your security and compliance responsibilities.

In general, Oracle provides security of cloud infrastructure and operations, such as cloud operator access controls and infrastructure security patching. You are responsible for securely configuring your cloud resources.

Oracle's Responsibilities

- **Physical security:** Oracle is responsible for protecting the global infrastructure of Oracle Cloud Infrastructure to which Compute Cloud@Customer is connected. Oracle is also responsible for ensuring the secure delivery, installation, and maintenance of the Compute Cloud@Customer rack that's installed in your data center.
- **Tamper indicators:** Oracle is responsible for applying, maintaining, and tracking serialized tamper labels that are applied to all rack access panels and doors. A broken, missing, or damaged label provides a visual indication of a potential hardware breach.
- **Software Upgrades:** Oracle is responsible for creating upgrades that include bug fixes and feature enhancements. Oracle is also responsible for upgrading Compute Cloud@Customer on regular intervals, during upgrade schedule windows of your choosing.

Your Responsibilities

Your security responsibilities include the following areas:

- **Physical Security:** While Oracle is responsible for the installation and maintenance of the rack, you are responsible for the physical security of the Compute Cloud@Customer rack in your data center. You must ensure that nobody, other than authorized Oracle personnel, access or open the Compute Cloud@Customer rack.
  - Provide a location for the rack that is in a locked, restricted-access room.
  - Store spare field-replaceable units (FRUs) or customer-replaceable units (CRUs) in a locked cabinet. Restrict access to the locked cabinet to authorized personnel.
  - Limit SSH listener ports to the management and private networks. Use SSH protocol 2 (SSH-2) and FIPS 140-2 approved ciphers.
  - Limit SSH allowed authentication mechanisms. Inherently insecure methods are disabled.
  - Keep hardware activation keys and licenses in a secure location that's easily accessible to the system managers in the case of a system emergency.
- **Cloud resources:** You are responsible for securing your workloads and securely configuring your cloud resources (such as compute, network, storage, and database).You are responsible for actions in your instances, and day-to-day management of databases and applications that run on your instances.
- **Access Control:** Limit privileges as much as possible. Users must only be granted the access necessary to perform their work.
- **Encryption and Confidentiality:** Use encryption keys and secrets to protect your data and connect to secured resources. Rotate these keys regularly.
- **Information and Data:** You always retain control over information and data. You control how and when this data is used. The cloud provider (Oracle) has zero visibility into customer data, and all data access is under your control by design.
- **Application Logic and Code:** Regardless of how cloud resources are spun up, you secure and control the your proprietary applications during the entire application life cycle. This includes securing code repositories from malicious misuse or intrusion, application build testing during the development and integration process, ensuring secure production access, and maintaining the security of any connected systems.
- **Identity and Access:** You are always responsible for all aspects of identity and access management (IAM). This includes authentication and authorization mechanisms, any single sign-on (SSO) access, multifactor authentication (MFA), access keys, certificates, the user creation processes, and password management.
- **Platform and Resource Configuration:** When cloud resources are created, you control the operating environment. How control is maintained over those environments varies, based on whether instances are server-based or serverless (PaaS). A server-based instance requires more hands-on control over security, including OS and application hardening, maintaining OS and application patches, and so on. Server-based instances in the cloud behave like physical servers, and function as an extension of your data center. For serverless resources, the provider’s control plane gives you access to the setup of the configuration. In all cases, you are responsible for knowing how to configure customer instances in a secure manner.

  Additionally, you maintain responsibility for securing everything in your organization that connects with the cloud. This includes:

  - The on-premises infrastructure stack and user devices.
  - Customer-owned networks and applications.
  - The communication layers that connect your users, both internal and external, to the cloud and to each other.

  You also need to set up monitoring and alerting for security threats, incidents, and responses for domains that remain under customer control.
- **Software Upgrades:** You are responsible for creating upgrade schedules that enable Oracle to upgrade Compute Cloud@Customer in time frames that are least disruptive to your operations.

Initial Security Tasks

Use this section to identify initial tasks to secure Compute Cloud@Customer in an Oracle Cloud Infrastructure tenancy.

Task	More Information
Secure the rack in your data center.	Guidelines for Securing the Hardware on page 84

Task	More Information
Secure your tenancy in Oracle Cloud Infrastructure.	Securing Your Tenancy
Secure the network.	Guidelines for Securing the Network on page 84 Network Port and Protocol Matrix on page 85
In Oracle Cloud Infrastructure, secure IAM services (compartments, users, and groups). This includes using IAM federation to manage Console sign ins.	Securing IAM

Guidelines for Securing the Hardware

After installation of Compute Cloud@Customer, secure the hardware by restricting access to the hardware. You must ensure that nobody, other than authorized Oracle personnel, access or open the Compute Cloud@Customer rack.

We recommend the following practices to restrict access:

- Have the Compute Cloud@Customer and related equipment installed in a locked, restricted-access room.
- Store any site-located spare parts in a locked cabinet. Restrict access to the locked cabinet to authorized personnel.
- Limit SSH listener ports to the management and private networks. Use SSH protocol 2 (SSH-2) and FIPS 140-2 approved ciphers.
- Limit SSH allowed authentication mechanisms. Inherently insecure methods are disabled.

Guidelines for Securing the Network

During initialization, the core network components are integrated with your existing data center network design. Uplink ports in the Compute Cloud@Customer switches connect to your next-level data center switches to provide a redundant high-speed and high-bandwidth physical connection that carries all traffic into and out of Compute Cloud@Customer.

This environment has several consequences when it comes to securing Compute Cloud@Customer networking:

- The rack us set up inside your data center, and connected directly to your on-premises network. There is no need for a secure tunnel over the internet to allow your cloud resources and your on-premises network to communicate. Access is enabled through a gateway between your virtual cloud network (VCN) and your on-premises network.
- When it comes to internet access, inbound or outbound, resources in your cloud environment have no direct internet access. In contrast with a public cloud environment, no gateway is capable of enabling direct internet connectivity for a VCN. The configuration of the networking components in the data center determines how cloud resources can connect to the internet and whether they can be reached from the internet.
- Generally, this environment means that the Compute Cloud@Customer virtual network is well-isolated from public internet threats. However, It's possible to use public IP addresses inside this private network. A public IP makes a resource reachable from outside the VCN it resides in. The IP addresses that are considered "public" are really part of the data center's private range.

Nevertheless, there are steps you can take to control cloud network security and access to compute instances:

- Use private subnets if instances don't require a public IP address.
- Configure firewall rules on the instance to control traffic into and out of an instance at the packet level. However, Compute Cloud@Customer-provided images that run Oracle Linux automatically include default rules that allow ingress on TCP port 22 for SSH traffic. In addition, the Microsoft Windows images include default rules that allow ingress on TCP port 3389 for Remote Desktop access.
- Configure gateways and route tables to allow only required connectivity. This can control traffic flow to "outside" destinations such as your on-premises network or another VCN.
- Use IAM policies to control access to Compute Cloud@Customer interfaces. You can control which cloud resources can be accessed and which type of access is allowed. For example, you can control who can set up your network and subnets, or who can update route tables, network security groups, or security lists.

For more network security information, see Network Security Guidelines on page 89.

Network Port and Protocol Matrix

Compute Cloud@Customer requires access permissions to be granted for certain IP addresses, ports, and protocols.

The default security posture for almost all firewalls is to deny access. This applies to the firewalls used between the Compute Cloud@Customer rack and the customer data center.

To allow certain Compute Cloud@Customer features to operate correctly, access must be granted for certain IP address and related services. An "allow all" rule such as `0.0.0.0/0` is too broad for security purposes, so the best practice is to explicitly list addresses, ports, and protocols to allow.

Compute Cloud@Customer is installed with connections to different networks for different purposes (see Customer Site Network Requirements on page 48). For security purposes, Compute Cloud@Customer isolates the administration network from the customer data network.

During the Compute Cloud@Customer installation, Oracle configures the isolated networks, and works with your network administrator to configure the network ports so they work within your environment.

The following table lists the access permissions for certain IP addresses, ports, and protocols that are granted for data center and admin network isolation.

Table key:

- Customer – Customer administrator access for Compute Cloud@Customer resource management
- Oracle – Oracle administrator access, which is only accessible by Oracle when granted access by the customer using Oracle Operator Access Control on page 29.

Source IP Address	Destination IP Address	Port	Protocol	Description
All customer networks	Customer VIP		ICMP	Type 0/Echo reply
Oracle administrators	Oracle VIP		ICMP	Type 0/Echo reply
All customer networks	Management Nodes		ICMP	Type 0/Echo reply
All customer networks	Object Storage IP		ICMP	Type 0/Echo reply
All customer networks	Customer VIP		ICMP	Type 3/Unreachable
Oracle administrators	Oracle VIP		ICMP	Type 3/Unreachable
All customer networks	Management Nodes		ICMP	Type 3/Unreachable
All customer networks	Object Storage IP		ICMP	Type 3/Unreachable
All customer networks	Customer VIP		ICMP	Type 8/ping to VIP
Oracle administrators	Oracle VIP		ICMP	Type 8/ping to VIP
Oracle administrators	Management Nodes		ICMP	Type 8/ping to node
All customer networks	Object Storage IP		ICMP	Type 8/ping to VIP
Oracle administrators	Oracle VIP	22	TCP	SSH to Active Management Node
Oracle administrators	Management Nodes	22	TCP	SSH to Specific Management Node

Source IP Address	Destination IP Address	Port	Protocol	Description
Initial installation DNS IPs	Customer VIP	53	UDP	Authoritative Zones
Initial installation DNS IPs	Customer VIP	53	TCP	Authoritative Zones
Initial installation DNS IPs	Oracle VIP	53	UDP	Administrative Zone
Initial installation AdminDNS IPs	Oracle VIP	53	TCP	Administrative Zone
Management Nodes	Initial installation DNS IPs	53	UDP	External DNS Resolution for Data Network
Management Nodes	Initial installation DNS IPs	53	TCP	External DNS Resolution for Data Network
Management Nodes	Initial installation AdminDNS IPs	53	UDP	External DNS Resolution for Admin Network
Management Nodes	Initial installation AdminDNS IPs	53	TCP	External DNS Resolution for Admin Network
Oracle administrators	Oracle VIP	443	TCP	Oracle API Endpoints and BUI
All Compute Cloud@Customer users	Customer VIP	443	TCP	Compute Cloud@Customer API Endpoints and BUI
All Compute Cloud@Customer users	Customer VIP	8079	TCP	Image download repository
Oracle administrators	Oracle VIP	30006	TCP	Admin CLI
Oracle administrators	Management Nodes	30006	TCP	Admin CLI
Customer VIP	DNS Recursive Servers	53	UDP	DNS Forwarding
Customer VIP	DNS Recursive Servers	53	TCP	DNS Forwarding
Oracle VIP	DNS Recursive Servers	53	UDP	DNS Forwarding
Oracle VIP	DNS Recursive Servers	53	TCP	DNS Forwarding
Oracle VIP	Customer NTP Servers	123	UDP	NTP
Oracle VIP	Usually *transport.oracle.com*	443	TCP	ASR Targets
Oracle VIP	Oracle Grafana Notification Targets	443	TCP	Grafana Notification Targets

Source IP Address	Destination IP Address	Port	Protocol	Description
Oracle VIP	Oracle Local ULN Mirror	443	TCP	patching
Customer VIP	Local image repository	443	TCP	Custom image import from-object-uri
Management Nodes	Load balancer public IP address	6443	TCP	Load balancer public IP address endpoint
All customer networks	Instance console	1443	TCP	Connecting to a Compute Instance on page 341

Routine Security Tasks

Conducting routine security tasks keeps Compute Cloud@Customer safe from breaches and unauthorized access, and provides a means of recovery for unplanned data loss.

Perform the following tasks regularly:

- Take Regular Backups on page 87
- Ensure the System is Upgraded on page 88
- Monitor your Identity and Access Management (IAM) configuration and establish authentication best practices. See Securing IAM.

Take Regular Backups

The following table lists the Compute Cloud@Customer backup and restore features that you can use to ensure data durability.

Backup and Restore Features	Links
Compute Instances You can create backups of your instances, and use the backups to restore or create new 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 Object Storage bucket, and use it to create instances.	Backing Up and Restoring an Instance on page 344
Block Volumes You can make a backup of a volume when the volume is attached to an instance or while it is detached. You can take manual or create scheduled backups.	Backing Up Block Volumes on page 422
File Storage You can create snapshots which are a consistent, point-in-time view of your file systems. You can create file system clones, which are a new file systems that are created based on a snapshot of an existing file system.	Managing File System Snapshots on page 468 Managing File System Clones on page 471

Backup and Restore Features	Links
Object Storage You can use Object versioning, which provides data protection against accidental or malicious object updates, overwrites, or deletions. You can use Object retention rules, which 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.	Managing Object Versioning on page 492

Ensure the System is Upgraded

Oracle performs all Compute Cloud@Customer upgrades and patching.

You can determine when the upgrades and patches are applied using these:

- Configuring upgrade schedules. If you don't create an upgrade schedule, Oracle can upgrade Compute Cloud@Customer at any time. See Managing Upgrade Schedules on page 94.
- Enabling Oracle personnel to access the Compute Cloud@Customer infrastructure. See Oracle Operator Access Control on page 29.

IAM Policies

Oracle Cloud Infrastructure Identity and Access Management (IAM) provides authentication of users, and authorization to access resources on Compute Cloud@Customer.

> **Note:**
>
> For Compute Cloud@Customer, IAM resources are managed in OCI within your tenancy, and synchronized to Compute Cloud@Customer approximately every ten minutes. IAM resources can't be managed on the Compute Cloud@Customer infrastructure.

For detailed recommendations for securing IAM resources, See Securing IAM.

For IAM policies that are unique to Compute Cloud@Customer, see Compute Cloud@Customer Policy Reference on page 110.

Limit Access to Compute Cloud@Customer by Assigning it to its own Compartment

Consider assigning each Compute Cloud@Customer infrastructure to its own compartment. Limit compartment access to only the users who are supposed to have access to the Compute Cloud@Customer infrastructure.

Securing Compute Cloud@Customer Resources

Compute Cloud@Customer resources are the virtual cloud networking, compute instances, and storage for which your workloads reside. Securing these resources secures your workloads.

You secure Compute Cloud@Customer resources the same way you secure resources in Oracle Cloud Infrastructure, with a slight, but important difference:

- **IAM resources**: When you create users, groups, compartments, and policies to secure resources, you configure the IAM resources in your tenancy in Oracle Cloud Infrastructure.

  The IAM resources are automatically synchronized to Compute Cloud@Customer, but IAM can't be changed in Compute Cloud@Customer.

For information about managing IAM, see IAM with Identity Domains.

- **Resource security features**: When you use security features that are part of the resource, for example, network security groups for securing networking resources, you configure the resource feature on Compute Cloud@Customer.

  The following table lists the resources that have security features that you can use to secure the resource.

Resources with Security Features	For More Information
Networking: VCNs and DNS	Securing Networking: VCN, Load Balancers, and DNS
Load balancers	LBaaS Security on page 90
Compute instances	Securing Compute
Block volumes	Securing Block Volume
File storage	Securing File Storage
Object storage	Object Storage Security Guidelines on page 92

Network Security Guidelines

Secure network access to your resources in Compute Cloud@Customer.

For comprehensive information about security networking, see Securing Networking: VCN, Load Balancers, and DNS.

Use VCN Security Features

Use security lists, network security groups, or a combination of both to control packet-level traffic in and out of the resources in your VCN. See Configuring VCN Rules and Options on page 145.

When you create a subnet in a VCN, by default the subnet is considered public and internet communication is permitted. Use private subnets to host resources that don't require internet access. You can also configure a service gateway in your VCN to allow resources on a private subnet to access other cloud services. See Accessing Oracle Services through a Service Gateway on page 175.

Compute Cloud@Customer has a collection of features for enforcing network access control and securing VCN traffic. These features are listed in the following table:

VCN Feature	Security Description
VCNs and Subnets	A VCN can be partitioned into subnets. Instances inside private subnets can't have public IP addresses. Instances inside public subnets can optionally have public IP addresses at your discretion. See Managing VCNs and Subnets on page 140
Security Rules	Security rules provide stateful and stateless firewall capability to control network access to your instances. To implement security rules in your VCN, you can use Network Security Groups or Security Lists. For more information, see: • Comparison of Security Lists and Network Security Groups • Configuring VCN Rules and Options on page 145

VCN Feature	Security Description
Gateways	Gateways let resources in a VCN communicate with destinations outside the VCN. The gateways include: • Providing Public Access through an Internet Gateway on page 170: For internet connectivity (for resources with public IP addresses in public subnets). • Enabling Public Connections through a NAT Gateway on page 168: For internet connectivity without exposing the resources to incoming internet connections (for resources in private subnets). • Connecting to the On-Premises Network through a Dynamic Routing Gateway (DRG) on page 174: For connectivity to your data center network. • Accessing Oracle Services through a Service Gateway on page 175: For private connectivity to Oracle services such as Object Storage. • Local VCN Peering using Local Peering Gateways: For connectivity to a peered VCN in the same region.
VCN Route Tables	Route tables control how traffic is routed from your VCN subnets to destinations outside the VCN. Routing targets can be VCN gateways or a private IP address in the VCN. See Working with Route Tables on page 155.
IAM Policies for Networking	IAM policies specify access and actions permitted by IAM groups to resources in a VCN. For example, IAM polices can give administrative privileges to network administrators who manage the VCNs, and scoped-down permissions to normal users. **Attention:** For Compute Cloud@Customer, IAM resources are managed within OCI, and synchronized to Compute Cloud@Customer. IAM resources can't be managed on the Compute Cloud@Customer infrastructure. See IAM Policies for Networking

LBaaS Security

On Compute Cloud@Customer, you can configure and secure the Load Balancer as a Service (LBaaS).

A load balancer (LB) connects a client's applications and a customer's VCN. LBs use TLS1.2 by default. You can configure LBs to use Compute Cloud@Customer public or private IP addresses.

Compute Cloud@Customer LBs use TLS and SSL for security. You can create LBs with one of three different types of connections:

- **SSL Termination:** The load balancer handles incoming SSL traffic and passes the unencrypted request to a backend server.
- **End-to-End (Point-to-Point) SSL:** The load balancer ends the SSL connection from an incoming traffic client, and then initiates another SSL connection to a backend server.
- **SSL Tunneling:** If the load balancer listener is configured for TCP traffic, the load balancer tunnels incoming SSL connections to the application servers.

You can configure LBs with two types of protocols:

- **HTTP:** LBs using HTTP or HTTPS end the TLS connection at the LB itself.
- **TCP:** LBs using TCP end the TLS connection at a backend server.

LBs listen to an IP address for these types of protocols.

For information about managing the LBaaS service, see Load Balancer as a Service on page 211.

SSL Certificates

On Compute Cloud@Customer, LBaaS requires the use of secure certificates. You must supply a certificate to use standard SSL for the LBs and their resources. You can upload SSL certificates if none are available.

> **Note:**
>
> We recommend that you upload the certificates you want to use before you create the listeners or backend sets you want to associate them with.

Compute Cloud@Customer LBaaS doesn't generate SSL certificates. It can only import an existing certificate that you already own. The certificate can be one issued by a vendor, such as Verisign or GoDaddy. You can also use a self-signed certificate that you generate with an open source tool, such as OpenSSL or Let's Encrypt. Refer to the corresponding tool's documentation for instructions on how to generate a self-signed certificate.

If you submit a self-signed certificate for backend SSL, you must submit the same certificate in the corresponding CA Certificate field.

Compute Cloud@Customer accepts x.509 type certificates in PEM format only. The following is an example PEM encoded certificate:

```
-----BEGIN CERTIFICATE-----
<Base64_encoded_certificate>
-----END CERTIFICATE-----
```

For more information about certificates, see Managing SSL Certificates on page 236.

Security Lists

On Compute Cloud@Customer, you use VCN network security groups (NSGs) or security lists to control network access to load balancers. This method provides the Compute Cloud@Customer with functionality similar to traditional LB firewalls. You configure the load balancer rules by configuring the NSGs or the subnet security lists for the LBs.

When you create a listener, you must also update the VCN security list to permit traffic to that LB listener.

The ingress rule for the security list must be edited to permit the following:

- Source CIDR (Enter a block that covers the networks using the LB. Can be 0.0.0.0/0.)
- Protocol TCP
- Destination port range 80 (the listener port)

If you created other listeners, then you must add an ingress rule for each listener port to permit traffic to reach the listener. For example, if you created a listener on port 443, you must permit traffic for port 443.

For more information about NSGs and security lists, see Controlling Traffic with Security Lists on page 158 and Controlling Traffic with Network Security Groups on page 163.

Backend Server Security

For public LBs on Compute Cloud@Customer, you must configure the VCN NSGs or security lists for the backend servers to accept traffic only for the public LBs. You can update the security lists used by the backend server subnets to permit ingress traffic from the LB subnet.

For example, if the LB is in subnet 10.0.4.0/24 and you're balancing traffic for web servers, then you must add a stateful ingress rule to permit traffic from source IP address 10.0.4.0/24, using the TCP protocol, from all source ports, and to destination port 80. This new rule permits ingress traffic from the LB subnet.

VCN subnet Security Lists (or NSGs, if used) can be associated with an NSG and need to be updated after an LB, Listener and backends have been created. Typically a Load balancer would be created in a different subnet than the backend servers. For example, a public LB created in a public subnet forwards traffic to backend servers in a private subnet.

However (though not recommended), the LB could be created in the same subnet as a backend server. In either case, the security lists for all subnets that involve the LB and backend servers need to be updated.

For the LB subnet, the Security List (or NSG) must be updated to permit egress traffic from the load balancer to each backend server's subnet. For example if the backend servers are in Subnet1 (10.0.1.0/24) and Subnet2 (10.0.0.0/24), then updates to the security list for the LB subnet would be:

- Allow egress traffic to the backend server on Subnet1
- Allow egress traffic to the backend server on Subnet2

In addition to the updates to permit egress traffic to the backend subnets, updates must be made to permit the listener to accept traffic. For example, if a public LB permits traffic from anywhere to reach the LB on port 80, you must add the following ingress rule to the subnet that hosts the LB:

- Source Type: CIDR
- Source CIDR: 0.0.0.0/0
- IP Protocol: TCP
- Destination Port Range: 80 (listener port)

If the LB is in subnet 10.0.4.0/24, then stateful rule updates to the security lists used by the backend server subnets are needed to permit ingress traffic from the load balancer subnet:

- Source Type: CIDR
- Source CIDR: 10.0.4.0/24
- IP Protocol: TCP
- Destination Port Range: 80 (listener port)

This new stateful ingress rule permits TCP traffic to reach the backend servers. The stateful nature permits responses.

Finally, delete all egress rules. There can be no egress rules for the backend servers.

Route Table Considerations

On Compute Cloud@Customer, the VCN route table must route requests to the various backend sets of the LB. You can assign a virtual hostname to a listener or create route rules for this purpose.

For more information on these LB topics, see:

- Managing Virtual Hostnames on page 246
- Managing Listeners on page 253
- Managing Backend Sets on page 238

For more information on route table configuration in general, see Working with Route Tables on page 155.

Object Storage Security Guidelines

On Compute Cloud@Customer, the Object Storage service stores and provides access to large amounts of unstructured data of any content type. Content stored in object storage can be accessed by compute instances and from the data center through the Compute Cloud@Customer domain name.

The Object Storage service provides these security capabilities:

- Encrypts data on disk (at rest) using Advanced Encryption Standard (AES) 128-bit algorithm. Encryption is on by default and can't be turned off. The encryption keys are themselves encrypted with a master encryption key.
- Data in transit between clients (for example, SDKs and CLIs) and Object Storage public endpoints is encrypted with TLS 1.2 by default.
- Data integrity is facilitated by a checksum generated for each uploaded object and, for multipart uploads, a checksum generated for each part.

- Objects can have revokable preauthenticated requests built with varying authorization levels.
- Object versioning is available to retain changes to objects.
- Write-once Read-many (WORM) retention rules can be set to ensure an object retains its original contents.
- Users that are authenticated to Compute Cloud@Customer can manage the Object Storage service and resources that are managed within the security policy framework.

For more information about the Object Storage service, see Object Storage on page 478.

Assign Least Privileged Access

> **Note:**
>
> For Compute Cloud@Customer, IAM resources are managed in OCI within your tenancy, and synchronized to Compute Cloud@Customer approximately every ten minutes. IAM resources can't be managed on the Compute Cloud@Customer infrastructure.

Assign least privileged access for users and groups to resource types in *object-family* (such as objectstorage-namespaces, buckets, and objects).

The following IAM verbs enable you to tailor Object Storage access privileges:

- `inspect` verb gives the least privilege. It lets you check to see if a bucket exists (`HeadBucket`) and list the buckets in a compartment (`ListBucket`).
- `manage` verb gives all permissions on the resource. You can create IAM security policies to give appropriate bucket and object access to various IAM groups.

For more information about IAM verbs and permissions for Object Storage buckets and objects, see Details for Object Storage, Archive Storage, and Data Transfer.

For users without IAM credentials, we recommend that you use preauthenticated requests (PARs) to give time-bound access to objects or buckets.

Preauthenticated Requests

Preauthenticated requests provide a mechanism to give access to objects and buckets without requiring the client or user to log into Compute Cloud@Customer. As a result, objects and buckets can be used without an user defined on the Compute Cloud@Customer.

An user with appropriate privileges for accessing objects creates a URL that grants time-bound access to objects. For more information on these grants, see Using Preauthenticated Requests on page 495. Note the following points:

- The creator of a preauthenticated request must have `PAR_MANAGE` permission and the appropriate IAM permissions for the access type that you're granting. You can create a preauthenticated request that grants read, write, or read/write access to one of the following:
  - All objects in the bucket.
  - A specific object in the bucket.
  - All objects in the bucket that have a specified prefix.

  For requests that apply to multiple objects, you can also decide whether you want to let users list those objects.
- Preauthenticated request accesses to a bucket are logged in Audit logs. Preauthenticated request accesses to an object are logged in Service logs.

> **Important:**
>
> The unique URL provided by the system when you create a preauthenticated request is the only way a user can access the request target. Copy the URL to durable storage. The URL is displayed only at the time of creation, isn't stored in Object Storage, and can't be retrieved later.

Data Durability and Integrity

The Object Storage service provides a variety of ways to ensure data remains consistent and intact when stored. Be aware of the following points:

- Data loss can be minimized by preventing inadvertent deletes by an authorized user or malicious deletes:
  - Use Object Versioning to automatically create an object version each time a new object is uploaded, an existing object is overwritten, or when an object is deleted.
  - Give `BUCKET_DELETE` and `OBJECT_DELETE` permissions to a minimum set of users and groups. Grant delete permissions only to tenancy and compartment administrators.
- Write once read many (WORM) compliance requires that objects can't be deleted or modified. Use retention rules to achieve WORM compliance. Retention rules are configured at the bucket level and are applied to all individual objects in the bucket. You can't update, overwrite, or delete objects or object metadata until the retention rule is deleted (indefinite rule) or for the duration specified (time-bound rules).

  For an independent assessment of the Object Storage retention rules feature's ability to meet regulatory requirements for record management and retention, see Cohasset Associate's SEC 17a-4(f), FINRA 4511(c), CFTC 1.31(c)-(d) and MiFID II Compliance Assessment PDF document.
- Objects are stored in a ZFS file system with a SHA-256 checksum to avoid phantom reads and detect invalid data returned from devices.

In addition to facilities provided for durability, Object Storage provides mechanisms to ensure data integrity:

- A checksum is provided for all objects uploaded to Object Storage. The checksum can be used in two ways:
  - To verify that the object stored is the object that was uploaded
  - To verify that the object retrieved is the one that was originally stored
- Multipart uploads are used by the Object Storage service for efficiency and resiliency on large object uploads. In a multipart upload, a large object is broken up into smaller parts by specifying a part size in MiB. Each part is uploaded separately. Object Storage then combines all the parts to re-create the original object. If any of the parts fail to upload, only the failed parts need to be retried for upload, not the entire object. In a multipart upload, checksum values are computed for each part of the upload, and a single checksum is computed over all the individual checksum values to get the checksum value reported by the object upload. To verify the value returned for a multipart upload, follow the same process for offline checksum calculation.

Using the checksum returned when the object is stored, the contents of the object can be verified when it's downloaded again or against the original object to verify uploaded object correctness.

OSs have various tools for checksum verification. The checksum returned by an object upload (or viewing an object in the bucket) is a Base-64 encoded value. This value might have to be converted to hexadecimal depending on the tools used for checksum comparison.

Managing Upgrade Schedules

Use Upgrade Schedules to define time periods when a Compute Cloud@Customer infrastructure may be upgraded by Oracle.

Upgrade schedules consist of upgrade windows, also known as events. Use separate upgrade windows to make defining the upgrade schedule easier. An upgrade window can only be used in one upgrade schedule.

When an upgrade is available, Oracle schedules it during a suitable upgrade window.

To receive email notifications when an upgrade starts and finishes, subscribe to the Compute Cloud@Customer notification topics. See Subscribing to Upgrade Notifications on page 96.

Creating a Compute Cloud@Customer Upgrade Schedule

Create an upgrade schedule to allow Oracle to upgrade Compute Cloud@Customer hardware and software during defined time periods. After the upgrade schedule is created, attach the schedule to the infrastructures you want upgraded with the schedule.

You must set at least one 48 hour window every quarter for upgrades. This means that at least one of the upgrade windows must be at least 48 hours in duration with a recurrence pattern of every 3rd month or shorter. You can also define shorter windows, either recurring or one-off, to accommodate smaller upgrades at convenient times. We recommend having several upgrade windows per calendar quarter.

If you don't set an upgrade schedule, Oracle can upgrade the infrastructure at any time.

All times in Compute Cloud@Customer are in UTC.

Using the Console

1. Navigate to Upgrade Schedules:
   a. In the Oracle Cloud Console, open the navigation menu, select **Hybrid & Multicloud**, and then select **Oracle Compute Cloud@Customer**.
   b. In the left panel, select **Oracle Compute Cloud@Customer**, then select **Upgrade Schedules**.
2. Select the compartment that you want to create the upgrade schedule in.
3. Select **Create upgrade schedule**.
4. Enter the following information:
   a. **Name:** Enter a name for the schedule so it can be assigned to one or more infrastructures. Avoid entering confidential information.
   b. **Description:** Enter an optional description. Avoid entering confidential information.
   c. **Compartment:** Select the compartment you want to create the schedule in.
5. Under **Upgrade windows:** Define the following upgrade window parameters. These are time periods during which Oracle can upgrade the infrastructure. You're likely to need several upgrade windows in a schedule.
   a. Select **Add upgrade window**.
   b. **Description:** enter a description to identify the upgrade window. Avoid entering confidential information.
   c. **Start time:** Select the calendar icon and select the start date and time for the window.
   d. **Duration:** Enter the duration in days and hours.
   e. (Optional) To repeat the time window on a regular schedule, select **Recurring window**. Configure the recurrence using one of the following methods:
      - Select **Build recurrence** to use the recurrence builder.
      - Manually enter an RFC 5545 recurrence pattern in the **Enter recurrence** box.

      Either way, you need the unit of frequency, repeat cycle, and calculation of day. The start time and duration set in the previous steps is used for each recurrence. For example, a window that recurs on the first Monday of every month would be defined as FREQ=MONTHLY;INTERVAL=1;BYDAY=MO; For more examples, see Upgrade Window Recurrence Patterns on page 99.
   f. If you created a recurrence using the recurrence builder, select **Build**.
   g. Select **Create new upgrade window**.
6. (Optional) To add tags, select **Show advanced options**.

   If you have permissions to create a resource, then you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags.
7. Select **Create upgrade schedule**.
8. Attach the upgrade schedule to the infrastructure:
   a. Select **Infrastructures**. The Infrastructures list page is displayed.
   b. On the infrastructure line, select the Actions menu (•••), and select **Edit**.
   c. Under **Upgrade schedule**, use the drop-down menu to select the upgrade schedule you created.

d. Select **Save**.

What's Next?

Subscribe to upgrade notifications. See Subscribing to Upgrade Notifications on page 96.

Using the CLI

Use the ccc upgrade-schedule create command and required parameters to create an upgrade schedule:

Create each upgrade window as a separate *CreateCccScheduleEvent* item, and supply these as a list with the `--events` option.

```
oci ccc upgrade-schedule create --compartment-id <compartment ocid> --
display-name <Descriptive name> --events <List of upgrade windows, of type
 CreateCccScheduleEvent>... [OPTIONS]
```

Avoid entering confidential information.

After the upgrade schedule is created, attach the schedule to the infrastructures you want associated with this schedule. To attach a schedule, use the `oci ccc infrastructure update` command with the `--upgrade-schedule-id` option. For more information, see the ccc infrastructure update command reference page.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

What's Next?

Subscribe to upgrade notifications. See Subscribing to Upgrade Notifications on page 96.

Using the API

Use the CreateCccUpgradeSchedule operation to create a Compute Cloud@Customer upgrade schedule.

Avoid entering confidential information.

After the upgrade schedule is created, attach the schedule to the infrastructures you want associated with this schedule. To attach a schedule, use the UpdateCccInfrastructure API operation.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

What's Next?

Subscribe to upgrade notifications. See Subscribing to Upgrade Notifications on page 96.

Subscribing to Upgrade Notifications

To receive email notifications when an upgrade starts and is finished, perform this procedure. The procedure involves creating a notification topic, then subscribing to the topic with an email address.

To use the Notification service for other purposes, see Notifications.

> **Important:**
>
> Create only one topic called `ComputeCloudCustomer-InfraWatch`, and at least one subscription to that topic.

Prerequisite

You must have notification policies configured as described in Add Required Policies on page 69.

1. Open the Oracle Cloud Console navigation menu and select **Developer Services**. Under **Application Integration**, select **Notifications**.
2. On the **Topics** page, select the Compute Cloud@Customer infrastructure root compartment.
3. Select **Create Topic**.
4. In the **Create Topic** panel, enter the following information:

- **Name:** Enter the following string exactly as shown. The string is case-sensitive:

```
ComputeCloudCustomer-InfraWatch
```

- **Description:** Enter a description such as "Upgrade notification topic." Avoid entering confidential information.

> **Note:**
>
> Ignore the message about creating an identity policy. The policies for this topic were created when the tenancy was prepared. See Add Required Policies on page 69.

5. Select **Create**.
6. Select the topic named **ComputeCloudCustomer-InfraWatch**.

   The Topic Details page is displayed.
7. Select **Create Subscription**.
8. In the **Create Subscription** panel, enter the following information:
   - **Protocol:** Select Email.
   - **Email**: Enter an email address.
9. Select **Create**.

   The Notifications service creates the email subscription and sends a confirmation URL to the email address. The subscription is pending until confirmation is received.

What's Next?

Learn about typical postinstallation administration. See Postinstallation Administration on page 80.

.

Editing a Compute Cloud@Customer Upgrade Schedule

Edit an upgrade schedule to change the time periods during which Oracle may upgrade Compute Cloud@Customer hardware and software.

> **Tip:**
>
> If the next upgrade needs to be delayed until after the next scheduled upgrade window, edit your upgrade schedule accordingly. Change the upgrade window to start at the date and time you want the next upgrade to occur. Recurring upgrade windows are scheduled using the start date and time as a reference point: the recurrence settings determine on which days or dates *after* the start date an upgrade window opens, at the specified start time.

Using the Console

1. Navigate to Upgrade Schedules:
   a. In the Oracle Cloud Console, open the navigation menu, select **Hybrid & Multicloud**, and then select **Oracle Compute Cloud@Customer**.
   b. In the left panel, select **Oracle Compute Cloud@Customer**, then select **Upgrade Schedules**.
2. For the upgrade schedule that you want to edit, select the Actions menu (•••), then select **Edit**.
3. Make the necessary changes. For a description of the fields, see Creating a Compute Cloud@Customer Upgrade Schedule on page 95.
4. Select **Update upgrade schedule**.

Using the CLI

Use the ccc upgrade-schedule update command and required parameters to update an existing upgrade schedule:

Supply a list of upgrade windows, each as a separate *UpdateCccScheduleEvent* item, as a list with the `--events` option.

```
oci ccc upgrade-schedule update --upgrade-schedule-id <upgrade schedule
 ocid> --display-name <Descriptive name> --events <List of upgrade windows,
 of type UpdateCccScheduleEvent>... [OPTIONS]
```

Avoid entering confidential information.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateCccUpgradeSchedule operation to update a Compute Cloud@Customer upgrade schedule.

Avoid entering confidential information.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Moving a Compute Cloud@Customer Upgrade Schedule Between Compartments

Move an existing Compute Cloud@Customer upgrade schedule to a different compartment within the same tenancy.

You must have the required permissions in both compartments. See Preparing Your Tenancy on page 67 for details.

Using the Console

1. Navigate to Upgrade Schedules:
   a. In the Oracle Cloud Console, open the navigation menu, select **Hybrid & Multicloud**, and then select **Oracle Compute Cloud@Customer**.
   b. In the left panel, select **Oracle Compute Cloud@Customer**, then select **Upgrade Schedules**.
2. For the upgrade schedule that you want to move, select the Actions menu (•••), then select **Move resource**.
3. Make the necessary changes. For a description of the fields, see Creating a Compute Cloud@Customer Upgrade Schedule on page 95.
4. Select the compartment to move the upgrade schedule to, and then select **Move resource**.

Using the CLI

Use the ccc upgrade-schedule change-compartment command and required parameters to move an upgrade schedule to a different compartment:

```
oci ccc upgrade-schedule change-compartment --compartment-id <compartment
 ocid of the new compartment> --upgrade-schedule-id <OCID of the upgrade
 schedule to be moved> ... [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ChangeCccUpgradeScheduleCompartment operation to move a Compute Cloud@Customer upgrade schedule to a different compartment in the tenancy.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Compute Cloud@Customer Upgrade Schedule

When you delete a schedule, all upgrade windows in the schedule are deleted.

> **Note:**
>
> You can't delete an upgrade schedule that's being used by an infrastructure. However, you can change an upgrade schedule to be an *any time* schedule. An any time schedule enables an upgrade to occur anytime, which provides the same functionality as not having an upgrade schedule. To create an any time schedule, see Editing a Compute Cloud@Customer Upgrade Schedule on page 97, and set the schedule to use an any time recurrence pattern as shown in Upgrade Window Recurrence Patterns on page 99.

Using the Console

1. Navigate to Upgrade Schedules:
   a. In the Oracle Cloud Console, open the navigation menu, select **Hybrid & Multicloud**, and then select **Oracle Compute Cloud@Customer**.
   b. In the left panel, select **Oracle Compute Cloud@Customer**, then select **Upgrade Schedules**.
2. For the upgrade schedule that you want to delete, select the Actions menu (•••), then select **Delete**.
3. Select **Delete**, and then select **Delete** to confirm.

Using the CLI

Use the ccc upgrade-schedule delete command and required parameters to delete an upgrade schedule and the upgrade windows associated with it:

```
oci ccc upgrade-schedule delete --upgrade-schedule-id <upgrade schedule
 ocid> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteCccUpgradeSchedule operation to delete a Compute Cloud@Customer upgrade schedule.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Upgrade Window Recurrence Patterns

Examples of RFC 5545 recurrence pattern definitions. Use these when setting upgrade windows.

> **Important:**
>
> Recurring upgrade windows are scheduled using the start date and time as a reference point. Upgrades don't occur on the specified start date and time. Rather, an upgrade window opens on a day or date calculated from the start date, based on the recurrence settings. That means a recurring upgrade window is scheduled on the selected days of the week or month *after* the start date, at the specified start time.
>
> For example, if a recurring window is configured to start on a Monday at 22:00 UTC with recurrence set to every other Friday, the next upgrade window doesn't open on Monday but on Friday at 22:00 UTC.

For more information on the format required, see RFC 5545.

Any time

The upgrade can occur anytime. This is the same functionality as not having an upgrade schedule.

```
FREQ=WEEKLY;BYDAY=SU,MO,TU,WE,TH,FRI,SA
```

Weekly, on Tuesdays and Fridays

```
FREQ=WEEKLY;BYDAY=TU,FR
```

Monthly, on the first Sunday of the month

```
FREQ=MONTHLY;BYDAY=1SU
```

Every third month on the last Wednesday of the month

```
FREQ=MONTHLY;INTERVAL=3;BYDAY=-1WE
```

Every 10 days

```
FREQ=DAILY;INTERVAL=10
```

Yearly, in April

The day is obtained from the first start date/time.

```
FREQ=YEARLY;BYMONTH=4
```

Disaster Recovery

The goal of disaster recovery (DR) is to provide high availability at the level of an installation site, and to protect critical workloads hosted on Compute Cloud@Customer against outages and data loss.

The current Compute Cloud@Customer controller software provides a Disaster Recovery service with orchestration of DR operations from within the Service Enclave, the infrastructure administration environment. The service is also called *native DR* because it is built directly into the infrastructure services layer.

Setting up disaster recovery requires that you open a support request (SR) with Oracle to establish a peer connection between systems, and provision an authorization group with the necessary access to all relevant operations in the highly restricted Service Enclave.

To open a Support Request. See Creating a Support Request. To access support, sign in to the Oracle Cloud Console as described in Sign In to the OCI Console.

Configuring the Disaster Recovery Service

Setting up disaster recovery requires initial setup by Oracle. Further management of the DR configuration and execution of DR plan operations is the responsibility of an authorized infrastructure administrator. The systems participating in the DR setup are fully operational environments on their own, running in different physical locations.

Peer Connection (Oracle)

A mutual peer connection must be established first, so the installations can operate as each other's standby or replica in case an outage occurs at one of the sites. It involves these steps:

1. Installing dedicated cabling at each site between the system's spine switches and the data center network.
2. Creating a local endpoint on each participating system. Traffic between peered systems flows through tunnels between endpoints.
3. Creating the peer connection on each participating system. Configuration parameters of each connected system must be included to complete the peering. When each system has accepted the connection from its peer, a mutual trust relationship is established and peering is complete.

Service Setup (Oracle)

The Disaster Recovery service is configured on top of an active peer connection. On each peered system the service is enabled with a single command that includes the serial number of its peer.

If the Disaster Recovery service needs to be disabled, the DR service setup must be deleted from each peered system.

DR Configurations (authorized administrator)

The Disaster Recovery service makes a clear distinction between resources and operations. *DR configurations* specify the resources that play a critical role in protecting Compute Cloud@Customer workloads against site-level incidents. A DR configuration contains compute instances under DR protection, and their required site mappings.

Site mappings determine how and where on the standby system the instances should be brought back up in case the primary system experiences an outage. They associate network resources and compartment hierarchies on the primary and standby systems with each other. Each site mapping consists of a source object – subnet or compartment – on the primary system, and a corresponding target object on the standby system.

Compute instances are added to the DR configuration after the site mappings. For each instance under DR protection, data and disks are stored in the ZFS storage project associated with the DR configuration, and replicated over the peer connection.

DR Plans (authorized administrator)

DR operations are defined in a *DR plan*, which outlines the steps to perform during a switchover, failover, or postfailover operation. These operations are performed on the resources described in the DR configuration the DR plan is associated with.

- When a **switchover** is performed, there is no outage, so both peered systems are online. The goal is to move all resources covered in the DR configuration from the primary system (A) to the standby system (B). When completed, system B becomes the primary and system A the standby for the resources in question.
- A **failover** is performed on the standby system, when one of the peered systems goes down. The goal is to recover all resources covered in the DR configuration on the standby system (B), allowing continuation of service. The failover steps are similar to the switchover plan, but none of the operations on the primary system (A) can be performed. The primary system cannot be cleaned up until it comes back online.
- A **postfailover** plan is performed after a failover, when the system that experienced an outage comes back online, and the peer connection is restored. The goal is to clean up the DR configuration on the primary system that went down (A), and set it up as the standby for the new primary system (B).

Working with DR Configurations and DR Plans

The implementation of the Native Disaster Recovery service is shared with Private Cloud Appliance, but explicit permission must be given to one or more administrator accounts to access the DR functionality of the Service Enclave.

From the Service CLI you create and maintain the DR configurations and DR plans. Follow the instructions in the Private Cloud Appliance documentation.

- Working with Disaster Recovery Configurations

  Learn how to create and manage DR configurations. It's important to keep site mappings up-to-date and ensure the list of compute instances under DR protection reflects your current requirements.
- Working with Disaster Recovery Plans

  Learn how to create and manage DR plans. These define the operations performed on protected resources when a controlled switchover is executed, and when a site level incident occurs that requires failover. The execution of a DR plan is not automated through failure detection. It must be initiated from the Service CLI on the standby system.

Supported Resources

It is important to understand what is covered under disaster recovery and what is not.

Disaster recovery supports:

- Compute instances
- The block volumes associated with these compute instances

The following limitations apply to the Disaster Recovery service:

- File systems are not supported
- Object storage is not supported
- OKE clusters are not supported
- Application and network load balancers are not supported
- SR-IOV instances are not supported

Storage Replication

In addition to the system level DR service, Console users can add a data recovery option by enabling storage replication. Replication to a peered system is available for boot and block volumes, volume groups, file systems and their snapshots. A replica can be used to restore operation when the original resource has become unavailable or corrupted.

For details and instructions, see the documentation sections on Block Volume Storage on page 380 and File Storage on page 433.

Accessing Compute Cloud@Customer User Interfaces

Use the Compute Cloud@Customer Console, CLI, and API to create and manage resources on an infrastructure you have configured and connected to Oracle Cloud Infrastructure.

If you need to change infrastructures or upgrade schedules, or control access to Compute Cloud@Customer resources using Identity and Access Management (IAM), use the Oracle Cloud Console. For more information, see Managing Compute Cloud@Customer Infrastructures on page 71 and Identity and Access Management (IAM) on page 108.

Otherwise, you manage Compute Cloud@Customer resources by accessing an infrastructure directly. See the following topics:

- Signing in to the Compute Cloud@Customer Console on page 102
- Using the CLI to Manage Compute Cloud@Customer Resources on page 103
- Using the API to Manage Compute Cloud@Customer Resources on page 107

Signing in to the Compute Cloud@Customer Console

Use the Compute Cloud@Customer Console to create and manage compute, storage and other resources on a Compute Cloud@Customer infrastructure.

The Compute Cloud@Customer Console is an intuitive, graphical interface to create and manage resources on the Compute Cloud@Customer infrastructure.

Prerequisites

You need to know the following Compute Cloud@Customer names and your password.

- Infrastructure name, as provided to Oracle when setting up the infrastructure.
- Domain name where the infrastructure's Console is hosted.
- Tenancy name for the infrastructure.
- Your Oracle Cloud Infrastructure username and password.

1. From a browser, enter the URL of the Compute Cloud@Customer infrastructure.

   For example, `https://console.<ccc_name>.example.com`

   where:

- *ccc_name* is the Compute Cloud@Customerinfrastructure name.
- *example.com* is the domain name.

2. On the Select Tenancy page, enter the tenancy name for the infrastructure and then select **Continue**.

   The Sign In page is displayed.
3. Enter your username and password, and then select **Sign In**.

   The dashboard is displayed and can be used to access common resources.
4. Use the navigation menu ( ) to access the full list of manageable resources.

> **Note:**
>
> The Compute Cloud@Customer Console provides access to the resources on the Compute Cloud@Customer hardware. You can't access your OCI tenancy using the Compute Cloud@Customer Console.
>
> To access your OCI tenancy, sign in to the Oracle Cloud Console. See Sign In to the OCI Console.

Using the CLI to Manage Compute Cloud@Customer Resources

You can use the OCI CLI to manage resources (compute, storage, and networking) on Compute Cloud@Customer the same way you use the OCI CLI to manage your tenancy resources.

To simplify the use the CLI on Compute Cloud@Customer, update your OCI CLI configuration as described in the following procedure.

Updating Your CLI Configuration to Access Compute Cloud@Customer

This procedure assumes that you already have the OCI CLI installed and configured for use in your OCI tenancy. These instructions describe how to update your OCI CLI environment to run OCI CLI commands on the Compute Cloud@Customer infrastructure.

> **Note:**
>
> The CLI environment can be configured in many different ways. This procedure describes only the key configuration items that enable CLI commands to run on the infrastructure. If you're familiar with configuring the CLI environment, you can adjust the steps to suit your CLI configuration practices.

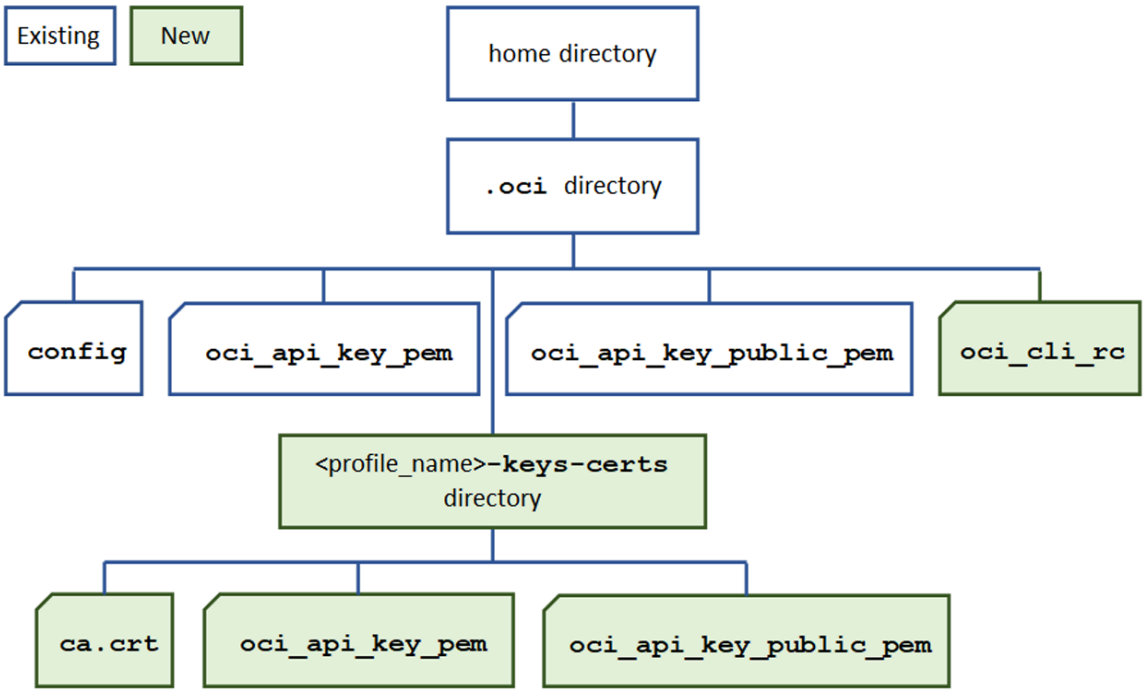
Before You Begin

- The OCI CLI must be installed on a Linux, macOS, or Microsoft Windows system that you use to run CLI commands. If the OCI CLI isn't installed, install and configure it to run in your tenancy before you perform the steps in this procedure. For instructions, see Command Line Interface (CLI).
- You have an OCI configuration file that's configured to run CLI commands on your tenancy. The configuration file path name is `<home>/.oci/config`. For more information, see Setting up the Configuration File and Configuration File Entries.

By the end of this procedure, the following updates are made to your CLI configuration:

- Creates a new directory for storing your API keys and `ca.cert` file.
- Adds a new `ca.crt` certificate file to enable the CLI to access the infrastructure.
- Copies your API keys to a new directory.
- Creates a new CLI profile in the `config` file for running CLI commands on the infrastructure.
- Adds the new profile to the `oci_cli_rc` file.

Your `.oci` directory has the following structure:



The examples are for Linux. If you're using Microsoft Windows, use backslash as the directory separator in path names, instead of the forward slash.

1. Create a new directory in your `<home>/.oci` directory for API key and certificate files.

   Suggestion: Include the new profile name (will be defined in the `config` file in a later step) in the directory name as shown in this example:

   ```
   cd ~/.oci
   mkdir cccadmin-keys-certs
   ```

2. Copy your private and public API keys to the new directory.

   Example:

   ```
   cp oci_api_key.pem  oci_api_key_public.pem ./cccadmin-keys-certs
   ```

   Note: The original key pair must remain in the `.oci` directory to keep your tenancy authentication functional.

3. Obtain the infrastructure certificate authority (CA) chain file that's provided in the certificate bundle.

   Obtain the certificate bundle using the following URL:

   ```
   https://iaas.<system_name>.<domain_name>/cachain
   ```

   Where *<system_name>* is the name of the infrastructure, typically the same as the display name. And *<domain_name>* is the domain the infrastructure is in.

   Save the CA chain in a file with this path name: `<home>/.oci/<profile_name>-keys-certs/ca.cert`

4. Edit your `<home>/.oci/config` file, and copy and paste your current OCI profile so that it appears twice in the file.

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/.oci/oci_api_key.pem
tenancy=ocid1.tenancy...unique_ID
region=tenancy_region

[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/.oci/oci_api_key.pem
tenancy=ocid1.tenancy...unique_ID
region=tenancy_region
```

5. In the copied profile, change these parameters:

Parameter	Value
Profile name (such as `[DEFAULT]`)	Change the this to a unique profile name, for example: `[CCCADMIN]`.
`user`	No change needed.
`fingerprint`	No change needed.
`key_file`	Change the path to `/home/.oci/<profile_name>-keys-certs`
`tenancy`	No change needed.
`region`	Change the region so that it points to the Compute Cloud@Customer infrastructure using this format: `region=<compute-cloud-customer_system_name>.<domain_name>` where *<compute-cloud-customer_system_name>* is the name of the Compute Cloud@Customer infrastructure, and *<domain_name>* is the domain the infrastructure is in. Example: `region=ccc1.us.example.com`

> **Note:**
>
> Your `user` OCID and `tenancy` information is part of your IAM configuration in OCI. If you change any of these parameters, you might need to wait 10 to 15 minutes for the changes to be synchronized on the infrastructure. See Identity and Access Management (IAM) on page 108.

6. Modify or create a `<home>/.oci/oci_cli_rc` file.

   If you don't have an `oci_cli_rc` file, you can create one with a text editor, or use the `oci-cli-rc` command. See oci-cl-rc CLI Reference page.

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

   The following example shows the addition of the `CCCADMIN` profile with the path to the new `ca.crt` file.

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

[CCCADMIN]
```

```
cert-bundle=/home/username/.oci/cccadmin-keys-certs/ca.crt
```

> **Note:**
>
> The `oci_cli_rc` file is used to provide various OCI CLI default values and other configuration information. If you have other parameters specified for your tenancy profile, you can include them for the new profile that you're adding to the `oci_cli_rc` file.

7. Verify your changes, then save the `config` file.

   Example `config` file:

```
[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/.oci/oci_api_key.pem
tenancy=ocid1.tenancy...unique_ID
region=tenancy_region

[CCCADMIN]
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/.oci/oci_api_key.pem
tenancy=ocid1.tenancy...unique_ID
region=ccc1.us.example.com
```

   You can create as many profiles as you like. For example, if you have several infrastructures, you can create a profile for each infrastructure.

8. Test your new OCI CLI profile by specifying the `--profile <profile_name>` option with a command.

   Example:

```
oci iam user list --profile CCCADMIN
```

> **Note:**
>
> You might encounter a warning message stating the permissions on the `config` or `oci_cli_rc` file are too open. If this happens, follow the instructions in the warning message to resolve the issue.

Running CLI Commands on the Infrastructure

The required CLI arguments you need to use to reach the Compute Cloud@Customer infrastructure depend on how you've configured your CLI environment. Here are some common scenarios:

- If you updated your CLI configuration to access Compute Cloud@Customer, use the (`--profile profile_name`) option on the command line. The profile provides the CLI with the infrastructure's region and certificate information.

  Example of listing images in a compartment on an infrastructure:

```
oci compute image list --compartment-id ocid.compartment...uniqueID --
profile CCCADMIN
```

- If you didn't update your CLI configuration, you might need to include these CLI arguments:
  - `--region <compute-cloud-customer_system_name>.<domain_name>`
  - `--cert-bundle <full_path_to_CA_certificate_bundle>`

Example of listing images in a compartment on an infrastructure:

```
oci compute image list --compartment-id ocid.compartment...uniqueID --
region ccc1.us.example.com --cert-bundle <path_to_cert_file>
```

For more information about running CLI commands, see these resources:

- Get help on the command line:

```
oci --help

oci network --help

oci network vcn create --help
```

- Get Started with the Command Line Interface
- Command Line Reference
- Compute Cloud@Customer CLI Reference Section

Obtaining the Certificate Authority Bundle

Whether you configured the CLI manually or used the automated tool, you must obtain the Compute Cloud@Customer device external silo CA chain before you can run commands.

The external silo CA chain must be copied to the system where you're 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.

Using the API to Manage Compute Cloud@Customer Resources

The Compute Cloud@Customer API is a subset of the Oracle Cloud Infrastructure (OCI) REST API that uses HTTPS requests and responses.

To see the API references for Compute Cloud@Customer, use the links in the following table:

Description	API Reference
The API used to manage resources on the Compute Cloud@Customer infrastructure in your data center. For example, API operations to manage instances, virtual cloud networks, and storage resources.	REST API for Oracle Private Cloud Appliance
The API used to manage Compute Cloud@Customer specific resources in your OCI tenancy. For example, API operations to manage infrastructures and upgrade schedules.	Compute Cloud@Customer API

- Endpoints follow this format

  ```
  https://<service>.<system_name>.<domain_name>
  ```

  Where *<service>* is the name of the OCI service providing the API you want to use. For example, `iaas` for the core services. *<system_name>* is the name of the infrastructure, typically the same as the display name. *<domain_name>* is the domain the infrastructure is in.
- For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Identity and Access Management (IAM)

On Compute Cloud@Customer, the Oracle Cloud Infrastructure Identity and Access Management (IAM) service lets you control who has access to the cloud resources within your tenancies.

Identity and Access Management (IAM) provides features such as authentication, single sign-on (SSO), and identity lifecycle management for Oracle Cloud.

Where to Manage IAM

> **Note:**
>
> For Compute Cloud@Customer, IAM resources are managed in OCI within your tenancy, and synchronized to Compute Cloud@Customer approximately every ten minutes. IAM resources can't be managed on the Compute Cloud@Customer infrastructure.

Manage the following IAM features and resources in Oracle Cloud Infrastructure:

- Tenancy management
- Identity domain types (IAM with identity domains)
- Federating with identity providers
- Users, groups, credentials, and roles
- Compartments
- Policies
- Tagging

The way you manage IAM depends on the type of IAM you have – with Identity Domains or without Identity Domains. For information about managing IAM in OCI, see these resources:

- IAM with Identity Domains
- IAM without Identity Domains

Manage the following features and resources in OCI using the Oracle Cloud Console, OCI CLI, and OCI API:

- Tenancy details
- Identity domain types
- Compartments
- Federated identity providers
- Users and groups
- Policies
- Defined tags

The way you work with IAM in OCI depends on the type of IAM that's configured in your tenancy – either with Identity Domains or without Identity Domains. For more information, see these topics:

- Determining the Tenancy Type
- IAM with Identity Domains

- IAM without Identity Domains
- Managing User Capabilities for Federated Users
- Managing Oracle Identity Cloud Service Users and Groups in the Oracle Cloud Infrastructure Console

IAM Information Specific to Compute Cloud@Customer

For key IAM information that's specific to Compute Cloud@Customer, see these topics:

- Federated Identity Providers (IAM in OCI) on page 109
- Identity Domains (IAM in OCI) on page 109
- Compute Cloud@Customer Policy Reference on page 110
- Tagging Resources (IAM in OCI) on page 115

Federated Identity Providers (IAM in OCI)

The Compute Cloud@Customer service uses the same federated identity provider that provides your identity services to Oracle Cloud Infrastructure. The identity provider manages usernames, passwords, and authentication to access the service.

Federation involves setting up a trust relationship between the identity provider and Oracle Cloud Infrastructure. When the administrator has established that relationship, any user who accesses the Compute Cloud@Customer Console is prompted with a single sign-on experience offered by the identity provider.

For more information, see Establish a Federated Identity Provider on page 68.

Identity Provider Administration

Your federated identity provider is configured in the tenancy, and Oracle ensures that the same identity provider is configured in Compute Cloud@Customer during installation (see Establish a Federated Identity Provider on page 68.)

If you are federating with Oracle Identity Cloud Service, see Federating with IDCS.

> **Important:**
>
> If you change your identity provider configuration in Oracle Cloud Infrastructure, Oracle must make the same administrative changes on Compute Cloud@Customer. In this situation, open an Oracle Support Request to request help. See Creating a Support Request.

Identity Domains (IAM in OCI)

Compute Cloud@Customer supports the use of one or two Oracle Cloud Infrastructure (OCI) identity domains for managing users, roles, user federation, and OAuth administration. This assumes that your OCI tenancy is configured to use IAM identity domains.

This section describes unique identity domain aspects that apply to Compute Cloud@Customer. For more general information, see IAM with Identity Domains.

By default, if Compute Cloud@Customer is associated with an OCI tenancy that uses IAM identity domains, the default identity domain is automatically synchronized to the Compute Cloud@Customer infrastructure in your data center. This enables you to manage all your IAM resources in one place.

Secondary, Nondefault Identity Domains

Optionally, during or after installation, you can have one additional OCI identity domain synchronized to Compute Cloud@Customer, each with different identity and security requirements to protect your applications and resources.

Having a secondary identity domain enables you to maintain the isolation of administrative control over each identity domain. This is necessary if, for example, security standards prevent certain user IDs from existing in the production environment, or require that different administrators have control over different environments.

Key points:

- Before you can have a secondary domain synchronized on Compute Cloud@Customer, both identity domains must exist in your tenancy.
- No matter how many identity domains are in your tenancy, you can only select one to serve as a secondary domain on Compute Cloud@Customer.
- You can select which identity domain is the default identity domain and which is the secondary identity domain. Only one domain can be a default domain.
- The default identity domain and the secondary identity domain continue to be managed in your OCI tenancy.
- Users in a secondary identity domain can perform operations on Compute Cloud@Customer resources based on their permissions. However, if a user views IAM resources on Compute Cloud@Customer, only default domain resources are displayed. To see secondary domain IAM resources, sign in to your OCI tenancy.

Requesting Secondary Identity Domain Administration

Open a support request to have Oracle enable or disable the synchronization of a secondary identity domain on Compute Cloud@Customer infrastructure. See Creating a Support Request. To access support, sign in to the Oracle Cloud Console as described in Sign In to the OCI Console.

Compute Cloud@Customer Policy Reference

Use policies to control access to Compute Cloud@Customer infrastructure and upgrade schedule operations.

Information in these sections provide policy information specifically for Compute Cloud@Customer infrastructures and upgrade schedules. For detailed information about Oracle Cloud Infrastructure IAM and policies, see the following topics:

- Overview of IAM
- Getting Started with Policies
- Oracle Cloud Infrastructure Policy Reference

> **Note:**
>
> Creating a policy requires proper privileges. Work with your tenancy administrator to either obtain the privileges or have the policies created for you.

Resource-Types

Compute Cloud@Customer introduces additional resource-types that enable you to manage the Compute Cloud@Customer infrastructures and upgrade schedules.

Aggregate Resource-Type

An aggregate resource-type covers the list of individual resource-types that directly follow. For example, writing one policy to allow a group to have access to the `ccc-family` is equivalent to writing separate policies for the group that would grant access to the `ccc-infrastructure`, and `ccc-upgrade-schedule`. For more information, see Permissions Required for Each API Operation on page 113.

Family Name	Member Resources
ccc-family	ccc-infrastructure ccc-upgrade-schedule

Family Name	Member Resources
ccc-mp-family	ccc-fault ccc-remote-resource ccc-resource-principal ccc-trust-information ccc-infrastructure-notifier

Individual Resource-Types

Resource Types	Permissions
ccc-infrastructure	CCC_INFRASTRUCTURE_INSPECT (list with summaries) CCC_INFRASTRUCTURE_READ (view resource) CCC_INFRASTRUCTURE_UPDATE (modify settings) CCC_INFRASTRUCTURE_CREATE (provision new CCC infrastructure) CCC_INFRASTRUCTURE_DELETE (delete CCC infrastructure) CCC_INFRASTRUCTURE_MOVE (move the infrastructure)
ccc-upgrade-schedule	CCC_UPGRADE_SCHEDULE_INSPECT CCC_UPGRADE_SCHEDULE_READ CCC_UPGRADE_SCHEDULE_UPDATE CCC_UPGRADE_SCHEDULE_CREATE CCC_UPGRADE_SCHEDULE_DELETE CCC_UPGRADE_SCHEDULE_MOVE
ccc-fault	CCC_FAULT_INSPECT (list faults) CCC_FAULT_CREATE (provision new CCC installation) CCC_FAULT_READ (view settings) CCC_FAULT_UPDATE (modify settings) CCC_FAULT_DELETE (teardown CCC installation) CCC_FAULT_MOVE (change fault compartments)

Resource Types	Permissions
ccc-remote-resource	CCC_REMOTE_RESOURCE_INSPECT (list remote resources) CCC_REMOTE_RESOURCE_CREATE (create a remote resource) CCC_REMOTE_RESOURCE_READ (view a remote resource) CCC_REMOTE_RESOURCE_UPDATE (update a remote resource) CCC_REMOTE_RESOURCE_DELETE (delete a remote resource) CCC_REMOTE_RESOURCE_MOVE (change resource compartments)
ccc-resource-principal	CCC_RESOURCE_PRINCIPAL_UPDATE (update to a refreshed resource principal token)
ccc-trust-information	CCC_TRUST_INFORMATION_CREATE (push trust information to OCI)
ccc-infrastructure-notifier	no permissions (no API related to this resource)

Supported Variables

Compute Cloud@Customer, supports the Oracle Cloud Infrastructure general variables.

See General Variables for All Requests

Details for Verb+Resource-Type Combinations

You use permissions and verbs to write policies to give a group access to a particular resource-type. Compute Cloud@Customer provides resource-types and permissions that are unique to Compute Cloud@Customer, but use the Oracle Cloud Infrastructure verbs.

The following tables show the Permissions and API operations covered by each verb, using the following notations:

- The level of access is cumulative as you go from `inspect` > `read` > `use` > `manage`.
- A plus sign (+) indicates incremental access compared to the cell directly above it.
- "no extra" indicates no incremental access.

`ccc-infrastructure`

Verbs	Permissions	APIs Fully Covered	APIs Partially Covered
inspect	CCC_INFRASTRUCTURE_INSPECT	ListCccInfrastructures	none
read	INSPECT + CCC_INFRASTRUCTURE_READ	GetCccInfrastructure	none
use	READ + CCC_INFRASTRUCTURE_UPDATE	UpdateCccInfrastructure	none

Verbs	Permissions	APIs Fully Covered	APIs Partially Covered
manage	USE + CCC_INFRASTRUCTURE_CREATE CCC_INFRASTRUCTURE_DELETE CCC_INFRASTRUCTURE_MOVE	no extra	CreateCccInfrastructure (also needs use subnets) DeleteCccInfrastructure (also needs use subnets) ChangeCccInfrastructureCompartment

`ccc-upgrade-schedule`

Verbs	Permissions	APIs Fully Covered	APIs Partially Covered
inspect	CCC_UPGRADE_SCHEDULE_INSPECT	ListCccUpgradeSchedules	none
read	INSPECT + CCC_UPGRADE_SCHEDULE_READ	GetCccUpgradeSchedule	none
use	READ + CCC_UPGRADE_SCHEDULE_UPDATE	UpdateCccUpgradeSchedule	none
manage	USE + CCC_UPGRADE_SCHEDULE_CREATE CCC_UPGRADE_SCHEDULE_DELETE CCC_UPGRADE_SCHEDULE_MOVE	CreateCccUpgradeSchedule DeleteCccUpgradeSchedule ChangeCccUpgradeScheduleCompartment	none

Permissions Required for Each API Operation

The following tables list the API operations and which permissions are required to use the operation.

Compute Cloud@Customer Infrastructure Operations

API Operation	Permissions Required to Use the Operation
ListCccInfrastructures	CCC_INFRASTRUCTURE_INSPECT
CreateCccInfrastructure	CCC_INFRASTRUCTURE_CREATE and CLIENT_SUBNET_UPDATE
GetCccInfrastructure	CCC_INFRASTRUCTURE_READ
UpdateCccInfrastructure	CCC_INFRASTRUCTURE_UPDATE
DeleteCccInfrastructure	CCC_INFRASTRUCTURE_DELETE and CLIENT_SUBNET_UPDATE
ChangeCccInfrastructureCompartment	CCC_INFRASTRUCTURE_MOVE

Upgrade Schedule Operations

API Operation	Permissions Required to Use the Operation
ListCccUpgradeSchedules	CCC_UPGRADE_SCHEDULE_INSPECT
CreateCccUpgradeSchedule	CCC_UPGRADE_SCHEDULE_CREATE
GetCccUpgradeSchedule	CCC_UPGRADE_SCHEDULE_READ
UpdateCccUpgradeSchedule	CCC_UPGRADE_SCHEDULE_UPDATE
DeleteCccUpgradeSchedule	CCC_UPGRADE_SCHEDULE_DELETE
ChangeCccUpgradeScheduleCompartment	CCC_UPGRADE_SCHEDULE_MOVE

Fault Operations

API Operation	Permissions Required to Use the Operation
ListCccFaults	CCC_FAULT_INSPECT
CreateCccFaults	CCC_FAULT_CREATE
GetCccFaults	CCC_FAULT_READ
UpdateCccFaults	CCC_FAULT_UPDATE
DeleteCccFaults	CCC_FAULT_DELETE
ChangeCccFaultCompartment	CCC_FAULT_MOVE

Remote Resource Operations

API Operation	Permissions Required to Use the Operation
ListCccRemoteResources	CCC_REMOTE_RESOURCE_INSPECT
CreateCccRemoteResources	CCC_REMOTE_RESOURCE_CREATE
GetCccRemoteResources	CCC_REMOTE_RESOURCE_READ
UpdateCccRemoteResources	CCC_REMOTE_RESOURCE_UPDATE
DeleteCccRemoteResources	CCC_REMOTE_RESOURCE_DELETE
ChangeCccRemoteResourcesCompartment	CCC_REMOTE_RESOURCE_MOVE

Resource Principal Operations

API Operation	Permissions Required to Use the Operation
UpdateCccResourcePrincipal	CCC_RESOURCE_PRINCIPAL_UPDATE

Trust Information Operations

API Operation	Permissions Required to Use the Operation
CreateCccTrustInformationActionExchange	CCC_TRUST_INFORMATION_UPDATE

Compute Cloud@Customer Sample Policies

Allow Full Administration Anywhere in a Tenancy

```
Allow group CCCAdministrators to manage ccc-infrastructure in tenancy
Allow group CCCAdministrators to manage ccc-upgrade-schedule in tenancy
```

Allow a Compute Cloud@Customer Infrastructure to be Upgraded and to Generate Faults

```
Allow group CCCAdministrators to manage ccc-mp-family in SampleTenancy
```

Allow a Compartment Administrator to View Infrastructures in a Compartment

```
Allow group CCCMonitors to read ccc-infrastructure in compartment
 SampleCompartment
```

Allow a Compute Cloud@Customer Administrator Access to Manage the Upgrade Schedules in a Compartment

```
Allow group CCCEngineeringAdministrators to manage ccc-upgrade-schedule in
 compartment Engineering
```

Tagging Resources (IAM in OCI)

On Compute Cloud@Customer, tagging enables you to add metadata to resources by defining key/value pairs that are assigned to resources. You can use the tags to organize and list resources based on your business needs.

On Compute Cloud@Customer, tags are applied using the same methods as other resources in Oracle Cloud Infrastructure (OCI), using the following practices:

- Defined tags are set up in your tenancy by an administrator. Only users granted permission to work with the defined tags can apply them to resources. Defined tags provide a key/value map and are organized by combining the tag namespaces with tag keys using dot notation. For example, a tag namespace called `HumanResources` could have a key named `CostCenter`. You then associate the namespace and key `HumanResource.CostCenter` and then assign the tag.

  To apply defined tags to resources on Compute Cloud@Customer, first create the namespace and defined tags in the OCI tenancy where the Compute Cloud@Customer infrastructure is located. After one to ten minutes, the namespace and defined tags are synchronized on Compute Cloud@Customer, and can be applied to Compute Cloud@Customer resources.
- Free-form tags can be applied by any user with permissions on the resource. Free-form tags are a key/value map. These tags can be applied directly to Compute Cloud@Customer resources. Unlike defined tags, free-form tags don't require any configuration at the tenancy level.

- You can add tags to resources during resource creation. See Adding Tags at Resource Creation (IAM in OCI) on page 118.
- You can use tags to automatically add user names and creation dates to Compute Cloud@Customer resources. See Using Tags to Automatically Add User Names and Creation Dates to Resources (IAM in OCI) on page 116

> **Note:**
>
> Defined tags and tag namespaces must be configured in your OCI tenancy. See Defined tags.

> **Note:**
>
> On the Compute Cloud@Customer infrastructure you can't manage resource tags using bulk add, bulk update, or bulk remove operations.

For more information, refer to these OCI resources:

- Conceptual information: Tagging Overview
- Managing tags: Resource Tags

Using Tags to Automatically Add User Names and Creation Dates to Resources (IAM in OCI)

On Oracle Cloud Infrastructure (OCI), all resources are automatically tagged with the name of the user that created the resource and the time the resource was created. This task describes how to achieve the same result on Compute Cloud@Customer.

This task can be done using either the Oracle Cloud Console or the CLI. Only the CLI is shown. Ensure that you create the tag namespace and tags in the OCI tenancy where the Compute Cloud@Customer infrastructure is located.

1. Create the `Oracle-Tags` tag namespace.

   Get the OCID of the compartment in which you want to create the tag namespace. Because the goal of this task is for these tag defaults to be applied to every resource in every compartment, create the tag namespace in the tenancy (root compartment). Use the following command to find the OCID of the tenancy:

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

   Create the tag namespace.

   ```
   $ oci iam tag-namespace create -c ocid1.tenancy.unique_ID --name "Oracle-
   Tags" \
   --description "OCI compatibility: add creator and create date of resource"
   {
     "data": {
       "compartment-id": "ocid1.tenancy.unique_ID",
       "defined-tags": {},
       "description": "OCI compatibility: add creator and create date of
    resource",
       "freeform-tags": {},
       "id": "ocid1.tag_namespace.unique_ID",
       "is-retired": false,
       "lifecycle-state": "ACTIVE",
       "name": "Oracle-Tags",
       "time-created": "2022-06-07T14:51:23.256790+00:00"
     },
     "etag": "a000d250-3aea-4faf-b0e6-b6db486ffb42"
   }
   ```

2. Create a `CreatedBy` tag in the `Oracle-Tags` tag namespace.

Use the value of the `id` property from the `tag-namespace create` command for the tag namespace ID. The `--validator` option is not required.

```
$ oci iam tag create --tag-namespace-id ocid1.tag_namespace.unique_ID \
--name CreatedBy --description "user that created this resource" \
--validator '{"validator-type":"ENUM","values":["${iam.principal.name}"]}'
{
  "data": {
    "compartment-id": "ocid1.tenancy.unique_ID",
    "defined-tags": {},
    "description": "user that created this resource",
    "freeform-tags": {},
    "id": "ocid1.tag.unique_ID",
    "is-cost-tracking": false,
    "is-retired": false,
    "lifecycle-state": "ACTIVE",
    "name": "CreatedBy",
    "tag-namespace-id": "ocid1.tag_namespace.unique_ID",
    "tag-namespace-name": "Oracle-Tags",
    "time-created": "2022-06-07T15:32:22.226554+00:00",
    "validator": {
      "validator-type": "ENUM",
      "values": [
        "${iam.principal.name}"
      ]
    }
  },
  "etag": "fa18d128-fc7b-420d-87d6-c44a574e522a"
}
```

3. Create a `CreatedOn` tag in the Oracle-Tags tag namespace.

```
$ oci iam tag create --tag-namespace-id ocid1.tag_namespace.unique_ID \
--name CreatedOn --description "date this resource was created" \
--validator '{"validator-type":"ENUM","values":["${oci.datetime}"]}'
```

4. Create a `CreatedBy` tag default.

   To apply this tag default to every new resource created in any compartment, create the tag default in the tenancy.

   Use the value of the `id` property from Step 2 for the tag definition ID. The `--value` option is required.

```
$ oci iam tag-default create -c ocid1.tenancy.unique_ID
--tag-definition-id ocid1.tag.unique_ID --value "\${iam.principal.name}"
{
  "data": {
    "compartment-id": "ocid1.tenancy.unique_ID",
    "id": "ocid1.tag_default.unique_ID",
    "is-required": null,
    "lifecycle-state": "ACTIVE",
    "tag-definition-id": "ocid1.tag.unique_ID",
    "tag-definition-name": "CreatedBy",
    "tag-namespace-id": "ocid1.tag_namespace.unique_ID",
    "time-created": null,
    "value": "${iam.principal.name}"
  },
  "etag": "None"
}
```

5. Create a `CreatedOn` tag default.

   Use the value of the `id` property from Step 3 for the tag definition ID.

```
$ oci iam tag-default create -c ocid1.tenancy.unique_ID
```

```
--tag-definition-id ocid1.tag.unique_ID --value "\${oci.datetime}"
```

Whenever a new resource is created in the tenancy or any subcompartment of the tenancy, the following tags are applied, even if no tags are specified when the resource is created:

```
"defined-tags": {
  "Oracle-Tags": {
    "CreatedBy": "flast",
    "CreatedOn": "2022-06-07T16:09:47.01Z"
  }
}
```

For more information about CLI tag commands, refer to these CLI Reference pages:

- tag
- tag-default
- tag-namespace

Adding Tags at Resource Creation (IAM in OCI)

On Compute Cloud@Customer, 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 to create the resource.

> **Note:**
>
> To apply *defined tags* and *tag namespaces* during resource creation, they must be configured in your OCI tenancy ahead of time. See Defined tags.
>
> *Free-form* tags can be created and applied during resource creation. See Freeform tags

Using the Console

1. In the Create dialog for the resource, scroll to the Tagging section.
2. Select the Tag Namespace or select None (to 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, select the Additional Tag button.

   You can't 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, select the Tags tab to display the tags that are applied to this resource.

Using the 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 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 appropriate tags. Specify the result in the final command as shown in the following step.

2. Run the resource `create` or `launch` command.

   To add one or more defined tags, use the `--defined-tags` option. 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 <compartment_OCID> --
   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.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

To add a tag to a resource when you create the resource, use the `create`<*resource*> or `launch`<*resource*> command.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Using Free-Form Tags for Extended Functionality

On Compute Cloud@Customer, you can use free-form tags to extend the functionality of some services.

Use of these tags counts against your tag limit.

The following list describes the special purpose free-form tags:

> **Note:**
>
> Don't use these tag names for other purposes.

- **`PCA_no_lm` – Control instance live migration.**

  Use this tag to instruct the Compute service not to live migrate an instance. The value can be either `True` or `False`.

  By default, an instance can be live migrated, such as when you need to evacuate all running instances from a compute node. Live migration can be a problem for some instances. For example, live migration must not be used for instances in a Microsoft Windows cluster. To prevent an instance from being live migrated, set this tag to `True` on the instance.

  Specify this tag in the Tagging section of the Create Instance or Edit ***`instance_name`*** dialog, in the `oci compute instance launch` or `oci compute instance update` command, or using the API.

  The following is an example option for the `oci compute instance launch` command:

  ```
  --freeform-tags '{"PCA_no_lm": "True"}'
  ```

  Setting this tag to True on an instance doesn't prevent the instance from being moved when you change the fault domain. Changing the fault domain isn't a live migration. When you change the fault domain of an instance, the instance is stopped, moved, and restarted.
- **`PCA_blocksize` – Create new volumes with a specific disk block size.**

  Use this tag to change the LUN *block size* used by the underlying storage when a block volume is created. Note – Using this tag doesn't apply to the size of the block volume.

  The default block size is 8192 bytes. To specify a different block size, specify the `PCA_blocksize` tag in the Tagging section of the Create Block Volume dialog, in the `oci bv volume create` command, or using the API. Valid values are a power of 2 between 512 and 1M bytes, specified as a string and fully expanded.

  The following is an example option for the `oci bv volume create` command:

  ```
  --freeform-tags '{"PCA_blocksize": "65536"}'
  ```

  The block size can't be changed after the volume has been created.

Creating OraclePCA Tags (IAM in OCI)

On Compute Cloud@Customer you can use the OraclePCA tag namespace to enable resource attributes that aren't available as CLI options or API attributes.

> **Note:**
>
> This section covers the Tag Key Definitions for Block Volume and File System attributes, and SR-IOV. There are more Tag Key Definitions, and other administrative tasks required to enable Kubernetes Engine (OKE) on Compute Cloud@Customer. For those details, see Cluster Administrator Tasks on page 513.

You can configure the OraclePCA tag namespace to enable these attributes:

Block Volume Attributes

- **Synchronous Write Bias**

  You can enable the selection of `LATENCY` or `THROUGHPUT` when a block volume is created.
- **Secondary Cache**

  You can enable the selection of `ALL`, `METADATA`, or `NONE` when a block volume is created.

File System Attributes

- **Quota**

  You can enable the setting of a quota limitation when a file system is created.
- **Database record size**

  You can enable the selection of different database record sizes when a file system is created.
- **Backing store pool**

  You can enable the selection of `PCA_POOL_HIGH` and `poolName` when a file system is created.

The following procedures describe how to configure the OraclePCA tags so that the resource attributes are available when you create your resource, such as a block volume or file system.

Creating the OraclePCA Tag Namespace

1. In the Oracle Cloud Console, open the navigation menu, select **Governance & Administration**, and then select **Tag Namespaces**.
2. If OraclePCA isn't shown in the **Tag Namespaces** list, select **Create Tag Namespace**.

   If the OraclePCA namespace already exists, don't continue. Instead, go to Creating the OraclePCA Tag Key Definitions on page 121.
3. In the **Create Tag Namespace** dialog box, enter the following information:
   - **Create in Compartment**: Select the compartment in which you want to create the namespace definition.
   - **Namespace Definition Name**: Enter **OraclePCA**.
   - **Description**: Enter a description. For example, `Support resource attributes that are available.`
4. Select **Create Tag Namespace**.

   The details page for the new OraclePCA tag namespace definition is displayed.

Next, create tag key definitions. See Creating the OraclePCA Tag Key Definitions on page 121.

Creating the OraclePCA Tag Key Definitions

1. On the OraclePCA tag namespace details page, select **Create Tag Key Definition**.
2. In the **Create Tag Key Definition** dialog box, enter values for an attribute you want to enable. The attributes and values are shown in the following table.

> **Note:**
>
> You don't need to configure all the following tag key definitions.
> Configure only the tag key definitions for the attributes you want to enable.

Attribute	Value

Block Volume Synchronous Write Bias Tag Key Definition	• **Tag Key:** Enter `logBias`.
	• **Description:**Enter a description. For example, `Control the use of the write cache flash devices for a share or LUN.` Avoid entering confidential information.
	• **Tag Value Type:** Select **A List of Values**.
	• **Values:** Enter the following values, each on a separate line:
	  • `LATENCY`
	  • `THROUGHPUT`

Block Volume Secondary Cache Tag Key Definition	• **Tag Key**: Enter `secondaryCache`.
	• **Description:** Enter a description. For example, `Control the use of the read cache flash devices for a share or LUN`. Avoid entering confidential information.
	• **Tag Value Type:** Select **A List of Values**.
	• **Values:** Enter the following values, each on a separate line:
	  • `ALL`
	  • `METADATA`
	  • `NONE`

File System Quota	• **Tag Key**: Enter `quota`. • **Description**: Enter a description. For example, `The quota value in gigabytes includes the data in the file system and all snapshots created under the file system.` Avoid entering confidential information. • **Tag Value Type**: Select **Static Value**.

File System Database Record Size	• **Tag Key**: Enter `databaseRecordSize`.
	• **Description**: Enter a description. For example, `The size of a database record in bytes.` Avoid entering confidential information.
	• **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`

Attribute	Value

File System Backing Store Pool	• **Tag Key**: Enter `poolName`.
	• **Description**: Enter a description. For example, `Whether to use the default storage pool, or use a high performance pool as the backing store pool.` Avoid entering confidential information.
	• **Tag Value Type**: Select **A List of Values**.
	• **Values**: Enter the following values, each on a separate line:
	  • `PCA_POOL`
	  • `PCA_POOL_HIGH`

Network Configuration for SR-IOV	• **Tag Key**: Enter `networkType`. • **Description**: Enter a description. For example, `DRG and VCN network configuration required for SR-IOV.` • **Tag Value Type**: Select **Static Value**.

3. Select **Create Tag Key Definition**.

   The details page for the new tag key definition is displayed.
4. (Optional) Repeat this procedure to create more tag key definitions.

Configuring Resources to Use Attributes Provided by OraclePCA Tags

Depending on the resource you want to create, see one of these sections:

- Creating a Block Volume on page 384
- Creating a File System on page 445

Compute Cloud@Customer Metrics

You can monitor Compute Cloud@Customer infrastructure resources using metrics. Compute Cloud@Customer metrics provide you with the total and available OCPU, memory, and storage resources for the infrastructure.

The following topics describe the metrics emitted by the Compute Cloud@Customer infrastructure in the `oci_ccc` metric namespace:

- Viewing Compute Cloud@Customer Default Metric Charts on page 124
- Creating a Query with Metrics Explorer on page 124
- Example of Creating an Alarm on Compute Cloud@Customer on page 125
- Compute Cloud@Customer Metrics Reference on page 126

> **Note:**
>
> The monitoring information in the topics in this section only describe the metrics for the Compute Cloud@Customer infrastructure.
>
> For more general information about metrics, alarms, and the Metrics Explorer for other resources, see these topics:
>
> - Overview of Monitoring
> - Managing Alarms
> - Building Metric Queries

Required IAM Policy

To monitor resources, you must have the required type of access in a policy written by an administrator, whether you're using the Oracle Cloud Console or the REST API with an SDK, CLI, or other tool.

The policy must give you access to the monitoring services and the resources being monitored. For more information about user authorizations for monitoring, see Metric Access for Groups.

Viewing Compute Cloud@Customer Default Metric Charts

You can view the default metric charts for the Compute Cloud@Customer infrastructure. The charts show metrics for total and available OCPUs, memory, and storage resources in the infrastructure.

For metric specifications, see Compute Cloud@Customer Metrics Reference on page 126.

1. In the Oracle Cloud Console navigation menu, select **Observability & Management**.
2. Under **Monitoring**, select **Service Metrics**.
3. From the Console header, select the region that contains the metric data that you want.

   For more information about regions, see Understand Regions.
4. On the **Service Metrics** page, select the compartment where the Compute Cloud@Customer infrastructure was created.
5. For **Metric namespace**, select **oci_ccc**.

   The default metric charts are displayed.
6. (Optional) To change the time range:
   a. Select a period of time from **Quick Selects**. For example, **Last hour**.
   b. To specify the start or end of a period time, select **Start time** or **End time** and then type a value.

Creating a Query with Metrics Explorer

On Compute Cloud@Customer, you can create your own metric queries using Metrics Explorer.

Use the information in this section to create a metric query that's specifically for Compute Cloud@Customer Infrastructure resources using the metric oci_ccc namespace. For general information about metric queries, see Building Metric Queries.

1. Open the Oracle Cloud Console navigation menu, and select **Observability & Management**. Under **Monitoring**, select **Metrics Explorer**.

   The **Metrics Explorer** page displays an empty chart with fields to build a query.
2. From the Oracle Cloud Console header, select the region that contains the metric data that you want.

   For more information about regions, see Understand Regions.
3. **(Optional)** The default query period is set to **Last hour**. To change it, go to the top panel, and set the query period by selecting a **Start time** and **End time**, or select a period from the **Quick selects** menu.
4. In the **Query** area under the chart, select values for the following items:
   - **Compartment:** Select the compartment where the Compute Cloud@Customer infrastructure was created.
   - **Metric namespace:** Select **oci_ccc** (the Compute Cloud@Customer namespace).
   - **Metric name:** From the drop-down menu, select one of the Compute Cloud@Customer metrics. For descriptions of each metric, see Compute Cloud@Customer Metrics Reference on page 126.
   - **Interval:** Change the default interval to a value greater than collection frequency. For queries using the oci_ccc namespace, choose an interval of 15 minutes or longer.
   - **Statistic:** Select a statistic. For queries using the oci_ccc namespace, Max works well.
5. **(Optional)** In the **Metric dimensions** area, to further refine the query, select values for the following items:
   - **Dimension name:** From the drop-down menu, select a dimension value. For example, to see metrics for a particular fault domain, select **faultDomain**.
   - **Dimension value:** From the drop-down menu, select a dimension value. For example, you might select a specific fault domain for this query.
6. Select **Update Chart**.

   The chart shows data points for the custom metric in a graph view.

   To change the metric chart, update the query values, then select **Update Chart** again.

The query is a one-time query. If you want a persistent query, you can take advantage of the following services:

- Configure a query in a **Console Dashboard**. The Console Dashboard service enables you to create custom dashboards to monitor resources, diagnostics, and key metrics for your tenancy. See Console Dashboards.
- View the default metric charts. See Viewing Compute Cloud@Customer Default Metric Charts on page 124.

Example of Creating an Alarm on Compute Cloud@Customer

On Compute Cloud@Customer, you can create alarms to notify you when Compute Cloud@Customer metrics meet specified triggers.

This example shows how to create a basic alarm that's based on Compute Cloud@Customer metrics. This example creates a critical alarm when the total storage utilization on the Compute Cloud@Customer crosses the specified threshold. The alarm is configured to email the administrators if an alarm is triggered.

For information about managing alarms, see Managing Alarms.

For detailed descriptions about each alarm option, see Creating a Basic Alarm.

To see the Compute Cloud@Customer metrics you can use to create alarms, see Compute Cloud@Customer Metrics Reference on page 126.

Example values are shown in parenthesis like this: (**`value`**)

1. In the Oracle Cloud Console, open the navigation menu and select **Observability & Management**.
2. Under **Monitoring**, select **Alarm Definitions**.
3. Select **Create Alarm**.
4. Enter the following information:
   - **Define alarm**:
     - **Alarm name:** Enter a descriptive name. Avoid entering confidential information. (**`Storage space alarm`**)
     - **Alarm severity:** Select the alarm level. (**`Critical`**)
     - **Alarm body:** Enter the notification content. (**`Low storage space in your Compute Cloud@Customer instance. See action plan instructions at https://example.us.runbook-lowstorage.`**)
   - **Tags**: No tags are applied for this example.
   - **Metric description**:
     - **Compartment:** Select the name of the compartment where the Compute Cloud@Customer infrastructure was created. The selected compartment also becomes the compartment for the alarm that create. (**`ccc-compartment`**)
     - **Metric namespace:** Select the metric namespace. For alarms that are based on Compute Cloud@Customer metrics, the namespace is `oci_ccc`. See Compute Cloud@Customer Metrics Reference on page 126 (**`oci_ccc`**)
     - **Resource group:** No resource group is selected for this example.
     - **Metric name:** Select the metric name. The drop-down menu gives you the choice to select any valid Compute Cloud@Customer metric name. (**`StorageSpaceAvailable`**)
     - **Interval:** Select the interval to check this metric. The interval should be longer than the collection frequency. See Compute Cloud@Customer Metrics Reference on page 126. (**`1 hour`**)
     - **Statistic:** Select the statistic type. This example uses min, but you can select other values. See Selecting the Statistic for an Alarm Query. (**`min`**)
   - **Metric dimensions:**
     - **Dimension name:** Select a dimension. For this example, the alarm dimension is set to `ressourceid` which requires you to specify the resource ID in the next field. (**`resourceid`**)
     - **Dimension value:** From the drop-down menu, select the ID of the Compute Cloud@Customer infrastructure. (**`uniqueID`**)

- **Trigger rule**:
  - **Operator**: Select the operator used in the condition threshold. Less than is selected in this example to trigger when `StorageSpaceAvailable` drops below the specified value. (**`less than`**)
  - **Value**: Select the value to use for the condition threshold. The field automatically displays the unit such as GB. (**`10000`**)
  - **Trigger delay minutes**: Select the number of minutes that the condition must be maintained before the alarm is in the firing state. (**`75`**)
- **Define alarm notifications:**
  - **Destination service:** Select Notifications to send alarm notifications to a topic. (**`Notifications`**)
  - **Compartment:** Select a compartment that you want to store the new topic in. This compartment can be a different compartment than the one specified for the alarm and metric. By default, the first accessible compartment is selected. (**`my-compartment`**)
  - **Topic:Select Create a topic**, and fill in these fields:
    - **Topic Name:** Enter a descriptive name. Avoid entering confidential information. (**`notify_administrators`**)
    - **Topic description:** Enter a topic description. (**`Email alarm notifications to the administrators when storage space is low.`**)
    - **Subscription protocol:** Select a protocol. (**`Email`**)
    - **Subscription Email:** Enter the email address. (**`admins@example.com`**)
- **Message grouping:** Select `Group notifications across metric streams` to collectively track metric status across all metric streams. Send a message when metric status across all metric streams changes. If you're monitoring many resources, this selection is less likely to flood the subscription protocol. (**`Group notifications across metric streams`**)
- **Message Format:** Select an option for the appearance of messages that you receive. (**`Send formatted messages`**)
- Don't select **Repeat notification** or **Suppress notifications**.

5. Ensure that **Enable this alarm** is selected.
6. Select **Save alarm**.

More alarm resources:

- Monitoring Overview
- Viewing an Alarm Metric Chart
- Best Practices for Your Alarms

Compute Cloud@Customer Metrics Reference

See a list of metrics emitted by Compute Cloud@Customer using the `oci_ccc` metric namespace.

Each metric includes the following dimensions:

- **RESOURCEID**: The OCID of the Compute Cloud@Customer infrastructure
- **RESOURCENAME**: The name of the Compute Cloud@Customer infrastructure

Metric Name	Metric Display Name	Unit	Description	Collection Frequency (minutes)	Dimensions

`ComputeOCPUsTotal`	OCPU Total	integer	Total number of OCPUs	15	`resourceId`
					`resourceName`
					`faultDomain`
					`computeNode`

Metric Name	Metric Display Name	Unit	Description	Collection Frequency (minutes)	Dimensions

`ComputeOCPUsAvailable`	OCPU Available	integer	Number of available OCPUs	15	`resourceId`
					`resourceName`
					`faultDomain`
					`computeNode`

`ComputeMemoryTotal`	Memory Total	GB	Total amount of compute memory	15	`resourceId`
					`resourceName`
					`faultDomain`
					`computeNode`

`ComputeMemoryAvailable`	Memory Available	GB	Amount of available compute memory	15	`resourceId`
					`resourceName`
					`faultDomain`
					`computeNode`

`StorageSpaceTotal`	Storage Total	GB	Total storage capacity	15	`resourceId`
					`resourceName`
					`storageNode`

`StorageSpaceAvailable`	Storage Available	GB	Amount of available storage capacity	15	`resourceId`
					`resourceName`
					`storageNode`

Virtual Cloud Networks

On Compute Cloud@Customer, networking enables you to set up virtual versions of traditional network components.

When you work with Compute Cloud@Customer, one of the first steps is to set up a virtual cloud network (VCN) for your cloud resources.

The infrastructure that provides the necessary services to deploy cloud workloads is configured to operate within the network environment of your data center. During initialization, the infrastructure's core network components are integrated with your existing data center network design.

Virtual Network Interface Cards (VNICs)

The compute nodes in Compute Cloud@Customer have physical network interface cards (NICs). When you create a compute instance on one of the servers, the Networking service ensures that a VNIC is created on top of a physical interface, so that the instance can communicate over the network. Each instance has a primary VNIC that is automatically created and attached. The primary VNIC resides in the subnet you specify when creating the instance. It can't be removed from the instance.

A VNIC enables an instance to connect to a VCN and determines how the instance communicates with endpoints inside and outside of the VCN.

For more information, see Configuring VNICs on page 181.

IP Addressing

Instances use IP addresses for communication. Each instance has at least one private IP address and optionally one or more public IP addresses. A private IP address enables the instance to communicate with other instances inside the VCN. A public IP address enables the instance to communicate with hosts outside of the cloud network environment.

Certain types of resources are designed to be directly reachable from outside the secure Compute Cloud@Customer network environment, and therefore automatically come with a public IP address. For example: a NAT gateway. Other types of resources are directly reachable only if you configure them to be. For example: specific instances in your VCN. Direct public connectivity also requires that the VCN has an internet gateway and that the public subnet has correctly configured route tables and security lists.

> **Note:**
>
> You can only configure IPv4 IP addresses. You can't configure IPv6 IP addresses.

For more information, see Managing Private IP Addresses on page 186 and Managing Public IP Addresses on page 191.

DHCP Options

The Networking service uses DHCP to automatically provide configuration information to instances when they boot up. Although DHCP lets you change some settings dynamically, others are static and never change. For example, when you create an instance, either you specify the instance's private IP address or the system chooses one for you. Each time the instance boots up or you restart the instance's DHCP client, DHCP passes that same private IP address to the instance. The address never changes during the instance's lifetime.

The Networking service provides DHCP options to let you control certain types of configuration on the instances in your VCN. You can change the values of these options at your discretion, and the changes take effect the next time you restart an instance's DHCP client or reboot the instance.

For more information, see Working with DHCP Options on page 151.

Traffic Management Steering Policies

Steering Policies enable you to configure policies to serve intelligent responses to DNS queries, meaning different answers (endpoints) may be served for the query depending on the logic defined in the policy.

For more information, see Managing Traffic with Steering Policies on page 204

Name Resolution

On Compute Cloud@Customer, the Domain Name System (DNS) translates human-readable domain names to machine-readable IP addresses.

For example, when you type a domain name into a browser, the OS queries several DNS name servers until it finds the authoritative name server for that domain. The authoritative name server then responds with an IP address or other requested record data; the answer is relayed back to your browser and the DNS record is resolved to the web page.

For DNS name resolution within your VCN provides two options. You choose an option for each subnet in the VCN, using the subnet's set of DHCP options.

The default choice is the **Internet and VCN Resolver**. It consists of two parts: the Internet Resolver and the VCN Resolver. The Internet Resolver lets instances resolve host names that are publicly published on the internet, without requiring the instances to have internet access by way of either an internet gateway or a connection to the on-premises network. The VCN resolver lets instances resolve host names, which you can assign, of other instances in the same VCN.

Or, you can use a **Custom Resolver**. This enables you to use up to three DNS servers of your choice for name resolution. These could be DNS servers that are available through the internet, in your VCN, or in your on-premises network, which is connected to your VCN through a DRG.

Domains and Host Names

When you create a VCN and subnets, you can specify DNS labels for each. Subnet DNS labels can only be set if the VCN itself is created with a DNS label. The labels, along with the parent domain of `oraclevcn.com` form the VCN domain name and subnet domain name. Next, when you create an instance, you can assign a host name. The host name is assigned to the primary VNIC that is automatically created during instance creation. Along with the subnet domain name, the host name forms the instance's fully qualified domain name (FQDN).

- **VCN domain name: `<VCN_DNS_label>.oraclevcn.com`**
- **Subnet domain name: `<subnet_DNS_label>.<VCN_DNS_label>.oraclevcn.com`**
- **Instance FQDN: `<host_name>.<subnet_DNS_label>.<VCN_DNS_label>.oraclevcn.com`**

The FQDN resolves to the instance's private IP address. The Internet and VCN Resolver also enables reverse DNS lookup, which lets you decide the host name corresponding to the private IP address.

If you add a secondary VNIC to an instance, you can specify a host name. The resulting FQDN resolves to the VNIC primary private IP address. The resulting FQDN resolves to that private IP address.

DNS labels and host names must meet these requirements:

- VCN and subnet labels must start with a letter and have a maximum length of 15 alphanumeric characters. Hyphens and underscores are not allowed. The value can't be changed later.
- Host names can be up to 63 characters in length and must be compliant with RFCs 952 and 1123. The value can be changed later.
- VCN DNS labels must be unique across the VCNs in your tenancy. Subnet DNS labels must be unique within the VCN, and host names must be unique within the subnet.

Host Name Validation and Generation

If you set DNS labels for VCN and subnet, the host name is validated for compliance during instance creation. If you haven't specified a host name, the system tries to use the instance display name. If the display name doesn't pass validation, or if you didn't specify one, the system generates a DNS-compliant host name.

If you create an instance using the CLI or SDK, and you haven't specified a host name or display name, the system doesn't generate them for you. This means the instance isn't resolvable using the Internet and VCN Resolver.

If you add a secondary VNIC to an instance, the system never generates a host name. You must provide a valid host name if you want the private IP address to be resolvable using the Internet and VCN Resolver.

Public DNS Zones

To enable DNS name resolution for instances deployed inside your VCNs, from outside the Compute Cloud@Customer network environment, Compute Cloud@Customer provides configuration support for public DNS zones. Within your tenancy, you create the DNS zones, or sections of the DNS namespace, that you intend to manage. The edge network of the Compute Cloud@Customer handles all DNS queries for the domains.

For more information, see Managing Public DNS Zones on page 198.

Networking Scenarios

Learn about various networking scenarios for Compute Cloud@Customer.

The following sections describe basic networking scenarios to help you understand the networking service and how networking components operate together.

> **Important:**
>
> Regardless of the selected networking scenario, always ensure that the IP address ranges configured as Compute Cloud@Customer public IPs don't conflict with IP addresses used within the data center network.

> When configuring virtual cloud networks (VCNs) with a dynamic routing gateway (DRG), ensure that the VCN IP address ranges don't overlap with each other or with the on-premises network.

- Scenario A – Public Subnet on page 130 describes a setup consisting of a VCN and a public subnet on Compute Cloud@Customer.
- Scenario B – Private Subnet on page 131 describes a setup consisting of a VCN and a private subnet on Compute Cloud@Customer.
- Scenario C – Public and Private Subnets on page 132 describes a multitier setup consisting of a VCN with a public and a private subnet on Compute Cloud@Customer.
- Scenario D - Transit Routing Through a Hub VCN on page 134 describes a network topology where your on-premises network connects to a VCN on Compute Cloud@Customer to reach resources beyond that VCN.

Scenario A – Public Subnet

This scenario describes a setup consisting of a VCN and a public subnet on Compute Cloud@Customer.

For external connectivity the VCN needs an internet gateway. Your on-premises network also uses this gateway to communicate with resources inside the VCN. The IP addresses used in this scenario must be public. In a private cloud context, this means a unique address directly reachable from the on-premises network.

The subnet uses the default security list, which has default rules that are designed to simplify getting started. The rules enable typical required access; for example inbound SSH connections and any type of outbound connections. Remember that security list rules only allow traffic. Any traffic not explicitly covered by a security list rule is implicitly denied. In this scenario, you add more rules to the default security list. You could instead create a custom security list for those rules. You would then set up the subnet to use both the default security list and the custom security list.

The subnet uses the default route table, which contains no rules when the VCN is created. In this scenario, the table has only a single rule: to route traffic intended for all destinations (0.0.0.0/0) through the internet gateway.

To set up this networking scenario, you perform the following steps:

1. Create the VCN.

   Choose a compartment you have permission to work in. Specify one or more non-overlapping CIDR blocks for the VCN; for example: 172.16.0.0/16. Optionally, enable DNS and specify a DNS label for the VCN.
2. Create the public subnet.

   Specify a single, contiguous CIDR block within the VCN CIDR block; for example: 172.16.10.0/24. Select the default route table. Ensure the subnet is a public subnet, so that instances can obtain public IP addresses. If you enabled DNS at the VCN level, you can choose to assign host names in the subnet and specify a subnet DNS label as well.
3. Create the internet gateway.

   When you create the internet gateway, it's enabled immediately. However, you must add a route rule to allow traffic to flow to the gateway.
4. Update the default route table to use the internet gateway.

   The default route table starts out with no rules. No route rule is required to route traffic within the VCN itself. You must add a rule that routes all traffic destined for addresses outside the VCN to the internet gateway. Enter these parameters:

   - Target Type: Internet Gateway
   - Destination CIDR block: 0.0.0.0/0

     This means that all non-intra-VCN traffic that's not already covered by other rules in the route table goes to the target specified in this rule.
   - Target: The internet gateway you created.

   Because the subnet was set up to use the default route table, the resources in the subnet can now use the internet gateway. The existence of this rule also enables inbound connections to the subnet, through the internet gateway.

The next step is to specify the types of traffic you want to allow into and out of the instances you later create in the subnet.

5. Update the default security list.

   You set up the subnet to use the VCN default security list. Now you add security list rules that allow the types of connections that the instances in the VCN need.

   For example, if the instances in your subnet are web servers, they likely need to receive inbound HTTPS connections. To enable that traffic, add an ingress rule to the default security list using these parameters:

   - Source Type: CIDR
   - Source CIDR: 0.0.0.0/0
   - IP Protocol: TCP
   - Source Port Range: All
   - Destination Port Range: 443

6. Create instances.

   Your next step is to create one or more instances in the subnet. Each instance automatically gets a private IP address. With the network setup in this scenario, you must give each instance a public IP address, otherwise you can't access them through the internet gateway.

Scenario B – Private Subnet

This scenario describes a setup consisting of a VCN and a private subnet on Compute Cloud@Customer.

For connectivity to your on-premises network, the VCN needs a dynamic routing gateway (DRG).

The subnet uses the default security list, which has default rules that are designed simplify getting started. The rules enable typical required access; for example inbound SSH connections and any type of outbound connections. Remember that security list rules only allow traffic. Any traffic not explicitly covered by a security list rule is implicitly denied. In this scenario, you add more rules to the default security list. You could instead create a custom security list for those rules. You would then set up the subnet to use both the default security list and the custom security list.

The subnet uses the default route table, which contains no rules when the VCN is created. In this scenario, the table has only a single rule: to route traffic intended for all destinations (0.0.0.0/0) through the DRG.

To set up this networking scenario, you perform the following steps:

1. Create the VCN.

   Choose a compartment you have permission to work in. Specify one or more non-overlapping CIDR blocks for the VCN; for example: 172.16.0.0/16. Optionally, enable DNS and specify a DNS label for the VCN.

2. Create the private subnet.

   Specify a single, contiguous CIDR block within the VCN CIDR block; for example: 172.16.10.0/24. Make the subnet private; the instances you create can't obtain a public IP address. Select the default route table. If you enabled DNS at the VCN level, you can choose to assign host names in the subnet and specify a subnet DNS label as well.

3. Update the default security list.

   You set up the subnet to use the VCN default security list. Now you add security list rules that allow the types of connections that the instances in the VCN will need.

   For example, if your subnet contains Microsoft Windows instances and you intend to access them using RDP, add an ingress rule to the default security list using these parameters:

   - Source Type: CIDR
   - Source CIDR: 0.0.0.0/0
   - IP Protocol: TCP
   - Source Port Range: All
   - Destination Port Range: 3389

4. Create a dynamic routing gateway (DRG) and attach it to your VCN.

   When you create the DRG, it is in "Provisioning" state for a short period. Ensure provisioning is done before continuing. Next, attach the DRG you just created to your VCN. For this scenario you can ignore the advanced attachment options. The DRG attachment will be in "Attaching" state for a short period before it's ready.

   To allow traffic to flow to the DRG, you must add a route rule.
5. Update the default route table to use the DRG.

   The default route table starts out with no rules. No route rule is required to route traffic within the VCN itself. You must add a rule that routes all traffic destined for addresses in your on-premises network to the DRG. Enter these parameters:

   - Target Type: Dynamic Routing Gateway.

     The VCN attached DRG is automatically selected as the target.
   - Destination CIDR block: 0.0.0.0/0

     This means that all non-intra-VCN traffic that is not already covered by other rules in the route table goes to the target specified in this rule.

   Because the subnet was set up to use the default route table, the DRG now enables traffic between the resources in the subnet and in your on-premises network.
6. Create instances.

   Your next step is to create one or more instances in the subnet. Each instance automatically gets a private IP address. With the network setup in this scenario, no additional configuration is required to access the instances from your on-premises network.

Scenario C – Public and Private Subnets

This scenario describes a multitier setup consisting of a VCN with a public and a private subnet on Compute Cloud@Customer.

The public subnet holds public instances such as web servers, and the private subnet holds private instances such as database servers. The VCN has a dynamic routing gateway (DRG) for connectivity to your on-premises network. Instances in the public subnet have external access through an internet gateway.

> **Note:**
>
> In a public cloud environment, instances in the private subnet could be allowed to initiate external connections using a NAT gateway, for example to get software updates. However, in Compute Cloud@Customer the NAT gateway would provide access to the on-premises network, which the DRG already enables for those instances. The combination of a NAT gateway and DRG could cause issues because of nondeterministic routing.

Each subnet uses the default security list, which has default rules that are designed to simplify getting started. The rules enable typical required access; for example inbound SSH connections and any type of outbound connections. Remember that security list rules only allow traffic. Any traffic not explicitly covered by a security list rule is implicitly denied.

Each subnet also has its own custom security list and custom route table with rules specific to the needs of the subnet's instances. In this scenario, the VCN default route table, which is always empty to start with, is not used.

To set up this networking scenario, you perform the following steps:

1. Create the VCN.

   Choose a compartment you have permission to work in. Specify one or more nonoverlapping CIDR blocks for the VCN; for example: 172.16.0.0/16. Optionally, enable DNS and specify a DNS label for the VCN.
2. Add the necessary gateways to the VCN.

   The instances in the public subnet need an internet gateway for incoming and outgoing public traffic. The instances in the private subnet need a NAT gateway to be able to reach the data center network and the internet.

These gateways are enabled immediately upon creation, but you must add route rules to allow traffic to flow to them.

To reach the private instances in the VCN from your on-premises network, you create a DRG and attach it to the VCN. When you create the DRG, it's in "Provisioning" state for a short period. Ensure provisioning is done before attaching it to the VCN. For this scenario you can ignore the advanced attachment options. The DRG attachment will be in an "Attaching" state for a short period before it's ready. To allow traffic to flow to the DRG, you must also add a route rule.

3. Create custom route tables for the subnets that you create later.
   a. For the public subnet, create a route table and add a rule that routes all traffic destined for addresses outside the VCN to the internet gateway. Enter these parameters:
      - Target Type: Internet Gateway
      - Destination CIDR block: 0.0.0.0/0

        This means that all non-intra-VCN traffic that's not already covered by other rules in the route table goes to the target specified in this rule.
      - Target: The internet gateway you created.
   b. For the private subnet, create a route table and add two rules: one that routes traffic destined for the on-premises network to the DRG, and one that routes all other traffic leaving the VCN to the NAT gateway.

      Create the route rule for the NAT gateway with these parameters:
      - Target Type: NAT Gateway
      - Destination CIDR block: 0.0.0.0/0

        This means that all nonintra-VCN traffic that's not already covered by other rules in the route table goes to the target specified in this rule.
      - Target: The NAT gateway you created.

      Create the route rule for the DRG with these parameters:
      - Target Type: Dynamic Routing Gateway.

        The VCN's attached DRG is automatically selected as the target.
      - Destination CIDR block: 10.25.0.0/16

        This means that traffic intended for an address in the on-premises network (10.25.x.y) goes to the DRG target specified in this rule.
4. Update the default security list.

   Add security list rules that allow the types of connections that the instances in the VCN will need.

   Edit each of the existing stateful ingress rules so that the Source CIDR is **not** 0.0.0.0/0, but the CIDR for your on-premises network; in this example: 10.25.0.0/16.
5. Create custom security lists for the subnets you will create later.
   a. Create a custom security list for the public subnet and add rules to allow the types of connections that the public instances need. For example, web servers likely need to receive HTTP and HTTPS ingress traffic. For HTTP, use the following settings. For HTTPS, add another similar rule for TCP port 443.
      - Source Type: CIDR
      - Source CIDR: 0.0.0.0/0
      - IP Protocol: TCP
      - Source Port Range: All
      - Destination Port Range: 80
   b. Create a custom security list for the private subnet and add rules to allow the types of connections that the private instances need. For example, database servers likely need to receive SQL*Net (TCP port 1521) ingress traffic from clients in the private and the public subnet. For clients in the public subnet, use the following settings. For clients in the private subnet, add another similar rule for the CIDR of the private subnet (172.16.1.0/24).

- Source Type: CIDR
- Source CIDR: 172.16.2.0/24
- IP Protocol: TCP
- Source Port Range: All
- Destination Port Range: 1521

6. Create the subnets in the VCN.
   - Public subnet:

     Specify a single, contiguous CIDR block within the VCN CIDR block; for example: 172.16.2.0/24. Ensure the subnet is a public subnet, so that instances can obtain public IP addresses. Select the custom public subnet route table you created earlier.

     Select two security lists: both the default security list and the public subnet security list you created earlier. If you enabled DNS at the VCN level, you can choose to assign host names in the subnet and specify a subnet DNS label as well.
   - Private subnet:

     Specify a single, contiguous CIDR block within the VCN CIDR block; for example: 172.16.1.0/24. Make the subnet private; the instances you create in this subnet can't obtain a public IP address. Select the custom private subnet route table you created earlier.

     Select two security lists: both the default security list and the private subnet security list you created earlier. If you enabled DNS at the VCN level, you can choose to assign host names in the subnet and specify a subnet DNS label as well.
7. Create instances.

   Your next step is to create instances in the subnets. Each instance automatically gets a private IP address. For each instance in the public subnet, ensure to assign the instance a public IP address. Otherwise, you won't be able to reach the instance from your on-premises network. The DRG you set up allows you to reach the instances in the private subnet from your on-premises network without any additional configuration.

Scenario D - Transit Routing Through a Hub VCN

This scenario describes a network topology where the on-premises network connects to a VCN on the Compute Cloud@Customer to reach resources beyond that VCN. Routing rules in this VCN are configured so that traffic *transits through* the VCN to its destination.

As a basic example of why you might use transit routing: you have a large organization with different departments, each with their own VCN. The on-premises network needs access to the different VCNs, but you do not want the administration overhead of maintaining a secure connection from each VCN to the on-premises network. Instead, you use VCNs in a hub-and-spoke topology: one VCN acts as the hub to which the on-premises network connects, and the spoke VCNs are peered with the hub VCN. Traffic between the on-premises network and the peered VCNs transits through the hub VCN. The VCNs can be in different tenancies.

Important Aspects of Transit Routing

Transit routing introduces specific component configurations that a network administrator must take into account. These and other important aspects are discussed in this section.

Notable Concepts

If you are already familiar with the Networking service and VCN peering, these are the most important concepts to understand when implementing a transit routing scenario:

- There are several different gateways involved in transit routing. The hub VCN has a dynamic routing gateway (DRG), which is the communication path with the on-premises network. For each peered spoke VCN, there is a pair of local peering gateways (LPGs) that anchor the peering connection. One LPG is on the hub VCN, and the other is on the spoke VCN.
- The Networking service does not allow local VCN peering between two VCNs with overlapping CIDRs. That means each spoke must not overlap with the hub. However, the Networking service does not validate whether

the spoke VCNs themselves overlap with each other, or if any of the VCNs overlap with the on-premises network. You must ensure that CIDRs for all the subnets that need to communicate with each other don't overlap. Otherwise, traffic might be dropped.

- For each spoke VCN subnet that needs to communicate with the on-premises network, you must update the subnet's route table with a rule that sets the target, or the next hop, as the spoke VCN's LPG for all traffic destined for the on-premises network.
- You must add a route table to the hub VCN and associate it with the DRG attachment. For each spoke VCN, add a route rule to the DRG attachment route table for all traffic destined for a given spoke VCN, or a subnet in the spoke VCN; set the target to the hub VCN's LPG for that spoke.
- You must add another route table to the hub VCN and associate it with the hub VCN's LPG for a given spoke. Add a route rule to the spoke LPG route table for all traffic destined for the on-premises network, or a subnet in that network; set the target to the DRG. The same route table setup is required for each spoke VCN that needs to communicate with the on-premises network.

Routing Restrictions

There are important restrictions to be aware of when implementing transit routing in your environment.

- **Route table for DRG attachment**

  A route table that is associated with a DRG attachment can have only rules that target an LPG or a private IP.

  A DRG attachment can exist without an associated route table. However, after you associate a route table with a DRG attachment, there must always be a route table associated with it. You can associate a different route table. You can also edit the table's rules, or delete some or all of the rules.
- **Route table for LPG**

  A route table that is associated with an LPG can have only rules that target a DRG.

  An LPG can exist without an associated route table. However, after you associate a route table with an LPG, there must always be a route table associated with it. You can associate a different route table. You can also edit the table's rules, or delete some or all of the rules.
- **Traffic through the hub VCN**

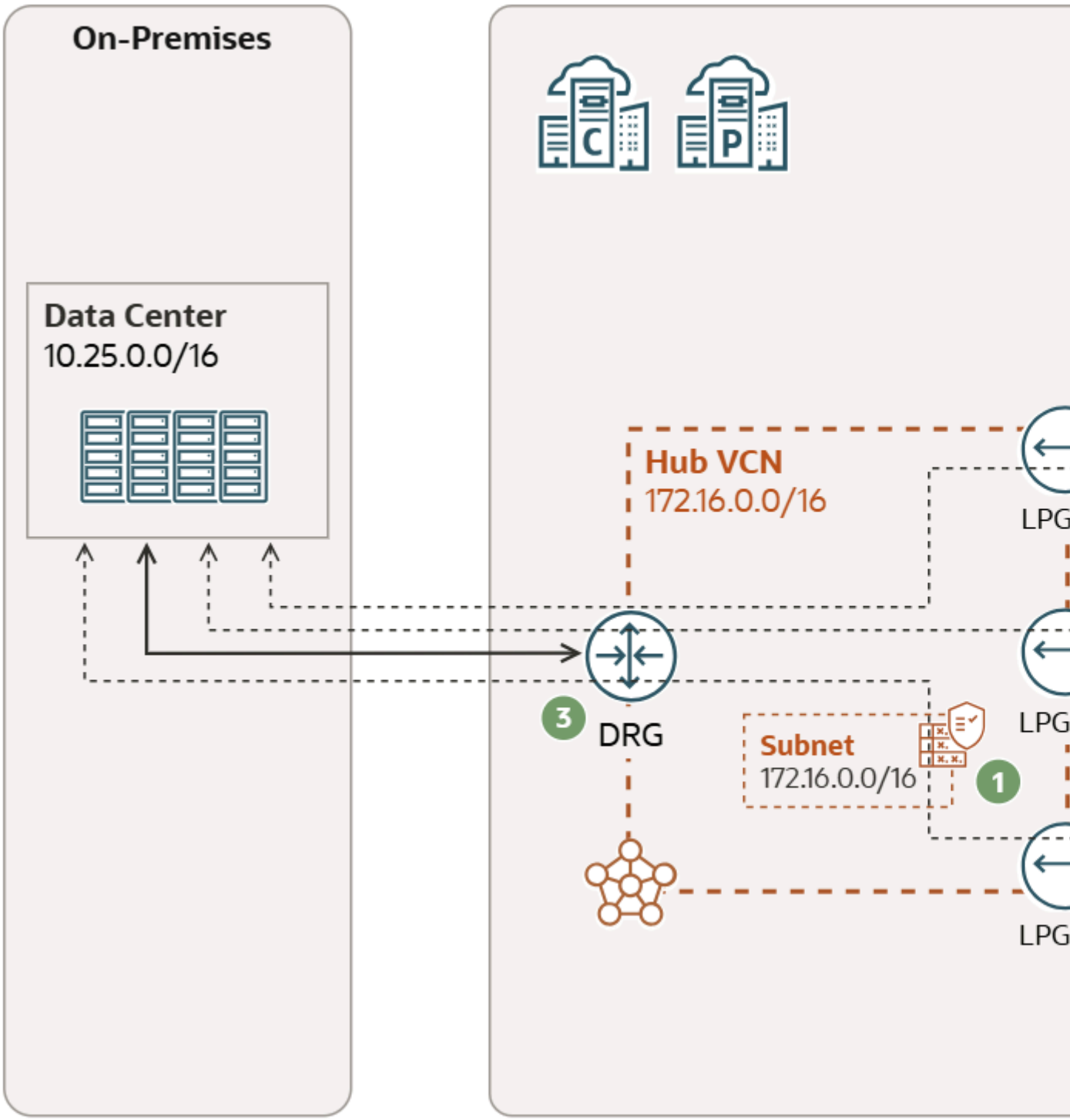
  The route tables discussed here are intended only for moving traffic *through* the hub VCN between locations in the on-premises network and locations in a spoke VCN. Routing traffic through a private IP in the hub is not supported on Compute Cloud@Customer.
- **Inbound traffic to the hub VCN**

  Although the purpose of transit routing is to move traffic through the hub VCN, inbound traffic to subnets within the hub VCN is always allowed. No explicit rules are required in the DRG attachment route table or the hub LPG route table.

  When this kind of inbound traffic reaches the DRG or the hub LPG, the traffic is automatically routed to its destination in the hub VCN by the VCN local routing. Consequently, for any route table belonging to a given VCN, you cannot create a rule that lists that VCN's CIDR, or a sub-section, as the rule's destination.

Routing Paths in Detail

To further illustrate how transit routing takes place, the different paths of traffic are displayed in the diagram, and described in detail below.



The callouts in the diagram indicate the route tables to be added for this configuration. The route rules required in this example are listed in the table that follows.

Route Tables

Callout	Route Table	Route Rules
1	Hub subnet	Destination CIDR Route Target 10.25.0.0/16 DRG 10.1.0.0/16 LPG-H1 10.2.0.0/16 LPG-H2 10.3.0.0/16 LPG-H3
2	Spoke subnet	For all spoke subnets, the route target must be the LPG of the parent VCN. Destination CIDR Route Target 172.16.0.0/16 LPG-S1 10.25.0.0/16 LPG-S1
3	DRG attachment	Destination CIDR Route Target 10.1.0.0/16 LPG-H1 10.2.0.0/16 LPG-H2 10.3.0.0/16 LPG-H3
4	Hub LPG	For each hub LPG (H1, H2, H3), traffic intended for the on-premises network must be sent through the DRG. Destination CIDR Route Target 10.25.0.0/16 DRG

These are the networking elements used in the example:

- On-premises network: 10.25.0.0/16
- Hub VCN: 172.16.0.0/16

  The hub VCN has a DRG to connect with the on-premises network. It also has an LPG for each peering connection to a spoke VCN, for example: LPG-H1, LPG-H2 and so on.
- Spoke VCN-S1: 10.1.0.0/16

  The spoke VCN has an LPG for the peering connection to the hub VCN: LPG-S1

  The spoke VCN contains subnets, for example: 10.1.1.0/24, 10.1.2.0/24 and so on.
- Spoke VCN-S2: 10.2.0.0/16

  The spoke VCN has an LPG for the peering connection to the hub VCN: LPG-S2

  The spoke VCN contains subnets, for example: 10.2.1.0/24, 10.2.2.0/24 and so on.
- Additional spoke VCNs follow the same pattern as VCN-S1 and VCN-S2.

Traffic from the on-premises network to the spoke VCN

1. Traffic leaves the on-premises network and reaches the DRG. The traffic's destination is in Subnet-1 of VCN-S1, for example: 10.1.1.5.
2. The DRG attachment's associated route table has a rule for 10.1.0.0/16. It matches the destination and sends the traffic to the route target: LPG-H1.

3. LPG-H1 receives the traffic.

   Egress traffic leaving a VCN through an LPG is automatically routed to the LPG's peered LPG, which is LPG-S1 in this situation. That routing occurs automatically because of the peering connection between the two LPGs.
4. LPG-S1 receives the traffic.

   Traffic coming in to a VCN through the LPG is automatically routed to the destination within the VCN because of VCN local routing. No explicit route rules are required.

Traffic from the spoke VCN to the on-premises network

1. Traffic comes from an instance in Subnet-1 of spoke VCN-S1. The traffic's destination is in the on-premises network, for example: 10.25.112.17.
2. Subnet-1's associated route table has a rule for 10.25.0.0/16. It matches the destination and sends the traffic to the route target: LPG-S1.
3. LPG-S1 receives the traffic.

   Egress traffic leaving a VCN through an LPG is automatically routed to the LPG's peered LPG, which is LPG-H1 in this situation. That routing occurs automatically because of the peering connection between the two LPGs.
4. LPG-H1 receives the traffic.

   LPG-H1's associated route table has a rule for 10.25.0.0/16. It matches the destination and sends the traffic to the route target: the hub VCN's DRG.
5. The DRG receives the traffic.

   Egress traffic leaving a VCN through the DRG is automatically routed to the destination in the connected on-premises network. No explicit route rules in the DRG attachment's route table are required.

Subnet-1 in the spoke VCN and LPG-H1 both have route rules with 10.25.0.0/16 as the destination CIDR. Those rules are not required to use the exact same CIDR block, but you must ensure that both rules cover the traffic you want to route from the spoke to the on-premises network. The rule in Subnet-1's route table controls which traffic is routed from Subnet-1 to LPG-H1. The rule in LPG-H1's route table controls which traffic is routed from the spoke VCN to the on-premises network. If LPG-H1's route rule is more restrictive than Subnet-1's route rule, some traffic leaving the subnet could be dropped and not reach the DRG.

Traffic from the spoke VCN to a subnet in the hub VCN

Depending on your situation, you might want to enable traffic between instances in the hub VCN and a spoke VCN, in addition to the traffic between the on-premises network and a spoke VCN. This is possible as long as routing only occurs directly between gateways, and not through a private IP in the hub VCN.

This is how traffic would flow from the spoke VCN to a destination with an address in the hub VCN:

1. Traffic comes from an instance in Subnet-1 of spoke VCN-S1. The traffic's destination is in HubSubnet-A of the hub VCN, for example: 172.16.2.85.
2. Subnet-1's associated route table has a rule for 172.16.0.0/16. It matches the destination and sends the traffic to the route target: LPG-S1.
3. LPG-S1 receives the traffic.

   Egress traffic leaving a VCN through an LPG is automatically routed to the LPG's peered LPG, which is LPG-H1 in this situation. That routing occurs automatically because of the peering connection between the two LPGs.
4. LPG-H1 receives the traffic.

   Traffic coming in to a VCN through the LPG is automatically routed to the destination within the VCN because of VCN local routing. No explicit route rules are required.

A similar series of routing steps occurs for traffic going from HubSubnet-A to Subnet-1, but in the reverse direction. HubSubnet-A's route table has a rule that matches the spoke VCN's CIDR (10.1.0.0/16) and sends the traffic to LPG-H1, which forwards it on to LPG-S1.

Setting Up VCN Transit Routing

To set up a transit routing scenario using direct traffic between gateways, you perform the tasks described below.

Task 1: Set up the hub VCN

1. Create a new VCN.

   Specify one or more non-overlapping CIDR blocks for the VCN; for example: 172.16.0.0/16.
2. Optionally, create one or more subnets in the hub VCN. The subnets can contain cloud resources that your on-premises network or the spoke VCN need to use.

   Specify a single, contiguous CIDR block within the VCN's CIDR block; for example: 172.16.1.0/24.

Task 2: Connect the hub VCN with the on-premises network

1. Create a DRG and attach it to the hub VCN.
2. Update the hub VCN route table to allow traffic to flow to the DRG.

   Select the on-premises network as the destination CIDR; for example: 10.25.0.0/16.

   The route table to be associated with the DRG attachment is created in a later task.

Task 3: Set up a spoke VCN with at least one subnet

1. Create a new VCN. In this example it is named VCN-S1.

   Specify one or more non-overlapping CIDR blocks for the VCN; for example: 10.1.0.0/16.
2. Create one or more subnets in the spoke VCN.

   Specify a single, contiguous CIDR block within the VCN's CIDR block; for example: 10.1.1.0/24.

Task 4: Set up local peering between the hub VCN and the spoke VCN

1. Add an LPG to the hub VCN and spoke VCN. In this example they are named LPG-H1 and LPG-S1 respectively.
2. Establish the peering connection between the two LPGs.
3. Update the hub VCN route table to allow traffic to flow to LPG-H1.

   Select the spoke VCN-S1 as the destination CIDR: 10.1.0.0/16.

   The route table to be associated with LPG-H1 is created in a later task.
4. Create a route table for Subnet-1 in spoke VCN-S1.

   Add a route rule to direct traffic destined for the hub VCN (172.16.0.0/16) to LPG-S1.
5. Associate the route table with Subnet-1 in spoke VCN-S1.
6. If other subnets in the spoke VCN need to communicate with the on-premises network, perform steps 4 and 5 with different parameters for each of those subnets in order to set up and associate a similar route table.

Task 5: Add a route rule to the spoke VCN's subnet

1. Edit the route table associated with Subnet-1 in spoke VCN-S1. You created this route table in the previous task.
2. Add a route rule for traffic destined for the on-premises network (10.25.0.0/16). Set the rule target to the LPG of this spoke VCN: LPG-S1.
3. Repeat these steps for other subnets in the spoke VCN that need to communicate with the on-premises network.

Task 6: Set up ingress routing for the DRG and LPG on the hub VCN

1. Create a route table for the DRG attachment.

   Add a route rule for traffic destined for the spoke VCN-S1 (10.1.0.0/16). Set the rule target to the hub VCN's LPG for the spoke in question: LPG-H1.
2. Associate the DRG route table with the hub VCN's DRG attachment.
3. Create a route table for the hub VCN's LPG for this spoke: LPG-H1.

   Add a route rule for traffic destined for the on-premises network (10.25.0.0/16). Set the rule target to the hub VCN's DRG.
4. Associate the route table with the hub VCN's LPG for the spoke in question: LPG-H1.

Later: adding more spoke VCNs

1. Repeat tasks 3-5 for the new spoke VCN.
2. Repeat task 6 with these changes:
    - For Step 1: Instead of creating a new route table for the DRG attachment, update the existing route table to include a new rule for the new spoke VCN. The destination CIDR is the spoke VCN's CIDR, or a subnet within). The target is the hub VCN's LPG for the new spoke.
    - For Step 2: Skip this step entirely because the DRG attachment is already associated with its route table.
    - For the entire procedure: use convenient names in accordance with the spokes you are adding. For example, if you are adding spoke VCN-S4, it is helpful to call the two peering gateways LPG-S4 and LPG-H4, and use those gateway names to identify the route tables you associate with them.

Turning Off Transit Routing

To turn off transit routing, remove the rules from:

- The route table associated with the DRG attachment.
- The route table associated with each LPG on the hub VCN.

A route table can be associated with a resource but have no rules. Without at least one rule, a route table does nothing.

A DRG attachment or LPG can exist without a route table associated with it. However, after you associate a route table with a DRG attachment or LPG, there must always be a route table associated with it. But, you can associate a *different* route table. You can also edit the table's rules, or delete some or all rules.

Managing VCNs and Subnets

On Compute Cloud@Customer, a virtual cloud network, or VCN, is a software-defined equivalent of a traditional network, with firewall rules and various types of communication gateways.

Virtual Cloud Network (VCN) Overview

A VCN resides in the single region of and covers one contiguous CIDR block of your choice. The size of a VCN is /16 to /30. The CIDR block can **NOT** be changed after the VCN is created. The maximum number of private IPs within a VCN is 64,000. Oracle recommends using a private IP address range as specified in RFC 1918 (10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16). This documentation uses the term private IP address when referring to IP addresses in your VCN CIDR.

You can privately connect a VCN to another VCN so that traffic doesn't leave the secure network environment of Compute Cloud@Customer. However, the CIDRs of the two VCNs must not overlap. The concept of privately connecting VCNs is called *peering*. It involves setting up a type of virtual router known as a Local Peering Gateway. For more information about the use of gateways in VCNs, see Configuring VCN Gateways on page 168.

Subnets

A VCN is subdivided into subnets. Each subnet in a VCN consists of a contiguous range of IPv4 addresses that don't overlap with other subnets in the VCN. The first two IPv4 addresses and the last in the subnet CIDR are reserved by the Networking service. You can't change the size of the subnet after creation.

Subnets act as a unit of configuration: all instances in a particular subnet use the same route table, security lists, and DHCP options. Subnets can be either public or private. This is defined when the subnet is created, and can't be changed later. In a private subnet, instances can't be assigned a public IP address.

You can think of a compute instance as residing in a subnet. However, to be precise, each instance is attached to a virtual network interface card (VNIC), which in turn resides in the subnet and enables a network connection for that instance.

Creating a VCN

On Compute Cloud@Customer, a *virtual cloud network* (VCN) a virtual, private network that closely resembles a traditional network. 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.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. Select **Create Virtual Cloud Network**.
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. **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
5. Select **Create Virtual Cloud Network**.

   The details page of the new VCN is displayed.

Using the CLI

Use the oci network vcn create command and required parameters to create a VCN.

```
oci network vcn create --compartment-id compartment_OCID --cidr-block
 10.0.0.0/16 --dns-label vcn1 --display-name VCN1 [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Procedure

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.

   To use DNS host names in the VCN, include the DNS label in the create command. It can't be added later.

   Optionally, set a descriptive name for the VCN.

```
$ oci network vcn create --compartment-id <compartment_OCID> \
--cidr-blocks '["<cidr-block_eg-10.0.0.0/16>"]' --dns-label <dns_label> --
display-name <display_name>
{
  "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": "2023-04-27T04:34:58.722835+00:00",
    "vcn-domain-name": "vcn1.oraclevcn.com"
  },
  "etag": "unique_ID"
}
```

Using the API

Use the CreateVCN operation to create a VCN.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Creating a Subnet

On Compute Cloud@Customer, a subnet is a subdivision of a VCN. For each VCN, you create one or more subnets.

VCNs can be divided into subnets. Although it's 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.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN in which you want to create a subnet.
3. Select the name of the VCN in which you want to create a new subnet.

   The VCN details page is displayed.
4. In the **Resources** section, select **Subnets**.
5. Select **Create Subnet**.
6. 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.

     > **Important:**
     >
     > The CIDR block range is an important subnet parameter. The range must be within the VCN CIDR block and must not overlap with other subnet CIDR block ranges. These IP addresses can't 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 to assign a DNS hostname when you launch an instance in this subnet. If you check the box, enter a DNS label that's 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, select +Add Security List. Select a security list to associate with the subnet. You might need to change the compartment selection. If you want

another security list, select +Add Security List and select another security list. If you don't select a security list, the VCN default security list is used.

7. **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
8. Select **Create Subnet**.

   The details page of the new subnet is displayed.

Using the CLI

Use the oci network subnet create command and required parameters to create a subnet in a specified VCN.

```
oci network subnet create --compartment-id compartment_OCID --vcn-
id vcn_OCID --cidr-block cidr_block --dns-label dns_label --display-
name display_name [OPTIONS]
```

> **Important:**
>
> The CIDR block range is an important subnet parameter. The range must be within the VCN CIDR block and must not overlap with other subnet CIDR block ranges. These IP addresses can't be shared.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateSubnet operation to create a subnet in a specified VCN.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Editing a Subnet

On Compute Cloud@Customer, you can edit subnets. You can change the subnet name, the route tables and security lists used by the subnet, and DHCP options.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN and subnet.
3. Select the name of the VCN that contains the subnet you want to edit.

   The VCN details page is displayed.
4. Under **Resources**, select **Subnets**.
5. Select the Actions menu (⋮), then select **Edit**.
6. Make the changes to the subnet. The following items can be changed:
   - **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.
7. **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
8. Select **Save Changes**.

   The subnet properties are updated.

Using the CLI

Use the oci network subnet update command and required parameters to update a 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 [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateSubnet operation to update a subnet.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Subnet

On Compute Cloud@Customer, to delete a subnet, it must contain no resources (instances, VNICs).

For more information about deleting VCNs and Subnets, see Subnet or VCN Deletion

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN and subnet.
3. Select the name of the VCN that contains the subnet you want to delete.
4. Under **Resources**, select **Subnets**.
5. Select the Actions menu (⋮), select **Delete**.
6. Confirm the deletion.

Using the CLI

Use the oci network subnet delete command and required parameters to delete a subnet.

```
oci network subnet delete --subnet-id subnet_OCID [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteSubnet operation to delete a subnet.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a VCN

On Compute Cloud@Customer, a VCN can only be deleted if it's empty. Before deleted a VCN, ensure that all subnets, route tables, gateways, and other resources have been deleted.

For more information about deleting VCNs and Subnets, see Subnet or VCN Deletion

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN.
3. Select the name of the VCN that you want to delete.

   The VCN details page is displayed.
4. Under **Resources**, ensure that the VCN doesn't have any resources.

If all the resources such as Subnets, Route Tables, and so on include a zero in parenthesis, then the VCN doesn't have any resources.

If a resource exists, you must delete the resource before you can delete the VCN.

5. Select **Delete**.
6. Confirm the deletion.

Using the CLI

Use the oci network vcn delete command and required parameters to delete a VCN.

```
oci network vcn delete --vcn-id vcn_OCID [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteVCN operation to delete a VCN.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Configuring VCN Rules and Options

On Compute Cloud@Customer, 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.

VCN Security Rules and Security Lists

On Compute Cloud@Customer, VCN security involves creating and using rules that are gathered into security lists or network security groups (NSGs).

The rules can be stateful or stateless. Stateful rules presume various things about an instance interaction. For example, a stateful rule that applies to a request to an instance assumes that there is a response and automatically allows this response. In contrast, stateless rules don't presume anything about any interactions. Stateless rules are messages from one instance to another, and apply to everything regardless of what has gone before. Stateful rules require overhead processing to track states that stateless rules don't. Instances with heavy and widely varied traffic, such as web sites, should use stateless rules where possible to minimize system loads.

> **Note:**
>
> Care is needed when creating stateless and stateful rules. When a message appears that fits both stateful and stateless rule criteria, the stateless rule is applied. This results in responses being blocked to requests that seemingly should be allowed.

Your VCN might have subnets that use the default security list. Don't delete any of the list's default security rules unless you've first confirmed that resources in your VCN don't require them. Otherwise, you might disrupt VCN connectivity.

Remember that:

- A security rule is stateful by default, but can also be configured to be stateless. A common practice is to use stateless rules for high-performance applications. In a case where network traffic matches both stateful and stateless security lists, the stateless rule takes precedence. For more information about configuring VCN security rules, see Security Rules.
- To prevent unauthorized access or attacks on Compute instances, we recommend that you use a VCN security rule to allow SSH or RDP access only from authorized CIDR blocks rather than leave them open to the internet (0.0.0.0/0). For additional security, you can temporarily enable SSH (port 22) or RDP (port 3389) access on an as-needed basis using the VCN API `UpdateNetworkSecurityGroupRules` (if you're using network security groups) or `UpdateSecurityList`(if you're using security lists).

For more information about enabling RDP access, see **To enable RDP access** in Recommended Networking Launch Types.

For performing instance health checks, we recommend that you configure VCN security rules to allow ICMP pings. For more information, see Rules to Enable Ping.

VCN network security groups (NSGs) and security lists enable security-critical network access control to Compute instances, and it's important to prevent any unintended or unauthorized changes to NSGs and security lists. To prevent unauthorized changes, we recommend that you use IAM policies to allow only network administrators to make NSG and security list changes.

The default security list for a VCN comes without stateless rules: all the default security list rules are stateful. In most cases, the default rules should be changed to allow only inbound traffic from authorized subnets.

Default Security List Rules

- **Stateful ingress:** Allow TCP traffic on destination port 22 (SSH) from authorized source IP addresses and any source port. This rule makes it easy for you to create a new cloud network and public subnet, open a Linux instance, and then immediately use SSH to connect to that instance without needing to write any security list rules yourself.
- **Stateful ingress:** Allow ICMP traffic type 3 code 4 from authorized source IP addresses. This rule enables instances to receive Path MTU Discovery fragmentation messages.
- **Stateful ingress:** Allow ICMP traffic type 3 (all codes) from your VCN CIDR block. This rule makes it easy for your instances to receive connectivity error messages from other instances within the VCN.
- **Stateful egress:** Allow all traffic. This allows instances to initiate traffic of any kind to any destination. Notice that this means the instances with public IP addresses can talk to any internet IP address if the VCN has a configured internet gateway. And because stateful security rules use connection tracking, the response traffic is automatically allowed regardless of any ingress rules.

Security List Rule Modifications

RDP

The default security list doesn't include a rule to allow Remote Desktop Protocol (RDP) access. If you're using Microsoft Windows images, ensure to add a stateful ingress rule for TCP traffic on destination port 3389 from authorized source IP addresses and any source port.

Instance Pings

The default security list doesn't include a rule to allow ping requests. If you plan to ping an instance, ensure that the instance's applicable security list includes an additional stateful ingress rule to allow ICMP traffic type 8 from the source network you plan to ping from. To allow ping access from the data center, use `0.0.0.0/0` for the source. Note that this rule for pinging is separate from the default ICMP-related rules in the default security list. Don't remove those rules.

Path MTU Discovery Fragmentation Messages

If you decide to use stateless security rules to allow traffic to and from endpoints outside the VCN, it's important to add a security rule that allows ingress ICMP traffic type 3 code 4 from source `0.0.0.0/0` and any source port. This rule enables instances to receive Path MTU Discovery fragmentation messages. This rule is critical for establishing a connection to instances. Without it, you can experience connectivity issues.

UDP

Instances can send and receive UDP traffic. In some cases, UDP packets might be fragmented on links with smaller MTU sizes. If a UDP packet is too large for the connection size limit, it's fragmented. However, only the first fragment from the packet contains the protocol and port information. If the security rules that allow this ingress or egress traffic specify a particular port number (source or destination), then the fragments after the first one are dropped. If you expect your instances to send or receive large UDP packets, set both the source and destination ports for the applicable security rules to ALL (instead of a particular port number).

Subnets and Security Rules

On their own, security lists define a set of security rules that apply to all the VNICs in an entire subnet. To use a specific security list with a particular subnet, associate the security list with the subnet either during subnet creation or later. A subnet can be associated with a maximum of five security lists. Any VNICs that are created in that subnet are subject to the security lists associated with the subnet.

Virtual Firewall

On Compute Cloud@Customer, the Networking service offers two virtual firewall features that both use security rules to control traffic at the packet level – security lists and network security groups (NSGs). They offer different ways to apply security rules to a set of virtual network interface cards (VNICs).

- **Security lists:**

  A security list defines security rules at the subnet level, which means that all VNICs in a particular subnet are subject to the same rules. Each VCN comes with a default security containing default rules for essential traffic. The default security list is automatically used with all subnets, unless a custom security list is specified. A subnet can have up to five associated security lists.
- **Network security groups (NSGs):**

  A network security group defines security rules based on membership. Its security rules apply to resources that are explicitly added to the NSG. A VNIC can be added to a maximum of five NSGs. An NSG is intended to provide a virtual firewall for a set of cloud resources with the same security posture. For example: a group of instances that perform the same tasks and thus need to use the same set of ports.

Oracle recommends using NSGs instead of security lists because NSGs let you separate the VCN subnet architecture from your application security requirements. However, NSGs are only supported for specific services. It is possible to use both security lists and NSGs together, depending on your particular security needs.

If you have security rules that you want to enforce for all VNICs in a VCN, the easiest solution is to put the rules in one security list, and then associate that security list with all subnets in the VCN. This way you can ensure that the rules are applied, regardless of who in your organization creates a VNIC in the VCN. Or, you can add the required security rules to the VCN default security list.

If you choose to combine security lists and network security groups, the set of rules that applies to a particular VNIC is the union of these items:

- The security rules in the security lists associated with the VNIC subnet
- The security rules in all NSGs that the VNIC is in

A packet in question is allowed if *any rule* in any of the relevant lists and groups allows the traffic, or if the traffic is part of an existing connection being tracked because of a stateful rule.

Security Rules

On Compute Cloud@Customer, you can configure security rules for security lists and network security groups (NSGs).

This section explains important aspects of security rules that you need to understand to implement them. How you create, manage, and apply security rules varies between security lists and network security groups.

Parts of a Security Rule

A security rule allows a particular type of traffic into or out of a VNIC. For example, a commonly used security rule allows ingress TCP port 22 traffic for establishing SSH connections to the instance. Without security rules, no traffic is allowed into and out of VNICs in the VCN.

Each security rule specifies the following items:

- **Direction (ingress or egress):** Ingress is inbound traffic to the VNIC; egress is outbound traffic from the VNIC.

  The REST API model for security lists is different from network security groups. With security lists, there is an `IngressSecurityRule` object and a separate `EgressSecurityRule` object. With network security

groups, there is only a `SecurityRule` object, and the object's `direction` attribute determines whether the rule is for ingress or egress traffic.

- **Stateful or stateless:** If stateful, connection tracking is used for traffic matching the rule. If stateless, no connection tracking is used. See Stateful and Stateless Rules in this section.
- **Source type and source:** For ingress rules only; the source you provide depends on the source type you choose. These source types are allowed:

Source Type	Allowed Source
CIDR	The CIDR block where the traffic originates from. Use 0.0.0.0/0 to indicate all IP addresses. The prefix is required. For example, include the /32 if specifying an individual IP address.

- **Destination type and destination:** For egress rules only; the destination you provide depends on the destination type you choose. These destination types are allowed:

Destination Type	Allowed Destination
CIDR	The CIDR block that the traffic is destined for. Use 0.0.0.0/0 to indicate all IP addresses. The prefix is required. For example, include the /32 if specifying an individual IP address.

- **IP Protocol:** Either a single IPv4 protocol or "all" to cover all protocols.
- **Source port:** The port where the traffic originates from. For TCP or UDP, you can specify all source ports, or optionally specify a single source port number, or a range.
- **Destination port:** The port where the traffic is destined to. For TCP or UDP, you can specify all destination ports, or optionally specify a single destination port number, or a range.
- **ICMP type and code:** For ICMP, you can specify all types and codes, or optionally specify a single type with an optional code. If the type has multiple codes, create a separate rule for each code you want to allow.
- **Description:** NSG security rules contain an optional attribute to include a description of the rule. This is currently not supported for security list rules.

Stateful and Stateless Rules

When you create a security rule, you choose whether it is stateful or stateless. The default is stateful. Stateless rules are recommended if you have a high-volume internet-facing website, for the HTTP/HTTPS traffic.

Marking a security rule as **stateful** indicates that you want to use connection tracking for any traffic that matches that rule. This means that when an instance receives traffic matching the stateful ingress rule, the response is tracked and automatically allowed back to the originating host, regardless of any egress rules applicable to the instance. And when an instance sends traffic that matches a stateful egress rule, the incoming response is automatically allowed, regardless of any ingress rules.

Marking a security rule as **stateless** indicates that you do NOT want to use connection tracking for any traffic that matches that rule. This means that response traffic is not automatically allowed. To allow the response traffic for a stateless ingress rule, you must create a corresponding stateless egress rule.

If you use both stateful and stateless rules, and there is traffic that matches both a stateful and stateless rule in a particular direction, the stateless rule takes precedence and the connection is not tracked. You would need a corresponding rule in the other direction, either stateless or stateful, for the response traffic to be allowed.

If you decide to use stateless security rules to allow traffic to/from endpoints outside the VCN, it is important to add a security rule that allows ingress ICMP traffic type 3 code 4 from source 0.0.0.0/0 and any source port. This rule enables your instances to receive Path MTU Discovery fragmentation messages. This rule is critical for establishing a connection to your instances. Without it, you can experience connectivity issues.

Best Practices for Security Rules

- **Use network security groups**

  Oracle recommends using NSGs for components that all have the same security posture. For example, in a multitier architecture, you would have a separate NSG for each tier. A tier's VNICs would all belong to that tier's NSG.

  Within a tier, you might have a particular subset of the tier's VNICs that have additional, special security requirements. Therefore you would create another NSG for those additional rules, and place that subset of VNICs into both the tier's NSG and the NSG with additional rules.
- **Understand default security list rules**

  Each VCN automatically comes with a default security list that contains several default security rules to help you get started using the Networking service. Those rules exist because they enable basic connectivity.

  Even if you choose not to use security lists or the default security list, get familiar with the rules so you better understand the types of traffic that your networked cloud resources require. You might want to use those rules in your NSGs or any custom security lists that you set up.

  There is no default rule to allow ping requests. If you want to ping an instance, add a stateful ingress rule to specifically allow ICMP traffic type 8 from the source network you plan to ping from. To allow ping access from the internet, use 0.0.0.0/0 for the source. Note that this rule for pinging is separate from the default ICMP-related rules in the default security list. Don't remove those rules.
- **Do not delete default security rules indiscriminately**

  Your VCN might have subnets that use the default security list by default. Don't delete any of the list's default security rules unless you have first confirmed that resources in your VCN don't require them. Otherwise, you might disrupt your VCN connectivity.
- **If necessary, add rules to allow ping requests**

  There is no default rule to allow ping requests. If you want to ping an instance, add a stateful ingress rule to specifically allow ICMP traffic type 8 from the source network you plan to ping from. To allow ping access from the internet, use 0.0.0.0/0 for the source. Note that this rule for pinging is separate from the default ICMP-related rules in the default security list. Don't remove those rules.
- **If necessary, add rules to handle fragmented UDP packets**

  Instances can send or receive UDP traffic. If a UDP packet is too large for the connection, it's fragmented. However, only the first fragment from the packet contains the protocol and port information. If the security rules that allow this ingress or egress traffic specify a particular (source or destination) port number, the fragments after the first one are dropped. If you expect your instances to send or receive large UDP packets, set both the source and destination ports for the applicable security rules to ALL instead of a particular port number.
- **Align OS firewall rules with security rules**

  Your instances running images provided with Compute Cloud@Customer also have OS firewall rules that control access to the instance. When troubleshooting access to an instance, ensure that all the following items are set correctly:

  - The rules in the network security groups that the instance is in
  - The rules in the security lists associated with the instance's subnet
  - The instance's OS firewall rules

Security Lists

On Compute Cloud@Customer, a security list acts as a virtual firewall for an instance, with ingress and egress rules that specify the types of traffic allowed in and out.

Each security list is enforced at the VNIC level. However, you configure your security lists at the subnet level, which means that all VNICs in a particular subnet are subject to the same set of security lists.

The security lists apply to a particular VNIC whether it's communicating with another instance in the VCN or a host outside the VCN. Each subnet can have multiple security lists associated with it, and each list can have multiple rules.

Each VCN comes with a default security list. If you don't specify a custom security list for a subnet, the default security list is automatically used with that subnet. You can add and remove rules in the default security list. It has an initial set of stateful rules, which should be changed to only allow inbound traffic from authorized subnets. The default rules are:

- **Stateful ingress:** Allow TCP traffic on destination port 22 (SSH) from authorized source IP addresses and any source port.

  This rule enables you to create a new cloud network and public subnet, create a Linux instance, and then immediately use SSH to connect to that instance without needing to write any security list rules yourself.

  The default security list doesn't include a rule to allow Remote Desktop Protocol (RDP) access. If you're using Compute Cloud@Customer images, add a stateful ingress rule for TCP traffic on destination port 3389 from authorized source IP addresses and any source port.
- **Stateful ingress:** Allow ICMP traffic type 3 code 4 from authorized source IP addresses.

  This rule enables instances to receive Path MTU Discovery fragmentation messages.
- **Stateful ingress:** Allow ICMP traffic type 3 (all codes) from your VCN CIDR block.

  This rule enables instances to receive connectivity error messages from other instances within the VCN.
- **Stateful egress:** Allow all traffic.

  This allows instances to initiate traffic of any kind to any destination. This implies that instances with public IP addresses can talk to any internet IP address if the VCN has a configured internet gateway. And because stateful security rules use connection tracking, the response traffic is automatically allowed regardless of any ingress rules.

The general process for working with security lists is as follows:

1. Create a security list.
2. Add security rules to the security list.
3. Associate the security list with one or more subnets.
4. Create resources, such as compute instances, in the subnet.

   The security rules apply to all the VNICs in that subnet.

When you create a subnet, you must associate at least one security list with it. It can be either the VCN default security list or one or more other security lists that you already created. You can change which security lists the subnet uses at any time. You can add and remove rules in the security list. It's possible for a security list to contain no rules.

For more information, see Controlling Traffic with Security Lists on page 158.

Network Security Groups

On Compute Cloud@Customer, a network security group (NSG) provides a virtual firewall for a set of cloud resources, within a single VCN, that all have the same security posture. For example: a group of compute instances that all perform the same tasks and thus all need to use the same set of ports.

Rules in an NSG are enforced on VNICs, but their NSG membership is determined through their parent resources. Not all cloud services support NSGs. Currently, the following types of parent resources support the use of NSGs:

- **Compute instances:** When you create an instance, you can specify one or more NSGs for the instance's primary VNIC. If you add a secondary VNIC to an instance, you can specify one or more NSGs for that VNIC. You can also change the NSG membership of existing VNICs.
- **Load balancers:** When you create a load balancer, you can specify one or more NSGs for the load balancer (not the backend set). You can also update an existing load balancer to use one or more NSGs.
- **Mount targets:** When you create a mount target for a file system, you can specify one or more NSGs. You can also update an existing mount target to use one or more NSGs.

For resource types that don't yet support NSGs, continue to use security lists to control traffic to and from those parent resources.

> **Note:**
>
> You can't associate an Internet Gateway with an NSG.

An NSG contains two types of elements:

- **VNICs:** One or more VNICs; for example, the VNICs attached to the set of compute instances that all have the same security posture. All the VNICs must be in the VCN that the NSG belongs to. A VNIC can be in a maximum of five NSGs.
- **Security rules:** Rules that define the types of traffic allowed into and out of the VNICs in the group. For example: ingress TCP port 22 SSH traffic from a particular source.

The general process for working with NSGs is as follows:

1. Create an NSG.

   When you create an NSG, it is initially empty, without any security rules or VNICs. After the NSG is created, you can add or remove security rules to allow the types of ingress and egress traffic that the VNICs in the group require.
2. Add security rules to the NSG.
3. Add parent resources, or more specifically VNICs, to the NSG.

   When you manage an NSG VNIC membership, you do it as part of working with the parent resource, not the NSG itself. You can do this when you create the parent resource, or you can update the parent resource and add it to one or more NSGs.

   When you create a compute instance and add it to an NSG, the instance's primary VNIC is added to the NSG. You can create secondary VNICs separately, and optionally add them to NSGs.

There are some differences in the REST API model for NSGs compared to security lists:

- With security lists, there is an `IngressSecurityRule` object and a separate `EgressSecurityRule` object. With network security groups, there is only a `SecurityRule` object, and the object's `direction` attribute determines whether the rule is for ingress or egress traffic.
- With security lists, the rules are part of the `SecurityList` object, and you work with the rules by calling the security list operations; for example: `UpdateSecurityList`. With NSGs, the rules aren't part of the `NetworkSecurityGroup` object. Instead you use separate operations to work with the rules for a particular NSG; for example: `UpdateNetworkSecurityGroupSecurityRules`.
- The model for updating existing security rules is different between security lists and NSGs. With NSGs, each rule in a particular group has a unique identifier. When you call `UpdateNetworkSecurityGroupSecurityRules`, you provide the IDs of the specific rules that you want to update. With security lists, the rules have no unique identifier. When you call `UpdateSecurityList`, you must pass in the entire list of rules, including rules that aren't being updated in the call.
- There is a limit of 25 rules when calling the operations to add, remove, or update security rules.

For more information, see Controlling Traffic with Network Security Groups on page 163.

Working with DHCP Options

On Compute Cloud@Customer, 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 don't 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 the instances in that subnet.

Each subnet in a VCN can have a single set of DHCP options associated with it. That set of options applies to all instances in the subnet. Each VCN comes with a default set of DHCP options with initial values that you can change. Unless you create and assign a different set of DHCP options, every subnet uses the VCN default set.

These are the available DHCP options you can configure:

- **Domain Name Server**

The default setting is DNS Type = Internet and VCN Resolver.

If instead you set DNS Type = Custom Resolver, you can specify up to three DNS servers of your choice.

- **Search Domain**

  If you set up your VCN with a DNS label, the default value for the Search Domain option is the VCN domain name: ***<VCN_DNS_label>***`.oraclevcn.com`. Otherwise, the Search Domain option isn't present in the default set of DHCP options.

  In general, when any set of DHCP options is initially created – the default set or a custom set –, the Networking service automatically adds the Search Domain option and sets it to the VCN domain name (***<VCN_DNS_label>***`.oraclevcn.com`) if all of these are true:

  - The VCN has a DNS label.
  - DNS Type = Internet and VCN Resolver.
  - You did NOT specify a search domain of your choice during creation of the set of DHCP options.

  After the set of DHCP options is created, you can always remove the Search Domain option or set it to a different value. You can specify only a single search domain in a set of DHCP options.

Important notes about your instances and DHCP options:

- Whenever you change the value of one of the DHCP options, you must either restart the DHCP client on the instance, or reboot the instance for the changes to take effect. This applies to all existing instances in the subnets associated with that set of DHCP options.
- Keep the DHCP client running so you can always access the instance. If you stop the DHCP client manually or disable NetworkManager, the instance can't renew its DHCP lease and will become inaccessible when the lease expires; typically within 24 hours. Do not disable NetworkManager unless you use another method to ensure renewal of the lease.
- Stopping the DHCP client might remove the host route table when the lease expires. Also, loss of network connectivity to your iSCSI connections might result in loss of the boot drive.
- Any changes you make to the `/etc/resolv.conf` file are overwritten whenever the DHCP lease is renewed or the instance is rebooted.
- Changes you make to the `/etc/hosts` file are overwritten whenever the DHCP lease is renewed or the instance is rebooted. To persist your changes to the `/etc/hosts` file in Oracle Linux or CentOS instances, add the following line to `/etc/oci-hostname.conf`: `PRESERVE_HOSTINFO=2`. If the `/etc/oci-hostname.conf` file doesn't exist, create it.

Creating a Set of DHCP Options

On Compute Cloud@Customer, you can create a set of DHCP options.

For conceptual information, see:

- Name Resolution on page 128
- Working with DHCP Options on page 151

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN.
3. Select the name of the VCN for which you want to create a set of DHCP options.

   The VCN details page is displayed.
4. Under **Resources**, select **DHCP Options**.
5. Select **Create DHCP Options**.
6. 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.
- **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.
- **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.

7. Select **Create DHCP Options**.

   You can specify this set of options when creating or updating a subnet.

Using the CLI

Use the oci network dhcp-options create command and required parameters to create a new set of DHCP options for the specified VCN.

```
oci network dhcp-options create --compartment-id <compartment_OCID>  --vcn-
id <vcn_OCID> --options <JSON_formatted_values> [OPTIONS]
```

--options <*JSON_formatted_values*> is a set of DHCP options. 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 using the `file://` syntax.

Example:

```
'[{"type": "DomainNameServer", "customDnsServers": ["202.44.61.9"],
 "serverType": "CustomDnsServer"}]'
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateDhcpOptions operation to create a new set of DHCP options for the specified VCN.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing VCN DHCP Option Sets

On Compute Cloud@Customer, every VCN has a default set of DHCP options. If you create more sets, then you can choose which set to assign to a subnet.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN.
3. Select the name of the VCN for which you want to list DHCP Options sets.
4. Under **Resources**, select **DHCP Options**.

   The list of DHCP options sets is displayed.
5. Select the Actions menu (⋮) for the a particular DHCP set, and select **Edit**.

The options are displayed.

Using the CLI

Use the oci network dhcp-options list command and required parameters to list DHCP option sets.

```
oci network dhcp-options list --compartment-id ocid1.compartment.unique_ID
 --vcn-id ocid1.vcn.unique_ID [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListDhcpOptions operation to list DHCP option sets.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Updating a Set of DHCP Options

On Compute Cloud@Customer, you can change DHCP options.

To update the DHCP options for the instances in a subnet, do one of the following:

- Update the DHCP Options object that's 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 on page 143.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN.
3. Select the name of the VCN for the DHCP options that you want to edit.
4. Under **Resources**, select **DHCP Options**.

   The list of DHCP options sets is displayed.
5. Select the Actions menu (⋮) for the set of options you want to change, then select **Edit**.
6. Change options.
7. Select **Save Changes**.

Using the CLI

Use the oci network dhcp-options update command and required parameters to update the specified set of DHCP options. You can update the display name or the options themselves.

> **Important:**
>
> The options object you provide replaces the entire existing set of options.

```
oci network dhcp-options update --dhcp-id <dhcp_OCID> <options_to_update>
 [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateDhcpOptoins operation to update the specified set of DHCP options. You can update the display name or the options themselves. Avoid entering confidential information.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Set of DHCP Options

On Compute Cloud@Customer, you can delete a set of DHCP options as long as the DHCP options are not 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 on page 143. You can't delete a VCN default set of DHCP options.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN.
3. Select the name of the VCN for which you want to delete a DHCP Options set.
4. Under **Resources**, select **DHCP Options**.
5. Select the Actions menu (⋮) for the set you want to delete, then select **Delete**.
6. Confirm the deletion.

Using the CLI

Use the oci network dhcp-options delete command and required parameters to delete the specified set of DHCP options, but only if it's not associated with a subnet.

```
oci network dhcp-options delete --dhcp-id ocid1.dhcpoptions.unique_ID
 [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteDhcpOptions operation to delete the specified set of DHCP options, but only if it's not associated with a subnet.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Working with Route Tables

On Compute Cloud@Customer, each VCN automatically comes with an empty default route table. Subnets use the default route table of the parent VCN, unless you explicitly assign them a different route table.

When you add route rules to a VCN, you can add them to the default table if that suits your needs. However, if you need both a public subnet and a private subnet, you instead create a separate custom route table for each subnet.

The primary routing scenario is for sending subnet traffic to destinations outside the VCN. A subnet has a single route table of your choice associated with it at the time of creation. All VNICs in that subnet are subject to the rules in the route table. Traffic within the VCN is automatically handled by the VCN internal routing. No route rules are required to enable that traffic. You can change which route table the subnet uses at any time. You can also edit a route table's rules, or remove all the rules from the table.

A route rule specifies a destination CIDR block and the target, or the next hop, for any traffic that matches that CIDR. When selecting the target you also specify its compartment. Supported target types for a route rule are the different VCN gateways.

Route rules must be configured carefully to ensure that the network traffic reaches the intended destination and isn't dropped. Moving route rules between compartments is not supported.

Creating a Route Table

On Compute Cloud@Customer, 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.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN in which you want to create a subnet.
3. Select the name of the VCN for which you want to create a route table.

   The VCN details page is displayed.
4. Under **Resources**, select **Route Tables**.
5. Select **Create Route Table**.
6. Enter the required information:
   - **Name:** Specify a user-friendly name for the route table. The name doesn't have to be unique, and you can change it later. Avoid entering confidential information
   - **Create in Compartment:** Select the compartment where you want to create the route table. You're not required to create the route table in the same compartment as the VCN.
7. To add a rule, select **+New Rule**, and enter the following information:
   - **Target Type:** Select from the list.
   - **CIDR Block**: The destination CIDR block for the traffic. A value of 0.0.0.0/0 means that all non-intra-VCN traffic that is not already covered by other rules in the route table goes to the target specified in this rule.
   - **Target:** The label for this value is the value that you selected for Target Type. Select the arrow and select the target. You might need to change the compartment just above the arrow.
   - **Description:** An optional description of the rule.
8. **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
9. Select **Create Route Table**.

   The details page of the new route table is displayed. You can specify this route table when creating or updating a subnet.

Using the CLI

Use the oci network route-table create command and required parameters to create a new route table for the specified VCN. In the request you must also include at least one route rule for the new route table.

```
oci network route-table create --compartment-id compartment_OCID --vcn-
id vcn_OCID --route-rules route_rules_json [OPTIONS]
```

The --route-rules option includes the collection of rules used for routing destination IPs to network devices. 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 through the `file://` syntax.

Example:

```
'[{"cidrBlock":"0.0.0.0/0","networkEntityId":"ocid1.internetgateway.oc1.phx.UniqueID"}]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateRouteTable operation to create a new route table for the specified VCN.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing a Route Table

On Compute Cloud@Customer, you can list the route tables in the specified VCN.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN in which you want to create a subnet.
3. Select the name of the VCN for which you want to view route tables.

   The VCN details page is displayed.
4. Under **Resources**, select **Route Tables**.

   The list of route tables is displayed.
5. Select the name of the route table to view its route rules.

Using the CLI

Use the oci network route-table list command and required parameters to list the route tables in the specified VCN and specified compartment. If the VCN ID is not provided, then the list includes the route tables from all VCNs in the specified compartment. The response includes the default route table that automatically comes with each VCN in the specified compartment, plus any route tables you've created.

```
oci network route-table list --compartment-id ocid1.compartment.unique_ID --
vcn-id ocid1.vcn.unique_ID [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListRoutTables operation to list the route tables in the specified VCN.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Updating Route Table Rules

On Compute Cloud@Customer, you can change the name of a route table and add, edit, or delete rules in a route table.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN in which you want to create a subnet.
3. Select the name of the VCN for which you want to update the route table.

   The VCN details page is displayed.
4. Under **Resources**, select **Route Tables**.
5. Select the name of the route table that you want to update.
6. Edit the route rules:
   - To change the name of the route table, select **Edit**, change the name in the dialog box, and select **Save Changes**.
   - To create a route rule, select **Add Route Rules** and enter the information.
   - To edit an existing rule, select the Actions menu (⋮) for that rule, and then select **Edit**.
   - To delete a rule, select the Actions menu (⋮) for that rule, and then select **Delete**.

Using the CLI

Use the oci network route-table update command and required parameters to update the specified route table's display name or route rules. Avoid entering confidential information.

```
oci network route-table update --rt-id <route_table_OCID> --route-
rules <options_to_change> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateRouteTable operation to update the specified route table's display name or route rules. Avoid entering confidential information.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Route Table

On Compute Cloud@Customer, you can delete a route table as long as it is not associated with a subnet. You can't delete a VCN default route table.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN in which you want to create a subnet.
3. Select the name of the VCN for which you want to delete a route table.

   The VCN details page is displayed.
4. Under **Resources**, select **Route Tables**.
5. Select the Actions menu (⋮) for the route table that you want to delete, then select **Delete**.
6. Confirm the deletion.

Using the CLI

Use the oci network route-table delete command and required parameters to delete the specified route table.

```
oci network route-table delete --rt-id ocid1.routetable.unique_ID [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteRouteTable operation to delete the specified route table.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Controlling Traffic with Security Lists

On Compute Cloud@Customer, 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 decide 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 on page 163.

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 don't 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's associated with the VNIC subnet
- Any rule in any NSG that the VNIC is in

For more information about security lists, see Security Lists on page 149.

Creating a Security List

On Compute Cloud@Customer, you can create a security list for a VCN.

Before you create a security list, view the security rules that are already defined in the default security list and any other security list for this VCN. See Viewing Security Lists on page 161.

A security list must have at least one rule. A security list isn't required to have both ingress and egress rules.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN in which you want to create a subnet.
3. Select the name of the VCN for which you want to create a security list.

   The VCN details page is displayed.
4. Under **Resources**, select **Security Lists**.
5. Select **Create Security List**.
6. In the Create Security List dialog box, enter the following information:
   - **Name:** Provide a descriptive name for the security list. The name doesn't have to be unique. Avoid entering confidential information. (The name can't be changed later in the Console but it can be changed with the CLI).
   - **Create in Compartment:** Select the compartment where you want to create the security list.
7. Add at least one rule.

   To add one or more ingress rules, select **+New Rule** in the **Allow Rules for Ingress** 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.
   - **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.
8. **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
9. Select **Create Security List**.

   The details page of the new security list is displayed. You can specify this security list when creating or updating a subnet.

Using the CLI

Use the oci network security-list create command and required parameters to create a new security list for the specified VCN.

```
oci network security-list create --compartment-id <compartment_OCID> --vcn-
id <vcn_OCID> --ingress-security-rules <ingress_rules> --egress-security-
rules <egress_rules> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Procedure

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.

   Or, 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:

   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": "unique_ID",
    "vcn-id": "ocid1.vcn.unique_ID"
  },
  "etag": "unique_ID"
}
```

Using the API

Use the CreateSecurityList operation to create a new security list for the specified VCN.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing Security Lists

On Compute Cloud@Customer, you can view security lists that are associated with a VCN.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN in which you want to create a subnet.
3. Select the name of the VCN for which you want to view security lists.

   The VCN details page is displayed.
4. Under **Resources**, select **Security Lists**.

   The list of security lists is displayed.
5. Select the name of the security list to view its ingress and egress rules.

Using the CLI

Use the oci network security-list list command and required parameters to list security lists.

If you specify both the VCN OCID and the compartment OCID arguments, all security lists that belong to the specified VCN and are in the specified compartment are listed.

```
oci network security-list list --compartment-id ocid1.compartment.unique_ID
 --vcn-id ocid1.vcn.unique_ID [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListSecurityLists operation to list security lists.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Updating a Security List

On Compute Cloud@Customer, 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.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN in which you want to create a subnet.
3. Select the name of the VCN for which you want to update a security list.

   The VCN details page is displayed.
4. Under **Resources**, select **Security Lists**.
5. For the security list that you want to update, do one of the following:

   - Select the Actions menu (⋮), then select **Edit** to open the **Edit Security List** dialog box. Update rules in the **Allow Rules for Ingress** and **Allow Rules for Egress** sections.

     To delete a rule, select the trash can icon. To add a rule, select **+New Rule**. You can also update the security list name and tags. Select **Save**.
   - Select the Actions menu (⋮), then select **View Details** to open the security list details page.
     - Select **Edit** to open the Edit Security List dialog box.
     - To edit only the rules, scroll to the **Resources** section and select either **Ingress Rules** or **Egress Rules**.

       To create a new rule, select **Create Security Rule**. To update a rule, select the Actions menu (⋮) for that rule and then select **Edit**.

       To delete a rule, select the Actions menu and then select **Delete**.

Using the CLI

Use the oci network security-list update command and required parameters to update the specified security list's display name or rules. Avoid entering confidential information.

```
oci network security-list update --security-list-id <security_list_OCID>
 [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateSecurityList operation to update the specified security list's display name or rules.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Security List

On Compute Cloud@Customer, you can delete a security list as long as it isn't associated with a subnet. You can't delete a VCN default security list.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN in which you want to create a subnet.
3. Select the name of the VCN for which you want to delete a security list.

   The VCN details page is displayed.
4. Under **Resources**, select **Security Lists**.
5. Select the Actions menu (⋮) for the security list that you want to delete, then **Delete**.
6. Confirm the deletion.

Using the CLI

Use the oci network security-list delete command and required parameters to delete the specified security list, but only if it's not associated with a subnet.

```
oci network security-list delete [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteSecurityList operation to delete the specified security list, but only if it's not associated with a subnet.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Controlling Traffic with Network Security Groups

On Compute Cloud@Customer, 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 decide 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 on page 158.

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 particular VNIC is allowed if any rule in any applicable security list or NSG allows the traffic:

- Any rule in any security list that's associated with the VNIC subnet
- Any rule in any NSG that the VNIC is in

For more information, see Network Security Groups on page 150.

Creating a Network Security Group

On Compute Cloud@Customer, you can create Network Security Groups (NSGs).

General Process for Working with NSGs

1. Create an NSG as described in this section.

2. Add security rules to the NSG. See Managing Network Security Group Rules on page 165.
3. Add parent resources (or more specifically, VNICs) to the NSG.

   You can do this when you create the parent resource, or you can update the parent resource and add it to one or more NSGs.

   When you create a Compute instance and add it to an NSG, the instance's primary VNIC is added to the NSG. Separately, you can create secondary VNICs and optionally add them to NSGs.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN in which you want to create a subnet.
3. Select the name of the VCN for which you want to create an NSG.

   The VCN details page is displayed.
4. Under **Resources**, select **Network Security Groups**.
5. Select **Create Network Security Group**.
6. In the **Create Network Security Group** dialog box, enter the following information:
   - **Name:** Enter a descriptive name for the NSG. The name doesn't have to be unique, and it can be changed later. Avoid entering confidential information.
   - **Create in Compartment:** Select the compartment where you want to create the NSG.
7. **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
8. Select **Create Network Security Group**.

   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 Configuring VCN Rules and Options on page 145 and Configuring VNICs on page 181.

Using the CLI

Use the oci network nsg create command and required parameters to create a new network security group for the specified VCN.

```
oci network nsg create --compartment-id <compartment_OCID> --vcn-
id <vcn_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateNetworkSecurityGroup operation to create a new network security group for the specified VCN.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing Network Security Groups

On Compute Cloud@Customer, NSGs are available for you to create and use.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN in which you want to create a subnet.
3. Select the name of the VCN for which you want to view Network Security Groups.

   The VCN details page is displayed.
4. Under **Resources**, select **Network Security Groups**.

The list of NSGs is displayed.

5. Select the name of the NSG to view its details, including security rules and attached VNICs.

Using the CLI

Use the oci network nsg list command and required parameters to list either the network security groups in the specified compartment, or those associated with the specified VLAN.

> **Note:**
>
> You must specify either a `vlanId` or a `compartmentId`, but not both. If you specify a `vlanId`, all other parameters are ignored.

```
oci network nsg list --compartment-id <compartment_OCID> | --VLAN-
id <vlan_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListNetworkSecurityGroups operation to list either the network security groups in the specified compartment, or those associated with the specified VLAN.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Managing Network Security Group Rules

On Compute Cloud@Customer, you can add, update, and remove NSG rules.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN in which you want to create a subnet.
3. Select the name of the VCN for which you want to manage rules in an NSG.

   The VCN details page is displayed.
4. Under **Resources**, select **Network Security Groups**.
5. In the list of NSGs, select the name of the NSG for which you want to manage rules. The NSG details page is displayed.
6. Under **Resources**, select **Security Rules**.
7. You can add, edit, and delete rules.

   - To add a rule, select **Create Security Rules**. To add one or more ingress rules, select +**New Rule** in the **Allow Rules for Ingress** box. To add one or more egress rules, select **+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.
     - **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. Avoid entering confidential information.
   - To edit a rule, select the Actions menu (⋮) for the Egress or Ingress rule, select **Edit**, make the necessary changes, and then select **Update**.

- To delete a rule, select the Actions menu (⋮) for the Egress or Ingress rule, select **Remove**, and then select **Confirm**. While you're editing a rule, select the trash can icon to delete the rule.

Using the CLI

Use these CLI commands to manage NSG rules:

- Use the oci network nsg rules add command and required parameters to add one or more security rules to the specified network security group.

```
oci network nsg rules add [OPTIONS]
```

- Use the oci network nsg rules list command and required parameters to list the security rules in the specified network security group.

```
oci network nsg rules list [OPTIONS]
```

- Use the oci network nsg rules update command and required parameters to update one or more security rules in the specified network security group.

```
oci network nsg rules update [OPTIONS]
```

- Use the oci network nsg rules remove command and required parameters to remove one or more security rules from the specified network security group.

```
oci network nsg rules remove [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the AddNetworkSecurityGroupSecurityRules operation to add one or more security rules to the specified network security group.

Use the ListNetworkSecurityGroupSecurityRules operation to list the security rules in the specified network security group.

Use the UpdateNetworkSecurityGroupSecurityRules operation to update one or more security rules in the specified network security group.

Use the RemoveNetworkSecurityGroupSecurityRules operation to remove one or more security rules from the specified network security group.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Attaching VNICs to a Network Security Group

On Compute Cloud@Customer, 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 on page 316
- Creating and Attaching a Secondary VNIC on page 183
- Updating a VNIC on page 185

Using the Console

You can view NSGs and VNICs in different ways:

- To view the list of NSGs that a VNIC is attached to, perform the following steps.
  1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
  2. At the top of the page, select the compartment that contains the instance that has the VNIC.

3. Select the name of the instance.

   The instance details page is displayed.
4. Under **Resources**, and select **Attached VNICs**.
5. Select the name of the VNIC.
6. Under **Resources**, select **Network Security Groups**.

- To view the list of VNICs that are attached to an NSG, perform the following steps.

  1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
  2. At the top of the page, select the compartment that contains the VNC.
  3. Select the name of the VCN.

     The VCN details page is displayed.
  4. Under **Resources**, select **Network Security Groups**.
  5. In the list, select the name of the NSG.
  6. Under **Resources**, select **VNICs**.
- To change the list of NSGs that a VNIC is attached to, update the VNIC. See Updating a VNIC on page 185.

Using the CLI

Use the oci network vnic get command and required parameters to view the list of NSGs that a VNIC is attached to.

```
oci network vnic get [OPTIONS]
```

Use the oci network nsg list command and required parameters to view the list of VNICs that are attached to an NSG.

```
oci network nsg vnics list [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

To change the list of NSGs that a VNIC is attached to, update the VNIC.

Using the API

Use the GetVnic operation to view the list of NSGs that a VNIC is attached to.

Use the ListNetworkSecurityGroupVnics operation to view the list of VNICs that are attached to an NSG.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Network Security Group

On Compute Cloud@Customer, you can delete Network Security Groups (NSGs) as long as the group doesn't contain any VNICs.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN in which you want to create a subnet.
3. Select the name of the VCN for which you want to delete an NSG.

   The VCN details page is displayed.
4. Under **Resources**, select **Network Security Groups**.
5. Select the Actions menu (⋮) for the NSG that you want to delete, then select **Delete**.
6. Confirm the deletion.

Using the CLI

Use the oci network nsg delete command and required parameters to delete the specified network security group.

```
oci network nsg delete [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteNetworkSecurityGroup operation to delete the specified network security group.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Configuring VCN Gateways

Learn about the different types of gateways you can configure on Compute Cloud@Customer.

This section describes the role and usage of each gateway type.

Network Address Translation (NAT) Gateways

A NAT gateway gives cloud resources without a public IP access to the on-premises network, which is an external public network from the point of view of a VCN, without exposing those resources. You create a NAT gateway in the context of a specific VCN, so the gateway is automatically attached to that VCN upon creation.

See Enabling Public Connections through a NAT Gateway on page 168.

Internet Gateways

An internet gateway connects the edge of the VCN with the on-premises network. The ultimate target of the traffic routed through an internet gateway can be the internet.

See Providing Public Access through an Internet Gateway on page 170.

Local Peering Gateways

VCN peering is the process of connecting multiple virtual cloud networks (VCNs) so that resources can communicate using private IP addresses. You can use VCN peering to divide your network into multiple VCNs, for example, based on departments or lines of business, with each VCN having direct private access to the others. You can also place shared resources into a single VCN that all the other VCNs can access privately.

See Connecting VCNs through a Local Peering Gateway (LPG) on page 172.

Dynamic Routing Gateways (DRGs)

a dynamic routing gateway, or DRG, provides a path for private network traffic between the VCN and an on-premises network. This traffic is routed to the data center network and on to its destination.

See Creating a Dynamic Routing Gateway on page 174.

Service Gateways

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.

See Accessing Oracle Services through a Service Gateway on page 175.

Enabling Public Connections through a NAT Gateway

On Compute Cloud@Customer, a NAT gateway gives cloud resources without a public IP access to the on-premises network, which is an external public network from the point of view of a VCN, without exposing those resources.

You create a NAT gateway in the context of a specific VCN, so the gateway is automatically attached to that VCN upon creation.

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 Compute Cloud@Customer traffic to communicate with on-premises data center networks.

The gateway allows hosts to initiate connections to the on-premises network and receive responses, but prevents them from receiving inbound connections initiated from the on-premises network. NAT gateways are highly available and support TCP, UDP, and ICMP ping traffic. The Networking service automatically assigns a public IP address to the NAT gateway. You can't choose its public IP address.

When a host in the private network initiates a connection to the on-premises network, the NAT device's public IP address becomes the source IP address for the outbound traffic. The response traffic from the on-premises network therefore uses that public IP address as the destination IP address. The NAT device then routes the response back to the private network, to the host that initiated the connection.

VCN routing is controlled at the subnet level, so you can specify which subnets use a NAT gateway. You can only configure one NAT gateway per VCN.

For the purpose of access control, when creating a NAT gateway, you must specify the compartment where you want the gateway to reside. If you're not sure which compartment to use, put the gateway in the same compartment as the VCN.

By default, a NAT gateway allows traffic at the time of creation. However, you can block or allow traffic through the gateway at any time. Blocking a NAT gateway prevents all traffic, regardless of any existing route rules or security rules in the VCN.

Configuring a NAT Gateway

On Compute Cloud@Customer, you can configure NAT gateways for VCNs.

After created, the NAT Gateway is ready for the addition of route rules or security settings. Note that the name of the gateway (`natgatewayUniqueID`) is assigned automatically and not by a parameter, and that the IP address of the device is also assigned automatically.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN where the NAT gateway will be created.
3. Select the name of the VCN.

   The VCN details page is displayed.
4. Under **Resources**, select NAT Gateways.
5. Select **Create NAT Gateway**.
6. Enter the following information:

   - **Name:** Provide a descriptive name for the NAT gateway. Avoid entering 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's not configured.
     - **(No: Traffic Blocked)**: You can set the NAT gateway not see traffic until it's explicitly enabled to do so.
   - **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
7. Select **Create NAT Gateway**.

   The NAT Gateway is now ready for the addition of route rules or security settings. See Configuring VCN Rules and Options on page 145.

Using the CLI

Use the oci network nat-gateway create command and required parameters to create a new NAT gateway for the specified VCN. You must also set up a route rule with the NAT gateway as the rule's target.

```
oci network nat-gateway create [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateNatGateway operation to create a new NAT gateway for the specified VCN.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Providing Public Access through an Internet Gateway

On Compute Cloud@Customer, an internet gateway connects the edge of the VCN with the on-premises network. The ultimate target of the traffic routed through an internet gateway can be the internet.

However, on Compute Cloud@Customer, the internet gateway routes traffic to the on-premises network. Traffic to and from the internet is then managed by the routing configuration in the on-premises network.

You create an internet gateway in the context of a specific VCN, so the gateway is automatically attached to that VCN upon creation. To use the gateway, the hosts on both ends of the connection must have public IP addresses for routing. Connections that originate in the VCN and are destined for a public IP address, either inside or outside the VCN, go through the internet gateway. Connections that originate outside the VCN and are destined for a public IP address inside the VCN also go through the internet gateway.

A VCN can have only one internet gateway. You control which public subnets in the VCN can use the gateway by configuring the subnet associated route table. The public subnet security list rules determine the specific types of traffic that are allowed to and from the resources in the subnet. An internet gateway can be disabled, meaning no traffic flows to or from the internet, regardless of any existing route rules that enable such traffic.

For the purpose of access control, when creating an internet gateway, you must specify the compartment where you want the gateway to reside. If you're not sure which compartment to use, put the gateway in the same compartment as the VCN.

To configure an Internet gateway, see Configuring an Internet Gateway on page 170.

Configuring an Internet Gateway

On Compute Cloud@Customer, you can configure an Internet Gateway (IGW) which provides the VCN with outside access through the on-premises data center network.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN where the internet gateway will be created.
3. Select the name of the VCN.

   The VCN details page is displayed.
4. Under **Resources**, select **Internet Gateways**.
5. Select **Create Internet Gateway**.
6. Enter the following information:

   - **Name:** Provide a descriptive name for the internet gateway. Avoid entering confidential information
   - **Create in Compartment:** Select the compartment in which to create the Internet Gateway.
   - **Enabled:** Use the toggle to decide 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's explicitly enabled to do so.

- **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.

7. Select **Create Internet Gateway**.

   The Internet Gateway is now ready for the addition of route rules or security settings. See Configuring VCN Rules and Options on page 145.

Using the CLI

Use the oci network internet-gateway create command and required parameters to create a new internet gateway for the specified VCN.

```
oci network internet-gateway create [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateInternetGateway operation to create a new internet gateway for the specified VCN.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Enable or Disable an Internet Gateway

On Compute Cloud@Customer, you can enable or disable an Internet gateway.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN that has the internet gateway.
3. Select the name of the VCN.

   The VCN details page is displayed.
4. Under **Resources**, select **Internet Gateways**.
5. Select the Actions menu (⋮), then select **Edit**.
6. Set the **Enabled** toggle:
   - Slide the toggle right to enable the internet gateway.
   - Slide the toggle left to disable the internet gateway.
7. Select **Update**.

Using the CLI

Use the oci network internet-gateway update command and required parameters to update the specified internet gateway. You can disable/enable it, or change its display name or tags.

```
oci network internet-gateway update [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateInternetGateway command and required parameters to the specified internet gateway. You can disable/enable it, or change its display name or tags.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting an Internet Gateway

On Compute Cloud@Customer, If you have previously configured an IGW, you can delete it.

The internet gateway doesn't have to be disabled, but there must not be a route table that lists it as a target.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN that has the internet gateway.
3. Select the name of the VCN.

   The VCN details page is displayed.
4. Under **Resources**, select **Internet Gateways**.
5. Select the Actions menu (⋮), then select **Delete**.
6. Confirm the deletion.

Using the CLI

Use the oci network internet-gateway delete command and required parameters to delete the specified internet gateway.

```
oci network internet-gateway delete [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteInternetGateway operation to delete the specified internet gateway.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Connecting VCNs through a Local Peering Gateway (LPG)

On Compute Cloud@Customer, You can configure local peering gateways. VCN peering is the process of connecting multiple virtual cloud networks (VCNs) so that resources can communicate using private IP addresses.

You can use VCN peering to divide your network into multiple VCNs, for example, based on departments or lines of business, with each VCN having direct private access to the others. You can also place shared resources into a single VCN that all the other VCNs can access privately. Two peered VCNs can be in the same tenancy or different ones.

Policies

Peering between two VCNs requires explicit agreement from both parties in the form of IAM policies that each party implements for their own VCN compartment or tenancy. If the VCNs are in different tenancies, each administrator must provide their tenancy OCID and put in place special coordinated policy statements to enable the peering.

To implement the IAM policies required for peering, the two VCN administrators must designate one administrator as the requester and the other as the acceptor. The requester must be the one to initiate the request to connect the two LPGs. In turn, the acceptor must create a particular IAM policy that gives the requester permission to connect to LPGs in the acceptor's compartment. Without that policy, the requester’s request to connect fails. Either VCN administrator can delete a peering connection by deleting their LPG.

Routing and Traffic Control

As part of configuring the VCNs, each administrator must update the VCN routing to enable traffic to flow between the VCNs. In practice, this looks just like routing you set up for any gateway, such as an internet gateway or dynamic routing gateway. For each subnet that needs to communicate with the other VCN, you update the subnet route table. The route rule specifies the destination traffic's CIDR and your LPG as the target. Your LPG routes traffic that matches that rule to the other LPG, which in turn routes the traffic to the next hop in the other VCN.

You can control packet flow over the peering connection with route tables in your VCN. For example, you can restrict traffic to only specific subnets in the other VCN. Without deleting the peering, you can stop traffic flow to the other VCN by removing route rules that direct traffic from your VCN to the other VCN. You can also effectively stop the traffic by removing any security list rules that enable ingress or egress traffic with the other VCN. This does not stop traffic flowing over the peering connection, but stops it at the VNIC level.

Security Rules

Each VCN administrator must ensure that all outbound and inbound traffic with the other VCN is intended, expected and well-defined. In practice, this means implementing security list rules that explicitly state the types of traffic your VCN can send to the other and accept from the other. If your subnets use the default security list, there are two rules allowing SSH and ICMP ingress traffic from anywhere, thus also the other VCN. Evaluate these rules and whether you want to keep or update them.

In addition to security lists and firewalls, evaluate other OS-based configuration on the instances in your VCN. There could be default configurations that do not apply to your own VCN CIDR, but inadvertently apply to the other VCN CIDR.

Connecting VCNs through a Local Peering Gateway

On Compute Cloud@Customer, a Local Peering Gateway (LPG) is a way to connect VCNs so that elements in each VCN can communicate, even using private IP address.

The following components are required to set up a peering connection:

- Two VCNs with nonoverlapping CIDRs
- A local peering gateway (LPG) on each VCN in the peering relationship
- A connection between the two LPGs
- Route rules to enable traffic over the peering connection to and from the desired subnets in the respective VCNs
- Security rules to control the types of traffic allowed to and from the instances in the subnets in question

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, under **Networking**, select **Virtual Cloud Networks**.

   A list of previously configured VCNs in compartments is displayed. If the compartment you're creating the local peering gateway in isn't displayed, use the drop-down menu to select the correct compartment.
2. Select the name of the VCN.
3. Under the **Resources** menu, select **Local Peering Gateways**.
4. Select **Create Local Peering Gateway**.
5. Enter the required information:

   - **Name:** Enter a name. Avoid entering confidential information.
   - **Create in Compartment:** Select the compartment in which to create the local peering gateway.
   - **Route Table Association (Optional)** Optionally, you can associate a route table with the local peering gateway. A list of configured route tables for the selected compartment is in a drop-down menu. You can change the compartment by selecting **(change)** next to the compartment name.
   - **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
6. Select **Create Local Peering Gateway**.

   The Local Peering Gateway is now ready for connecting VCNs with Establish Peering Connection, and ready for the addition of route rules or security settings.

Using the CLI

Use the oci network local-peering-gateway create command and required parameters to create a new local peering gateway (LPG) for the specified VCN.

```
oci network local-peering-gateway create --compartment-id <compartment_OCID>
 --vcn-id <vcn_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateLocalPeeringGateway operation create a new local peering gateway (LPG) for the specified VCN.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Connecting to the On-Premises Network through a Dynamic Routing Gateway (DRG)

On Compute Cloud@Customer, a dynamic routing gateway, or DRG, provides a path for private network traffic between the VCN and an on-premises network. This traffic is routed to the data center network and on to its destination.

The data center network is considered a public network because it's external to the Compute Cloud@Customer environment. However, no form of tunneling is required because traffic doesn't travel over the internet. This is a significant difference with public cloud environments.

For the purpose of access control, when creating a DRG, you must specify the compartment where you want the DRG to reside. If you aren't sure which compartment to use, put the DRG in the same compartment as the VCN.

A DRG is a standalone object. To use it, you must attach it to a VCN. In the API, the process of attaching creates a DRG Attachment object with its own OCID. To detach the DRG, you delete the attachment object.

A VCN can be attached to only one DRG at a time, though a DRG can be attached to multiple VCNs. You can detach a DRG and reattach it at any time. After attaching a DRG, you must update the routing in the VCN to use the DRG. Otherwise, traffic from the VCN will not flow to the DRG.

The basic routing scenario sends traffic from a subnet in the VCN to the DRG. When sending traffic from the subnet to your on-premises network, you set up a rule in the subnet route table. The rule's destination CIDR is the CIDR for the on-premises network or a subnet within, and the rule's target is the DRG.

Creating a Dynamic Routing Gateway

On Compute Cloud@Customer, a DRG is the 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).

After the DRG is configured, the DRG can be attached to more than one VCN and, similar to 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.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, under **Networking**, select **Dynamic Routing Gateway**s (DRGs).
2. Select **Create Dynamic Routing Gateway**.
3. Fill in the required 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.
   - **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
4. Select **Create Dynamic Routing Gateway**.

   The Dynamic Routing Gateway is now ready for the addition of DRG attachments. See Attaching VCNs to a Dynamic Routing Gateway on page 175.

Using the CLI

Use the oci network drg create command and required parameters to create a dynamic routing gateway.

```
oci network drg create [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateDrg operation to create a dynamic routing gateway.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Attaching VCNs to a Dynamic Routing Gateway

On Compute Cloud@Customer, you can connect many VCNs to a DRG, but each VCN can have only one DRG attached. Ensure that the route tables and security lists allow communication.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, under **Networking**, select **Dynamic Routing Gateway**s (DRGs).
2. At the top of the page, select the compartment that contains the DRG.
3. Select the name of the DRG that you plan to attach to a VCN.
4. Select **Attach to Virtual Cloud Network**.
5. From the list of VCNs in the drop-down list, select the VCN.
6. Select the VCN to attach the DRG to from the list of VCNs in the drop down list.
7. Select the correct compartment.
8. Select **Attach to DRG**.

   The DRG is attached to the selected VCN.
9. Repeat the process to attach the other VCNs to the DRG.

Using the CLI

Use the oci network drg-attachment create command and required parameters to attach the specified VCN to a DRG.

```
oci network drg-attachment create [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateDrgAttachment operation to attach the specified VCN to a DRG.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Accessing Oracle Services through a Service Gateway

On Compute Cloud@Customer, a VCN can have only one service gateway. You create a service gateway in the context of a specific VCN, so the gateway is automatically attached to that VCN upon creation. A service gateway allows traffic to and from all subnets at the time of creation. There's no mechanism to block or disable this traffic.

Certain services, such as the Object Storage service, are exposed internally over a conceptual *Services Network*. Normally, these services would use public IP addresses that can be reached over a public network – or in a public cloud model, over the internet. Instead, the purpose of a service gateway is to enable a VCN to privately access the services in the Services Network, meaning resources in a private subnet, within a tightened down VCN with no external access, can still connect to those services.

In Compute Cloud@Customer, these services are implemented at the infrastructure level through the management node cluster. Technically, the service endpoints, which are fully qualified domain names, are reachable from the

entire on-premises network by default. This means the service gateway has no real function. Specifically for private cloud use, there's no need to configure a service gateway and associated route rules to enable private access to the service endpoints. However, to maintain compatibility with Oracle Cloud Infrastructure, the concept of a service gateway does exist.

The service gateway uses a service CIDR label, which is a string that represents the endpoints for the service or group of services of interest. This means that you don't have to know the specific endpoints. If the service's endpoint changes in the future, you don't need to make any adjustments. You use the service CIDR label when you configure the service gateway. You can create the service gateway and configure route rules involving a *"Service CIDR Block"*. However, remember they serve no purpose other than compatibility.

Security rules aren't implemented for service gateways.

Creating a Service Gateway

On Compute Cloud@Customer, you can create one service gateway per VCN. The service gateway is automatically attached to the VCN it's 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 CIDR block as the rule destination and the service gateway as the rule target.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. At the top of the page, select the compartment that contains the VCN for which you want to create a Service Gateway.
3. Select the name of the VCN.
4. Under **Resources**, select **Service Gateways**.

   The number of configured Service Gateways in parentheses must be zero (0).
5. Select **Create Service Gateway**.
6. Fill in the required information:

   - **Name:** Provide a name or description for the Service Gateway.

     Avoid entering confidential information.
   - **Create in Compartment:** Select the compartment in which to create the Service Gateway.
   - **Services:** Select the service from the list.
   - **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
7. Select **Create Service Gateway**.

   The Service Gateway is now ready for the addition of route rules or security settings.

Using the CLI

Use the oci network service-gateway create command and required parameters to create a new service gateway in the specified compartment.

```
oci network service-gateway create [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateServiceGateway operation to create a new service gateway in the specified compartment.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Network Gateway Example: Internet Gateway

VCNs are the basic networking unit of Compute Cloud@Customer, 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 aren't allowed. 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 Console or the CLI.

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

Internet Gateway Setup

To configure an internet gateway on Compute Cloud@Customer, start by setting up the IGW.

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 be determined. These include ingress HTTPS connections, ingress ICMP pings, or other types of traffic. The IGW primarily responds to ingress network protocol requests.

To set up an IGW, see Providing Public Access through an Internet Gateway on page 170.

After the IGW is 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

For each public subnet that uses the internet gateway, you must update the subnet 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's not shown in this example. The route rule uses CIDR block 0.0.0.0/0 so that all traffic not covered by other rules in the route table go to the IGW target specified in the new rule.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. If needed, change the compartment to find the resource you want.
3. Select the name of the VCN for which you want to create a route table. The VCN details page is displayed.
4. Under **Resources**, select **Route Tables**.
5. Go to the details page of the Default Route Table and select **Add Route Rule**.
6. Select **+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. Select 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."
7. Select **Create Route Table Rule**.

   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 CLI

Use the oci network route-table update command and required parameters to update the route rules.

> **Note:**
>
> The `routeRules` object you provide replaces the entire existing set of rules.

```
oci network route-table update --rt-id <route-table_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Procedure

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 specified 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
```

Using the API

Use the UpdateRouteTable operation to update the specified route table's display name or route rules. Avoid entering confidential information.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Establish the Internet Gateway Security Rules

After the IGW is 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 isn't 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).

For more information about security lists and NSGs, see Controlling Traffic with Security Lists on page 158 and Controlling Traffic with Network Security Groups on page 163.

> **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. We recommend that you become familiar with this basic access set of rules. If you don't use the default security list, ensure 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 aren't allowed. 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 on page 159.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. If needed, change the compartment to find the resource you want.
3. Select the name of the VCN for which you want to add the rule to a security list. The VCN details page is displayed.
4. Under **Resources**, select **Security Lists**.
5. Update or add new rules as follows:
   - For the security list that you want to add the rule, select the Actions menu (⋮), then select **Edit**. Make changes as described below.
   - To add a new rule, in the **Allow Rules** sections, select **+New Rule**. You can also update the security list name and tags. Make changes as described below.
6. Select **Save Changes**.

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. Ensure that the stateless box is unchecked. For more information about stateless and stateful rules, see Controlling Traffic with Security Lists on page 158.
- **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."

You can always edit the new rule at any time.

Using the CLI

Use the oci network security-list update command and required parameters to update the specified security list's display name or rules. Avoid entering confidential information.

> **Note:**
>
> The `routeRules` object you provide replaces the entire existing set of rules.

```
oci network security-list update --security-list-id <security_list_OCID>
 [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Procedure

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 on page 159. 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
```

Using the API

Use the UpdateSecurityList operation to update the specified security list's display name or rules.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Configuring VNICs

On Compute Cloud@Customer, the compute nodes have physical network interface cards (NICs). When you create a compute instance, the Networking service ensures that a VNIC is created on top of a physical interface, so that the instance can communicate over the network.

Each instance has a primary VNIC that's automatically created and attached. The primary VNIC resides in the subnet you specify when creating the instance. It can't be removed from the instance.

A VNIC enables an instance to connect to a VCN and determines how the instance communicates with endpoints inside and outside of the VCN. Each VNIC resides in a subnet within a VCN and includes these items:

- One primary private IPv4 address from the subnet the VNIC is in.
- Up to 31 optional private IPv4 addresses from the subnet the VNIC is in.
- An optional public IPv4 address for each private IP, assigned at your discretion.
- An optional host name for DNS for each private IP address.
- A MAC address.
- A flag to enable or disable the source/destination check on the VNIC network traffic.
- Optional membership of one or more network security groups (NSGs).
- An Oracle-assigned identifier (OCID).
- An optional friendly name you can choose and assign.

You can add secondary VNICs to an instance after the instance is created. To be able to use a secondary VNIC, you must also configure the instance OS for it. The maximum number of VNICs for an instance varies by shape. Each secondary VNIC can be in a different subnet than the primary VNIC, either within the same VCN or a different one. A secondary VNIC can also be in the same subnet as the primary VNIC. However, attaching multiple VNICs from the same subnet CIDR block to an instance can introduce asymmetric routing, especially on Linux instances.

> **Note:**
>
> To avoid asymmetric routing in a configuration with multiple IP addresses from the same subnet, Oracle recommends assigning multiple private IP addresses to one VNIC, or using policy-based routing.
>
> If traffic comes in to a service on the instance through a secondary VNIC and the service replies, the reply packets automatically have that VNIC IP address as the source IP address. Policy-based routing is required for that reply to go back out on the same interface and find the correct default gateway.

Secondary VNICs must always be attached to an instance and can't be moved. The process of creating a secondary VNIC automatically attaches it to the instance. The process of detaching a secondary VNIC automatically deletes it. They're automatically detached and deleted when you delete the instance. An instance's bandwidth is fixed regardless of the number of VNICs attached. You can't specify a bandwidth limit for a particular VNIC on an instance.

By default, every VNIC performs the source/destination check on its network traffic. The VNIC looks at the source and destination listed in the header of each network packet. If the VNIC isn't the source or destination, then the packet is dropped. If the VNIC needs to forward traffic – for example, if it needs to perform Network Address Translation (NAT) – you must disable the source/destination check on the VNIC.

VNICs reside in a subnet but attach to an instance. The VNIC attachment to the instance is a separate object from the VNIC or the instance itself. The VNIC and subnet always exist together in the same compartment, but the VNIC attachment to the instance always exists in the instance's compartment. This distinction affects your IAM policies if you set up an access control scenario where network administrators manage the network and other users manage instances.

VNICs are rate limited to prevent very active applications from reducing the available bandwidth of other applications to unacceptable levels. Rate limiting applies to each instance’s VNIC interface.

The rate is limited to a value below the maximum network bandwidth associated with each instance shape. Generally, the bandwidth limit increases with the maximum number of VNICs a shape uses, but there is no maximum bandwidth guarantee.

You can't specify the bandwidth limits directly. Rate limiting is a system function. The rate applied to each VNIC is based on the shape option during instance creation.

Viewing VNICs and VNIC Attachments

On Compute Cloud@Customer, you can view VNIC attachments and for a particular instance and view VNIC details.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. At the top of the page, select the compartment that contains the instance in which you want to view VNIC attachments.
3. Select the name of the instance.

   The instance details page is displayed.
4. Under **Resources**, select **Attached VNICs**.

   The list of attached VNICs for that instance is displayed.
5. To view the details page for the VNIC, select the name of the attached VNIC.

Using the CLI

Use the oci compute vnic-attachment list command and required parameters to list the VNIC attachments in the specified compartment. A VNIC attachment resides in the same compartment as the attached instance.

```
oci compute vnic-attachment list [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListVnicAttachments operation to list the VNIC attachments in the specified compartment. A VNIC attachment resides in the same compartment as the attached instance.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Creating and Attaching a Secondary VNIC

On Compute Cloud@Customer, you can add more VNICs to an instance.

The number of secondary VNICs that you can add to an instance depends on the shape of the instance. See Compute Shapes on page 301.

After you attach a secondary VNIC, log onto the instance to configure the instance OS to use the new interface. See Configuring the Instance OS for a Secondary VNIC on page 184.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. At the top of the page, select the compartment that contains the instance to which you want to add a secondary VNIC.
3. Select the name of the instance.
4. Under **Resources**, select **Attached VNICs**.

   The primary VNIC and any secondary VNICs attached to the instance are displayed.
5. Select **Create VNIC Attachment**.
6. 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.

   > **Note:**
   >
   > Specifying the same subnet for this VNIC that's specified for another VNIC for this instance can introduce asymmetric routing.
   >
   > 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's in this subnet.
7. 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 isn't 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.
8. If you selected a public subnet, you can specify whether to automatically assign a public IPv4 address object to the VNIC private IP address object.
9. (Optional) Specify the following private IP information.

   - **Private IP Address**: An address that's within the CIDR block range assigned to the subnet and not already in use. If you don't 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.
10. (Optional) Add this VNIC to an NSG.

    By default, the new VNIC isn't 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. Select **Add Another NSG** to attach to another NSG.
c. To remove an NSG from the list, select 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 on page 163 for information about NSGs.

11. Select **Create Attachment**.

The secondary VNIC is created and then displayed on the **Attached VNICs** list for the instance.

12. Configure the instance OS to use the secondary VNIC. See Configuring the Instance OS for a Secondary VNIC on page 184.

Using the CLI

Use the oci compute instance attach-vnics command and required parameters to create a secondary VNIC and attaches it to the specified instance. You can specify either –subnet-id or –vlan-id for this create request, but not both.

```
oci compute instance attach-vnic [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the AttachVnic operation to create a secondary VNIC and attaches it to the specified instance.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Configuring the Instance OS for a Secondary VNIC

On Compute Cloud@Customer, after you create a secondary VNIC, log in to the instance OS and configure the OS to use the new VNIC.

For Oracle Linux and Microsoft Windows, you can use utilities and scripts that are provided by Oracle. The scripts use information from the instance metadata. See Retrieving Instance Metadata from Within the Instance on page 324 for instructions about how to view that data on the instance OS, including how to show the VNIC data.

For other OSs, consult the OS documentation.

Oracle Linux Instance OS Configuration

Use the oci-network-config utility to perform the OS configuration required for secondary VNICs on instances that run Oracle Linux.

Microsoft Windows Instance OS Configuration

You must configure the secondary VNIC within the OS. You can configure the secondary VNIC manually in the Microsoft Windows OS, or create a PowerShell script.

Ensure that the script includes these actions:

- Check if the network interface has an IP address and a default route.
- Enable the OS to recognize the secondary VNIC, the **script must overwrite the IP address and default route with static settings (which effectively disables DHCP).** The script should prompt you with a choice: to overwrite with the static settings, or exit.

Overall Process to Add a Secondary VNIC

1. Create and attach a secondary VNIC to the instance. Keep the VNIC OCID handy so you can provide it later when you run the configuration script.
2. Connect to the instance with Remote Desktop.
3. Run your script as an administrator.

Oracle Solaris Instance OS Configuration

Use the `ipadm` command to configure network interfaces persistently.

More Information about Configuring OSs for VNICs

See this OCI page: Virtual Network Interface Cards (VNICs).

Updating a VNIC

On Compute Cloud@Customer, 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.

Avoid entering confidential information in names.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. At the top of the page, select the compartment that contains the instance for which you want to update a VNIC.
3. Select the name of the instance.

   The instance details page is displayed.
4. Under **Resources**, select **Attached VNICs**.

   The list of attached VNICs for that instance is displayed.
5. For the VNIC that you want to update, select the Actions menu (⋮) and then select **Edit**.
6. In the **Update VNIC** dialog box, 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 on page 183 for information about the Skip Source/Destination Check selection.

   If you change the Enable Network Security Groups box from cleared to selected, 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 select Add Another NSG to attach to another NSG.

   If more than one NSG is already listed, you can select the trash can next to an existing NSG to detach this VNIC from that NSG. To detach the last NSG or all NSGs, clear the Enable Network Security Groups box.

   See Controlling Traffic with Network Security Groups on page 163 for information about NSGs.
7. Select **Update VNIC**.

Using the CLI

Use the oci network vnic update command and required parameters to update the specified VNIC.

```
oci network vnic update [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateVnic operation to update the specified VNIC.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Secondary VNIC

On Compute Cloud@Customer, you can delete a secondary VNIC.

This operation detaches and deletes the specified secondary VNIC.

You can't delete an instance's primary VNIC.

When you delete an instance, all attached VNICs (primary and secondary) are automatically detached and deleted.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. At the top of the page, select the compartment that contains the instance for which you want to delete a VNIC.
3. Select the name of the instance.

   The instance details page is displayed.
4. Under **Resources**, select **Attached VNICs**.

   The list of attached VNICs for that instance is displayed.
5. For the VNIC that you want to delete, select the Actions menu (⋮), then select **Delete**.
6. Select **Confirm**.

   The VNIC state changes to **Detached**. After a few seconds, the VNIC is removed from the list.
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 VNIC. See Configuring the Instance OS for a Secondary IP Address on page 189.

Using the CLI

Use the oci compute instance detach-vnic command and required parameters to detach and delete the specified secondary VNIC.

```
oci compute instance detach-vnic [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DetachVnic operation to detach and delete the specified secondary VNIC.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Managing Private IP Addresses

On Compute Cloud@Customer, a private IP address enables communication with resources on the VCN.

Each instance has at least one private IP address, and optionally, one or more public IP addresses.

The Networking service defines a private IP object, identified by an OCID, which consists of a private IPv4 address and optional host name for DNS. Each instance receives a primary private IP object during creation. The Networking service uses DHCP to assign a private IP address. This address doesn't change during the instance's lifetime and can't be removed from the instance. The private IP object is deleted when the instance is deleted.

If an instance has any secondary VNICs attached, each of those VNICs also has a primary private IP. A private IP can have a public IP assigned to it at your discretion. You can't use a private IP as the target of a route rule in a VCN.

Secondary Private IPs

After an instance is created, you can add a *secondary private IP* to one of its VNICs. You can add it to either the primary VNIC or a secondary VNIC on the instance. The secondary private IP address must be in the subnet to which the VNIC belongs. You can move a secondary private IP from a VNIC on one instance to a VNIC on another instance on condition that both VNICs belong to the same subnet.

Typical use cases for a secondary private IP include:

- **Instance failover:** You assign a secondary private IP to an instance. If a problem occurs with the instance, you can easily reassign its secondary private IP to a standby instance in the same subnet. In addition, if the secondary private IP has an associated public IP, that public IP moves along with the private IP.

- **Running multiple services or endpoints on a single instance:** For example, you could have an instance that runs multiple container pods, where each pod uses its own IP address from the VCN CIDR. The containers have direct connectivity to other instances and services in the VCN. Another example: you could run multiple SSL web servers with each one using its own IP address.

Secondary private IP addresses have the following properties:

- They're supported for all shapes and OS types.
- They can be assigned only after the instance is created, or the secondary VNIC is created and attached.
- They're automatically deleted when you delete the instance or detach and delete the secondary VNIC.
- Deleting a secondary private IP from a VNIC returns the address to the pool of available addresses in the subnet.
- A VNIC can have a maximum of 32 secondary private IPs.
- The instance's bandwidth is fixed regardless of the number of private IP addresses attached. You can't specify a bandwidth limit for a particular IP address on an instance.
- A secondary private IP can have a reserved public IP associated with it at your discretion.

> **Note:**
>
> After assigning a secondary private IP to a VNIC, you must configure the instance OS to use it. See Configuring the Instance OS for a Secondary IP Address on page 189

Viewing Private IP Addresses

On Compute Cloud@Customer, you can see the private IP addresses for instances.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. At the top of the page, select the compartment that contains the instance for which you want to view IP address information.
3. Select the name of the instance.

   The instance details page is displayed.
4. View networking information or VNIC information.

   - Select the **Networking** tab. The primary private IP address and any attached public IP address are shown in the Instance Access column.
   - Under **Resources**, select **Attached VNICs**. Select the name of the VNIC for which you want to view IP addresses.

     On the VNIC details page, Under **Resources**, select **IP Addresses**. The primary private IP address and any secondary private IP addresses, and any attached public IP addresses, are shown in the table.

Using the CLI

Use the oci network private-ip list command and required parameters to list private IP address objects.

```
oci network private-ip list [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListPrivateIps operation to list private IP address objects.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Assigning a Secondary Private IP Address

On Compute Cloud@Customer, 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 Configuring VNICs on page 181. Instead, you can create a secondary private IP address for the existing VNIC that's in the subnet that you want.

See information about secondary private IP addresses, including use cases, in Managing Private IP Addresses on page 186.

After you perform the following 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 on page 189.

Avoid entering confidential information in names and tags.

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 previously mentioned, 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 can't move a VNIC 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 Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. At the top of the page, select the compartment that contains the instance for which you want to add a secondary private IP address.
3. Select the name of the instance to which you want to add a secondary private IP address.

   The instance details page is displayed.
4. Under **Resources**, select **Attached VNICs**.

   The primary VNIC and any secondary VNICs attached to the instance are displayed.
5. Select the name of the attached VNIC to which you want to add a secondary private IP address.
6. On the VNIC details page, under **Resources**, select **IP Addresses**.
7. Select **Assign Secondary Private IP Address**.
8. In the **Attach Private IP** dialog box, all input fields are optional.

   - **IP Address:** If you don't 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's already assigned to another VNIC in the subnet. You can't enter a primary private IP address.

     If you enter an IP address that's already assigned, see the following option.
   - **Unassign if assigned:** In the previous option, if you entered a secondary private IP address that's already assigned, select this item to *move* that private IP address. The address will be unassigned from the VNIC where it's currently assigned and reassigned to this VNIC.

     If you entered an IP address that's already assigned and you don't select this item, 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.

9. Select **Attach IP Address**.

   The new secondary private IP address is shown in the table.

10. Configure the new secondary private IP address in the instance. See Configuring the Instance OS for a Secondary IP Address on page 189.

Using the CLI

Use the network vnic assign-private-ip command and required parameters to assign a secondary private IP address to the specified VNIC. The secondary private IP must be in the same subnet as the VNIC.

```
oci network vnic assign-private-ip --vnic-id <vnic_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the PrivateIp operation to assign a secondary private IP address to the specified VNIC. The secondary private IP must be in the same subnet as the VNIC.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Configuring the Instance OS for a Secondary IP Address

On Compute Cloud@Customer, after you create a secondary private IP address on a VNIC, sign 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 subinterface 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 user interface.

Updating a Secondary Private IP Address

On Compute Cloud@Customer, you can update the hostname and change the IP address of a secondary private IP address object. You can't update a VNIC primary private IP address.

To change the hostname, use the Console, CLI, or API as described in this section.

To change the IP address, delete the secondary IP address. Then create a new secondary IP address. See:

- Deleting a Secondary Private IP Address on page 190
- Assigning a Secondary Private IP Address on page 188

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. At the top of the page, select the compartment that contains the instance that has the secondary private IP address object that you want to update.
3. Select the name of the instance.

   The instance details page is displayed.
4. On the instance details page, under **Resources**, select **Attached VNICs**.

   The primary VNIC and any secondary VNICs attached to the instance are displayed.
5. Select the name of the attached VNIC that has the secondary private IP address object that you want to update.
6. On the VNIC details page, under **Resources**, select **IP Addresses**.
7. For the secondary private IP address object that you want to update, select the Actions menu (⋮), then select **Edit**.
8. In the **Attach Private IP** dialog box, update the host name.
9. Select **Attach IP Address**.

Using the CLI

Use the oci network private-ip update command and required parameters to change the display name or the hostname for a secondary private IP.

```
oci network private-ip update [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdatePrivateIp operation to change the display name or the hostname for a secondary private IP.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Secondary Private IP Address

On Compute Cloud@Customer, you can delete a secondary IP address. You can't delete a VNIC 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 become available for assignment.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. At the top of the page, select the compartment that contains the instance for which you want to delete a secondary private IP address.
3. Select the name of the instance.

   The instance details page is displayed.
4. Under **Resources**, select **Attached VNICs**.

   The primary VNIC and any secondary VNICs attached to the instance are displayed.
5. Select the name of the attached VNIC for which you want to delete a secondary private IP address.
6. On the VNIC details page, under **Resources**, select **IP Addresses**.
7. For the secondary private IP address that you want to delete, select the Actions menu (⋮) and select **Delete**.

   Confirm the deletion.
8. 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 VNIC on page 184.

Using the CLI

Use the network vnic unassign-private-ip command and required parameters to unassign a secondary private IP address from a VNIC.

```
oci network vnic unassign-private-ip [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Refer to the PrivateIp Reference for information about unassigning a secondary private IP address from a VNIC.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Managing Public IP Addresses

On Compute Cloud@Customer, a public IP address enables communication outside the VCN, including to the data center network.

All of the following network configurations 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 can't 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.

When you create an instance with a public IP address, the instance is assigned a public IP address from an address pool. Technically, the assignment is to a private IP object on the instance, and the VNIC that the private IP is assigned to must reside in a public subnet. An instance can have multiple secondary VNICs; each with a (primary) private IP. Consequently, you can assign an instance multiple public IPs across their VNICs.

The Networking service defines a public IP object, identified by an OCID, which consists of a private IPv4 address and additional properties that further define the public IP type and behavior. There are two types of public IPs:

- An **ephemeral** public IP is temporary and exists for the lifetime of the instance.
- A **reserved** public IP is persistent and exists beyond the lifetime of the instance it is assigned to. It can be unassigned and then reassigned to another instance.

The following table summarizes the differences between both types of IP addresses:

Characteristics	Ephemeral Public IP	Reserved Public IP
Allowed assignment	to a VNIC primary private IP limits: one per VNIC, two per instance	to a VNIC primary private IP limit: one per VNIC
Creation	Optionally created and assigned during instance creation or secondary VNIC creation. You can create and assign one later if the VNIC doesn't already have one.	You create one at any time. You can then assign it when you like.
Unassignment	You can unassign it at any time, which deletes it. You might do this if whoever created the instance included a public IP, but you do not want the instance to have one. When you stop an instance, its ephemeral public IPs remain assigned to the instance.	You can unassign it at any time, which returns it to the tenancy's pool of reserved public IPs.
Moving to a different resource	You can't move an ephemeral public IP to a different private IP.	You can move it at any time by unassigning and then reassigning it to another private IP. Can be in a different VCN or availability domain.
Automatic deletion	Its lifetime is tied to the private IP lifetime. Automatically unassigned and deleted when: • its private IP is deleted • its VNIC is detached or deleted • its instance is deleted	Never. Exists until you delete it.
Scope	Availability domain	Regional (can be assigned to a private IP in any availability domain in the region)
Compartment and availability domain	Same as the private IPs	Can be different from the private IPs

When you create an instance in a public subnet, the instance gets a public IP by default, unless you specify otherwise. After you create a public IP, you can't change which type it is. For example, if you create an instance that is assigned an ephemeral public IP with address 203.0.113.2, you can't convert the ephemeral public IP to a reserved public IP with address 203.0.113.2.

Resources that are designed to be directly publicly reachable automatically get a public IP address assigned from a pool upon creation. For NAT gateways, the assigned address is a regional ephemeral public IP. As with other ephemeral public IPs, it is automatically unassigned and deleted when you delete its assigned resource. However, unlike other ephemeral public IPs, you can't edit it or unassign it yourself.

Viewing Public IP Addresses

On Compute Cloud@Customer, you can use the Console, CLI, and API to view the IP addresses assigned to instances.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. At the top of the page, select the compartment that contains the instance for which you want to view the private IP address.
3. Select the name of the instance.

The instance details page is displayed.

4. View networking information or VNIC information:
   - Select the **Networking** tab. The primary private IP address and any attached public IP address are shown in the Instance Access column.
   - Under **Resources**, select **Attached VNICs**. Select the name of the VNIC for which you want to view IP addresses.

     On the VNIC details page, under **Resources**, select **IP Addresses**. The primary private IP address and any secondary private IP addresses, and any attached public IP addresses, are shown in the table.

Using the CLI

Use the oci network public-ip list command and required parameters to list the PublicIp objects in the specified compartment.

```
oci network public-ip list [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListPublicIps operation to list the PublicIp objects in the specified compartment.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Assigning an Ephemeral Public IP Address to an Instance

On Compute Cloud@Customer, you assign a public IP address to an instance by assigning 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.

An ephemeral public IP address can't be unassigned and can't 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 deleted.
- Its instance is deleted.

Avoid entering confidential information in names.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. At the top of the page, select the compartment that contains the instance for which you want to assign a public IP address.
3. Select the name of the instance.

   The instance detail page is displayed.
4. Under **Resources**, select **Attached VNICs**.
5. Select the name of the VNIC for which you want to assign a public IP address.
6. On the VNIC details page, under **Resources**, select **IP Addresses**. The primary private IP address and any secondary private IP addresses, and as any attached public IP addresses, are shown in the table.

7. If the primary private IP address doesn't already have a public IP address assigned, select the Actions menu (⋮) for the primary private IP address, and then select **Edit Public IP**.
8. In the dialog box, select **Ephemeral Public IP**.
9. (Optional) Give the ephemeral public IP address a name.
10. Select **Reserve Public IP**.

    You might need to refresh the page to see the new public IP address.

    The new public IP address is displayed 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 CLI

Use the oci network public-ip create command and required parameters to create a public IP. Use the *lifetime* property to specify whether it's an ephemeral or reserved public IP.

```
oci network public-ip create [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreatePublicIp operation to create a public IP.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Reserving a Public IP Address

On Compute Cloud@Customer, you can reserve a public IP address that's available to assign to a private IP address object at a later time.

There are two types of public IPs:

- **Ephemeral:** Think of it as temporary and existing for the lifetime of the instance.
- **Reserved:** Think of it as persistent and existing beyond the lifetime of the instance it's assigned to. You can unassign it and then reassign it to another instance whenever you like.

For more information, see Public IP Addresses.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Reserved Public IPs**.
2. Select **Reserve Public IP Address**.
3. Enter the following information:
   - **Reserved Public IP Address Name:** (Required) Enter a name for the reserved address. Avoid entering confidential information.
   - **Create in Compartment:** Select the compartment in which to reserve the address.
   - **IP Address Source:** Select the source from which to reserve the address.
4. Select **Reserve Public IP**.

   The new reserved public IP is created and displayed on the page. You can assign it to an instance if you like. See Assigning a Reserved Public IP Address to an Instance on page 195.

Using the CLI

Use the oci network public-ip create command with the `--lifetime` option and required parameters to define how the public IP handled when it is deleted.

```
oci network public-ip create --lifetime EPHEMERAL|RESERVED [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreatePublicIp operation to define how the public IP handled when it is deleted.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Assigning a Reserved Public IP Address to an Instance

On Compute Cloud@Customer, you can 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 on page 193. A reserved public IP address can be assigned to any private IP address.

See Reserving a Public IP Address on page 194 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 on page 195 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.

Avoid entering confidential information in names.

Updating a Public IP Address

On Compute Cloud@Customer, you can manage public IPs.

You can use the public IP update command to do any of the following actions:

- 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.

Avoid entering confidential information in names.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. At the top of the page, select the compartment that contains the instance for which you want to update a public IP address.
3. Select the name of the instance.

   The instance details page is displayed.
4. Under **Resources**, select **Attached VNICs**.

   Select the name of the VNIC for which you want to assign a public IP address.
5. On the VNIC details page, under **Resources**, select **IP Addresses**.

   The primary private IP address and any secondary private IP addresses, and any attached public IP addresses, are shown in the table.
6. For the private IP address for which you want to add or update a public IP address, select the Actions menu (⋮), then select **Edit Public IP**.
7. In the **Reserve Public IP** dialog box, select one of the following choices:

   - **No public IP**

     Select Reserve Public IP 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**

  Select one of the following choices:

  - **Reserve existing public IP**

    a. Select an existing public IP address. You might need to change the compartment.
    b. Select **Reserve Public IP** 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. Select **Reserve Public IP**.

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 CLI

Use the oci network public-ip update command and required parameters to update the specified public IP.

```
oci network public-ip update [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdatePublicIp operation to update the specified public IP.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Public IP Address

On Compute Cloud@Customer, in certain circumstances, you can delete public IP addresses.

An ephemeral public IP address object can't be unassigned and can't 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 deleted.
- Its instance is deleted.

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 deleted, or its instance is deleted.

Use the procedures in this section to delete a reserved public IP address object.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci network private-ip delete command and required parameters to unassign and delete the specified public IP.

```
oci network public-ip delete [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeletePublicIp operation to unassign and delete the specified public IP.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Configuring SR-IOV for Virtual Networking

On Compute Cloud@Customer, single root I/O virtualization (SR-IOV) technology enables instances to achieve low latency and high throughput simultaneously on 1 or more physical links. This technology is ideal for low-latency workloads such as video streaming, real-time applications, and large or clustered databases. Hardware-assisted (SR-IOV) networking uses the VFIO driver framework.

VCNs, DRGs, and instances must be configured and enabled for SR-IOV.

You configure a VCN to support SR-IOV by adding the `networkType`: `VFIO` tag in the OraclePCA defined tag namespace. You can't add this tag to an existing VCN. This tag can't be changed or removed from the VCN. The only way to remove the tag is to delete the VCN.

DRGs compatible with SR-IOV support also use the `networkType`: `VFIO` tag in the OraclePCA defined tag namespace. You can't add this tag to an existing DRG. This tag can't be changed or removed from the DRG. The only way to remove the tag is to delete the DRG. Only SR-IOV DRGs can attach to SR-IOV VCNs.

Linux instances natively support SR-IOV, however, you must configure a secondary VNIC on your instance, in addition to the primary network interface, as a path to the physical NIC. Only secondary vNICs can be used for SR-IOV connectivity. The vNIC type is determined by the VCN's `networkType` tag. You can use the script available in the Oracle blog *Automating SR-IOV/VFIO bond creation on PCA/C3* available at https://blogs.oracle.com/oracle-systems/post/automating-sriovvfio-bond-creation-on-oracle-compute-cloudcustomer-and-private-cloud-appliance to automate creation of the network bond interfaces taking into account the variability in hardware type and SR-IOV/VFIO type VNIC count.

> **Important:**
>
> Instances configured with SR-IOV networking are non-migratable instances. These types of instances can't be live migrated. Compute Cloud@Customer infrastructure upgrades can't migrate SR-IOV instances. You must manually shut down these types of instance before an upgrade. If you have SR-IOV instances, inform Oracle before Compute Cloud@Customer infrastructure is upgraded. See Creating a Support Request. To access support, sign in to the Oracle Cloud Console as described in Sign In to the OCI Console.

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.

Guidelines and Limitations

- Compute Cloud@Customer supports up to 84 Virtual Functions (VFs) per compute node.

- VFIO vNICs are limited to 6 per instance, assuming 1 paravirtualized vNIC.

  > **Note:**
  >
  > Each paravirtualized vNIC requires a single PCIe slot, while each VFIO vNIC required 4 PCIe slots. Attaching multiple paravirtualized vNICs may reduce the number of slots available for VFIO vNICs. and vis-a-versa. As mentioned previously, each instance is required to have one paravirtualized vNIC attached.

- IPv4 and IPv6 SR-IOV vNICs are not supported on Suse Linux.
- GPU Shapes are limited to 5 SR-IOV vNICS,1 primary paravirtualized vNIC, 3 secondary vNICs.

Configure SR-IOV Networking

1. Ensure you have the OraclePCA.networkType tag defined on the system. See Creating OraclePCA Tags (IAM in OCI) on page 120.

   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 can't 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 don't have the OraclePCA.networkType:VFIO tag set.

2. Create a VCN with SR-IOV functionality enabled.

   Create a VCN. See Creating a VCN on page 141. In the Tagging section, add the OraclePCA.networkType tag with the value VFIO.

   You must create a VCN with SR-IOV support enabled, you can't convert an existing VCN to include SR-IOV functionality.

3. If you plan to use a DRG in your SR-IOV configuration, you must create a DRG with SR-IOV functionality. Only SR-IOV DRGs can attach to SR-IOV VCNs.

   a. Create a DRG as described in Creating a Dynamic Routing Gateway on page 174. 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 can't add SR-IOV functionality to an existing DRG.

   b. Attach the SR-IOVs VCNs to the DRG as described in Attaching VCNs to a Dynamic Routing Gateway on page 175.

4. Prepare an instance for SR-IOV functionality.

   a. Create and launch an instance. See Creating an Instance on page 316.
   b. Create and attach a secondary VNIC to the instance to use as the SR-IOV network interface. The primary VNIC of the instance can't be the SR-IOV VNIC. See Creating and Attaching a Secondary VNIC on page 183.
   c. Configure the network bond interfaces, including the secondary IP address on an 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.

Managing Public DNS Zones

On Compute Cloud@Customer, the Domain Name System (DNS) lets computers use hostnames instead of IP addresses to communicate with each other.

In its most basic form, DNS returns an IP address (if known) when given a string in the DNS namespace 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.

After you create a DNS zone inside a compartment, you can't move the zone to another compartment.

When creating a DNS zone, you specify the name of the domain it manages – for example: *example.com*. You select whether the zone is primary or secondary. A primary zone contains its own DNS records, while a secondary zone retrieves its records from another zone. To access the external zone's records, the secondary zone needs at least one server IP address for the external zone. In addition, a TSIG (Transaction Signature) key may be required. TSIG keys are shared secrets used for authentication of secondary DNS zones. You can store these keys in the compartment of your choice.

Each DNS zone you create automatically contains two essential records:

- The SOA (Start of Authority) record specifies authoritative information about the DNS zone. This information includes the primary name server, the domain administrator email address, the domain serial number, and several timers related to refreshing the zone. For more information about SOA records, see RFC 1035.
- The NS (Name Server) record lists the authoritative name servers for a zone. For more information about NS records, see RFC 1035.

You configure the DNS zone by adding specific domain information in the form of resource records. For example, using an address record, you make a domain name resolve to the public IP address of an instance in a public subnet of a VCN. For a list of record types, see Public DNS Resource Record Types.

Creating a Public DNS Zone

On Compute Cloud@Customer, DNS zones are created in a compartment to associate IP addresses with portions of the DNS namespace. Zones are created in a compartment using the DNS service.

After a public DNS zone is created, you can add resource tags. You can also edit the `externalMasters` field of a `SECONDARY` zone.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **DNS**, then select **Zones**.
2. Select **Create Zone**.
3. Enter the required zone information:
   - **Zone Name:** Provide a name or description for the DNS zone.
   - **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 namespace. 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.
   - **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
4. Select **Create Zone**.

   The zone is now ready for the addition of zone records or for the configuration of TSIG Keys or Steering Policies. See Zone Records on page 200, Transaction Signature (TSIG) Keys on page 202, and Managing Traffic with Steering Policies on page 204.

Using the CLI

Use the oci dns zone create command and required parameters to create a DNS zone.

```
oci dns zone create [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateZone operation to create a DNS zone.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Zone Records

On Compute Cloud@Customer, 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 namespace, email server details, and so on.

Creating a Zone Record

On Compute Cloud@Customer, you can create a DNS 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 Console

1. In the Compute Cloud@Customer Console navigation menu, select **DNS**, then select **Zones**.

   A list of previously configured zones in compartments is displayed.
2. At the top of the page, select the compartment that contains the designated DNS zone.
3. Select 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 only SOA and NS records.
4. Enter the required zone record information:

   - **Zone Record:** Select the type of zone record you're 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.
   - **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 don't 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 want 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's what you are creating.
  - **A - IPv4 Address**: If you're creating an A type zone record, then the data is a 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.

5. Select **Create Record**.

   The zone record is now added to the zone. If you select 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 CLI

Use the oci dns record zone update command and required parameters to replace records in the specified zone with the records specified in the request body.

```
oci dns record zone update [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateZoneRecords operation to replace records in the specified zone with the records specified in the request body.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Editing a Zone Record

On Compute Cloud@Customer, you can update a DNS 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

On Compute Cloud@Customer, you can delete many, but not all DNS zone records. The initial SOA and NS records created by default when the zone is created can't be deleted.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **DNS**, then select **Zones**.

   A list of previously configured zones in compartments appears.
2. At the top of the page, select the compartment that contains the DNS zone with the record you want to delete.
3. Select 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.
4. Select the Actions menu (⋮) for the zone record that you're deleting.
5. Select **Delete**.

Using the CLI

Use the oci dns record rrset delete command and required parameters to delete all records in the specified RRSet.

```
oci dns record rrset delete [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteRRSet command and required parameters to delete all records in the specified RRSet.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Editing a Public DNS Zone

On Compute Cloud@Customer, you can add resource tags to an existing zone.

You can also edit the `externalMasters` field of a `SECONDARY` zone.

Transaction Signature (TSIG) Keys

On Compute Cloud@Customer, you can create, add, and delete TSIG 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 can't 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. Several 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.

Creating a TSIG Key

On Compute Cloud@Customer, you can create TSIG keys to ensure that DNS packets originate from an authorized sender by using shared secret keys and one-way hashing to add a cryptographic signature to the DNS packets.

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.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **DNS**, then select **TSIG Keys**.
2. Select **Create Key**.
3. Enter the required TSIG Key information:
   - **Name:** Provide a name or description for the TSIG key.
   - **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:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
4. Select **Create TSIG Key**.

   The TSIG key now available for use in the DNS zone between TSIG client and DNS server.

Using the CLI

Use the oci dns tsig-key create command and required parameters to create a new TSIG key in the specified compartment.

```
oci dns tsig-key create [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateTsigKey operation to create a new TSIG key in the specified compartment.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a TSIG Key

On Compute Cloud@Customer, you can delete a TSIG key.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **DNS**, then select **TSIG Keys**.
2. Select the Actions menu (⋮) for the TSIG key that you want to delete, and select **Delete**.

   The TSIG key is removed from the list.

Using the CLI

Use the oci dns tsig-key delete command and required parameters to delete the specified TSIG key.

```
oci dns tsig-key delete [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteTsigKey operation to delete the specified TSIG key.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Public DNS Zone

On Compute Cloud@Customer, you can delete a public DNS zone.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **DNS**, then select **Zones**.
2. At the top of the page, select the compartment that contains the public DNS zone you want to delete.
3. Select the zone name that you want to remove from the drop-down list of zones.
4. Select **Delete**.

   The DNS zone is removed from the list.

   The DNS zone is deleted.

Using the CLI

Use the oci dns zone delete command and required parameters to delete the specified DNS zone.

```
oci dns zone delete [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteZone operation to delete the specified DNS zone.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Managing Traffic with Steering Policies

On Compute Cloud@Customer, offers two types of traffic steering policies based on load balancing and some value of the IP address prefix.

DNS can do more than return an IP address (if known) when given a string in the DNS namespace 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 fully-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.

Compute Cloud@Customer supports these types of traffic management steering policies:

Policy Type	Description
Load Balancer	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	IP Prefix steering policies enable you to steer DNS traffic based on the IP Prefix of the originating query. You can divide users into groups based on the subnets from where requests originate, and steer them to specific resources based on the subdivision you made. For example, you can serve different responses for your internal users as opposed to external users.

A steering policy contains rules to answer DNS queries. You use those rules to filter answers based on the properties of the DNS request. When multiple answers are served in response to DNS queries, that group of answers is called an answer pool. The answers within a pool are sorted by priority and are marked eligible or ineligible. Ineligible answers are omitted from the response.

When attached to a DNS zone, a steering policy takes precedence over all resource records of the zone it covers, and causes DNS responses to be constructed from the steering policy rules instead. For example, if a steering policy attached to the *example.com* DNS zone contains a rule covering the domain *application.example.com* and an answer for the A (address) record type, then the steering policy responds with that answer regardless of any relevant A records that exist in the zone. If a steering policy has no answer for the record type being requested, the DNS request is passed on to the next steering policy or the base zone records.

Steering policies only support records of the types A, AAAA, CNAME. A domain can have at most one steering policy attachment covering a given record type. This means a DNS zone (*example.com*) may have multiple attached steering policies covering different record types for a given domain – for example, one A record policy and one CNAME record policy for *application.example.com*. A DNS zone may also have multiple attached steering policies covering a given record type for different domains – for example, an A record policy for *application.example.com* and an A record policy for *database.example.com*. However, multiple steering policies for the same domain and record type are not supported, because the answer can be provided by only one policy.

Creating a Load Balancer with Steering Policies

On Compute Cloud@Customer, 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.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **DNS**, then select **Steering Policies**.
2. Select **Create Steering Policy**.
3. Select **Load Balancer** 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).
   - **Answers:** Supply the answer or answers to the DNS request for FILTER, WEIGHED, and LIMIT rules. You don't have to specify which condition the answers is for: that's 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, select the toggle to change the Disabled value to TRUE.
6. **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
7. Select **Save Changes**.

   The load balancing steering policy is created.

Using the CLI

Use the oci dns steering-policy create command and required parameters to create a new steering policy in the specified compartment.

```
oci dns steering-policy create [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateSteeringPolicy operation to create a new steering policy in the specified compartment.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Creating an IP Prefix Steering Policy

On Compute Cloud@Customer, an IP prefix steering policy dynamically routes DNS request traffic to different servers based on the originating IP prefix.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu select **DNS**, then select **Steering Policies**.
2. Select **Create Steering Policy**.
3. Select IP Prefix Steering and enter the following information:
   - **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 value is 604800 (equal to 168 hours or 7 days).

4. In the Answers 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:** Select 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.
5. In the IP Prefix Steering Rules box, supply the following properties:
   - **+Add Rule:** Select 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 selecting **+Add Rule**.
6. **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
7. Select **Save Changes**.

   The IP prefix steering policy is created.

Using the CLI

Use the oci dns steering-policy create command and required parameters to create a new steering policy in the specified compartment.

```
oci dns steering-policy create [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateSteeringPolicy operation to create a new steering policy in the specified compartment.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Editing a Steering Policy

On Compute Cloud@Customer, you can edit steering policies using the Console, CLI, and API.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **DNS**, then select **Steering Policies**.
2. For the policy that you want to update, select the Actions menu (⋮), and select **Edit**.
3. Make the necessary changes on the **Edit Steering Policy** dialog box.
4. Select **Save Changes**.

   The details page for this steering policy is displayed with the updated information.

Using the CLI

Use the oci dns steering-policy update command and required parameters to update the configuration of the specified steering policy.

```
oci dns steering-policy update [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateSteeringPolicy operation to update the configuration of the specified steering policy.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Moving a Steering Policy to a Different Compartment

On Compute Cloud@Customer, you can move a steering policy to a different compartment.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci dns steering-policy change-compartment command and required parameters to move a steering policy to a different compartment.

```
oci dns steering-policy change-compartment -c <destination_compartment_OCID>
 \
--steering-policy-id <steering_policy_OCID>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

This task is not available in the API.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Attaching a Domain to a Steering Policy

On Compute Cloud@Customer, 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 Console

1. In the Compute Cloud@Customer Console navigation menu, select **DNS**, then select **Steering Policies**.
2. At the top of the page, select the compartment that contains the steering policy.
3. Select the name of the policy to which you want attach a domain.
4. Under **Resources**, select **Attached Domains**.
5. In the list of attached domains, select **Add Attached Domain**.
6. In the Add Attached Domain dialog box, enter the domain name and select a zone.
7. Select **Submit**.

   The new domain is added to the **Attached Domains** list for this steering policy.

Using the CLI

Use the oci dns steering-policy-attachment create command and required parameters to create a new attachment between a steering policy and a domain, giving the policy permission to answer queries for the specified domain.

```
oci dns steering-policy-attachment create [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateSteeringPolicyAttachment operation to create a new attachment between a steering policy and a domain, giving the policy permission to answer queries for the specified domain.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Editing an Attached Domain

On Compute Cloud@Customer, you can update a steering policy attachment with new information.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **DNS**, then select **Steering Policies**.
2. At the top of the page, select the compartment that contains the steering policy.
3. Select the name of the policy for which you want to edit an attached domain.
4. Under **Resources**, select **Attached Domains**.
5. Select the name of the attached domain that you want to edit.
6. On the top of the details page for the attached domain, select **Edit**.
7. Make the necessary changes on the **Edit Steering Policy Attachment** dialog box.
8. Select **Save Changes**.

   The details page for this steering policy attachment is displayed with the updated information.

Using the CLI

Use the oci dns steering-policy-attachment update command and required parameters to update the specified steering policy attachment with new information.

```
oci dns steering-policy-attachment update [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateSteeringPolicyAttachment operation to update the specified steering policy attachment with new information.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Steering Policy Attachment

On Compute Cloud@Customer, you can delete a steering policy attachment.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **DNS**, then select **Steering Policies**.
2. At the top of the page, select the compartment that contains the steering policy.
3. Select the name of the policy for which you want to delete an attachment.
4. Under **Resources**, select **Attached Domains**.
5. For the attached domain that you want to delete, select the Actions menu (⋮), then select **Delete**.
6. Confirm the deletion.

   The steering policy attachment is removed from the **Attached Domains** list.

Using the CLI

Use the oci dns steering-policy-attachment delete command and required parameters to delete the specified steering policy attachment. A *204* response indicates that the delete has been successful.

```
oci dns steering-policy-attachment delete [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteSteeringPolicyAttachment operation to delete the specified steering policy attachment.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Steering Policy

On Compute Cloud@Customer, you can delete a steering policy that isn't attached to any zones.

To detach a policy from a zone, see Deleting a Steering Policy Attachment on page 208.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **DNS**, then select **Steering Policies**.
2. At the top of the page, select the compartment that contains the steering policy.
3. Select the name of the policy that you want to delete.
4. Under **Resources**, select **Attached Domains**, and ensure that this policy has no attached domains.

   To deleted attached domains, see Deleting a Steering Policy Attachment on page 208.
5. At the top of the steering policy details page, select **Delete**.
6. Confirm the deletion.

Using the CLI

Use the oci dns steering-policy delete command and required parameters to delete the specified policy.

```
oci dns steering-policy delete [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteSteeringPolicy operation to delete the specified policy.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Load Balancers

On Compute Cloud@Customer, 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.

Load Balancer Types

Two major types of load balancers are available on Compute Cloud@Customer:

- Load Balancer as a Service on page 211 – 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 Balancers on page 262 – 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 Compute Cloud@Customer. 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 let the system assign an IP address.

Subnet Prerequisites

Load balancers in a Compute Cloud@Customer environment require several IP addresses for their deployment. This applies to standalone configurations, and setups used for Kubernetes Engine (OKE) control plane and worker node load balancing.

In the subnet where a load balancer is deployed, an IP address must be assigned to the gateway and to each of the three management nodes. This is in addition to the standard network and broadcast addresses. Each load balancer in the subnet also needs three IP addresses for its operation.

Here's a practical example for illustration purposes:

- Subnet overhead:

```
cidrBlock:          10.196.43.64/28
ipRange:            10.196.43.64 - 10.196.43.79
netIp:              10.196.43.64
netBroadcast:       10.196.43.79

netGw:              10.196.43.65
mn1-ip:             10.196.43.69
mn2-ip:             10.196.43.70
mn3-ip:             10.196.43.71
```

- Load balancers:

```
lb1-ip1:            10.196.43.66
lb1-ip2:            10.196.43.67
lb1-ipv:            10.196.43.68

lb2-ip1:            10.196.43.72
lb2-ip2:            10.196.43.73
lb2-ipv:            10.196.43.74
```

As the example shows, the deployment of a single load balancer requires 7 IP addresses. A /29 subnet only has 6 usable IPs, so to provide the required address range the subnet must be /28 or larger.

Characteristics Compared

The two load balancer types apply to different use cases. This section compares their characteristics side by side.

The following table summarizes the major differences.

LBaaS and NLB Major Characteristics Compared

Major Characteristic	LBaaS	NLB
Visibility	Public or Private	Public or Private

Major Characteristic	LBaaS	NLB
IP Address	Ephemeral or Reserved IP address	Ephemeral or Reserved IP address
Policy Parameters	Weighted Round Robin, IP hash, Least Connections	IP hash mapped from 5, 3, or 2 header fields
Layer 4 Functioning	Yes	Yes
Layer 7 Functioning	Yes	**No**
TLS Support	Yes	**No**

If a VCN uses network security groups (NSGs), you can associate the load balancer with an NSG. An NSG has a set of security rules that controls allowed types of inbound and outbound traffic. The rules apply only to the resources in the group. An NSG isn't a security list, where the rules apply to all the resources in any subnet that uses the list. See Controlling Traffic with Network Security Groups on page 163.

If you prefer to use security lists for the VCN, see Controlling Traffic with Security Lists on page 158.

Other differences are of an operational nature, or involved in configuration limits. Many of the NLB limitations are because of the functioning at Layer 4 and no higher. These differences are listed in the following table.

Other LBaaS and NLB Characteristics Compared

Characteristic	LBaaS	NLB
Routing of Requests	Yes	No
Persistence of Sessions	Yes	No
SSL Certificates	Yes	No
Cipher Suites	Yes	No
Listener Protocol	HTTP, HTTP2, TCP, HTTPS	TCP
Health Check Protocol	HTTP, TCP	HTTP, HTTPS, TCP
Certificate Support	Yes	No

Depending on the type of load balancer you want to use, see one of these sections:

- Load Balancer as a Service on page 211
- Network Load Balancers on page 262

Load Balancer as a Service

On Compute Cloud@Customer, you can configure the Load Balancing service (LBaaS) to automatically distribute network traffic.

The Load Balancing 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; provisioned bandwidth isn't user-configurable.

A load balancer improves resource usage, facilitates scaling, and helps ensure high availability. You can configure multiple load balancing policies and application-specific health checks to ensure that the load balancer directs traffic only to healthy instances.

Load Balancer Essentials

When working with load balancers, it's critical that you understand its building blocks and related key terminology.

Backend Server

An application server responsible for generating content in reply to the incoming TCP or HTTP traffic.

You typically identify application servers with a unique combination of private IPv4 address and port – for example: 10.25.4.101:8080, 10.25.4.102:8080, etc.

Backend Set

A logical entity defined by a list of backend servers, a load balancing policy, and a health check policy.

The backend set determines how the load balancer directs traffic to the collection of backend servers. SSL configuration is optional.

Load Balancing Policy

A load balancing policy tells the load balancer how to distribute incoming traffic to the backend servers.

Common load balancer policies include: round robin, least connections, IP hash.

Health Check

A test to confirm the availability of backend servers.

You configure the health check policy when you create a backend set. You can configure TCP-level or HTTP-level health checks for backend servers. Test results are reported through health status indicators.

Listener

A logical entity that checks for incoming traffic on the load balancer IP address.

You configure a listener's protocol and port number, and the optional SSL settings. To handle TCP and HTTP traffic, you must configure multiple listeners. Supported protocols include: TCP, HTTP/1.x, HTTP/2.

Cipher Suite

A logical entity for a set of algorithms, or *ciphers*, using Transport Layer Security (TLS) to determine the security, compatibility, and speed of HTTPS traffic.

Path Route Set

A set of path route rules to route traffic to the correct backend set without using multiple listeners or load balancers.

Shape

A template that determines the load balancer total pre-provisioned maximum capacity (bandwidth) for ingress plus egress traffic. Compute Cloud@Customer only provides the 400 Mbps shape.

Pre-provisioned maximum capacity applies to aggregated connections, not to a single client attempting to use the full bandwidth.

SSL Certificate

If you use HTTPS or SSL for the listener, you must associate an SSL server certificate (X.509) with the load balancer. A certificate enables the load balancer to terminate the connection and decrypt incoming requests before passing them to the backend servers. Traffic between the load balancer and the backend can also be SSL-encrypted.

Work Request

An object that reports on the current state of a Load Balancing request.

The Load Balancing service handles requests asynchronously. Each request returns a work request ID (OCID) as the response. You can view the work request item to see the status of the request.

Implementation Options

> **Note:**
>
> The controller software on Roving Edge Infrastructure currently doesn't support session persistence. The option applies to Compute Cloud@Customer and Private Cloud Appliance.

The default implementation of application load balancers doesn't provide session persistence through cookies. However, you can optionally select a different implementation if you need a load balancer with cookie based session persistence. The choice is made for individual load balancers, and cannot be changed after creation. Both types can be deployed on the same system.

You deploy an application load balancer with session persistence by adding a specific defined tag during the creation steps.

1. Ensure that the OraclePCA tag namespace exists on your system. If not, create the namespace.

   For more information, see Creating OraclePCA Tags (IAM in OCI) on page 120.
2. In the OraclePCA tag namespace, create the defined tag key `sessionPersistence` with two possible values: `True` or `False`.
3. When creating an application load balancer, apply the defined tag:
   - Console: in the Tagging section, select OraclePCA.sessionPersistence with value True.
   - CLI: in the `oci lb load-balancer create` command, include this parameter:

     ```
     --defined-tags '{"OraclePCA": {"sessionPersistence": True}}'
     ```

Frontend Configuration

Compute Cloud@Customer supports different types of load balancers.

In the context of load balancers, the term *frontend* refers to the components that a client can see and send requests to. The entry point of client requests is the outward-facing floating IP address of the load balancer. Incoming traffic is checked by listeners, which are configured for specific protocols and ports. Different types of rules can be defined to categorize incoming requests and route them accordingly to different sets of backend servers.

Load Balancer Types

The Load Balancing service allows you to create a public or private load balancer within a VCN. Load balancers are deployed in pairs: one active and one standby instance sharing a floating IP address. A public load balancer has a public IP address that's accessible from outside the Compute Cloud@Customer network environment. A private load balancer has an IP address from the hosting subnet, which is visible only within the VCN.

You can configure multiple listeners for an IP address to load balance transport Layer 4 and Layer 7 (TCP and HTTP) traffic. Both public and private load balancers act as reverse proxies and can route data traffic to any backend server that's reachable from the VCN.

All load balancers have a backend set to route incoming traffic to compute instances. The backend set is a logical entity that includes:

- a list of backend servers
- a load balancing policy
- a health check policy
- (optional) SSL handling

The backend servers – compute instances – associated with a backend set can exist anywhere, as long as the associated network security groups (NSGs), security lists, and route tables allow the intended traffic flow.

Private Load Balancer

To isolate a load balancer from the external network and simplify its security posture, you can create a private load balancer. The Load Balancing service assigns it a private IP address that serves as the entry point for incoming traffic.

When you create a private load balancer, the service requires only one subnet to host both the primary and standby load balancers. The load balancer is accessible only from within the VCN that contains the host subnet, or as further restricted by security rules.

The assigned floating private IP address is local to the host subnet. The primary and standby load balancers each require an extra private IP address from the host subnet.

See Creating a Load Balancer on page 227.

Public Load Balancer

To accept traffic from a network location outside of Compute Cloud@Customer, you create a public load balancer. A public load balancer must be deployed in a public subnet of a VCN that has an internet gateway configured. The service assigns the load balancer a public IP address that serves as the entry point for incoming traffic. You can associate the public IP address with a friendly DNS name through any DNS provider.

A Compute Cloud@Customer is a single Availability Domain by definition, which implies that the primary and standby load balancers can't be deployed in separate Availability Domains. Therefore, both the primary and standby load balancer are deployed in the same subnet, and they each are assigned a private IP address from that host subnet. In addition, a load balancer uses one floating public IP address that can be reassigned between standby and primary as required.

See Creating a Load Balancer on page 227.

Load Balancing Policies

On Compute Cloud@Customer, after you create a load balancer, you can apply policies to control traffic distribution to your backend servers. The primary policy types that the Load Balancing service supports, are: Round Robin, Least Connections, and IP Hash.

Round Robin

The Round Robin policy is the default load balancer policy. This policy distributes incoming traffic sequentially to each server in a backend set list. After each server has received a connection, the load balancer repeats the list in the same order.

Round Robin is a simple load balancing algorithm. It works best when all the backend servers have similar capacity and the processing load required by each request doesn't vary significantly.

Least Connections

The Least Connections policy routes incoming nonsticky request traffic to the backend server with the fewest active connections. This policy helps you maintain an equal distribution of active connections with backend servers. As with the round robin policy, you can assign a weight to each backend server and further control traffic distribution.

This policy works best with protocols using long sessions, such as LDAP and SQL. It's not recommended for short sessions as typically seen in HTTP use cases. If you select the Least Connections policy for HTTP connections, the load balancer might instead apply Round Robin internally, and even adjust server weighting dynamically.

> **Note:**
>
> In TCP use cases, a connection can be active but have no current traffic. Such connections don't serve as a good load metric.

IP Hash

The IP Hash policy uses an incoming request's source IP address as a hashing key to route nonsticky 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.

> **Caution:**
>
> Multiple clients that connect to the load balancer through a proxy or NAT router appear to have the same IP address. If you apply the IP Hash policy to your backend set, the load balancer routes traffic based on the incoming IP address and sends these proxied client requests to the same backend server. If the proxied client pool is large, the requests could flood a backend server.

When processing load or capacity varies among backend servers, you can refine each of these policy types with backend server *weighting*. Weighting affects the proportion of requests directed to each server. For example, a server weighted '3' receives three times the number of connections as a server weighted '1.' You assign weights based on criteria of your choosing, such as each server's traffic-handling capacity. Weight values must be from 1 to 100.

Load balancer policy decisions apply differently to TCP load balancers, cookie-based session persistent HTTP requests – also known as sticky requests –, and nonsticky HTTP requests.

- A TCP load balancer considers policy and weight criteria to direct an initial incoming request to a backend server. All subsequent packets on this connection go to the same endpoint.
- For nonsticky HTTP requests, the load balancer applies policy and weight criteria to every incoming request and determines an appropriate backend server. Multiple requests from the same client could be directed to different servers.

Listeners

On Compute Cloud@Customer,

The listener is a logical entity at the ingress side of the load balancer setup. It’s the key component that detects incoming traffic on the load balancer IP address. It listens for requests from clients using a particular protocol and port. Requests are subsequently routed to the appropriate backend servers based on the rules defined in the load balancer configuration. To handle TCP, HTTP, and HTTPS traffic, you must configure at least one listener per traffic type.

When you create a listener, ensure that your VCN security rules allow the listener to accept traffic.

> **Note:**
>
> To accommodate high-volume traffic, we recommend that you use *stateless* security rules for your load balancer subnets.

You can have one SSL certificate bundle per listener. You can configure two listeners, one each for ports 443 and 8443, and associate SSL certificate bundles with each listener. For more information about SSL certificates for load balancers, see Load Balancer SSL Certificates on page 234.

Cipher Suites

On Compute Cloud@Customer, a cipher suite is a set of algorithms or ciphers that help secure network connections using Transport Layer Security (TLS). You configure cipher suites for a load balancer to determine the security, compatibility and speed of HTTPS traffic. All ciphers are associated with at least one version of TLS (1.0, 1.1, 1.2).

Predefined Cipher Suites

On Compute Cloud@Customer, the Load Balancing service supports predefined cipher suites. Note that different ciphers are supported when session persistence is enabled on the load balancer.

oci-default-ssl-cipher-suite-v1 (without session persistence)

This cipher suite contains a restricted set of ciphers that are only supported in TLS version 1.2 and meet stricter compliance requirements.

- ECDHE-RSA-AES128-GCM-SHA256
- ECDHE-RSA-AES256-GCM-SHA384

oci-default-ssl-cipher-suite-v1 (with session persistence)

This cipher suite contains a restricted set of ciphers that are only supported in TLS version 1.2 and meet stricter compliance requirements.

- ECDHE-RSA-AES128-GCM-SHA256
- ECDHE-RSA-AES128-SHA256
- ECDHE-RSA-AES256-GCM-SHA384
- ECDHE-RSA-AES256-SHA384
- DHE-RSA-AES256-GCM-SHA384
- DHE-RSA-AES256-SHA256
- DHE-RSA-AES128-GCM-SHA256
- DHE-RSA-AES128-SHA256

oci-modern-ssl-cipher-suite-v1 (without session persistence)

This cipher suite offers a wider set of ciphers, but still limited to TLS version 1.2 only.

- ECDHE-ECDSA-AES128-GCM-SHA256
- ECDHE-RSA-AES128-GCM-SHA256
- ECDHE-ECDSA-AES128-SHA
- ECDHE-RSA-AES128-SHA
- ECDHE-ECDSA-AES256-GCM-SHA384
- ECDHE-RSA-AES256-GCM-SHA384
- ECDHE-RSA-AES256-SHA
- ECDHE-ECDSA-AES256-SHA
- AES128-GCM-SHA256
- AES128-SHA
- AES256-GCM-SHA384
- AES256-SHA

oci-modern-ssl-cipher-suite-v1 (with session persistence)

This cipher suite offers a wider set of ciphers, but still limited to TLS version 1.2 only.

- ECDHE-ECDSA-AES128-GCM-SHA256
- ECDHE-RSA-AES128-GCM-SHA256
- ECDHE-ECDSA-AES128-SHA256
- ECDHE-RSA-AES128-SHA256
- ECDHE-ECDSA-AES256-GCM-SHA384
- ECDHE-RSA-AES256-GCM-SHA384
- ECDHE-ECDSA-AES256-SHA384
- ECDHE-RSA-AES256-SHA384
- AES128-GCM-SHA256
- AES128-SHA256
- AES256-GCM-SHA384
- AES256-SHA256
- DHE-RSA-AES256-GCM-SHA384
- DHE-RSA-AES256-SHA256
- DHE-RSA-AES128-GCM-SHA256
- DHE-RSA-AES128-SHA256

oci-compatible-ssl-cipher-suite-v1 (without session persistence)

This cipher suite supports the broadest set of ciphers. It contains ciphers supported by TLS versions 1.1 and 1.2.

- ECDHE-ECDSA-AES128-GCM-SHA256
- ECDHE-RSA-AES128-GCM-SHA256
- ECDHE-ECDSA-AES128-SHA
- ECDHE-RSA-AES128-SHA
- ECDHE-ECDSA-AES256-GCM-SHA384
- ECDHE-RSA-AES256-GCM-SHA384
- ECDHE-RSA-AES256-SHA
- ECDHE-ECDSA-AES256-SHA
- AES128-GCM-SHA256
- AES128-SHA
- AES256-GCM-SHA384
- AES256-SHA

oci-compatible-ssl-cipher-suite-v1 (with session persistence)

This cipher suite supports the broadest set of ciphers. It contains ciphers supported by TLS versions 1.1 and 1.2.

- ECDHE-ECDSA-AES128-GCM-SHA256
- ECDHE-RSA-AES128-GCM-SHA256
- ECDHE-ECDSA-AES128-SHA256
- ECDHE-RSA-AES128-SHA256
- ECDHE-ECDSA-AES128-SHA
- ECDHE-RSA-AES128-SHA
- ECDHE-ECDSA-AES256-GCM-SHA384
- ECDHE-RSA-AES256-GCM-SHA384
- ECDHE-ECDSA-AES256-SHA384
- ECDHE-RSA-AES256-SHA384
- ECDHE-RSA-AES256-SHA
- ECDHE-ECDSA-AES256-SHA
- AES128-GCM-SHA256
- AES128-SHA256
- AES128-SHA
- AES256-GCM-SHA384
- AES256-SHA256
- AES256-SHA
- DHE-RSA-AES256-GCM-SHA384
- DHE-RSA-AES256-SHA256
- DHE-RSA-AES128-GCM-SHA256
- DHE-RSA-AES128-SHA256

oci-wider-compatible-ssl-cipher-suite-v1 (without session persistence)

This cipher suite contains all supported ciphers.

- ECDHE-ECDSA-AES128-GCM-SHA256
- ECDHE-RSA-AES128-GCM-SHA256
- ECDHE-ECDSA-AES256-GCM-SHA384
- ECDHE-RSA-AES256-GCM-SHA384
- AES256-GCM-SHA384
- ECDHE-ECDSA-AES128-SHA
- ECDHE-ECDSA-AES256-SHA
- ECDHE-RSA-AES128-SHA
- ECDHE-RSA-AES256-SHA

- AES128-GCM-SHA256
- AES128-SHA
- AES256-SHA
- DES-CBC3-SHA
- PSK-AES256-CBC-SHA
- PSK-AES128-CBC-SHA

oci-wider-compatible-ssl-cipher-suite-v1 (with session persistence)

This cipher suite contains all supported ciphers.

- ECDHE-ECDSA-AES128-GCM-SHA256
- ECDHE-RSA-AES128-GCM-SHA256
- ECDHE-ECDSA-AES128-SHA256
- ECDHE-ECDSA-AES256-GCM-SHA384
- ECDHE-RSA-AES256-GCM-SHA384
- ECDHE-ECDSA-AES256-SHA384
- ECDHE-RSA-AES256-SHA384
- AES128-SHA256
- AES256-GCM-SHA384
- AES256-SHA256
- DHE-RSA-AES256-GCM-SHA384
- DHE-RSA-AES256-SHA256
- DHE-RSA-AES128-GCM-SHA256
- DHE-RSA-AES128-SHA256
- DH-DSS-AES256-GCM-SHA384
- DHE-DSS-AES256-GCM-SHA384
- DH-RSA-AES256-GCM-SHA384
- DHE-DSS-AES256-SHA256
- DH-RSA-AES256-SHA256
- DH-DSS-AES256-SHA256
- ECDH-RSA-AES256-GCM-SHA384
- ECDH-ECDSA-AES256-GCM-SHA384
- ECDH-RSA-AES256-SHA384
- ECDH-ECDSA-AES256-SHA384
- DH-DSS-AES128-GCM-SHA256
- DHE-DSS-AES128-GCM-SHA256
- DH-RSA-AES128-GCM-SHA256
- DHE-DSS-AES128-SHA256
- DH-RSA-AES128-SHA256
- DH-DSS-AES128-SHA256
- ECDH-RSA-AES128-GCM-SHA256
- ECDH-ECDSA-AES128-GCM-SHA256
- ECDH-RSA-AES128-SHA256
- ECDH-ECDSA-AES128-SHA256
- ECDHE-ECDSA-AES128-SHA
- ECDHE-ECDSA-AES256-SHA
- ECDHE-RSA-AES128-SHA
- ECDHE-RSA-AES256-SHA
- AES128-GCM-SHA256
- AES128-SHA

- AES256-SHA
- DES-CBC3-SHA
- DHE-RSA-AES128-SHA
- DHE-RSA-CAMELLIA256-SHA
- DHE-RSA-CAMELLIA128-SHA
- DHE-RSA-SEED-SHA
- DHE-RSA-AES256-SHA
- DHE-DSS-AES256-SHA
- DH-RSA-AES256-SHA
- DH-DSS-AES256-SHA
- DHE-RSA-CAMELLIA256-SHA
- DHE-DSS-CAMELLIA256-SHA
- DH-RSA-CAMELLIA256-SHA
- DH-DSS-CAMELLIA256-SHA
- ECDH-RSA-AES256-SHA
- ECDH-ECDSA-AES256-SHA
- CAMELLIA256-SHA
- PSK-AES256-CBC-SHA
- DHE-RSA-AES128-SHA
- DHE-DSS-AES128-SHA
- DH-RSA-AES128-SHA
- DH-DSS-AES128-SHA
- DHE-RSA-CAMELLIA128-SHA
- DHE-DSS-CAMELLIA128-SHA
- DH-RSA-CAMELLIA128-SHA
- DH-DSS-CAMELLIA128-SHA
- ECDH-RSA-AES128-SHA
- ECDH-ECDSA-AES128-SHA
- CAMELLIA128-SHA
- PSK-AES128-CBC-SHA

Custom Cipher Suites

On Compute Cloud@Customer,

Instead of selecting from the predefined cipher suites, you can create a cipher suite of your own to match the specific requirements of your environment. You build a custom cipher suite by adding individual ciphers associated with the TLS versions used in your configuration. A custom cipher suite must contain at least one cipher. Include only ciphers for the TLS versions that your environment effectively supports.

> **Note:**

- Ensure compatibility between specified SSL protocols and configured ciphers in the cipher suite, otherwise the SSL handshake will fail.
- Ensure compatibility between configured ciphers in the cipher suite and configured certificates. For example: RSA-based ciphers require an RSA certificate, whereas ECDSA-based ciphers require ECDSA certificates.

Supported Ciphers

On Compute Cloud@Customer, the Load Balancing service supports specific ciphers. Note that different ciphers are supported when session persistence is enabled on the load balancer.

TLS Version 1.2 Ciphers (without session persistence)

- AES128-GCM-SHA256
- AES256-GCM-SHA384

- ECDHE-ECDSA-AES128-GCM-SHA256
- ECDHE-ECDSA-AES256-GCM-SHA384
- ECDHE-RSA-AES128-GCM-SHA256
- ECDHE-RSA-AES256-GCM-SHA384

TLS Version 1.2 Ciphers (with session persistence)

- AES128-GCM-SHA256
- AES128-SHA256
- AES256-GCM-SHA384
- AES256-SHA256
- DH-DSS-AES128-GCM-SHA256
- DH-DSS-AES128-SHA256
- DH-DSS-AES256-GCM-SHA384
- DH-DSS-AES256-SHA256
- DH-RSA-AES128-GCM-SHA256
- DH-RSA-AES128-SHA256
- DH-RSA-AES256-GCM-SHA384
- DH-RSA-AES256-SHA256
- DHE-DSS-AES128-GCM-SHA256
- DHE-DSS-AES128-SHA256
- DHE-DSS-AES256-GCM-SHA384
- DHE-DSS-AES256-SHA256
- DHE-RSA-AES128-GCM-SHA256
- DHE-RSA-AES128-SHA256
- DHE-RSA-AES256-GCM-SHA384
- DHE-RSA-AES256-SHA256
- ECDH-ECDSA-AES128-GCM-SHA256
- ECDH-ECDSA-AES128-SHA256
- ECDH-ECDSA-AES256-GCM-SHA384
- ECDH-ECDSA-AES256-SHA384
- ECDH-RSA-AES128-GCM-SHA256
- ECDH-RSA-AES128-SHA256
- ECDH-RSA-AES256-GCM-SHA384
- ECDH-RSA-AES256-SHA384
- ECDHE-ECDSA-AES128-GCM-SHA256
- ECDHE-ECDSA-AES128-SHA256
- ECDHE-ECDSA-AES256-GCM-SHA384
- ECDHE-ECDSA-AES256-SHA384
- ECDHE-RSA-AES128-GCM-SHA256
- ECDHE-RSA-AES128-SHA256
- ECDHE-RSA-AES256-GCM-SHA384
- ECDHE-RSA-AES256-SHA384

TLS Version 1.0/1.1 Ciphers Supported in TLS Version 1.2 (without session persistence)

- AES128-SHA
- AES256-SHA
- DES-CBC3-SHA
- ECDHE-ECDSA-AES128-SHA
- ECDHE-ECDSA-AES256-SHA

- ECDHE-RSA-AES128-SHA
- ECDHE-RSA-AES256-SHA
- PSK-AES128-CBC-SHA
- PSK-AES256-CBC-SHA

TLS Version 1.0/1.1 Ciphers Supported in TLS Version 1.2 (with session persistence)

- AES128-SHA
- AES256-SHA
- CAMELLIA128-SHA
- CAMELLIA256-SHADES-CBC3-SHA
- DH-DSS-AES128-SHA
- DH-DSS-AES256-SHA
- DH-DSS-CAMELLIA128-SHA
- DH-DSS-CAMELLIA256-SHA
- DH-DSS-DES-CBC3-SHA
- DH-DSS-SEED-SHA
- DH-RSA-AES128-SHA
- DH-RSA-AES256-SHA
- DH-RSA-CAMELLIA128-SHA
- DH-RSA-CAMELLIA256-SHA
- DH-RSA-DES-CBC3-SHA
- DH-RSA-SEED-SHA
- DHE-DSS-AES128-SHA
- DHE-DSS-AES256-SHA
- DHE-DSS-CAMELLIA128-SHA
- DHE-DSS-CAMELLIA256-SHA
- DHE-DSS-DES-CBC3-SHA
- DHE-DSS-SEED-SHA
- DHE-RSA-AES128-SHA
- DHE-RSA-AES256-SHA
- DHE-RSA-CAMELLIA128-SHA
- DHE-RSA-CAMELLIA256-SHA
- DHE-RSA-DES-CBC3-SHA
- DHE-RSA-SEED-SHA
- ECDH-ECDSA-AES128-SHA
- ECDH-ECDSA-AES256-SHA
- ECDH-ECDSA-DES-CBC3-SHA
- ECDH-ECDSA-RC4-SHA
- ECDH-RSA-AES128-SHA
- ECDH-RSA-AES256-SHA
- ECDH-RSA-DES-CBC3-SHA
- ECDH-RSA-RC4-SHA
- ECDHE-ECDSA-AES128-SHA
- ECDHE-ECDSA-AES256-SHA
- ECDHE-ECDSA-DES-CBC3-SHA
- ECDHE-ECDSA-RC4-SHA
- ECDHE-RSA-AES128-SHA
- ECDHE-RSA-AES256-SHA
- ECDHE-RSA-DES-CBC3-SHA
- ECDHE-RSA-RC4-SHA

- IDEA-CBC-SHA
- KRB5-DES-CBC3-MD5
- KRB5-DES-CBC3-SHA
- KRB5-IDEA-CBC-MD5
- KRB5-IDEA-CBC-SHA
- KRB5-RC4-MD5
- KRB5-RC4-SHA
- PSK-3DES-EDE-CBC-SHA
- PSK-AES128-CBC-SHA
- PSK-AES256-CBC-SHA
- PSK-RC4-SHA
- RC4-MD5
- RC4-SHA
- SEED-SHA

Request Routing

On Compute Cloud@Customer, the Load Balancing service enables you to route incoming requests to different backend servers based on certain properties of the request. The goal is to optimize the usage of backend resources.

Request routing can be configured using virtual hostnames path route sets, and rule sets.

When multiple applications are hosted on the backend, several virtual host names can be associated with the load balancer listeners, so that incoming HTTP requests matching a particular host name are routed to the right backend servers.

When backend applications use multiple endpoints or content types, represented by a distinct URI or path, HTTP requests for each of those endpoints can be routed to different backend servers based on path route rules. These path route rules contain a string that's matched against the URI of an incoming request to determine the appropriate backend destination.

Virtual host names and path route rule sets are frequently combined in request routing. Note that virtual host names take precedence over URI matching.

To apply actions to the traffic on a load balancer's listener, you can associate a rule set with the listener. Rules can add or remove HTTP request or response headers, or they can restrict access to the application hosted on the backend based on the source of the request.

Session Persistence

Session persistence is supported on Compute Cloud@Customer.

> **Tip:**
>
> Cookie-based session persistence is not provided by default. It's available for application load balancers if a defined tag is applied to enable that specific implementation. For more information, see LBaaS Implementation Options on page 213.

Session persistence is a method to direct all requests originating from a single logical client to a single backend web server. Backend servers that use caching to improve performance, or to enable log-in sessions or shopping carts, can benefit from session persistence.

You enable session persistence when you create a load balancer or when you create a backend set. You can also edit an existing backend set to enable, disable, or change the session persistence configuration.

The Load Balancing service offers two mutually exclusive cookie-based configurations for enabling session persistence: application cookie stickiness and load balancer cookie stickiness.

> **Note:**
>
> **IP address-driven session persistence**

> Some products offer session persistence support without cookies. These products depend on the IP address of the incoming request. ISP proxies and company exit gateways can issue many requests from a single IP address. In this case, a single backend server can be subject to high traffic volumes. Your backend fleet can become overwhelmed, one server at a time, even though effective load balancing is possible.
>
> Another weakness of IP address-driven session persistence is that the originating IP address can change. In this case, session persistence can be lost or the request redirected to the wrong backend server.

Application Cookie Stickiness

To configure application cookie session persistence, you only need to specify a cookie name. By default, traffic from a persistent session client is redirected to a different backend server when the original server is unavailable. This fallback behavior cannot be disabled.

The Load Balancing service activates application cookie session persistence (stickiness) when a backend server sends a Set-Cookie response header containing a recognized cookie name. The cookie name must match the name specified in the backend set configuration. If the configuration specifies a match-all pattern ("*") any cookie set by the server activates session persistence. Unless a backend server activates session persistence, the service follows the load balancing policy specified when you created the load balancer.

Requirements:

- The load balancer must operate in HTTP mode to support server side, cookie-driven session persistence.
- The client computer must accept cookies for load balancing session persistence to work.

How It Works:

The Load Balancing service calculates a hash of the configured cookie and other request parameters, and sends that value to the client in a cookie. The value stored in the cookie enables the service to route subsequent client requests to the correct backend server. If your backend servers change any of the defined cookies, the service recomputes the cookie's value and resends it to the client.

> **Note:**
>
> We recommend that you treat cookie data as an opaque entity. Do not use it in your applications.
>
> The backend server can stop application cookie persistence by deleting the session persistence cookie. If you used the match-all pattern, it must delete all cookies. You can delete cookies by sending a Set-Cookie response header with a past expiration date. The Load Balancing service routes subsequent requests using the configured load balancing policy.

Load Balancer Cookie Stickiness

When you configure load balancer cookie stickiness, the load balancer inserts a cookie into the response. The parameters configured within the cookie enable session stickiness. This method is useful when you have applications and web backend services that cannot generate their own cookies.

To configure load balancer cookie session persistence, you specify these parameters and settings:

The cookie name

If you do not specify a cookie name, the default name is `X-Oracle-BMC-LBS-Route`.

> **Note:**
>
> Ensure that the cookie name used at the backend application servers is different from the cookie name used at the load balancer. To minimize

the chance of name collision, We recommend that you use a prefix such as `X-Oracle-OCI-`.

If both a backend server and the load balancer insert cookies with the same name, the client or browser behavior can vary depending on the domain value associated with the cookie. If the name and domain values of the Set-cookie header (generated by a backend server) and the Set-cookie header (generated by the load balancer) are the same, the client or browser treats them as one cookie. The client returns only one of the cookie values in subsequent requests. If both Set-cookie names are the same, but the domain names are different, the client or browser treats them as two different cookies.

The domain in which the cookie is valid

The Set-cookie header inserted by the load balancer contains a domain attribute with the specified value. This attribute has no default value. If you do not specify a value, the load balancer does not insert the domain attribute into the Set-cookie header.

Note:

- RFC 6265 - HTTP State Management Mechanism describes client and browser behavior when the domain attribute is present or not present in the `Set-cookie` header.

  If the value of the `Domain` attribute is `example.com` in the `Set-cookie` header, the client includes the same cookie in the `Cookie` header when making HTTP requests to `example.com`, `www.example.com`, and `www.abc.example.com`. If the `Domain` attribute is not present, the client returns the cookie only for the domain to which the original request was made.
- Ensure that this attribute specifies the correct domain value. If the `Domain` attribute in the `Set-cookie` header does not include the domain to which the original request was made, the client or browser might reject the cookie. As specified in RFC 6265, the client accepts a cookie with the `Domain` attribute value `example.com` or `www.example.com` sent from `www.example.com`. It does not accept a cookie with the `Domain` attribute `abc.example.com` or `www.abc.example.com` sent from `www.example.com`.

Whether the `Set-cookie` header should contain the `Secure` attribute

The `Secure` attribute directs the client or browser to send the cookie only using a secure protocol.

Note:

If you set this field to true, you cannot associate the corresponding backend set with an HTTP listener.

Whether the `Set-cookie` header should contain the `HttpOnly` attribute

The `HttpOnly` attribute limits the scope of the cookie to HTTP requests. This attribute directs the client or browser to omit the cookie when providing access to cookies through non-HTTP APIs. For example, it restricts the cookie from JavaScript channels.

These parameters cannot be used:

The amount of time the cookie remains valid

The maximum age attribute (`maxAgeInSeconds`) of the Set-cookie header inserted by the load balancer must not be specified. It has no default value either. The client or browser typically retains the cookie until the current session ends, as defined by the client.

The URI path in which the cookie is valid

The path attribute of the Set-cookie header inserted by the load balancer must not be specified. The default value ("`/`") is always used.

Whether to disable fallback for unavailable servers

Refer to the Fallback section below.

Fallback

By default, the Load Balancing service directs traffic from a persistent session client to a different backend server when the original server is unavailable. You cannot configure the backend set to disable this fallback behavior.

> **Note:**
>
> The Load Balancing service considers a server marked *drain* available for existing persisted sessions. New requests that are not part of an existing persisted session are not sent to that server.

Connection Timeout Settings

On Compute Cloud@Customer, you can configure load balancer listeners to control the maximum idle time allowed during each TCP connection or HTTP request and response pair.

Load balancers support connection multiplexing. The load balancer can route many incoming requests from multiple clients to the destination backend server through a small number (one or multiple) of backend connections.

After your load balancer connects a client to a backend server, the connection can be closed because of inactivity. You can configure load balancer listeners to control the maximum idle time allowed during each TCP connection or HTTP request and response pair.

The following timeout settings affect the load balancer behavior:

- Keep-alive setting between load balancer and client
- Keep-alive setting between load balancer and backend server
- Idle timeout in seconds

Keep-Alive Settings

The Load Balancing service does not honor keep-alive settings from backend servers. The load balancer closes backend server connections that are idle for more than 300 seconds. We recommend that you do not allow your backend servers to close connections to the load balancer. To prevent possible 502 errors, ensure that your backend servers do not close idle connections in less than 310 seconds.

The Load Balancing service sets the keep-alive value to maintain the connection until it has been idle for 65 seconds. The value for this setting cannot be changed.

Idle Timeout Settings

When you create or edit a TCP or HTTP listener, you can specify the maximum idle time in seconds. This setting applies to the time allowed between two successive receive or two successive send network input/output operations during the HTTP request-response phase. If the configured timeout has elapsed with no packets sent or received, the client's connection is closed. For HTTP and WebSocket connections, a send operation does not reset the timer for receive operations and a receive operation does not reset the timer for send operations.

> **Note:**
>
> This timeout setting does not apply to idle time between a completed response and a subsequent HTTP request.

The default timeout values are: 300 seconds for TCP listeners and 60 seconds for HTTP listeners. The maximum timeout value is 7200 seconds.

Modify the timeout parameter if either the client or the backend server requires more time to transmit data; for example:

- The client sends a database query to the backend server and the database takes over 300 seconds to run. Therefore, the backend server does not transmit any data within 300 seconds.
- The client uploads data using the HTTP protocol. During the upload, the backend does not transmit any data to the client for more than 60 seconds.
- The client downloads data using the HTTP protocol. After the initial request, it stops transmitting data to the backend server for more than 60 seconds.
- The client starts transmitting data after establishing a WebSocket connection, but the backend server does not transmit data for more than 60 seconds.
- The backend server starts transmitting data after establishing a WebSocket connection, but the client does not transmit data for more than 60 seconds.

Managing Load Balancers

On Compute Cloud@Customer, load balancing is the method of sharing a workload equally among servers.

Load balancing prevents clients from overwhelming certain servers and helps the service to scale. When a flexible load balancer process on a system is configurable and available to all clients and servers then it's called Load Balancer as a Service (LBaaS).

LBaaS On Compute Cloud@Customer provides automated traffic distribution from one entry point to multiple servers reachable from the virtual cloud network (VCN). The service offers a load balancer with a choice of using a public or private IP address, and various load balancing policies.

When you create an LB using the CLI, you can either configure all the resources at the same time or create a minimal LB and supply other configuration details later. In other words, more than the basic LB is needed to assemble the complete LB after creation. These other components are added by editing the LB resources.

The two type of load balancers are:

- Private: A private load balancer is isolated from the network outside the Compute Cloud@Customer and security is therefore easier. A private load balancer gets a private IP address assigned from the address block that serves as the entry point for incoming traffic.
- Public: A public load balancer accepts traffic from a network location outside of Compute Cloud@Customer. Because of this outside traffic, a public load balancer is deployed in a public subnet of a VCN that has an internet gateway (IGW) configured. The service assigns the load balancer a public IP address that serves as the entry point for incoming traffic. You can associate the public IP address with a friendly DNS name through any DNS provider, but a public IP address must be available in order for public load balancer provisioning to succeed.

This chapter creates load balancers with minimal information, using the Console, API, or CLI, and then supplies the details for the other resources. An implied order can be used in the way things are configured in the Console (which differs from the LB resource listing) because some items must be configured before use in other places (for example, virtual hostnames are needed to complete listener configuration). After initial load balancer configuration, the order used in this section is:

- Cipher Suites and SSL Certificates
- Backend Sets and Backend Servers
- Virtual Hostnames
- Path Route Sets
- Listeners

This section also includes other parameters and commands that relate to these major configuration steps that aren't LB resources. They include:

- Backend-set health check
- Backend (server) health check
- LB health policy

This isn't an exhaustive list.

In addition, many configuration steps are processed as work requests. Work requests are tasks that can take some time to complete and therefore are tracked as they are processed and displayed as separate line items or CLI output. For example, the act of creating an LB or cipher suite is a work request that records and displays start and finish times, state (succeeded, failed, and so on), and other relevant details. Work requests are part of the Console LB resource list.

The status of a work request, such as "Succeeded," isn't the same as completion of the provisioning of the resource. In other words, load balancer creation can "succeed" while provisioning is still in progress.

Creating a Load Balancer

On Compute Cloud@Customer, you can create a load balancer (LB).

> **Tip:**
>
> Cookie-based session persistence is not provided by default. It's available for application load balancers if a defined tag is applied to enable that specific implementation. For more information, see LBaaS Implementation Options on page 213.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. Select **Create Load Balancer**.
3. Enter the following information:
   - **Name:** Enter a descriptive name for the LB. The name doesn't need to be unique, and you can change it. Avoid entering confidential information.
   - **Create in Compartment:** Select the compartment in which to create the LB. The LB doesn't 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.
   - Select **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 Balancers on page 209 for more information about private and public load balancers.
   - **Bandwidth:** Select the shape type (either Flexible or Dynamic) and bandwidth for this LB. The bandwidth cannot be changed after the LB is created.
     - **Flexible Shapes**. Enter a minimum bandwidth and a maximum bandwidth in megabits per second (Mbps) in the fields below the type selection. To specify a fixed shape size, set the minimum and maximum bandwidth to the same value.

       The minimum bandwidth is the amount of bandwidth that is always available to provide instant readiness for the workloads.

       The maximum bandwidth is the greatest amount of bandwidth the LB supports during time of peak workload. The maximum bandwidth is set for all listeners such that all listeners share the given bandwidth inside the LB. The maximum bandwidth is a limit and is not guaranteed to be achieved. The maximum value that you can specify is 8000 Mbps.

- **Dynamic Shapes**. Select the size from the menu below the type selection: Small (100 Mbps), Medium (400 Mbps), or Large (8000 Mbps).

> **Note:**
>
> Oracle recommends using flexible shapes. Dynamic shapes are being deprecated.

- **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. Select **Add Another NSG** if you want to attach to another NSG.
  c. To remove an NSG from the list, select the trash can to the right of that NSG. To remove the last NSG or all NSGs, clear the **Enable Network Security Groups** box.
- **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.

4. Select **Create Load Balancer**.

   The details LB details page is displayed.

What's next?

On the LB details page, under **Resources**, select resources to create to complete the configuration.

Using the CLI

Use the oci lb load-balancer create command and required parameters to create a private load balancer.

Procedure

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 examples show 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
```

```
oci lb load-balancer create --compartment-id compartment_OCID --display-
name load-balancer-name --shape-name [flexible | fixed-shape-name] --
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 Balancers on page 209 for more information about private and public load balancers.

Specify "flexible" for `--shape-name` and then specify the minimum and maximum bandwidth in megabits per second (Mbps) in the value of the `--shape-details` option. The maximum value that you can specify is 8000 Mbps. To specify a fixed bandwidth, specify the same value for both minimum and maximum.

```
--shape-details '{"maximumBandwidthInMbps": 400, "minimumBandwidthInMbps":
 400}'
```

The bandwidth can't be changed after the LB is created.

> **Note:**
>
> We recommend specifying the "flexible" shape name and not a fixed bandwidth shape name such as 400Mbps. Dynamic shapes (fixed bandwidth shape names) are being deprecated.

```
$ 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"
}
```

```
$ oci lb load-balancer create --compartment-id ocid1.compartment.unique_ID \
--display-name Private_LB1 --is-private true --shape-name flexible \
--shape-details '{"maximumBandwidthInMbps": 400, "minimumBandwidthInMbps":
 400}' \
```

```
--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": {
      "maximumBandwidthInMbps": 400,
      "minimumBandwidthInMbps": 400
    },
    "shape-name": "flexible",
    "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 didn't 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, use `oci lb -h`.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateLoadBalancer operation to create a load balancer.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Editing a Load Balancer

On Compute Cloud@Customer, You can change load balancer (LB) name and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. For the load balancer you want to edit, select the Actions menu (⋮), then select **Edit**.
4. Make your changes and select **Update Load Balancer**.

Using the CLI

Use the oci lb load-balancer update command and required parameters to edit load balancer properties.

```
oci lb load-balancer update --load-balancer-id <loadbalancer_OCID> --
display-name <name-of-load-balancer>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

If you didn't 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`.

Using the API

Use the UpdateLoadBalancer operation to update a load balancer configuration.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing Load Balancer Details

On Compute Cloud@Customer, you can view a list of existing load balancers and view their details.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the LB to go to its details page.

   Alternatively, select the Actions menu (⋮), then select the **View details** option.

Using the CLI

Use the oci lb load-balancer get command and required parameters to view load balancer details.

```
oci lb load-balancer get --load-balancer-id <loadbalancer_OCID>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the GetLoadBalancer operation to view the specified load balancer configuration information.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Load Balancer

On Compute Cloud@Customer, you can delete a load balancer (LB) and remove it from service.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. For the LB that you want to delete, select the Actions menu (⋮), and select **Terminate**.

4. Confirm the operation.

Using the CLI

Use the oci lb load-balancer delete command and required parameters to remove a load balancer from service.

```
oci lb load-balancer delete --load-balancer-id <loadbalancer_OCID>
Are you sure you want to delete this resource? [y/N]: y
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteLoadBalancer operation to stop a load balancer and remove it from service.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Managing Cipher Suites

On Compute Cloud@Customer, you can use cipher suites with a load balancer (LB) to determine the security, compatibility, and speed of HTTPS traffic.

For more details about the types of cipher suites you an use, see Cipher Suites on page 215.

Creating a Load Balancer SSL Cipher Suite

On Compute Cloud@Customer, a lload 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.

A series of predefined cipher suites exist 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 Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the Load Balancer for which you want to create the load balancer SSL cipher suite.
4. Under **Resources**, select **Cipher Suites**.

   The list of available cipher suites appears. If none are listed, you must create one.
5. To create a load balancer cipher suite, select **Create Cipher Suite**.
6. In the **Load Balancer SSL Cipher Suite** dialog box, 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.
7. Select the cipher suite components to be part of the SSL cipher suite.
8. Select **Create Cipher Suite**.

   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 CLI

Use the oci lb ssl-cipher-suite create command and required parameters to create a load balancer SSL cipher suite.

```
oci lb ssl-cipher-suite create --ciphers <ssl_ciphers_complex_type> --load-
balancer-id <load-balancer_OCID> --name <ssl_cipher_suite_name>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateSSLCipherSuite operation to create a load balancer SSL cipher suite.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing a Load Balancer SSL Cipher Suite Details

On Compute Cloud@Customer, you can 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 Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the LB for which you want to view existing cipher suites.
4. Under **Resources**, select **Cipher Suites**.

   The list of cipher suites is displayed.
5. 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 **View Details**.

Using the CLI

Use the oci lb ssl-cipher-suite get command and required parameters to view a load balancer SSL cipher suite details.

```
oci lb ssl-cipher-suite get --load-balancer-id <load-balancer_OCID> --
name <cipher-suite-name>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the GetSSLCipherSuite operation to get a load balancer SSL cipher suite details.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Editing a Load Balancer SSL Cipher Suite

On Compute Cloud@Customer, you can edit SSL cipher suites associated with a load balancer (LB) to add or remove ciphers. The name of the cipher suite can't be changed.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the Load Balancer that has the cipher suite you want to edit.
4. Under **Resources**, select **Cipher Suites**.

   The cipher suites are displayed.
5. 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 **Edit**.
   - Select the Actions menu (⋮) for the cipher suite and select **Edit**.
6. Select **Update ciphers**.

Using the CLI

Use the oci lb ssl-cipher-suite update command and required parameters to update the details of a load balancer SSL cipher suite.

```
oci lb ssl-cipher-suite update --load-balancer-id <load-balancer_OCID> --
name <cipher-suite-name>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateSSLCipherSuire operation to update the details of a load balancer SSL cipher suite.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Load Balancer SSL Cipher Suite

On Compute Cloud@Customer, you can delete an SSL cipher suite associated with a load balancer (LB).

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the Load Balancer that has the cipher suite you want to delete.
4. Under **Resources**, select **Cipher Suites**.

   The cipher suites are displayed.
5. 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 **Delete**.

Using the CLI

Use the oci lb ssl-cipher-suite delete command and required parameters to delete a load balancer cipher suite:

```
oci lb ssl-cipher-suite delete --load-balancer-id <load-balancer_OCID> --
name <cipher-suite-name>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteSSLCipherSuire operation to delete a load balancer cipher suite.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Load Balancer SSL Certificates

On Compute Cloud@Customer, you can import and manage SSL certificates through the Load Balancing service. The service doesn't generate any certificates.

An SSL certificate can be one issued by a vendor such as VeriSign or GoDaddy, or a self-signed certificate that you generate with a tool like OpenSSL or Let's Encrypt.

> **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.

If you configure HTTPS or SSL for a listener, an SSL server certificate must be associated with the load balancer. A certificate enables the load balancer to terminate the connection and decrypt incoming requests before passing them to the backend servers. You can apply the following SSL configurations to your load balancer:

- **SSL termination:** The load balancer handles incoming SSL traffic and passes the unencrypted request to a backend server.
- **Point-to-point SSL:** The load balancer terminates the SSL connection with an incoming traffic client, and then initiates an SSL connection to a backend server.
- **SSL tunneling:** If you configure the load balancer listener for TCP traffic, the load balancer tunnels incoming SSL connections to your application servers.

Load Balancing supports the TLS 1.2 protocol with a default setting of strong cipher strength.

To use standard SSL with a load balancer and its resources, you must supply a certificate. To use mutual TLS (mTLS) with your load balancer, you must add one or more certificate authority bundles (CA bundles) to your system. A certificate bundle includes the public certificate, the corresponding private key, and any associated Certificate Authority (CA) certificates. We recommend that you upload the certificate bundles before creating the listeners or backend sets you want to associate them with. Only X.509 certificates in PEM format are accepted.

Load balancers commonly use single domain certificates. However, load balancers with listeners that include request routing configuration might require a subject alternative name (SAN) certificate (also called multidomain certificate) or a wildcard certificate. The Load Balancing service supports each of these certificate types.

SSL Traffic Handling Configuration

On Compute Cloud@Customer, you can configure a load balancer to handle SSL traffic in different ways at different stages of the secured connection.

Terminating SSL at the Load Balancer

This configuration is known as *frontend SSL*. Your load balancer can accept encrypted traffic from a client. No encryption of traffic exists between the load balancer and the backend servers.

To terminate SSL at the load balancer, you must create a listener at a port such as 443, and then associate an uploaded certificate bundle with the listener.

Implementing Backend SSL

In a *backend SSL* configuration, the load balancer does not accept encrypted traffic from client servers. Traffic between the load balancer and the backend servers is encrypted.

To implement SSL between the load balancer and your backend servers, you must associate an uploaded certificate bundle with the backend set.

If you want to have more than one backend server in the backend set, sign your backend servers with an intermediate CA certificate. The intermediate CA certificate must be included as part of the certificate bundle.

Your backend services must be able to accept and terminate SSL.

Implementing Point-to-Point SSL

In a *point-to-point SSL* configuration, the load balancer accepts SSL-encrypted traffic from clients and encrypts traffic to the backend servers.

To implement point-to-point SSL, you must associate uploaded certificate bundles with both the listener and the backend set.

Peer Certificate Verification

On Compute Cloud@Customer, peer certificate verification establishes mutual authentication between a client and load balancer.

When SSL is set up on the load balancer, the listener responds to an incoming client request with a certificate that the client can verify for authenticity. Enabling peer certificate verification adds an extra layer of security: it requires the client (peer) to present a certificate that the listener can validate.

In case peer certificate verification is configured incorrectly, the client is unable to verify the certificate and returns a client SSL handshake failure. The error message varies by client type. You can use the OpenSSL utility to check the depth at which the validation fails:

```
$ openssl verify -verbose -CAfile root-cert.pem intermediate-cert.pem
error 20 at 0 depth lookup: unable to get local issuer certificate
```

To resolve these verification errors, confirm that the client certificate and certificate authority certificate match, and ensure the certificates are included in the correct order.

SSL Key Issues

On Compute Cloud@Customer, the following key-related problems are known to occur when configuring SSL for a load balancer.

Submitting Private Keys

If an error is returned when you submit a private key, these are the most common causes:

- You provided an incorrect passphrase.
- Your private key is malformed.
- The system does not recognize the encryption method used for your key.

Key Pair Mismatch

If the private and public key do not match, attempting to upload them results in a mismatch error. Use the following OpenSSL commands to confirm that the public and private key are part of the same pair:

```
openssl x509 -in certificate_name.crt -noout -modulus | openssl sha1
openssl rsa -in private_key_name.key -noout -modulus | openssl sha1
```

The `sha1` hash values returned from these commands must match exactly. If they are different, then the private key is not used to sign the public certificate, and this key cannot be used.

Private Key Consistency

In general, when errors occur that are related to a private key, you can use the OpenSSL utility to check the key's consistency. This command verifies that the key is intact, the passphrase is correct, and the file contains a valid RSA private key:

```
openssl rsa -check -in private_key.pem
```

Private Key Decryption

If the system does not recognize the encryption technology used for a private key, you should decrypt the key and upload the unencrypted version of the key with your certificate bundle. You can use the OpenSSL utility to decrypt a private key:

```
openssl rsa -in private_key.pem -out private_key_decrypted.pem
```

Managing SSL Certificates

On Compute Cloud@Customer, you can use secure socket layer (SSL) certificates with your load balancer.

Uploading Certificate Chains

Use the following process for certificate chains on Compute Cloud@Customer.

If there are multiple certificates that form a single certification chain – for example when intermediate certificate authority (CA) certificates are used –, then include all relevant certificates in a single PEM file in the correct order before uploading them to the system. The correct order begins with the certificate directly signed by the trusted root certificate authority at the bottom of the list. Any additional certificates are pasted above the signed certificate.

Combine the server certificate (`ssl-certificate.crt`) and the intermediate CA certificate (`intermediate-ca-cert.crt`) files into a single, concatenated PEM file:

```
cat ssl_certificate.crt intermediate-ca-cert.crt >> certbundle.pem
```

Adding a Load Balancer Certificate

On Compute Cloud@Customer, you can add a public SSL certificate to use with a load balancer.

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 Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the LB for which you want to add the certificate.
4. Under **Resources**, select **Certificates**.
5. Select **Create Certificate**.
6. In the **Load Balancer Create Certificate** dialog box, enter the following information:
   - **Name:** Enter a descriptive name for the certificate bundle. The name must be unique and can't 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.
   - **Certificate Authority:** Select 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.
   - **Private Key:** Select 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.
7. Select **Create Certificate**.

Using the CLI

Use the oci lb certificate create command and required parameters to create a public SSL certificate to use with a load balancer.

```
oci lb certificate create --certicate-name <certificate-name> --load-
balancer-id <load-balancer_OCID> --certificate-file <path_to_file>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateCertificate operation to create a load balancer certificate.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing a Load Balancer Certificate

On Compute Cloud@Customer, you can view a public SSL certificate used with a load balancer.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the LB for which you want to view the certificate.
4. Under **Resources**, select **Certificates**.

   All the details of the configured certificates are displayed.

Using the CLI

Use the oci lb certificate list command and required parameters to a view the public SSL certificate used with a load balancer.

```
oci lb certificate list --load-balancer-id <load-balancer_OCID>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListCertificate operation to a view the public SSL certificate used with a load balancer.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Load Balancer Certificate

On Compute Cloud@Customer, you can delete a public SSL certificate associated with an existing load balancer.

To change a certificate, delete the certificate and add a new certificate.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the LB for which you want to delete the certificate.
4. Under **Resources**, select **Certificates**.

   The configured certificates are displayed.
5. Select the Actions menu (⋮), then select **Delete**.
6. Confirm the deletion.

Using the CLI

Use the oci lb certificate delete command and required parameters to delete a public SSL certificate associated with a load balancer

```
oci lb certificate delete --certificate-name <certificate_name> --load-
balancer-id <load-balancer_OCID>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteCertificate operation to delete the public SSL certificate used with a load balancer.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Managing Backend Sets

On Compute Cloud@Customer, in the context of load balancers, the term backend refers to the components that receive, process, and respond to forwarded client requests.

Backend servers are grouped into backend sets; they receive client requests based on the configured load balancing policy. Health checks ensure that traffic only goes to healthy backend servers.

> **Tip:**
>
> Cookie-based session persistence is not provided by default. It's available for application load balancers if a defined tag is applied to enable that specific implementation. For more information, see LBaaS Implementation Options on page 213.
>
> Some backend configuration settings are specific to load balancers with session persistence. This is called out in the instructions.

Backend Sets

The backend set in a load balancer configuration consists of a list of backend servers, a load balancing policy, and a health check script.

A backend set must be associated with one or more listeners.

> **Caution:**
>
> Changing the backend set's load balancing policy interrupts traffic and may cause active connections to be dropped.

Backend Servers

Backend servers can be set up as individual compute instances or as instance pools. The transport protocol of a backend server is configured as part of the backend set. You can add and remove backend servers without disrupting traffic. When creating a load balancer, you must specify the backend servers to include in each backend set.

When you add backend servers to a backend set, you specify either the instance OCID or an IP address for the server to add. An instance with multiple VNICs attached can have multiple IP addresses pointing to it. If you identify a backend server by OCID, the Load Balancing service uses the primary VNIC primary private IP address. If you identify the backend servers to add to a backend set by their IP addresses, it is possible to point to the same instance more than once.

The load balancer routes incoming traffic to the backend servers based on the selected load balancing policy. To route traffic to a backend server, the Load Balancing service requires the IP address of the compute instance and the relevant application port. If the backend server resides within the same VCN as the load balancer, we recommend that you specify the compute instance's private IP address. The private IP address will also work if a local peering gateway enables traffic between the load balancer VCN and the backend server's VCN. If the backend server and load balancer reside in different VCNs without peering connection, you must specify the public IP address of the compute instance. You must also ensure that the VCN security rules allow external traffic.

To enable backend traffic, your backend server subnets must have appropriate ingress and egress security rules. When you add backend servers to a backend set, you can specify the applicable network security groups (NSGs). If you prefer to use security lists for your VCN, you can configure them through the Networking service.

> **Note:**
>
> To accommodate high-volume traffic, we recommend that you use stateless security rules for your load balancer subnets. See in the chapter Virtual Networking Overview.

Creating a Load Balancer Backend Set

On Compute Cloud@Customer, you can create a backend set for an existing load balancer. The backend set is a group of servers to which traffic is load balanced.

Using the CLI or API, you can create backend servers when you create the backend set. Using the Compute Cloud@Customer Console, you must add backend servers after the backend set is created.

For more information, including how to diagnose misconfigurations, see Health Checks on page 257.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the LB for which you want to create the backend set.
4. Under **Resources**, select **Backend Sets**.

   Any existing backend sets are listed, otherwise the list says **No data available**.
5. Select **Create Backend Set**.
6. Enter the following information:

   - **Name:** Enter a descriptive name for the LB backend set. This must be unique and can't be changed. Avoid entering confidential information.
   - **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.
  - **Session Persistence:** (only LB with session persistence) Select the box to enable session persistence.
    - **Disable session persistence:** No method is configured to send all requests from a single client to the same backend server. Each client request is independently load balanced.
    - **Enable application cookie persistence:** The load balancer uses an existing application-generated cookie to keep the client session on the same backend server. The application issues and manages the cookie. You enter the cookie name in the applicable field.
    - **Enable load balancer cookie persistence:** The load balancer inserts and manages its own cookie to ensure a client's requests go to the same backend server, with no application changes required. You enter the cookie name and configure to which domain and URL path it applies. Security settings determine how the cookie is exposed.
  - **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. These values are optional (the parameters have default values) except for protocol.
    - **Protocol:** Select the protocol to use: HTTP or TCP. Select 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.
    - **Response Body:** (only LB with session persistence) Enter the response body regex, such as .*, that the health checker uses.
7. Select **Create Backend Set**.

   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 CLI

Use the oci lb backend-set create command and required parameters to create a backend set for an existing load balancer.

```
oci lb backend-set create --health-checker-protocol [HTTP | TCP] --load-
balancer-id <load-balancer_OCID> --name <backend-set-name> --policy <load-
balancer-policy>
```

Example:

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.

   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"
}
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateBackendSet operation to create a backend set for an existing load balancer.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing a Load Balancer Backend Set

On Compute Cloud@Customer, you can 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 Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the LB for which you want to view the backend set details.
4. Under **Resources**, select the **Backend Sets**.

   The list of backend sets for this LB is shown.
5. Select the name of the backend set that you're interested in.
6. View the backend set details:

   - Select the **Backend Set Information** tab to see the load balancing policy and the overall health of the servers in the set.
   - Select the B**ackend Set Configuration** tab to see health checker and SSL configuration details.
   - Under **Resources**, view the list of backend servers.

Using the CLI

Use the oci lb backend-set get command and required parameters to view a list of the servers in a backend set of an existing load balancer and view their details.

```
oci lb backend-set get --load-balancer-id <loadbalancer_OCID>
--backend-set-name <backendset_name>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the GetBackendSet operation to get a list of the servers in a backend set of an existing load balancer and view their details.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Editing a Load Balancer Backend Set

On Compute Cloud@Customer, you can change load balancer (LB) backend set properties, such as the health checker protocol that's used.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the LB for which you want to modify a backend set.
4. Under **Resources**, select **Backend Sets**.
5. For the backend set that you want to modify, select the Actions menu (⋮), then select **Edit**.
6. In the Edit Load Balancer Backend Set dialog box, make your changes.
7. Select **Update Load Balancer Backend Set**.

Using the CLI

Use the oci lb backend-set update command and required parameters to change load balancer (LB) backend set properties, such as the health checker protocol used.

```
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>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateBackendSet operation to change load balancer backend set properties.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Load Balancer Backend Set

On Compute Cloud@Customer, you can delete a load balancer (LB) backend set and remove it from service.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the load balancer for which you want to delete the backend set.
4. Under **Resources**, select **Backend Sets**.
5. For the LB backend set you want to delete, select the Actions menu (⋮), then select **Terminate**.
6. Confirm the deletion.

Using the CLI

Use the oci lb backend-set delete command and required parameters to delete a load balancer backend set and remove it from service.

```
$ oci lb backend-set delete --load-balancer-id <oloadbalancer_OCID> --
backend-set-name <name> --force
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteBackendSet operation to delete a load balancer backend set and remove it from service.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Managing Backend Servers

On Compute Cloud@Customer, when creating a load balancer, you must specify the backend servers to include in each backend set.

Creating a Load Balancer Backend Server

On Compute Cloud@Customer, you can 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's 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 on page 173.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the load balancer for which you want to create a backend server.
4. Under **Resources**, select **Backend Sets**.
5. In the list of backend sets, select the name of the backend set for which you want to create a backend server.
6. On the backend set details page, select **Create Backend**.
7. 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 on page 214.
     - **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 on page 158.

       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 previous description under Compute Instances.

       To add another backend, select **Add IP Address**. You can select an instance or you can specify an IP address that's already a member of the backend set if you specify a different port.
8. Select **Submit**.

Using the CLI

Use the oci lb backend create command and required parameters to create a backend server to add to an existing load balancer backend set.

```
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>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateBackend operation to add a backend server to a backend set.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing Load Balancer Backend Server Details

On Compute Cloud@Customer, you can view a list of the backend servers in a load balancer (LB) backend set and view the configuration details of a specific backend.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the load balancer for which you want to list backend servers.
4. Under **Resources**, select Backend Sets to see the list of backend sets for this LB.
5. 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 CLI

Use the oci lb backend list command and required parameters to list the backend servers in a load balancer backend set.

```
oci lb backend list --backend-set-name <backend-set-name> --load-balancer-id <load-balancer_OCID>
```

Use the oci lb backend get command and required parameters to get the details of a load balancer backend server.

```
oci lb backend get --backend-name <backend_name> --backend-set-name <backend_set_name> --load-balancer-id <load_balancer_id> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListBackend operation to view a list of the backend servers in an existing load balancer backend set and view their details.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Editing a Load Balancer Backend Server

On Compute Cloud@Customer, you can change the weight, drain, offline, and backup attributes of a load balancer (LB) backend server.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the load balancer for which you want to edit backend server.
4. Under **Resources**, select **Backend Sets**.

5. Select the name of the backend set that contains the backend server that you want to edit.
6. Select the name of the LB backend server that you want to edit.
7. Select the Actions menu (⋮), then select **Edit**.
8. Make allowable changes in the dialog box.
9. Select **Submit**.

Using the CLI

Use the oci lb backend update command and required parameters to edit a load balancer backend server.

```
oci lb backend update --load-balancer-id <load-balancer_OCID> --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 don't have default values. You can also specify `--max-connections`. Use the `-h` option to see descriptions of these options.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateBackend operation to change the properties of a load balancer backend server that's a member of a backend set.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Load Balancer Backend Server

On Compute Cloud@Customer, you can delete a load balancer (LB) backend server from a backend set and remove it from the service.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the load balancer for which you want to delete the backend server.
4. Under **Resources**, select **Backend Sets**.
5. Select the name of the backend set that contains the backend server you want to delete.
6. Select the Actions menu (⋮), then select **Terminate**.
7. Confirm the deletion.

Using the CLI

Use the oci lb backend delete command and required parameters to remove a backend server from a given load balancer and backend set.

```
oci lb backend delete --backend-name <backend_server_name> --backend-set-
name <backendset_name> --load-balancer-id <load-balancer_OCID>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteBackend operation to remove a backend server from a given load balancer and backend set.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Managing Virtual Hostnames

On Compute Cloud@Customer, you can use virtual hostnames with a load balancer for one or more listeners.

Creating a Load Balancer Virtual Hostname

On Compute Cloud@Customer, 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 Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the LB for which you want to create the virtual hostname.
4. Under **Resources**, select **Hostnames**.
5. Select **Create Hostname**.
6. Enter the following information:
   - **Name:** A name for the LB virtual hostname.
   - **Hostname:** The virtual hostname.
7. Select **Create Hostname**.

Using the CLI

Use the oci lb hostname create command and required parameters to create a virtual hostname associated with a load balancer.

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 a Load Balancer Virtual Hostname on page 247.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

This task isn't available in the API.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing a Load Balancer Virtual Hostname

On Compute Cloud@Customer, you can view the virtual hostnames associated with a load balancer (LB).

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the Load Balancer for which you want to view the virtual hostnames.
4. Under **Resources**, select **Hostnames**.

Using the CLI

Use the oci lb hostname list command and required parameters to list all hostname resources associated with the specified load balancer..

```
oci lb hostname list --load-balancer-id <load-balancer_OCID>
```

Use the oci lb hostname get command and required parameters to get the specified hostname resource configuration information.

```
oci lb hostname get --load-balancer-id <load-balancer_OCID>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

This task isn't available in the API.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Editing a Load Balancer Virtual Hostname

On Compute Cloud@Customer, you can 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 (*) 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 Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the Load Balancer for which you want to edit the virtual hostnames.
4. Under **Resources**, select **Hostnames**.
5. In the Actions menu (⋮), select **Edit**.
6. Edit the virtual hostname.

   To change the name of the virtual hostname, delete the virtual hostname and create a new one.
7. Select **Save Changes**.

Using the CLI

Use the oci lb hostname update command and required parameters to change the virtual hostname associated with a load balancer.

```
oci lb hostname update --hostname <virtual-hostname> --load-balancer-
id <load-balancer_OCID> --name <virtual-hostname-friendly-name>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

This task is not available in the API.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Load Balancer Virtual Hostname

On Compute Cloud@Customer, you can delete a virtual hostname associated with a load balancer (LB).

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the load balancer for which you want to delete the virtual hostname.
4. Under **Resources**, select **Hostnames**.
5. Select the Actions menu (⋮), then select **Delete**.
6. Confirm the deletion.

Using the CLI

Use the oci lb hostname delete command and required parameters to delete the virtual hostname associated with a load balancer.

```
oci lb hostname delete --load-balancer-id <load-balancer_OCID> --
name <virtual-hostname-friendly-name>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

This task is not available in the API.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Managing Path Route Sets

On Compute Cloud@Customer, 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.

Creating Path Route Sets

On Compute Cloud@Customer, you can 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 Managing Path Route Sets on page 248 for information about URI patterns, pattern match types, and path route set limits.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. Select the name of LB for which you want to create the path route set.
3. Under **Resources**, select **Path Route Sets**.
4. Select **Create Path Route Set**.
5. Enter the following information:
   - **Name:** Enter a descriptive name for the Path Route Set. Avoid entering confidential information.
   - **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 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 **Create Path Route Set**.

Using the CLI

Use the oci lb path-route-set create command and required parameters to create a path route set for a load balancer resource to distinguish by unique URL paths such as `/admin/`, `/data/`, `/video/`, or `/cgi/`.

```
oci lb path-route-set create --name <name-of-path-route-set> --path-
routes [COMPLEX-TYPE] --load-balancer-id <load-balancer_OCID>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreatePathRouteSet operation to add a path route set to a load balancer.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing Path Route Sets

On Compute Cloud@Customer, you can view path route set details for a load balancer resource.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.

2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of load balancer for which to view path route set details.
4. Select Path Route Sets under the Resources for the load balancer.
5. Select the Actions menu (⋮), then select **View Details**.

Using the CLI

Use the oci lb path-route-set get command and required parameters to view path route set details for a load balancer resource.

```
oci lb path-route-set get --load-balancer-id <load-balancer_OCID> --path-
route-set-name <name-of-path-route-set>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the GetPathRouteSet operation to get the specified path route set's configuration information.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Editing Path Route Sets

On Compute Cloud@Customer, you can edit a path route set to update the details.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of load balancer for which to update the path route set details.
4. Under **Resources**, select **Path Route Sets**.
5. Select the Actions menu (⋮), then select **Edit**.
6. Select **Save Changes**.

Using the CLI

Use the oci lb path-route-set update command and required parameters to update path route set details for a load balancer resource.

```
oci lb path-route-set update --load-balancer-id <load-balancer_OCID> --path-
route-set-name <name-of-path-route-set>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdatePathRouteSet operation to change the specified path route set's configuration information.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting Path Route Sets

On Compute Cloud@Customer, you can delete a path route set for a load balancer resource.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of load balancer for which to delete the path route set.
4. Under **Resources**, select **Path Route Sets**.
5. Select the Actions menu (⋮), then select **Delete**.
6. Select **Confirm**.

Using the CLI

Use the oci lb path-route-set delete command and required parameters to delete a path route set resource.

```
oci lb path-route-set delete --load-balancer-id <load-balancer_OCID> --path-route-set-name <name-of-path-route-set>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeletePathRouteSet operation to delete a path route set resource.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Managing Load Balancer Rule Sets

On Compute Cloud@Customer, you associate rule sets with listeners. The rules define actions to be applied to traffic passing through the listener.

A rule set is a named set of rules associated with a load balancer and applied to one or more listeners on that load balancer. To apply a rule set to a listener, you first create the rule set that contains the rules. Rules are objects that represent actions applied to traffic at a load balancer listener. The rule set becomes a part of the load balancer's configuration. You can specify the rule set to use when you create or update a listener for the load balancer.

You can apply the same rule set to several listeners on the same load balancer. Rule sets aren't shared between load balancers. To use the same set of rules on another load balancer, you must create a new, identical rule set under that load balancer.

You can include the following types of rules in a rule set:

- Access control rules, which restrict access to application resources based on the source of the request.
- Request header rules, which add or remove HTTP request headers.
- Response header rules, which add or remove HTTP response headers.

Creating a Load Balancer Rule Set

On Compute Cloud@Customer, you create a rule set first, so you can associate it with a listener for a load balancer (LB).

Not all OCI rule set actions are supported on Compute Cloud@Customer.

- Allow rules can only specify source IP address.
- HTTP header rules can only add or remove request and response headers, not alter them.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. Select the name of LB for which you want to create the rule set.
3. Under **Resources**, select **Rule Sets**.
4. Select **Create Rule Set**.
5. Enter the following information:
   - **Name:** Enter a descriptive name for the Rule Set. Avoid entering confidential information.
   - **Rules:** Select a type from the list and enter the following information:
     - **Add HTTP Request Header:** Enter the name and value of the header to add.
     - **Remove HTTP Request Header:** Enter the name of the header to remove.
     - **Add HTTP Response Header:** Enter the name and value of the header to add.
     - **Remove HTTP Response Header:** Enter the name of the header to remove.
     - **Allow:** Enter the IP address from where the HTTP requests are forwarded to the backend. The listener responds to requests from other clients with 403: Forbidden.

   To create another rule, select the Add Rule button.

6. Select **Create Rule Set**.

Using the CLI

Use the oci lb rule-set create command and required parameters to create a rule set for a load balancer.

```
oci lb rule-set create --name <name-of-rule-set> --load-balancer-id <load-balancer_OCID> --items [COMPLEX-TYPE]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateRuleSet operation to create a rule set for a load balancer.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing a Load Balancer Rule Set

On Compute Cloud@Customer, you can view the rule sets associated with a load balancer.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of LB for which you want to view the rule sets.
4. Under **Resources**, select **Rule Sets**.
5. To view the details of a rule set, select the Actions menu (⋮), then select **View Details**.

Using the CLI

Use the oci lb rule-set list command and required parameters to display all rule sets for a load balancer.

```
oci lb rule-set list --load-balancer-id <load-balancer_OCID>
```

Use the oci lb rule-set get command and required parameters to display the details of a particular rule set associated with a load balancer.

```
oci lb rule-set get --load-balancer-id <load-balancer_OCID> --rule-set-name <name-of-rule-set>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListRuleSets operation to display all rule sets for a load balancer.

Use the GetRuleSet operation to display the details of a particular rule set associated with a load balancer.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Editing a Load Balancer Rule Set

On Compute Cloud@Customer, you can update an existing rule set for a load balancer.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of LB for which you want to update the rule set.
4. Under **Resources**, select **Rule Sets**.
5. Select the Actions menu (⋮), then select **Edit**.

6. Select **Save Changes**.

Using the CLI

Use the oci lb rule-set update command and required parameters to update a rule set for a load balancer.

```
oci lb rule-set update --rule-set-name <name-of-rule-set> --load-balancer-
id <load-balancer_OCID> --items [COMPLEX-TYPE]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateRuleSet operation to update a rule set for a load balancer.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Load Balancer Rule Set

On Compute Cloud@Customer, you can remove an existing rule set from a load balancer.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of LB for which you want to delete the rule set.
4. Under **Resources**, select **Rule Sets**.
5. Select the Actions menu (⋮), then select **Delete**.
6. Select **Confirm**.

Using the CLI

Use the oci lb rule-set delete command and required parameters to delete a rule set for a load balancer.

```
oci lb rule-set delete --rule-set-name <name-of-rule-set> --load-balancer-
id <load-balancer_OCID>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteRuleSet operation to delete a rule set for a load balancer.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Managing Listeners

On Compute Cloud@Customer, you can use listeners to check for incoming traffic on the load balancer IP address.

Creating a Load Balancer Listener

On Compute Cloud@Customer, you can 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 Console: On the LB details page, under Resources, select **Create Listener**. The supported traffic protocols are shown on the **Protocol** drop-down list.
   - Using the 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 on page 236 and Creating a Load Balancer SSL Cipher Suite on page 232.
4. At least one backend set must exist for this LB. See Managing Backend Sets on page 238.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the Load Balancer for which you want to create the listener.

   Any existing listeners are listed under the Load Balancer Information, otherwise the list says **No data available**.
4. Select **Create Listener**.
5. Enter the following information. Name, protocol, port, and backend set are required. Other parameters, such as hostnames, path route sets, rule 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.
   - **Rule Sets:** Select Add Rule Set and select a rule set from the list. Repeat to add more rule sets.
   - **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 selected and can't be cleared. For TCP protocol, this box isn't checked. If you check this box, the following parameters appear, just as they do if you selected HTTP/2 or HTTPS protocol.

       Selecting 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:** Select this box to enable peer certificate verification.
     - **TLS Version:** Select at least one TLS version.
     - **Cipher Suite:** Select a cipher suite from the list.
6. Select **Create Listener**.

   To display the details of the listener, you must view the details for the entire load balancer.

Using the CLI

Use the oci lb listener create command and required parameters to create a listener for an existing load balancer. You must list the traffic protocols that the load balancer accepts before you create the listener.

Procedure

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"
}
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateListener operation to create a listener for an existing load balancer.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Editing a Load Balancer Listener

On Compute Cloud@Customer, you can change load balancer (LB) listener properties, such as the listener communication port.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the LB for which you want to modify a backend set.
4. Under **Resources**, select **Listeners**.
5. Select the name of the load balancer listener that you want to edit.
6. For the LB listener that you want to edit, select the Actions menu (⋮), then select **Edit**.
7. In the **Edit Listener** dialog box, you can change anything except the name of the listener.
8. Select **Save Changes**.

Using the CLI

Use the oci lb listener update command and required parameters to change load balancer listener properties, such as the listener communication port used load.

```
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>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateListener operation to change load balancer listener properties.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Load Balancer Listener

On Compute Cloud@Customer, you can delete a load balancer (LB) Listener.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the LB for which you want to delete a listener.
4. Under **Resources**, select **Listeners**.
5. For the listener that you want to delete, select the Actions menu (⋮), then select **Delete**.
6. Confirm the deletion.

Using the CLI

Use the oci lb listener delete command and required parameters to delete a load balancer listener and remove it from service.

```
oci lb listener delete --listener-name <listener-name-text> --load-balancer-
id ocid1.loadbalancer.....….….….uniqueID
Are you sure you want to delete this resource? [y/N]: y
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteListener operation to delete load balancer listener.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Health Checks

Load balancer health checks are tests to confirm the availability of backend servers.

These tests occur in the form of a request or a connection attempt depending on the protocol. The health check policy includes a time interval you specify, to ensure that the backend servers are continuously monitored. If a server fails a health check, the load balancer takes the server temporarily out of rotation. If the server later passes a subsequent health check, then the LB returns that backend server to the balancing rotation.

The health check policy is configured when you create a backend set. You can configure TCP-level or HTTP-level health checks for your backend servers. For backend sets configured with SSL the health checks also use SSL encryption.

- TCP-level health checks attempt to make a TCP connection with the backend servers and validate the response based on the connection status.
- HTTP-level health checks send requests to the backend servers at a specific URI and validate the response based on the status code or entity data (body) returned.

Configure your health check protocol to match your application or service. If you run an HTTP service, then configure an HTTP-level health check. If you run a TCP-level health check against an HTTP service, then you might not get an accurate response. The TCP handshake can succeed and indicate that the service is up even when the HTTP service is incorrectly configured or having other issues. Although the health check returns no errors, you might experience transaction failures.

For example:

- The backend HTTP service has issues when communicating with the health check URL and the health check URL returns 5*nn* messages. An HTTP health check catches the message from the health check URL and marks the service as down. In this case, a TCP health check handshake succeeds and marks the service as healthy, even though the HTTP service might not be usable.
- The backend HTTP service responds with 4*nn* messages because of authorization issues or no configured content. A TCP health check does not catch these errors.

The service provides application-specific health check capabilities to help you increase availability and reduce your application maintenance window.

Health status indicators are used to report the general health of a load balancer and its backend servers/sets. The possible statuses are: `ok`, `warning`, `critical`, `unknown`. Health status is updated every three minutes. No finer granularity is available. Historical health data is not provided.

Interpreting Load Balancer Health Issues

At the highest level, load balancer health reflects the status of its components. The health status indicators provide information you might need to drill down into and investigate further. Below are several common issues that the health status indicators can help you detect and correct.

Health check misconfigured

All the backend servers for one or more of the affected listeners report as unhealthy. If your investigation finds that the backend servers do not have problems, then a backend set probably includes a misconfigured health check.

Listener misconfigured

All the backend server health status indicators report OK, but the load balancer does not pass traffic on a listener. The listener might be configured to listen on the wrong port, use the wrong protocol, or use the wrong policy. If your investigation shows that the listener is not at fault, check the security rule configuration.

Security rule misconfigured

Health status indicators help you diagnose these cases of misconfigured security rules:

- All health status indicators report OK, but traffic does not flow (as with misconfigured listeners). If the listener is not at fault, check the security rule configuration.

- All health status indicators report as unhealthy. You have checked your health check configuration and your services run properly on your backend servers. In this case, your security rules might not include the IP range for the source of the health check requests. The source IP for health check requests belongs to a compute instance managed by the Load Balancing service.

  > **Note:**
  >
  > Traffic may also be blocked because of misconfigured route tables in the compute instances.

Backend server unhealthy

A backend server might be unhealthy or the health check might be misconfigured. To see the corresponding error code, check the status field in the backend server's details through the Console or CLI.

Common Side Effects of Load Balancer Health Check Misconfiguration

Misconfiguration scenarios are known to occur regularly. This page helps with troubleshooting.

Wrong Port

In this scenario, all backend servers are reported as unhealthy. If you confirmed that there are no issues with the backend servers, you might have made a mistake setting the port. Traffic must be allowed, and the backend must be listening on that port.

Wrong Path

In this scenario, all backend servers are reported as unhealthy. If you confirmed that there are no issues with the backend servers, you might have made a mistake setting the path for the HTTP health check. It needs to match an actual application on the backend.

You can use the curl utility to run a test from a system within the same network. For example: `curl -i http://backend_ip_address/health`.

You receive the configured status code in the response:

```
"msg":"invalid statusCode","statusCode":404,"expected":"200"
```

Wrong Protocol

In this scenario, all backend servers are reported as unhealthy. If you confirmed that there are no issues with the backend servers, for an HTTP health check you might have made a mistake setting the status code. It must match the actual status code being returned from the backend. A typical mismatch is when a backend returns a 302 status code while a 200 status code is expected. This is often caused by the backend directing you to a login page or another location on the server. You can either make the backend return the expected code or use 302 in your health check configuration.

Error message: `msg:invalid statusCode, statusCode:nnn,expected:200` (where `nnn` represents the actual status code returned).

Wrong Regex Pattern

Applies only to load balancers with session persistence enabled. In this scenario, all backend servers are reported as unhealthy. If you confirmed that there are no issues with the backend servers, you might have made a mistake setting a regex pattern that's not consistent with the body, or the backend isn't returning the expected body. In this scenario, you can either change the backend to match the pattern or correct the pattern to match the backend.

Error message: `response match resulte: failed`.

Misconfigured Security Rules

In this scenario, all or some backend servers report as unhealthy. If you confirmed that there are no issues with the backend servers, then you might have improperly configured either the network security groups, security lists, or local firewalls (such as firewalld, iptables, or SELinux).

In this scenario, you can use either the curl or netcat utilities to run a test from a system within the same subnet and network security group (NSG) as your load balancer instance HTTP. For example: `curl -i http://backend_ip_address/health TCP or nc -zvw3 backend_ip_address 443`.

Local firewalls can be verified by using the command: firewall-cmd --list-all --zone=public. If the expected rules are missing from the firewall configuration, then add the required service. For example, to add HTTP port 80:

```
firewall-cmd --zone=public --add-service=http
firewall-cmd --zone=public --permanent --add-service=http
```

Viewing Health Status and Health Check Configuration

On Compute Cloud@Customer, you can 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 Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. 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.
4. On the LB details page, under **Resources**, select **Backend Sets**.
5. 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.
6. 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.
7. On the backend set details page, under Resources, select **Backends**.

   The overall health of each backend server is shown in the **Overall Health** column.

Using the CLI

Use the oci lb backend-set-health get command and required parameters to view the parameters used by the backend set of an existing load balancer to check the health of the backend servers in the set.

```
oci lb backend-set-health get --backend-set-name <backend-set-name-text> --
load-balancer-id <load-balancer_OCID>
```

Procedure

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
```

   Health check configuration parameters listed at the beginning of this section are shown.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the GetBackendSetHealth operation to view the parameters used by the backend set of an existing load balancer to check the health of the backend servers in the set.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Editing Backend Set Health Check Parameters

On Compute Cloud@Customer, you can change load balancer backend set properties, such as the health check interval.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the load balancer for which you want to list the backend set health check parameters.
4. Under **Resources**, select **Backend Sets**.
5. To edit the health check parameters of a backend set, you can:
   a. Select the Actions menu (⋮), then select **Edit**.
   b. Select the backend set to view its details, then select **Edit**.
6. Select **Save**.

Using the CLI

Use the oci lb backend-set update command and required parameters to change load balancer backend set properties, such as the health check interval.

```
oci lb backend-set update --backend-set-name [<name-of-backend-set>]--
backends [<complex-type>] --health-checker-protocol [protocol-text] --load-
balancer-id [loadbalancer_OCID] --policy [<policy-text>] --force
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateBackendSet operation to change load balancer backend set properties, such as the health check interval.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing Load Balancer Health Policy

On Compute Cloud@Customer, you can view the policy used by the load balancer to check backend set health.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the load balancer for which you want to list backend sets.
4. Under **Resources**, select **Backend Sets**.
5. To view the health check parameters of a backend set, you can:
   a. Select the Actions menu (⋮), then select **View Details**.
   b. Select the backend set.
6. On the Backend-Set Details page, under the backend-set name for which to you want to view the health policy, select **Backend Set Configuration**.

Using the CLI

Use the oci lb health-checker get command and required parameters to view the policy used by the load balancer to check backend set health.

```
oci lb health-checker get --backend-set-name [<backend-set-name-text>] --
load-balancer-id <load-balancer_OCID>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

This task is not available in the API.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Editing Load Balancer Health Check Properties

On Compute Cloud@Customer, you can change load balancer (LB) backend set health check configuration.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the LB for which you want to edit backend set health check parameters.
4. Under **Resources**, select **Backend Sets**.
5. For the backend set that you want to modify, select the Actions menu (⋮), and select **Edit**.
6. Make changes.

7. Select **Update Backend Set**.

Using the CLI

Use the oci lb health-checker update command and required parameters to change load balancer (LB) and backend server set health check properties, such as the health check interval.

Procedure

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
   ```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

This task is not available in the API.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Network Load Balancers

On Compute Cloud@Customer, you can configure the Network Load Balancing (NLB) feature to automatically distribute network traffic.

Network Load Balancer (NLB) 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.

NLB Essentials

When working with network load balancers (NLBs), understanding the building blocks and related key terminology is important.

Backend Server

An NLB has a set of backend servers to route incoming traffic to destinations while balancing the load placed on each individual server in the set.

A backend server is an application server responsible for generating content in reply to the incoming client traffic. You typically identify application servers with a unique combination of overlay (private) IPv4 address and port number. For example, `10.10.10.1:8080` and `10.10.10.2:8080`.

> **Note:**
>
> The backend server can't function as both a client and a backend simultaneously as is unable to be a traffic source to the NLB virtual IP (VIP).

Backend Set

The backend set is a logical entity that includes:

- A list of backend servers (compute instances) that accept traffic from the NLB.
- A network-level load balancing policy to distribute traffic to the backend servers in a consistent manner.
- A health check policy to decide if the backend servers are functioning and can accept traffic from the NLBs.

Network Load Balancing Policy

A network load balancing policy tells the NLB how to distribute incoming traffic to backend servers in the backend set.

A NLB uses an IP hash mapped from an *n-tuple*, which is an ordered and finite list of elements. These elements are typically drawn from the fields in a packet header or from items from the grid of a relational database. The *n* indicates how many fields the tuple contains.

Allowed NLB policies for the IP hash include:

- A hash on 5 fields in the TCP/IP header (called "5-Tuple Hash")
- A hash on 3 fields in the TCP/IP header (called "3-Tuple Hash")
- A hash on 2 fields in the TCP/IP header (called "2-Tuple Hash"

> **Note:**
>
> NLB policies are different than LBaaS policies. NLB policies don't handle IP addresses and port numbers directly. Hashing is a way to manipulate these and other packet header fields to provide a number that the NLB can use without processing the entire packet header. The more tuples that are used for the load balancing policy, the more processing required for each load balanced packet, but the wider the range of values that can be used by the NLB to distribute traffic.

Health Check

A health check is a test to confirm the availability of backend servers. A health check can be a request or a connection try. Based on a time interval you specify, the NLB applies the health check policy to continuously monitor backend servers. If a server fails the health check, the NLB takes the server temporarily out of rotation. If the server later passes the health check, the NLB returns it to the rotation.

You configure a health check policy when you create a backend set. TCP-level health checks try to make a TCP connection with the backend servers and validate the response based on the connection status

The health check results update the health status of the NLB. The health status is an indicator that reports the general health of NLBs and their components.

Listener

A logical entity that checks for incoming traffic on the NLB IP address. You configure a listener's port number and various optional settings. When creating a listener, the VCN security rules must let the listener accept traffic.

Work Request

An object that reports on the current state of an NLB request. Some NLB configuration tasks can take much longer to complete than others. Work requests let NLBs handle configuration steps asynchronously and

form a queue. This lets the user issue more configuration commands without waiting for the completion of a previous configuration directive.

NLB Frontend Configuration

On Compute Cloud@Customer, in the context of network load balancers (NLB), the term *frontend* refers to the components that a client can see and send requests to. The entry point of client requests is the outward-facing floating IP address of the NLB.

Three main configuration tasks are needed for the NLB frontend:

- Type (public or private)
- Policy (the number of tuples used for the hash value)
- Listeners (Port number and other parameters that the NLB uses when waiting for traffic)

See Managing Network Load Balancers on page 265.

NLB Types

On Compute Cloud@Customer, there are two types of network load balancers (NLBs): *public* and *private*.

Private NLBs use a private IP address to serve as the entry point for incoming traffic. The load balancing service requires only one subnet to host both the primary and secondary load balancers. However, the NLB is only accessible from within the VCN that contains the host subnet, or as further restricted by security rules.

Public NLBs use a public IP address and accept traffic from a network location outside of Compute Cloud@Customer. However, the public NLB must be deployed in a public subnet of a VCN that has an internet gateway (IGW) correctly configured.

NLB Policies

On Compute Cloud@Customer, a network load balancer (NLB) policy acts on an *n-tuple* mapped to an IP hash instead of directly on the packet header. A tuple is an ordered and finite list of elements. For an NLB, these elements to be hashed are drawn from the fields in a packet header. The *n* indicates how many fields the tuple contains.

Hashing generates a smaller number of bits than used in the concatenated fields before the hashing algorithm is applied. Packets that have the same hash value are treated the same way by the NLB. The more tuples that are used for the load balancing policy, the more processing required for each load balanced packet, but the wider the range of values that can be used by the NLB to distribute traffic.

For compatibility with OCI, these NLB policies can be selected:

- 5-Tuple Hash: a default policy based on a hash on 5 fields in the TCP/IP header (source IP, source port, destination IP, destination port, protocol)
- 3-Tuple Hash: a default policy based on a hash on 3 fields in the TCP/IP header (source IP, destination IP, protocol)
- 2-Tuple Hash: a default policy based on a hash on 2 fields in the TCP/IP header (source IP, destination IP)

However, due to implementation differences, any selected NLB policy in Compute Cloud@Customer is always mapped internally to an IP hash based only on the source IP address of an incoming request. The NLB 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.

NLB Listeners

On Compute Cloud@Customer, a listener is a logical entity at the ingress side of the network load balancer (NLB) setup. This key component detects incoming traffic on the NLB IP address. It listens for requests from clients using a particular protocol and port. Requests are then routed to the appropriate backend servers based on the rules defined in the NLB configuration. You must configure a listener.

When you create a listener, ensure that the VCN security rules let the listener accept traffic.

> **Note:**
>
> To accommodate high-volume traffic, we recommend that you use *stateless* security rules for NLB subnets. For more information, see Virtual Firewall on page 147.

See Managing NLB Listeners on page 277.

NLB Backend Configuration

On Compute Cloud@Customer, in the context of network load balancers (NLBs), the term *backend* refers to the components that receive, process, and respond to forwarded client requests. Backend servers are grouped into backend sets; they receive client requests based on the configured load balancing policy. Health checks ensure that traffic only goes to healthy backend servers.

NLB Backend Sets

On Compute Cloud@Customer, the backend set in a network load balancer (NLB) configuration consists of a list of backend servers, an NLB policy, and a health check script. Backend sets help to group backend servers together and make them easier to configure and manage form an NLB perspective.

A backend set must be associated with one or more listeners.

See Managing NLB Backend Sets on page 269.

NLB Backend Servers

On Compute Cloud@Customer, when creating a network load balancer (NLB), you must specify the backend servers to include in each backend set.

Backend servers can be set up as individual compute instances or as instance pools. You can add and remove backend servers without disrupting traffic.

TCP is the transport protocol of a backend server and is configured as part of the backend set.

When you add backend servers to a backend set, you specify either the instance OCID or an IP address for the server to add. An instance with multiple VNICs attached can have multiple IP addresses pointing to it. If you identify a backend server by OCID, the NLB uses the primary VNIC primary private IP address. If you identify the backend servers to add to a backend set by their IP addresses, it is possible to point to the same instance more than once.

The NLB routes incoming traffic to the backend servers based on the configured load balancing policy. To route traffic to a backend server, the NLB requires the IP address of the compute instance and the relevant application port. If the backend server resides within the same VCN as the NLB, we recommend that you specify the compute instance's private IP address. The private IP address also works if a local peering gateway enables traffic between the NLB VCN and the backend server VCN. If the backend server and NLB reside in different VCNs without a peering connection, you must specify the public IP address of the compute instance. You must also ensure that the VCN security rules allow external traffic.

To enable backend traffic, your backend server subnets must have appropriate ingress and egress security rules. When you add backend servers to a backend set, you can specify the applicable network security groups (NSGs). If you prefer to use security lists for your VCN, you can configure them through the Networking service.

See Managing NLB Backend Servers on page 273.

Managing Network Load Balancers

On Compute Cloud@Customer, you can create, view, edit and delete Network Load Balancers (NLBs).

For conceptual information about NLBs, see Network Load Balancers on page 262.

When you create an NLB, you can either configure all the resources at the same time or create a minimal NLB and supply other configuration details later. More than the basic NLB is needed to establish the NLB service. You add more components by editing the NLB resources.

Many configuration steps are processed as work requests. Work requests are tasks that can take some time to complete and therefore are tracked as they're processed. For example, the act of creating an NLB or cipher suite is a work request that records and displays start and finish times, state (succeeded, failed, and so on), and other relevant details. Work requests are part of the NLB resource list.

The status of a work request, such as succeeded, isn't the same as completion of the provisioning of the resource. NLB creation can succeed while provisioning is still in progress.

Creating a Network Load Balancer

On Compute Cloud@Customer, you can create a private Network Load Balancer (NLB) using the Console, CLI, or API.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Network Load Balancers**.
2. Select Create **Load Balancer**.
3. Enter the following information:
   - **Name:** Enter a descriptive name for the NLB. Avoid entering confidential information.
   - **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 don't select a public IP from the list, a public IP is automatically assigned from the configured public IP range.

     For more information about private and public load balancers see NLB Types on page 264.
   - **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. Select the Add Another NSG button to attach to another NSG.
     c. To remove an NSG from the list, select the trash can to the right of that NSG. To remove the last NSG or all NSGs, clear the Enable Network Security Groups box.
   - **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
4. Select **Create Network Load Balancer**. The details page of the new NLB is displayed.

What's Next?

On the NLB details page, under Resources, select resources to create to complete the configuration.

Using the CLI

Use the oci nlb network-load-balancer create command and required parameters to create a new private NLB in the specified compartment.

```
oci nlb network-load-balancer create --compartment-id <compartment_OCID>
 --display-name <name-of-network-load-balancer> --is-private true --subnet-
id <subnet_OCID>
```

Procedure

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 Balancers on page 209 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.

What's Next?

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`.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateNetworkLoadBalancer operation to create a new private NLB in the specified compartment.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing NLB Details

On Compute Cloud@Customer, you can view a list of existing network load balancers (NLBs) and view their details.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Network Load Balancers**.
2. At the top of the page, select the compartment that contains the NLB.
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 CLI

Use the following commands to view NLBs:

- Use oci nlb network-load-balancer list to see a list of NLBs.

```
oci nlb network-load-balancer list [OPTIONS]
```

- Use oci nlb network-load-balancer get to see the details of a particular NLB.

```
oci nlb network-load-balancer get --network-load-balancer-
id networkloadbalancer_OCID  [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the following operations to view NLBs:

- Use ListNetworkLoadBalancers to list all NLBs.
- Use GetNetworkLoadBalancer to get the details of a particular NLB.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Editing an NLB

On Compute Cloud@Customer, you can change the network load balancer (NLB) name and tags.

> **Note:**
>
> To add or update related resources such as backend sets or listeners, go to the NLB details page, under Resources, select the resource that you want to add or edit.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Network Load Balancers**.
2. At the top of the page, select the compartment that contains the NLB.
3. For the NLB that you want to edit, select the Actions menu (⋮), then select **Edit**.
4. Make your changes and select **Update Network Load Balancer** to update the NLB properties.
5. Select **Update Load Balancer**.

Using the CLI

Use the oci nlb network-load-balancer update command and required parameters to update the NLB.

```
oci nlb network-load-balancer update --network-load-balancer-id
 ocid1.networkloadbalancer.unique_ID --display-name <new_nlb_name> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateNetworkLoadBalancer operation to update the NLB.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting an NLB

On Compute Cloud@Customer, you can delete a network load balancer (NLB) to remove it from service.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Network Load Balancers**.
2. At the top of the page, select the compartment that contains the NLB.
3. For the NLB that you want to delete, select the Actions menu (⋮), and select **Terminate**.
4. Confirm the operation when prompted.

Using the CLI

Use the oci nlb network-load-balancer delete command and required parameters to delete an NLB.

```
oci nlb network-load-balancer delete --force --network-load-balancer-
id networkloadbalancer_OCID [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteNetworkLoadBalancer operation to delete an NLB.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Managing NLB Backend Sets

On Compute Cloud@Customer, you can use backend sets to create logical entities consisting of a network load balancing (NLB) policy, health check policy, and a list of backend servers for a Network Load Balancer resource.

Creating an NLB Backend Set

On Compute Cloud@Customer, you can create a backend set for an existing network load balancer (NLB). 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.

> **Note:**
>
> The Compute Cloud@Customer implementation of network load balancers is different from OCI.
>
> - The NLB policy is always mapped to an IP hash based on the source IP address of the incoming request. For OCI compatibility, you must select one of the available hashes (5-, 3-, or 2-tuple).
> - The source IP of the packet is not preserved. The `--is-preserve-source` parameter for backend sets is always disabled. Therefore, the `--is-preserve-source-destination` parameter cannot be enabled for the NLB.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Network Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the Network Load Balancer for which you want to create the network load balancer backend set.
4. Under **Resources**, select **Backend Sets**.

   Any existing backend sets are listed, otherwise the list says **No data available**
5. Select **Create Backend Set**.
6. Enter the following information:

   - **Name:** Specify a friendly name for the backend set. It must be unique within the network load balancer, and can't be changed. Valid backend set names include only alphanumeric characters, dashes, and underscores. Backend set names can't 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.

     Internally, any policy you select is mapped to IP Hash. For OCI compatibility, select one of these policies: following load balancing policies: 5-tuple hash, 3-tuple hash, 2-tuple hash.
   - **Source Header Preservation:** The default value *False* can't be changed.
   - **IP Protocol Version:** 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 often 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 try 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.
7. Select **Create Backend Set**.

   To display the details of the new backend set, view the backend set.

Using the CLI

Use the oci nlb backend-set create command and required parameters to add a backend set to a network load balancer.

```
oci nlb backend-set create [OPTIONS]
```

Procedure

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.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateBackendSet operation to add a backend set to a network load balancer.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing an NLB Backend Set

On Compute Cloud@Customer, you can view a list of the backends in an existing network load balancer backend set and view their details.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Network Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the network load balancer (NLB) to which you want to list existing backends.
4. Under **Resources**, select the name of the existing Backend Set to view its details, such as IP address, port, and weight.

Using the CLI

Use the oci nlb backend-set list command and required parameters to list all backend sets associated with a network load balancer.

```
oci nlb backend-set list [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the GetBackendSet operation to list details for a backend set.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Editing an NLB Backend Set

On Compute Cloud@Customer, you can change some properties of a network load balancer (NLB) backend that's a member of a backend set.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Network Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the network load balancer for which you want to edit backend properties.
4. Under **Resources**, select **Backend Sets**.
5. Select the name of the Backend Set with the backend that you want to edit.
6. Select the name of the NLB backend that you want to edit.
7. In the Actions menu (⋮), select **Edit**.
8. Make allowable changes in the dialog box.
9. Select **Update Network Load Balancer Backend**.

Using the CLI

Use the oci nlb backend-set update command and required parameters to update a backend set.

```
oci nlb backend-set update [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateBackendSet operation to update a backend set.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting an NLB Backend Set

On Compute Cloud@Customer, you can delete a Backend Set from a network load balancer.

Deleting a backend set removes its backend servers from the network load balancer.

Before you can delete a backend set, you must remove it from any active listeners.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Network Load Balancers**.
2. At the top of the page, select the compartment that contains the network load balancer.
3. Select the name of the network load balancer for which you want to delete the backend server.
4. Under **Resources**, select **Backend Sets**.
5. For the backend set you want to delete, select the Actions menu (⋮), and select **Delete**.
6. Confirm the operation when prompted.

Using the CLI

Use the oci nlb backend-set delete command and required parameters to delete the specified backend set from a network load balancer.

```
oci nlb backend-set delete [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteBackendSet operation to delete the specified backend set from a network load balancer.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Managing NLB Backend Servers

On Compute Cloud@Customer, manage the backend servers that receive incoming traffic based on the policies you specified for the backend set that contains it for the network load balancer (NLB).

When you create an NLB, you must specify the backend servers (compute instances) to include in each backend set. The NLB routes incoming traffic to these backend servers based on the policies you specified for the backend set.

For more conceptual information, see NLB Backend Servers on page 265.

Creating an NLB Backend Server

On Compute Cloud@Customer, you can create backend servers *after* you create the backend set, or at the same time. This topic creates only the backend server.

> **Note:**
>
> The Compute Cloud@Customer implementation of network load balancers is different from OCI. Backend servers cannot be set as backup units, cannot be drained, and cannot be marked as offline. These optional backend create and update parameters are not supported: `--is-backup`, `--is-drain`, `--is-offline`.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Network Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the Network Load Balancer for which you want to create the network load balancer backend server.
4. Under **Resources**, select **Backend sets**.

   Any existing backend servers are listed, otherwise the list displays **No data available**.
5. Select the hyperlink of the BackendSet to add the backend.
6. Select Create **Backend**.
7. Select the method for adding the backend server. Possible values are:
   - **Computed Instances** Backend servers are added by instance.
   - **IP Addresses** Backend servers are added by instance IP address.
8. For **Computed Instances**, enter the following information:
   - **Instance:** Enter the name of the backend server.
   - **Port:** Enter 22 (TCP).
   - **Name:** Leave blank to take the default (ip_address: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.
9. For **IP Addresses**, enter the following information:
   - **IP Address:** Enter the IP address of the backend server.
   - **Port:** Enter the port number.
   - **Name:** Enter the name of the backend.

- **Weight:** Enter a weight in the range 1 to 100.

10. Select **Submit**.

Using the CLI

Use the oci nlb backend create command and required parameters to add a backend server to a backend set.

```
oci nlb backend create [OPTIONS]
```

Procedure

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
      },
    ]
  }
}
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateBackend operation to add a backend server to a backend set.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing an NLB Backend Server

On Compute Cloud@Customer, you can view a list of the backend servers in an existing network load balancer backend set and view their details.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Network Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the network load balancer (NLB) to which you want to list existing backend servers.
4. Under **Resources**, select **Backend sets**.
5. Select the name of the backend set.
6. Under **Resources**, select **Backends**.

Using the CLI

Use the oci nlb backend list command and required parameters to list the backend servers for a particular network load balancer and backend set.

```
oci nlb backend list [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the GetBackend operation to list the backend servers for a particular network load balancer and backend set.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Editing an NLB Backend Server

On Compute Cloud@Customer, you can change some properties of a network load balancer (NLB) backend server.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Network Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the network load balancer for which you want to edit backend properties.
4. Under **Resources**, select **Backend Sets**.
5. Select the name of the backend set with the backend server that you want to edit.
6. For the backend server you want to edit, select the Actions menu (⋮), then select **Edit**.
7. Make allowable changes in the dialog box.
8. Select **Update Network Load Balancer Backend**.

Using the CLI

Use the oci nlb backend update command and required parameters to update the configuration of a backend server within the specified backend set.

```
SYNTX [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateBackend operation to update the configuration of a backend server within the specified backend set.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting an NLB Backend Server

On Compute Cloud@Customer, you can delete a network load balancer (NLB) Backend server from a Backend Set.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Network Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the network load balancer for which you want to delete the backend server.
4. Under **Resources**, select **Backend Sets**.
5. Select the name of the backend set that contains the backend server you want to delete.
6. For the backend server you want to delete, select the Actions menu (⋮), then select **Delete**.
7. Confirm the operation when prompted.

Using the CLI

Use the oci nlb backend delete command and required parameters to remove a backend server from a particular network load balancer and backend set.

```
oci nlb backend delete [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteBackend operation to remove a backend server from a particular network load balancer and backend set.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Managing NLB Listeners

On Compute Cloud@Customer, use listeners to check for incoming traffic on the network load balancer IP address.

Creating am NLB Listener

On Compute Cloud@Customer, you can create a listener for an existing network load balancer (NLB).

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 security rules allow the listener to accept traffic.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Network Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the name of the Network Load Balancer for which you want to create the listener.
4. On the network load balancer's details page, under **Resources**, select **Listeners**.
5. Select **Create Listener**.
6. Enter the following information:
   - **Name:** Enter a descriptive name for the Listener. Avoid entering confidential information.
   - **Protocol:** Select TCP, UDP, or TCP_AND_UDP 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 drop-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.
7. Select **Create Listener**.

   To display the details of the listener, you view the details for the network load balancer. See Viewing NLB Details on page 268.

Using the CLI

Use the oci nlb listener create command and required parameters to add a listener to a network load balancer.

```
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  [OPTIONS]
```

Procedure

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: TCP, UDP, or TCP_AND_UDP.

For information about optional parameters, use `oci nlb listener create -h`.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateListener operation to add a listener to a network load balancer.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Editing an NLB Listener

On Compute Cloud@Customer, you can change some network load balancer (NLB) listener properties.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Network Load Balancers**.
2. At the top of the page, select the compartment that contains the load balancer.
3. Select the Network Load Balancer for which you want to edit listeners.
4. Under **Resources**, select **Listeners**.
5. Select the name of the NLB listener that you want to edit.
6. For the NLB listener you want to edit, select the Actions menu (⋮), then select **Edit**.
7. Make allowable changes to the listener in the dialog box.
8. Select **Update Network Load Balancer Listener**.

Using the CLI

Use the oci nlb listener update command and required parameters to update a listener for a particular network load balancer.

```
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 [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateListener operation to update a listener for a particular network load balancer.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting an NLB Listener

On Compute Cloud@Customer, you can delete a network load balancer (NLB) Listener and remove it from service.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Network Load Balancers**.
2. At the top of the page, select the compartment that contains the NLB.
3. Select the name of the network load balancer (NLB) or which you want to list existing listeners.
4. Under **Resources**, select **Listeners**.
5. Select the name of the NLB listener you want to delete.

6. On the row with the NLB listener you want to delete, select the Actions menu (⋮), then select **Delete**.
7. Confirm the operation when prompted.

Using the CLI

Use the oci nlb listener delete command and required parameters to delete the NLB listener.

```
oci nlb listener delete [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteListener operation to delete the NLB listener.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Checking NLB Health

On Compute Cloud@Customer, use health status indicators to check the health of your network load balancers (NLBs) and their resources.

You can see health status indicators and summaries in the NLBs, backend sets, and backend servers.

On Compute Cloud@Customer, Network Load Balancer (NLB) health checks are tests to confirm the availability of backend servers. These tests occur in the form of an attempt to make a TCP connection with the backend servers and validate the response based on the connection status.

The health status of the specified backend set server is reported by the primary and standby NLB.

The health check policy includes a time interval you specify, to ensure that the backend servers are continuously monitored. If a server fails a health check, the NLB takes the server temporarily out of rotation. If the server later passes the health check, the NLB returns it to the rotation.

The health check policy is configured when you create a backend set.

The TCP handshake can succeed and indicate that the service is up even when the HTTP service is incorrectly configured or having other issues. Although the health check returns no errors, you might experience transaction failures.

The service provides application-specific health check capabilities to help you increase availability and reduce your application maintenance window.

Health status indicators are used to report the general health of an NLB and its backend servers and sets. The possible statuses are:

- `ok` – No attention required. The resource is functioning as expected.
- `warning` – Some reporting entities require attention. The resource is not functioning at peak efficiency or the resource is incomplete and requires further work.
- `critical` – Some or all reporting entities require immediate attention. The resource isn't functioning or unexpected failure is imminent.
- `unknown` – Health status cannot be determined. The resource isn't responding or is in transition and might resolve to another status over time.

Health status is updated every three minutes. No finer granularity is available. Historical health data isn't provided.

Viewing NLB Health Checker Summaries

On Compute Cloud@Customer, you can view health summaries for all the network load balancers in a compartment.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Network Load Balancers**.
2. At the top of the page, select the compartment that contains the NLB.

3. Select the name of the NLB for which you want to view health checker status.

Using the CLI

Use the oci nlb network-load-balancer-health list command and required parameters to retrieve the health status for all NLBs.

```
oci nlb network-load-balancer-health list [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListNetworkLoadBalancerHealths operation to retrieve the health status for all NLBs.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing NLB Health Checker Status Details

On Compute Cloud@Customer, you can view the health status details of a network load balancer (NLB).

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Network Load Balancers**.
2. At the top of the page, select the compartment that contains the NLB.
3. Select the name of the NLB for which you want to view existing load balancer backend set health parameters.
4. Under **Resources**, select **Backend Sets**.
5. Select the name of the Backend Set to view its health check details.

Using the CLI

Use the oci nlb health-checker get command and required parameters to see the health status details for the specified network load balancer.

```
oci nlb network-load-balancer-health get [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the GetNetworkLoadBalancerHealth operation to get the health status details for the specified network load balancer.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing an NLB Health Checker Policy

On Compute Cloud@Customer, you can view the health checker policy parameters used by a network load balancer (NLB) to check backend set health.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Network Load Balancers**.
2. At the top of the page, select the compartment that contains the NLB.
3. Select the name of the NLB for which you want to view heath check policy.
4. Under **Resources**, select **Backend Sets**.
5. Select the name of the backend set.
6. Under the backend-set name for which to you want to view the health check policy, select **Backend Set Configuration**.

Using the CLI

Use the oci nlb health-checker get command and required parameters to see the health check policy information for a particular network load balancer and backend set.

```
oci nlb health-checker get --backend-set-name backend-set-name-text \
 --network-load-balancer-id network-load-balancer_OCID [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the GetHealthChecker operation to get the health check policy information for a particular network load balancer and backend set.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Editing NLB Health Check Parameters

On Compute Cloud@Customer, you can update the health check policy, such as the health check interval, for a network load balancer and backend set.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Network Load Balancers**.
2. At the top of the page, select the compartment that contains the NLB.
3. Select the name of the NLB for which you want to edit backend set health check parameters.
4. Under **Resources**, select **Backend Sets**.
5. Select the name of the backend set to view the details page.
6. Select **Edit**.
7. Make changes.
8. Select **Save**.

Using the CLI

Use the oci nlb health-checker update command and required parameters to update the health check policy for a particular network load balancer and backend set.

```
oci nlb health-checker update [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateHealthChecker operation to update the health check policy for a particular network load balancer and backend set.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing the Health of an NLB Backend Set

On Compute Cloud@Customer, you can view the network load balancer (NLB) backend set health status.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Network Load Balancers**.
2. At the top of the page, select the compartment that contains the NLB.
3. Select the name of the NLB for which you want to view backend set health status.
4. Under **Resources**, select **Backend Sets**.
5. Select the Backend Set to view the backend set health status.

Using the CLI

Use the oci nlb backend-set-health get command and required parameters to see the health status for the specified backend set.

```
oci nlb backend-set-health get [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the GetBackendSetHealth operation to get the health status for the specified backend set.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing the Health of an NLB Backend Server

On Compute Cloud@Customer, you can view the network load balancer (NLB) backend server health status.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Network Load Balancers**.
2. At the top of the page, select the compartment that contains the NLB.
3. Select the name of the NLB for which you want to view backend set health status.
4. Under **Resources**, select **Backend Sets**.
5. Select the Backend Set to view the backend health stat.

Using the CLI

Use the oci nlb backend-health get command and required parameters to see the health status for the specified backend set.

```
oci nlb backend-set-health get [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the GetBackendHealth operation to get the health status for the specified backend set.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing NLB Work Request Errors

On Compute Cloud@Customer, you can view NLB work requests to see if and NLB work request encountered errors.

Many of the configuration steps used to create and configure network load balancer (NLB) operation don't 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's not always obvious that a configuration step has failed with an error. The failed step is often not revealed until the next step.

You can view NLB work request status indicators to find out if a work request has failed with an error. You can check the progress of each operation, whether it resulted in a failed state, which step it failed on, and the reason for the failure.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Network Load Balancers**.
2. At the top of the page, select the compartment that contains the NLB.
3. Select the name of the NLB for which you want to view work requests.

4. Under **Resources**, select **Work Requests**.
5. For each work request, you can view the following information:
   - Type of work request
   - State of the work request (Succeeded, Failed, and so on)
   - Start and finish timestamp
6. Select the name of a work request to see the following work request details:
   - General information about the work request, such as the type
   - OCID of the work request
   - Error detail of a failed work request (nothing is displayed for a nonfailure status)
   - Start and finish timestamp

Using the CLI

Use the oci nlb work-request-error list command and required parameters to see a paginated list of errors for a particular work request.

```
oci nlb work-request-error list [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListWorkRequestErrors operation to list errors for a particular work request.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Images for Compute Cloud@Customer Instances

On Compute Cloud@Customer, an image is a template of a virtual hard drive. The image provides the OS and other software for a compute instance. You specify an image to use when you create a compute instance.

These are the types of images you can use to create a compute instance:

- **Platform images:** Compute Cloud@Customer includes Oracle Linux and Oracle Solaris images. These included images are available in every compartment of every tenancy. These images don't need to be downloaded or imported to access them to create an instance. See Compute Cloud@Customer Platform Images on page 284.
- **Custom images:** You can create a custom image of a compute instance's boot disk and use it to create other compute instances. Instances you create from your image include the customizations, configuration, and software installed when you created the image. See Managing Custom Images on page 285.
- **Marketplace images:** You can request access to a variety of Marketplace images that are tested and approved for use on Compute Cloud@Customer. For more information, see Marketplace Images on page 292.
- **Bring your own image (BYOI):** You can bring your own versions of images to the cloud as long as the underlying hardware supports it. See Bring Your Own Image (BYOI) on page 294.

For a list of Oracle tested OSs, see Guest Operating System Matrix on page 33.

Listing Images and Viewing Details

On Compute Cloud@Customer, in both the Console and CLI, Oracle provided images are listed first, followed by custom images.

The list of Oracle provided 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 Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Images**.

2. At the top of the page, select the compartment that contains the images you want to see.
3. To see details of an image, select the name of the image in the list.

Using the CLI

Use the oci compute image list command and required parameters to list the images in the specified compartment.

```
oci compute image list [OPTIONS]
```

Use the oci compute image get command and required parameters to get the image details for just one image.

```
oci compute image get [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListImages and GetImage operations to list all images or get one image.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Compute Cloud@Customer Platform Images

You can use the platform images that are included with Compute Cloud@Customer to create instances. You can select from Oracle Linux and Oracle Solaris images.

> **Note:**
>
> The platform images provided with Compute Cloud@Customer have been specifically configured for use On Compute Cloud@Customer only. The following restrictions apply:
>
> - Compute Cloud@Customer images can only be used to create instances On Compute Cloud@Customer.
> - Oracle Cloud Infrastructure Platform images can't be used to create instances On Compute Cloud@Customer.

> **Note:**
>
> Only the three most recently published versions of each major distribution are listed in the Console and by the `compute image list` command. After an upgrade, older versions might no longer be listed, but they're still accessible. To use an older image version, use the 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's using the image.

To see the Compute Cloud@Customer platform images, see Listing Images and Viewing Details on page 283.

To create an instance using an image provided with Compute Cloud@Customer, see:

- Tutorial: Launching Your First Instance on page 307
- Creating an Instance on page 316

Initial User Account

After you create an instance from a platform image, you initially connect to the instance using `ssh` with the user account `opc`. The `opc` user has `sudo` privileges.

The SSH connection is authenticated using your SSH key pair that's used during instance creation. For more information, see Connecting to an Instance.

Remote Access

Access to the instance is permitted only over SSH v2 protocol. All other remote access services are disabled.

Image OS Administration

After an image is used to create an instance, the instance OS administration is managed according to the OS type and version. For OSs provided in the platform images, refer to the Oracle Linux and Oracle Solaris documentation available on Oracle Help Center's Operating Environments page.

Managing Custom Images

On Compute Cloud@Customer, you can create, edit, move, import, export, and delete custom images.

For a list of Oracle-tested OSs for Compute Cloud@Customer, see Guest Operating System Matrix on page 33.

Creating a Custom Image from an Instance

On Compute Cloud@Customer, you can create a custom image of a compute instance's boot disk and use that custom image to create other compute instances. Instances that you create from this image include the customizations, configuration, and software that were installed on the boot disk when you created the image.

Custom images don't include the data from any attached block volumes.

After the new custom image reaches the Available state, you can use it to create new instances. See Creating an Instance on page 316.

Limitations and Considerations

- Certain IP addresses are reserved for Compute Cloud@Customer use and can not be used in your address numbering scheme.
- When you create an image of an instance, the instance must be in the stopped state. After the custom image is created, you can restart the instance.
- You can't create extra custom images of a compute instance while the compute instance is engaged in the image creation process. You can, however, create images of different compute instances at the same time.
- Custom images are available to all users authorized for the compartment in which the image was created.
- Custom images inherit the compatible shapes that are set by default from the base image.

Avoid entering confidential information in names.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. At the top of the page, select the compartment that contains the instance you plan to use.
3. Select 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:
   a. On the details page for the instance, select **Controls** (upper right corner), and then select **Stop**.
   b. Wait for the instance status to change to Stopped. The status is displayed above the icon of the object.
5. Select **Controls** and then select **Create Custom Image**.
6. In the **Create Image From Instance** dialog box, 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. Select **Create Custom Image**.

   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, select **Compute** and then select **Custom Images**. Select the correct compartment, and select the image name in the list. On the details page for the image, under **Resources** select **Work Requests**.

9. (Optional) Restart the instance.

   When the instance status changes from Creating Image to Stopped, you can restart the instance.

Using the CLI

Use the oci compute image create command and required parameters to create an image from an instance.

```
oci compute image create [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateImage operation to create an image from an instance.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Updating an Image Name

On Compute Cloud@Customer, you change a custom image's display name.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Images**.
2. At the top of the page, select the compartment that contains the image you want to edit.
3. Use one of the following methods to open the **Update Image** dialog box.
   - For the image that you want to update, select the Actions menu (⋮), and select **Edit**.
   - Select the name of the image that you want to update. On the image details page, select **Controls** (upper right corner), then select **Edit Details**.
4. In the **Update Image** dialog box, change the image name.

   The image name doesn't need to be unique.

   Avoid entering confidential information
5. Select **Save Changes**.

Using the CLI

Use the oci compute image update command and required parameters to change an image name.

```
oci compute image update [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateImage operation to update the specified image.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Moving an Image to a Different Compartment

On Compute Cloud@Customer, you can move an image to another compartment using the CLI or API.

Custom images are available to all users authorized for the compartment in which the image was created. Ensure that you move the image to a compartment that has the appropriate authorized users.

Using the Console

This task can't be performed using the Console.

Using the CLI

Use the oci compute image change-compartment command and required parameters to move an image to another compartment.

```
oci compute image change-compartment [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ChangeImageCompartment operation to move an image to another compartment.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Uploading an Image to an Object Storage Bucket

On Compute Cloud@Customer, you can store images in an object storage bucket.

Advantages of storing an image in an Object Storage bucket are that you can implement object versioning or preauthenticated requests as described in Managing Object Versioning on page 492 and Using Preauthenticated Requests on page 495.

1. Create an Object Storage bucket as described in Creating a Bucket on page 479.
2. Upload an image from a local file system to the bucket. See Uploading an Object on page 485.
3. Import the image from the Object Storage bucket to a compartment so that the image is available to select when you create an instance. See Importing an Image from an Object Storage Bucket on page 287.

Importing an Image from an Object Storage Bucket

On Compute Cloud@Customer, you can import an image into a compartment from an Object Storage bucket.

> **Note:**
>
> Alternatively, you can import an image from a URL as described in Importing an Image from a URL on page 288.

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 on page 479.

If the bucket doesn't contain the image that you want to use, perform the procedure described in Uploading an Image to an Object Storage Bucket on page 287.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Custom Images**.
2. Select **Import Image**.
3. In the **Import Image** dialog box, enter the following information:
   - **Name:** Enter a descriptive name for the image.

     Avoid entering confidential information.
   - **Create in Compartment:** Select the compartment for the image.
   - **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're 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 can't be changed.
  - **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
4. Select **Import Image**.

   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 create instances as described in Creating an Instance on page 316.

Using the CLI

Use the oci compute image import from-object command and required parameters to import an image from an Object Storage bucket.

```
oci compute image import from-object --compartment-id compartment_OCID
 --bucket-name bucket_name --name bucket_image_object_name --
namespace namespace [OPTIONS]
```

> **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, ensure the value doesn't contain the case-insensitive string "Windows".

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

This operation can't be performed using the API.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Importing an Image from a URL

On Compute Cloud@Customer, 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 on page 287.

Before You Begin

Get the URL that you need for this procedure. Ensure that the URL is accessible from your tenancy.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Custom Images**.
2. On the **Custom Images** page, select **Import Image**.
3. In the **Import Image** dialog box, enter the following information:
   - **Name:** Enter a descriptive name for the image.

     Avoid entering confidential information.
   - **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 can't be changed.
- **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.

4. Select **Import Image**.

   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 create instances as described in Creating an Instance on page 316.

Using the CLI

Use the oci compute image import from-object-uri command and required parameters to import an image from a URL.

```
oci compute image import from-object-uri --compartment-id compartment_OCID
 --uri URL_for_image[OPTIONS]
```

> **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, ensure the value doesn't contain the case-insensitive string "Windows".

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

This operation can't be performed using the API.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Exporting an Image to an Object Storage Bucket

On Compute Cloud@Customer, you can export an image to an Object Storage bucket.

Exported images are a copy of the boot volume and metadata when the image was created.

> **Note:**
>
> As an alternative, you can export an image to an Object Storage URL as described in Exporting an Image to an Object Storage URL on page 290.

Prerequisites

- Ensure that a bucket is available. See Listing Buckets on page 480 and Creating a Bucket on page 479.
- You need write access to the to the export location.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Custom Images**.
2. At the top of the page, select the compartment that contains the image you want to export.
3. Select the name of the custom image that you want to export.
4. On the image details page, select **Controls** (upper right corner), and then select **Export Image**.
5. In the **Export Custom Image** dialog box, enter the following information:
   - **Export Destination:** Select the **Export to an Object Storage Bucket** option.
   - **Bucket:** Select a bucket. If needed, 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're 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`).
6. Select **Create Export**.

   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 create instances while the image is exporting, but you can't 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 don't 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 CLI

Use the oci compute image export to-object command and required parameters to export the specified image to an Object Storage bucket.

```
oci compute image export to-object --bucket-name bucketname --image-
id image_OCID --namespace namespace --name exported_image_object_name --
export-format VMDK [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ExportImage operation to export the specified image to an Object Storage bucket.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Exporting an Image to an Object Storage URL

On Compute Cloud@Customer, you can export an image to an Object Storage URL.

Exported images are a copy of the boot volume and metadata when the image was created.

> **Note:**
>
> As an alternative, you can export an image to a URL as described in Exporting an Image to an Object Storage Bucket on page 289.

Prerequisite

- You need write access to the to the export location.

Using the Console

1. In the cnvCommonPhrases navigation menu, select **Compute**, then select **Custom Images**.

2. At the top of the page, select the compartment that contains the image you want to export.
3. Select the name of the custom image that you want to export.
4. On the image details page, select **Controls** (upper right corner), and then select **Export Image**.
5. In the **Export Custom Image** dialog box, 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're 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`).
6. Select **Export Image**.

   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 create instances while the image is exporting, but you can't 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 don't 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 CLI

Use the oci compute image export to-object-uri command and required parameters to export the specified image to the Object Storage service using the Object Storage URL to identify the location to export to.

```
oci compute image export to-object-uri --image-id <image_OCID> --
uri <url_to_export_to> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ExportImage operation to export the specified image to the Object Storage service using the Object Storage URL to identify the location to export to.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Sharing Custom Images Across Tenancies

On Compute Cloud@Customer, 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 on page 289.
2. Create a preauthenticated request with read-only access for the image in the bucket. See Using Preauthenticated Requests on page 495.
3. In the destination tenancy, import the image. Use the preauthenticated request URL as the Object Storage URL. See Importing an Image from a URL on page 288.

Deleting an Image

On Compute Cloud@Customer, you can delete custom images.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Images**.
2. At the top of the page, select the compartment that contains the image you want to delete.
3. For the custom image that you want to delete, select Actions menu (⋮), then select **Delete image**.

   The image is deleted.

Using the CLI

Use the oci compute image delete command and required parameters to delete a custom image.

```
oci compute image delete [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteImage operation to delete a custom image.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Marketplace Images

Learn about Marketplace images that you can use on Compute Cloud@Customer.

The following Marketplace images are available for use on Compute Cloud@Customer.

Oracle Weblogic Server Enterprise Edition UCM

OS: Oracle Linux 8

Version: 24.3.2-ol8.8-23.12.13-240826-1

Oracle WebLogic Server provides the following features:

- A modern development platform for building applications
- A runtime platform for high performance and availability
- Management tooling for efficient and low-cost operations.

The Oracle WebLogic Server includes an implementation of the Java Platform, Enterprise Edition (Java EE) that provides a standard set of APIs for creating distributed Java applications that can access a wide variety of services, such as databases, messaging services, and connections to external enterprise systems.

Oracle WebLogic Server enables you to deploy mission-critical applications in a robust, secure, highly available, and scalable environment.

These features enable enterprises to configure clusters of Oracle WebLogic Server instances to distribute load and provide extra capacity in case of hardware or other failures. Security features protect access to services, keep enterprise data secure, and prevent malicious attacks.

For more information, see Oracle WebLogic Server for Oracle Cloud Infrastructure.

Oracle WebLogic Suite UCM

OS: Oracle Linux 8

Version: 24.3.2-ol8.8-23.12.13-240826-1

The Oracle WebLogic Suite for OCI UCM image provides binaries of Oracle WebLogic Server and the Oracle JDK along with the entitlement to create Oracle WebLogic Server domains in Compute Cloud@Customer instances.

The image can also be used to create an Oracle Forms compute instance which installs everything needed to host most Oracle Forms applications. By default, an Oracle Database is also installed within this instance for storing product metadata. However, you can alternatively elect to connect to Oracle's Database Cloud Service (DBCS) or autonomous Database (ADB) at provisioning time instead, in which case the local database isn't installed.

Oracle WebLogic Suite includes all entitlements included in the Oracle WebLogic Suite license.

For more information, see Oracle WebLogic Suite UCM Image.

Oracle SOA Suite (PAID)

OS: Oracle Linux 8

Version: 25.1.3 - SOA Suite 12.2.1.4

Oracle SOA Suite is a Platform as a Service (PaaS) solution for running applications in the cloud. It includes a complete set of service infrastructure components for designing, deploying, and managing composite applications.

For more information, see Using Oracle SOA Suite on Marketplace in Oracle Cloud Infrastructure.

For licensing information, see About the Oracle SOA Suite on Marketplace License

Oracle Tuxedo Enterprise Image UCM

OS: Oracle Linux 8

Version: tuxedo-enterprise-22.1.0.0.0-ol8.6-22.10.04_101524

Oracle Tuxedo Enterprise Image provides a solid foundation for enterprise applications with the following comprehensive features:

- Distributed transaction processing with highly reliable transaction integrity
- Effective work load balancing/management
- Clustering and failover capabilities
- Robust fault management
- Data-dependent routing
- Rich, enterprise messaging capabilities
- Advanced and flexible security infrastructure
- Metadata driven application development
- Ease of diagnostic

With this image, you can quickly provision an instance for developing and deploying mission critical applications in many programming languages including C, C++, COBOL, Java and so on.

RackWare Migration Manager (RMM) UCM

OS: Oracle Linux 8

Version: 7.4.8.2

RackWare provides a complete Workload Mobility Platform that extends across physical and virtual environments.

RackWare's RMM Migrations are an automated, easy and convenient process to move existing workloads between different data centers and clouds, creating an exact duplicate of a running image without the burden of rebuilding or recreating template images and applications. Migrating large complex environments through a simple interface and automated tasks reduces the time required from weeks to days.

For more information, see RackWare RMM Migration Getting Started for Oracle Marketplace.

RackWare Kubernetes Solution - SWIFT (Migration-Subscription) UCM

OS: Oracle Linux 8

Version: 1.3.37.18

Move and Protect your Kubernetes/OpenShift deployments with RackWare SWIFT. Migrate Kubernetes/OpenShift applications from any environment to the OKE Cloud. Protect critical workloads with SWIFT's Converged Disaster Recovery and Backup solution.

For more information, see RackWare Kubernetes Solution - SWIFT (Migration-Subscription) marketplace description.

To use a Marketplace image, you must request the image from Oracle.

Requesting a Marketplace Image from Oracle

Open a support request to have Oracle load the images on your Compute Cloud@Customer infrastructure. See Creating a Support Request. To access support, sign in to the Oracle Cloud Console as described in Sign In to the OCI Console.

Using Marketplace Images

After the Marketplace image is available on Compute Cloud@Customer, you can create instances using the image. The first time a user creates an instance using a Marketplace image, the user is prompted to sign a user agreement. The first time you create an instance using a Marketplace image, you must use the Compute Cloud@Customer Console so that you can accept the user agreement. After that, you can use the Console, CLI, and API to create instances with a Marketplace image.

Marketplace Image Restrictions

The following restrictions apply to Marketplace images and to instances created with Marketplace images:

- Only federated users can accept the user agreement and create instances with the Marketplace image.
- Market place images can't be deleted or updated
- You can't create an image from an Marketplace instance's boot-volume.
- You can't export a Marketplace image to Object Storage.
- You can't create another image from a running instance created with a Market place image.
- If a Marketplace boot volume is attached as a data volume to another instance, then that instance export isn't allowed.

Permissible Marketplace Image Administration

You can perform the following actions on instances that were created from Marketplace images:

- You can attach the Marketplace instance boot volume as data volume to another instance.
- You can clone a Marketplace instance boot volume.
- You can restore a Marketplace instance boot volume backup to a boot volume.
- You can restore a Marketplace instance volume-group-backup that includes a Marketplace boot-volume.
- You can create Marketplace instance boot volume backups using backup policies.
- You can create volume group that has boot-volume of Marketplace instance.

Bring Your Own Image (BYOI)

On Compute Cloud@Customer, the Bring Your Own Image (BYOI) feature enables you to bring your own versions of OSs to the Compute Cloud@Customer service as long as the underlying hardware supports it.

The BYOI feature provides these benefits:

- Enables cloud migration projects.
- Supports both old and new OSs.
- Encourages experimentation.
- Increases flexibility.

> **Important:**
>
> You must comply with all licensing requirements when you upload and start instances based on OS images that you supply.

A critical part of any lift-and-shift cloud migration project is the migration of on-premises virtual machines (VMs) to the cloud. You can import your on-premises virtualized root volumes into Compute Cloud@Customer using the custom image import feature, and then create compute instances using those images.

For a list of Oracle-tested OSs for Compute Cloud@Customer, see Guest Operating System Matrix on page 33.

For information about creating custom images, see Managing Custom Images.

For information about importing images, see Importing Custom Linux Images on page 295 and Importing Custom Microsoft Windows Images on page 296.

For information about creating instances, see Working with Instances on page 316.

Importing Custom Linux Images

You can bring your own Linux image to Compute Cloud@Customer as long as the image meets specific requirements.

Linux Source Image Requirements

Custom images must meet the following requirements:

- The maximum image size is 400 GB.
- The image must be set up for BIOS boot.
- Only one disk is supported, and it must be the boot drive with a valid master boot record (MBR) and boot loader. You can migrate additional data volumes after you import the image's boot volume.
- The boot process must not require more data volumes to be present for a successful boot.
- The boot loader must use LVM or a UUID to locate the boot volume.
- The disk image can't be encrypted.
- The disk image must be a VMDK or QCOW2 file. These images can be converted to `.oci` type images.
  - Create the image file by cloning the source volume, not by creating a snapshot.
  - VMDK files must be either the "single growable" (monolithicSparse) type or the "stream optimized" (streamOptimized) type, both of which consist of a single VMDK file. All other VMDK formats, such as those that use multiple files, split volumes, or contain snapshots, are not supported.
- The network interface must use DHCP to discover the network settings. When you import a custom image, existing network interfaces are not re-created. Any existing network interfaces are replaced with a single NIC after the import process is complete. You can attach more VNICs after you create the imported instance.
- The network configuration must not hard code the MAC address for the network interface.
- Oracle recommends that you enable certificate-based SSH, however this recommendation is optional.

Preparing Linux VMs for Import

Before you import a custom Linux image, you must prepare the image to ensure that instances created from the image can boot correctly and that network connections will work.

Perform these steps.

1. Ensure that the source image meets the requirements.

   See Importing Custom Linux Images on page 295.
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 isn't available the first time an imported instance boots.
4. Ensure that all network interfaces use DHCP, and that the MAC address and IP addresses aren't 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 on page 289.
2. Import the image from the bucket to your tenancy.

   See Importing an Image from an Object Storage Bucket on page 287.
3. Complete the post import tasks.

   See Post Import Tasks for Linux Images on page 296.

Post Import Tasks for Linux Images

After you import a custom Linux image, perform these steps.

1. Use the imported image to create an instance.

   For the image source, select Custom Images, and then select the image that you imported. See Creating an Instance on page 316.
2. If the instance requires any remotely attached storage, such as block volumes, create and attach the storage.

   See Creating and Attaching Block Volumes on page 383.
3. Create and attach any required secondary VNICs.

   See Configuring VNICs on page 181.
4. Test that all applications are working as expected.
5. Reconfigure any services that were set to start manually.

Importing Custom Microsoft Windows Images

When you bring your own Microsoft Window image to Compute Cloud@Customer, it must meet specific requirements.

Microsoft Windows Source Image Requirements

- The maximum image size is 400 GB.
- The image must be set up for a BIOS boot.
- Only one disk is supported, and it must be the boot drive with a valid master boot record (MBR) and boot loader. You can migrate additional data volumes after you import the image's boot volume.
- The minimum boot volume size is 256 GB.
- The boot process must not require other data volumes to be present for a successful boot.
- The disk image can't be encrypted.
- The disk image must be a VMDK or QCOW2 file. Create the image file by cloning the source volume, not by creating a snapshot. VMDK files must be either the "single growable" (monolithicSparse) type or the "stream optimized" (streamOptimized) type, both of which consist of a single VMDK file. All other VMDK formats, such as those that use multiple files, split volumes, or contain snapshots, are not supported.
- The network interface must use DHCP to discover the network settings. When you import a custom image, existing network interfaces are not re-created. Any existing network interfaces are replaced with a single NIC after the import process is complete. You can attach additional VNICs after you launch the imported instance.
- The network configuration must not hard code the MAC address for the network interface.

Preparing Microsoft Windows Systems for Import

The configuration described in this section is required so that Compute instances that are created from the Microsoft Windows system image can boot correctly and network connections will work.

> **Important:**
>
> The system drive configuration where the Microsoft Windows source instance (virtual machine or physical 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 the instance is created from the image. Then manually move the data on the noninstance 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.

   See Importing Custom Microsoft Windows Images on page 296.
2. Clone the existing instance, using the procedures specific to the current platform. Then perform all changes on the clone to prevent disrupting the production instance.
3. 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 OS 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.
4. Configure Remote Desktop Protocol (RDP) access.
   a. Enable Remote Desktop connections to the image. See To connect to a Windows instance from a Remote Desktop Client.
   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.
5. 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.
6. If you plan to use the custom image to create 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.
7. Create a backup of the root volume.
8. If the Windows instance 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 Cloud Infrastructure.

9. 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.
10. Download the Oracle Windows VirtIO drivers:
    a. Sign in to the Oracle Software Delivery Cloud site.
    b. In the All Categories drop-down, select Release.
    c. Type Oracle Linux 7.9 in the search box and select Search.
    d. Select REL: Oracle Linux 7.9.0.0.0.
    e. At the top right of the page, to the right of your cart, select Continue.
    f. In the Platforms/Languages list, select x86 64 bit. Select Continue.
    g. Accept the license agreement and then select Continue.
    h. Select the checkbox next to Oracle VirtIO Drivers Version for Microsoft Windows 1.1.x. Clear the other check boxes.
    i. Select Download and then follow the prompts.
11. Install the Oracle VirtIO drivers for Windows:
    a. Follow the prompts in the installation workflow. On the Installation Type page, select Custom.
    b. Reboot the Windows instance.
12. Perform the Creating and Exporting an Image on page 298 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 OS.
3. Export the image from your physical system or virtualization environment.
4. Perform the Importing a Microsoft Windows Image on page 298 procedure to import the image into Oracle Cloud Infrastructure.

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 above.

   After importing the image, do *not* perform the Post Import Tasks for Microsoft Windows Images on page 299 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 on page 316.
4. Connect to the console as described in Connecting to a Compute Instance on page 341.
5. Perform the Preparing Microsoft Windows Systems for Import on page 297 procedure.
6. Perform the Post Import Tasks for Microsoft Windows Images on page 299 procedure.

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 on page 289.
2. Import the image from the bucket to your tenancy.

   See Importing an Image from an Object Storage Bucket on page 287 and Importing an Image from a URL on page 288. Use the CLI procedure and specify the `--operating-system` option. Ensure 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 on page 299.

Post Import Tasks for Microsoft Windows Images

After you import a custom Microsoft Windows image, perform these steps.

1. Use the imported image to create an instance.

   For the image source, select Custom Images, and then select the image that you imported. See Creating an Instance on page 316.
2. Enable Remote Desktop Protocol (RDP) access to the Compute instance.

   See To connect to a Windows instance from a Remote Desktop Client.
3. Connect to the instance using RDP.

   See To connect to a Windows instance from a Remote Desktop 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 on page 383.
5. Create and attach any required secondary VNICs.

   See Configuring VNICs on page 181.
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 configuration every time you create an instance using this custom image, consider creating a new image from the fully configured instance. See Creating a Custom Image from an Instance on page 285.

Compute Instances

You can provision and manage compute instances On Compute Cloud@Customer.

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.

When you create a compute instance, you can select the most appropriate type of compute instance for your applications based on characteristics such as the number of CPUs, amount of memory, and network resources. See Tutorial: Launching Your First Instance on page 307 and Working with Instances on page 316.

After you create a compute instance, you can access it securely from your computer, restart it, attach and detach volumes, and delete it when you're done with it.

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. See Working with Instance Configurations on page 333.

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. See Instance Pools on page 349.

You can create a pool of compute instances (nodes) in a Compute Cloud@Customer Kubernetes Engine (OKE) cluster. See Kubernetes Engine (OKE) on Compute Cloud@Customer on page 505.

You can connect to a compute instance. See Connecting to a Compute Instance on page 341.

You can back up an instance and restore the instance from backup. See Backing Up and Restoring an Instance on page 344.

Components for Creating Instances

These components are required to create a compute instance:

Compartment

A collection of related resources that are only accessible by certain groups that have been given permission by an administrator in your organization. Compute instances are created in compartments. All compartments exist in a tenancy, which is the root compartment.

Virtual Cloud Network (VCN)

A virtual version of a traditional network—including subnets, route tables, and gateways—on which your compute instance runs. At least one cloud network must be set up before you create compute instances.

SSH Key Pair

If the image that's used to create the instance is configured to require Secure Shell (SSH) for authentication, then you need an SSH key pair before creating the instance. This requirement applies to instances created from Compute Cloud@Customer platform images and by most UNIX type images. If the image is configured to use passwords instead, you need the password instead of the key pair.

Image

A template of a virtual hard drive that determines the OS and other software for a compute instance. You can also create compute instances using these images:

- Compute Cloud@Customer platform images
- Custom images created from other instances
- Import your own image

For more information about images, see Images for Compute Cloud@Customer Instances on page 283.

Shape

A template that determines the number of CPUs, amount of memory, and other resources allocated to a newly created compute instance. See Compute Shapes on page 301.

Boot Volumes

When you launch a compute instance based on an Compute Cloud@Customer platform image or custom image, a new boot volume for the compute instance is created in the same compartment. That boot volume is associated with that compute instance until you delete the compute instance.

When you delete the compute instance, you can preserve the boot volume and its data. This feature gives you more control and management options for your compute instance boot volumes, and enables:

- **Instance scaling:** When you delete your compute instance, you can keep the associated boot volume and use it to launch a new compute instance using a different compute instance type or shape. This flexibility enables you to easily scale up or down the number of cores for a compute instance.
- **Troubleshooting and repair:** If you think a boot volume issue is causing a compute instance problem, you can stop the compute instance and detach the boot volume. Then you can attach it to another compute instance as a data volume to troubleshoot it. After resolving the issue, you can then reattach it to the original compute instance or use it to launch a new compute instance.

Boot volume Encryption

Boot volumes are encrypted by default, the same as other block storage volumes.

> **Important:**
>
> In most cases, encryption isn't supported for compute instances launched from custom images imported for "bring your own image" (BYOI) scenarios.

For more information about Compute Cloud@Customer boot volumes, see Managing Boot Volumes on page 398

For information about backing up boot volumes, see Backing Up Block Volumes on page 422.

Storage for Instances

You can expand the storage that's available for your compute instances with the following services:

- **Block Volume:** Lets you dynamically provision and manage block volumes that you can attach to one or more compute instances. See Block Volume Storage on page 380.
- **File Storage:** A durable, scalable, secure, enterprise-grade network file system that you can connect to from any compute instance in your virtual cloud network (VCN). See File Storage on page 433.
- **Object Storage:** An internet-scale, high-performance storage platform that lets you store a large amount of unstructured data of any content type. This storage not tied to any specific compute instance. See Object Storage on page 478.

Simplifying Compute Instance Management

You can simplify the management of your compute instances using these features:

- **Instance Configurations:** Are templates that define the settings to use when creating compute instances.
- **Instance Pools:** are a group of compute instances that are created from the same compute instance configuration and managed as a group.

Calling Services from an Instance

A Compute Cloud@Customer instance can be configured to enable applications running on the instance to call services and manage resources similar to the way users call services to manage resources.

An instance that can perform actions on service resources is called an *instance principal*.

You can authorize an instance to make API calls in Compute Cloud@Customer services. After you set up the required resources and policies, an application running on an instance can call Compute Cloud@Customer public services, removing the need to configure user credentials or a configuration file.

See Configuring Instances for Calling Services on page 330.

Metadata Key Restrictions

Metadata keys have the following restrictions, with the noted exceptions:

- Metadata can have a maximum of 128 keys.
- Key names can have a maximum of 255 characters.
- Most key values can have a maximum of 255 characters.

The value of the `ssh_authorized_keys` metadata key can be more than 255 characters. This value must be a valid public key in OpenSSH format. Use a newline character to separate multiple keys.

The value of the `user_data` metadata key can be a maximum of 16KB. 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 data formats.

Compute Shapes

A shape is a template that determines the type and amount of resources that are allocated to a compute instance. Compute Cloud@Customer offers a choice between a flexible shape for generic workloads, and dedicated shapes for GPU-accelerated workloads.

You choose the shape configuration when you create an instance. SeeCreating an Instance on page 316.

You can use the following shapes for instances that are created on Compute Cloud@Customer:

- **Flexible Shapes on page 302:** You choose the number of OCPUs and amount of memory that are allocated to an instance.
- **GPU Shapes on page 302:** Each shape has a fixed number of OCPUs, memory, and GPUs that are allocated to an instance when the instance is created.

Flexible Shapes

A flexible shape lets you customize the number of OCPUs and the amount of memory for an instance. When you update an instance, you can change these properties of the flexible shape. This flexibility lets you create instances that meet your workload requirements, while optimizing performance and using resources efficiently.

The VM.PCAStandard.E5.Flex shape can be selected only for Compute Cloud@Customer X10 systems.

The VM.PCAStandard.E6.Flex shape can be selected only for Compute Cloud@Customer X11 systems.

Shape	OCPUs	Memory (GB)	Maximum VNICs	Maximum Bandwidth (Gbps)
VM.PCAStandard1.Flex	1–32	64 GB maximum per OCPU 512 GB maximum per instance	1 OCPU: 2 VNICs 2 to 24 OCPUs: 1 VNIC per OCPU 25 to 32 OCPUs: 24 VNICs	1 to 24 OCPUs: 24.6 Gbps 25 to 32 OCPUs: 1 Gbps per OCPU
VM.PCAStandard.E5.Flex	1–96	64 GB maximum per OCPU 960 GB maximum per instance	1 OCPU: 2 VNICs 2 to 24 OCPUs: 1 VNIC per OCPU 25 to 96 OCPUs: 24 VNICs	1 to 24 OCPUs: 24.6 Gbps 25 to 40 OCPUs: 1 Gbps per OCPU 41 to 96 OCPUs: 40.0 Gbps
VM.PCAStandard.E6.Flex	1–96	64 GB maximum per OCPU 960 GB maximum per instance	1 OCPU: 2 VNICs 2 to 24 OCPUs: 1 VNIC per OCPU 25 to 96 OCPUs: 24 VNICs	1 to 24 OCPUs: 24.6 Gbps 25 to 40 OCPUs: 1 Gbps per OCPU 41 to 96 OCPUs: 40.0 Gbps

GPU Shapes

The GPU VM shapes are optimized for enterprise GPU-accelerated workloads. They can only be used if the Compute Cloud@Customer deployment includes a GPU expansion rack. Instances created with a GPU shape have direct (passthrough) access to 1-4 physical GPUs. The ratio between GPUs, OCPUs and memory is fixed.

For GPU-accelerated workloads, you have a choice between these shapes: VM.GPU.L40S.1, VM.GPU.L40S.2,VM.GPU.L40S.3, VM.GPU.L40S.4. To access these dedicated shapes, you must create an instance based on the Oracle Linux 8 or Oracle Linux 9 platform image.

> **Note:**
>
> No GPU drivers are included in the current Oracle Linux platform images. The instance OS detects the allocated GPUs, but to use them, you need the CUDA Toolkit from the NVIDIA developer site to install the required drivers.
>
> The large download and local repository installation need a large amount of disk space. The default 50GB boot volume is insufficient on Oracle Linux 9 and only just large enough on Oracle Linux 8. It is highly recommended to increase the boot volume size to at least 60GB, and extend the file system accordingly.

Installing GPU Drivers in an Oracle Linux 9 Instance

1. From the command line of the instance, download and install the CUDA Toolkit rpm for your OS.

```
$ wget https://developer.download.nvidia.com/
compute/cuda/12.8.0/local_installers/cuda-repo-
rhel9-12-8-local-12.8.0_570.86.10-1.x86_64.rpm
$ sudo rpm -i cuda-repo-rhel9-12-8-
local-12.8.0_570.86.10-1.x86_64.rpm
$ sudo dnf clean all
$ sudo dnf install cuda-toolkit-12-8
```

2. Enable the Oracle Linux 9 EPEL yum repository. Install the `dkms` package.

```
$ sudo yum-config-manager --enable
 ol9_developer_EPEL
$ sudo dnf install dkms
```

3. Install the GPU drivers.

```
$ sudo dnf install cuda-12-8
```

4. Verify the installation with the NVIDIA System Management Interface.

```
$ nvidia-smi
+-----------------------------------------------------------------------------
+
| NVIDIA-SMI 570.86.10              Driver
 Version: 570.86.10      CUDA Version: 12.8
 |
|-----------------------------------------
+------------------------
+----------------------+
| GPU  Name                 Persistence-M |
 Bus-Id          Disp.A | Volatile Uncorr. ECC
 |
| Fan  Temp   Perf          Pwr:Usage/Cap |
       Memory-Usage | GPU-Util  Compute M. |
|                                         |
                     |                MIG M. |
|=========================================
+========================
+======================|
|   0  NVIDIA L40S                    Off |
 00000000:00:05.0 Off |                    0 |
| N/A   26C    P8             23W /  350W |
   1MiB /  46068MiB |      0%      Default |
|                                         |
                     |                  N/A |
+-----------------------------------------
+------------------------
+----------------------+

+-----------------------------------------------------------------------------
+
| Processes:
                                          |
|  GPU   GI   CI              PID   Type
 Process name                        GPU Memory
 |
```

```
|        ID   ID
                                  Usage      |
|
=============================================================================
|  No running processes found
                                             |
+-----------------------------------------------------------------------------
+
```

Installing GPU Drivers in an Oracle Linux 8 Instance

1. From the command line of the instance, download and install the CUDA Toolkit rpm for your OS.

```
$ wget https://developer.download.nvidia.com/
compute/cuda/12.8.0/local_installers/cuda-repo-
rhel8-12-8-local-12.8.0_570.86.10-1.x86_64.rpm
$ sudo rpm -i cuda-repo-rhel8-12-8-
local-12.8.0_570.86.10-1.x86_64.rpm
$ sudo dnf clean all
$ sudo dnf install cuda-toolkit-12-8
```

2. Enable the Oracle Linux 8 EPEL yum repository. Install the `dkms` package.

```
$ sudo yum-config-manager --enable
 ol8_developer_EPEL
$ sudo dnf install dkms
```

3. Install the GPU drivers.

```
$ sudo dnf install cuda-12-8
```

4. Install the NVIDIA kernel module.

```
$ sudo scl enable gcc-toolset-13 bash
# dkms install nvidia-open -v 570.86.10
```

   If this `make` error appears while the kernel module is built, you can safely ignore it.

```
Cleaning build area...(bad exit status: 2)
Failed command:
make -C /lib/
modules/5.15.0-206.153.7.el8uek.x86_64/build
 M=/var/lib/dkms/nvidia-open/570.86.10/build
 clean
```

5. Verify the installation with the NVIDIA System Management Interface.

```
# nvidia-smi
+-----------------------------------------------------------------------------
+
| NVIDIA-SMI 570.86.10              Driver
 Version: 570.86.10      CUDA Version: 12.8
 |
|-----------------------------------------
+------------------------
+----------------------+
| GPU  Name                 Persistence-M |
 Bus-Id          Disp.A | Volatile Uncorr. ECC
 |
```

```
| Fan  Temp   Perf          Pwr:Usage/Cap |
       Memory-Usage | GPU-Util  Compute M. |
|                                          |
                     |               MIG M. |
|=========================================
+========================
+======================|
|   0  NVIDIA L40S                    Off |
 00000000:00:05.0 Off |                    0 |
| N/A   26C    P8             23W /  350W |
   1MiB /  46068MiB |      0%      Default |
|                                          |
                     |                  N/A |
+-----------------------------------------
+------------------------
+----------------------+

+-----------------------------------------------------------------------------
+
| Processes:                                                                  |
|  GPU   GI   CI              PID   Type
 Process name                             GPU Memory
 |
|        ID   ID
                                 Usage      |
|
=============================================================================
|  No running processes found
                                                                            |
+-----------------------------------------------------------------------------
+
```

VM.GPU.L40S.x

Specification	Possible Values
Shape name	• VM.GPU.L40S.1 • VM.GPU.L40S.2 • VM.GPU.L40S.3 • VM.GPU.L40S.4
GPUs	1-4 – corresponding with the shape name
OCPUs	27 per GPU
Memory	200 GB per GPU
VNICs	Up to 24
Bandwidth	Up to 400 Gbps

Compute Instance Lifecycle

On Compute Cloud@Customer, a compute instance can be in a variety of different lifecycle states.

This list describes the different lifecycle states for compute instances.

- **Launching:** Occurs when you create a compute instance. The instance is displayed in the Console in a provisioning state. Expect provisioning to take several minutes before the state updates to running. After the instance is running, wait another few minutes for the OS to boot before you attempt to connect.

- **Connecting:** You connect to a running Linux or Oracle Solaris instance using a Secure Shell (SSH) connection. Most Linux and UNIX-like OSs include an SSH client by default.
- **Backing up the boot volume:** You can back up the boot volume using the Block Volume backup feature using one of these methods:
  - **Manual backups:** You manually perform create, get, list, rename, and delete backup commands.
  - **Automatic backups:** You create a backup policy and a backup policy assignment that specifies the time and frequency of the volume backups. The system automatically performs the commands that back up the volume.
- **Stopping:** You can stop an instance using the Console, CLI, and API, or by using the commands available in the OS when you are logged in to the instance.

  If the applications that run on the instance take more than 15 minutes to shut down, they could be improperly stopped. To avoid this situation, shut down the instance using the commands available in the OS before you stop the instance.
- **Starting or restarting:** You can start or restart an instance as needed using the Console, CLI, and API.
- **Rebooting:** You can reboot an instance as needed using the Console, CLI, and API. By default, a reboot gracefully restarts the instance by sending a shutdown command to the OS. After waiting 15 minutes for the OS to shut down, the instance is powered off and then powered back on.
- **Terminating:** 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.

  By default, the instance's boot volume 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. If you no longer need the boot volume, you can permanently delete it.

For more information, see Working with Instances on page 316.

Extending Compute Resources

On Compute Cloud@Customer, you can extend storage resources and add more network interfaces.

Expanding Volumes

You can expand the size of block volumes and boot volumes. You can't decrease the size.

You have several options to increase the size of your volumes:

- 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.

> **Caution:**
>
> Before you resize a boot or block volume, create a backup of the volume. See Backing Up Block Volumes on page 422.

After a volume has been resized, the first backup of the resized volume is a full backup.

Offline Resizing of Block Volumes

With offline resizing, you detach the volume from a compute instance before you expand the volume size. After the volume is resized and reattached, you need to extend the partition, but you don't need to rescan the disk.

Whenever you detach and reattach volumes, there are complexities and risks for both Linux-based and Microsoft Windows-based compute instances. For more information.

Rescanning the Disk for a Block Volume or Boot Volume

The Block Volume service lets you expand the size of block volumes and boot volumes while they're online and attached to compute instances.

After the volume is provisioned, you need to run commands to rescan the disk so that the OS recognizes the expanded volume size. You run different rescan commands depending on the OS of the attached compute instance.

For more information, see Resizing Volumes on page 403.

Adding Another Network Interface

You can add additional VNICs to a compute instance. Each additional VNIC can be in a subnet in the same VCN as the primary VNIC, or in a different subnet that's either in the same VCN or a different one.

You might add a VNIC to connect a compute instance to subnets in multiple VCNs. For example, you might set up your own firewall to protect traffic between VCNs, so the compute instance needs to connect to subnets in different VCNs.

Secondary VNICs are supported for these types of compute instances:

- **Linux and Oracle Solaris**
- **Microsoft Windows**

Here are more details about additional VNICs:

- There's a limit to how many VNICs can be attached to a compute instance, and it varies by shape. See Compute Shapes on page 301.
- VNICs can be added only after the compute instance is launched.
- VNICs must always be attached to a compute instance and can't be moved. The process of creating an additional VNIC automatically attaches it to the compute instance. The process of detaching a secondary VNIC automatically deletes it.
- VNICs are automatically detached and deleted when you delete the compute instance.
- The compute instance's bandwidth is fixed regardless of the number of VNICs attached. You can't specify a bandwidth limit for a particular VNIC on a compute instance.
- Attaching multiple VNICs from the same subnet CIDR block to a compute instance can introduce asymmetric routing, especially on instances using a variant of Linux. If you need this type of configuration, assign multiple private IP addresses to one VNIC, or using policy-based routing.

For more information, see Configuring VNICs on page 181.

Tutorial: Launching Your First Instance

In this tutorial, you'll learn the basic features of Compute Cloud@Customer by performing some guided steps to create 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.

Prerequisites

To successfully perform this tutorial, you must have the following prerequisites in place.

- You must be able to sign in to the Compute Cloud@Customer Console using your Oracle Cloud Console (OCI) account.
- You must be able to sign in to Oracle Cloud Console using your Oracle Cloud Console (OCI) account. Or have access to a compartment where you can create resources.
- An SSH key pair that's used for user authentication in the instance. If you want to create a key pair for this tutorial, see Managing Key Pairs on Linux Instances.

Tutorial Tasks

Task 1: Create a Compartment on page 308

Task 2: Create a Virtual Cloud Network (VCN) on page 309

Task 3: Create a Subnet on page 309

Task 4: Create an Internet Gateway and Configure Route Rules on page 310

Task 5: Launch an Instance on page 310

Task 6: Get the Instance IP Address on page 312

Task 7: Connect to Your Instance on page 312

Task 8: Add a Block Volume on page 313

Task 9: Attach the Block Volume to an Instance on page 314

Task 10: (Optional) Clean Up Resources on page 314

Task 1: Create a Compartment

On Compute Cloud@Customer, 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 this tutorial, you use one compartment for all resources. However, when you're 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.

If you want to use an existing compartment rather than create a new compartment, you can do so. Skip this task and use your compartment for the tasks that follow.

For more information about compartments, Managing Compartments.

> **Note:**
>
> On Compute Cloud@Customer, IAM resources such as compartments are managed in Oracle Cloud Console (OCI), and synchronized to Compute Cloud@Customer.

1. Sign in to the Oracle Cloud Console in OCI using your federated identity credentials.

   For more information, see Sign In to the Console.
2. In the Oracle Cloud Console navigation menu, select **Identity & Security**. Under **Identity**, select **Compartments**.
3. On the Compartments page, select **Create Compartment**.
4. Enter the following information:

   - **Name:** Enter Sandbox.

     Avoid entering confidential information.
   - **Description:** (Required) Enter a description for the compartment.
   - **Create in Compartment:** Select the compartment in which to create this new compartment.
5. Select **Create Compartment**.

   The new compartment is displayed in the list.

   It might take 10 minutes for the new compartment to show up on the Compute Cloud@Customer.

> **Note:**
>
> The remaining tutorial tasks use the Compute Cloud@Customer Console instead of the Oracle Cloud Console, except for 10-4: Delete the Compartment on page 316.

Perform the next task:

Task 2: Create a Virtual Cloud Network (VCN) on page 309

Task 2: 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, see Managing VCNs and Subnets on page 140.

Avoid entering confidential information in names and tags.

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. Select **Create Virtual Cloud Network**.
3. In the **Create Virtual Cloud Network** dialog box, 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 doesn't use tags.
4. Select **Create Virtual Cloud Network**.

Perform the next task:

Task 3: Create a Subnet on page 309

Task 3: 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 directs traffic to an internet gateway. The subnet also uses a security list to control traffic in and out of the instance.

Avoid entering confidential information in names and tags.

1. Return to the **Virtual Cloud Networks** page.

   A quick way to do this is to select the name of page in the breadcrumb that's at the top of the page.

   If the VCN details page isn't in your breadcrumb, in the navigation menu, select **Networking**, **Virtual Cloud Networks**, then select the name of your VCN.
2. Scroll down to the **Resources** panel, select **Subnets**, and select **Create Subnet**.
3. In the **Create Subnet** dialog box, 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 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:** Select Add Security List and select the default security list.
- **Tagging:** Leave blank. This tutorial does not use tags.

4. Select **Create Subnet**.

Perform the next task:

Task 4: Create an Internet Gateway and Configure Route Rules on page 310

Task 4: 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. Ensure to only permit the required 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.

Avoid entering confidential information in names and tags.

1. Navigate to your VCN details page.
2. In the **Resources** panel, select **Internet Gateways**.
3. Select **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 doesn't use tags.
5. Select **Create**.
6. Under **Resources**, select **Route Tables**.
7. Select the name of the default route table.
8. Scroll down to the **Resources** panel and select **Add Route Rules**.
9. On the **Create Route Table Rule** dialog box, enter the required information:
   - **Target Type:** From the menu, select Internet Gateway.
   - **CIDR Block:** Enter: `0.0.0.0/0` (which means that all nonintra-VCN traffic that isn't already covered by other rules in the route table will go to the target specified in this rule).
   - **Internet Gateway:** From the menu, select the name of the Internet Gateway that you created.
   - **Description:** An optional description of the rule.
10. Select **Create Route Table Rule**.

Perform the next task:

Task 5: Launch an Instance on page 310

Task 5: Launch an Instance

In this task, launch an instance with an image and a 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 and instance components, see these topics:

- Images for Compute Cloud@Customer Instances on page 283
- Compute Shapes on page 301
- Working with Instances on page 316

Avoid entering confidential information in names and tags.

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. On the **Instances** page, select **Create Instance**.
3. In the **Launch Instance** dialog box, 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 an Oracle Linux image.
   - **Shape:** Select a flexible shape (with *Flex* in its name), and configure the number of OCPUs and memory. For this temporary instance, set the OCPU value to 1, and memory to 16. For descriptions of available shapes, see Compute Shapes on page 301.
   - **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:
     - Select 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 Paste the public keys, 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 checkbox clear.
   - **Instance Options:** Don't select **Leave the Legacy Instance Metadata Service Endpoints Disabled**
   - **Availability configuration:** Leave this area as is.
   - **Tagging:** Leave blank. This tutorial doesn't use tags.
4. Select **Create Instance**.
5. Monitor the state of the instance.

   The state is displayed under the icon of the object.

   Your instance begins in the PROVISIONING state. After the instance is in the RUNNING state, you can connect to it.

Perform the next task:

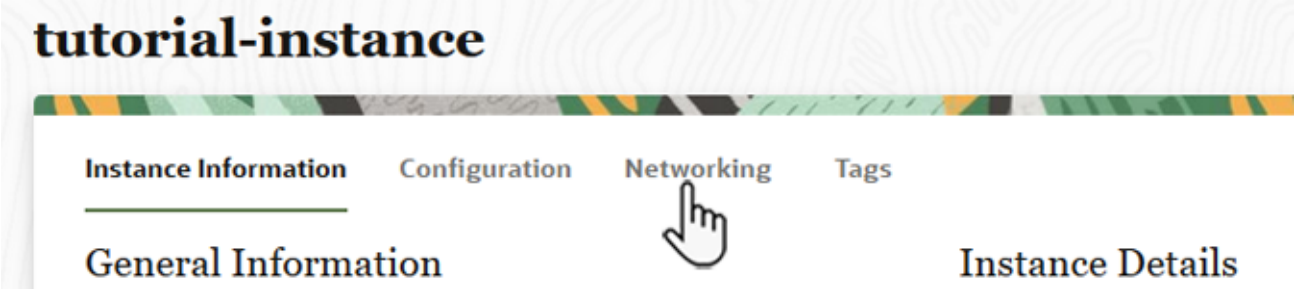
Task 6: Get the Instance IP Address on page 312

Task 6: Get the Instance IP Address

You connect to the instance using SSH with the instance IP address.

1. After creating the instance, you're likely on the instance details page. If not, navigate to the details page: In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**. Then select 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.

Perform the next task:

Task 7: Connect to Your Instance on page 312

Task 7: Connect to Your Instance

Connect to your instance using SSH.

Usually, 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 Compute Cloud@Customer, 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.

Before You Begin

- Get the public IP address of your instance, as described in Task 6: Get the Instance IP Address on page 312.
- 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 Compute Cloud@Customer, the default user is `opc`. See Initial User Account on page 284.

Using the `ssh` Command

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 username for the instance. For this tutorial, `opc` is the username.

- ***public-ip-address*** is your instance IP address.

  Example:

```
$ ssh -i /home/flast/.ssh/my_private_key opc@192.0.2.1
```

3. If asked whether you want to continue connecting, type `yes`.

   You are now logged in to your instance.

Perform the next task:

Task 8: Add a Block Volume on page 313

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 username for the instance. For this tutorial, the username 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 select **Auth**.
6. Select **Browse**, and then select your `.ppk` private key file.
7. Select **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 isn't cached in the registry. Select **Yes** or **Accept** to continue the connection.

   > **Tip:**
   >
   > If the connection fails, you might need to update your PuTTY proxy configuration.

Perform the next task:

Task 8: Add a Block Volume on page 313

Task 8: Add a Block Volume

Add additional storage by adding a block volume to your instance.

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.

Avoid entering confidential information in names and tags.

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Block Volumes**.
2. Select **Create Block Volume**.
3. In the **Create Block Volume** dialog box, 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:** Don't select a backup policy.
- **Tags:** Leave blank. This tutorial doesn't use tags.

4. Select **Create Block Volume**.
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.

Perform the next task:

Task 9: Attach the Block Volume to an Instance on page 314

Task 9: Attach the Block Volume to an Instance

Attach the block volume to an instance.

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. Ensure that the **Sandbox** compartment is selected at the top of the page.
3. In the **Instances** list, select the name of your instance to view its details.
4. Scroll down to the **Resources** panel, and select **Attached Block Volumes**.
5. Select **Attach Block Volume**.
6. In the **Attach Block Volume** dialog box, enter the following information:

   - **Select from Compartment:** Select the Sandbox compartment.
   - **Block Volume:** Select the block volume you created.
   - **Access:** Select Read/Write.

7. Select **Attach to Instance**.

   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 on page 389
- Configuring Volumes to Automatically Mount (Linux Instances) on page 391

Perform the next task:

Task 10: (Optional) Clean Up Resources on page 314

Task 10: (Optional) Clean Up Resources

After you've finished with the resources you created for this tutorial, you can delete and release the resources you no longer plan to use.

Perform the following tasks in the order listed:

10-1: Detach and Delete the Block Volume

After you detach a block volume, you can delete it as long as the volume isn't attached to any other instances.

> **Caution:**
>
> You can't undo a termination. Any data on a volume is permanently deleted when the volume is deleted.

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. Select the **Sandbox** compartment.
3. Select the name of your instance.
4. In the **Resources** panel, select **Attached Block Volumes**.
5. Find your volume, select the Actions menu (⋮), then select **Detach**. Confirm the detachment.

   You might need to refresh the web page to see that the block volume is no longer attached.
6. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Block Volumes**.
7. Select the **Sandbox** compartment.
8. Find your volume, select the Actions menu (⋮), and then select **Terminate**. Confirm the termination.

Perform the next task:

10-2: Terminate the Instance on page 315

10-2: Terminate the Instance

You can permanently terminate (delete) instances that you no longer need. Any attached VNICs and boot 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.

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. Select the **Sandbox** compartment.
3. Find the instance you created, select the Actions menu (⋮), then select **Terminate**.
4. In the **Confirm Instances termination** dialog box, move the "**Permanently delete the attached boot volume**" selector to the right, and select **Confirm**.

   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.

Perform the next task:

10-3: Delete the Subnet, Internet Gateway, and VCN on page 315

10-3: Delete the Subnet, Internet Gateway, and VCN

You can delete networking resources as long as the resource doesn't have any dependencies.

- **Subnets**: Must be empty (for example, no instance VNICs or route rules).
- **VCNs**: Must be empty and have no related resources or attached gateways (for example, no internet gateway, dynamic routing gateway, and so on).

1. In the Compute Cloud@Customer Console navigation menu, select **Networking**, then select **Virtual Cloud Networks**.
2. Select the **Sandbox** compartment.
3. Select the name of your VCN.
4. Under **Resources**, select **Route Tables**.
5. Select the name of the route table.
6. For the route rule you created, select the Actions menu (⋮), select **Delete**, and confirm the deletion.

   The route rule is deleted.
7. In the breadcrumb path at the top of the page, select the name of your VCN.

   The VCN details page is displayed.
8. Under **Resources**, select **Internet Gateways**.

9. For the internet gateway that you created, select the Actions menu (⋮), then select **Delete**, and confirm the deletion.

   The internet gateway is deleted.
10. Under **Resources**, select **Subnets**.
11. For the subnet you created, select the Actions menu (⋮), then select **Delete**, and confirm the deletion.

    The subnet is deleted.
12. At the top of the **VCN details** page, select **Terminate** and confirm the termination.

    The VCN is deleted.

Perform the next task:

10-4: Delete the Compartment on page 316

10-4: 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.

> **Note:**
>
> On Compute Cloud@Customer, IAM resources such as compartments are managed in Oracle Cloud Console (OCI), and synchronized to Compute Cloud@Customer.

1. Sign in to the Oracle Cloud Console in OCI using your federated identity credentials.

   For more information, see Sign In to the Console.
2. In the Oracle Cloud Console navigation menu, select **Identity & Security**, then under **Identity**, select **Compartments**.
3. For the Sandbox compartment, select the Actions menu (⋮), and then select **Delete**.
4. Confirm the deletion.

This concludes the tutorial.

Working with Instances

On Compute Cloud@Customer, 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 delete it.

Creating an Instance

On Compute Cloud@Customer, you can create an instance using the Console, CLI, and API.

Before You Begin

See Tutorial: Launching Your First Instance on page 307 for information about input you need to create an instance.

The following list describes the minimum information that you must provide to create an instance:

- A name for the instance
- The compartment where you want to create the instance
- An image or boot volume
- A shape – If you're using a platform image, specify a standard/flexible shape for generic workloads and applications. For GPU-accelerated workloads, select a dedicated GPU shape. For descriptions of available shapes, see Compute Shapes on page 301.
- 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 requires SSH keys for authentication, you must provide the public key when you create the instance. You can't provide the public SSH key after the instance is created.

To create an instance using the CLI, you need the same information as previously listed for the Console except that you don't need an instance name. If you don't 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 CLI. You can't change launch options or boot volume VPUs per GB after the instance is created.

Instance metadata options enable you to attach important information to instances such as an SSH key, information for cloud-init to use, and labels and proxies for nodes in a Kubernetes Engine cluster node pool. There are metadata key restrictions. See Metadata Key Restrictions on page 301. To configure metadata, create the instance using the CLI `--metadata` or `--extended-metadata` option.

An alternative way to create an instance is to create an instance configuration and use that configuration to launch an instance, as described in Working with Instance Configurations on page 333. 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 on page 334.

After you create an instance, you can optionally attach additional VNICs and assign private and public IP addresses. See Creating and Attaching a Secondary VNIC on page 183, Managing Private IP Addresses on page 186, and Managing Public IP Addresses on page 191.

> **Note:**
>
> If you specify a boot volume size that's larger than the default, you need to extend the volume to take advantage of the larger size. See Resizing Volumes on page 403.

Avoid entering confidential information in names and tags.

Using Marketplace Images

On Compute Cloud@Customer, the first time you create an instance using a Marketplace image, you must use the Compute Cloud@Customer Console so that you can accept the user agreement. After that, you can use the Console, CLI, and API to create instances with a Marketplace image.

To maintain the integrity of Marketplace images, there are restrictions and permissible actions. For more information, see Marketplace Images on page 292.

Using the Console

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. See Listing Images and Viewing Details on page 283 and Listing Boot Volumes on page 399.
   - A virtual cloud network (VCN) and subnet and the compartment where the VCN and subnet are located. See Creating a VCN on page 141.
   - A public Secure Shell (SSH) key if users will connect to the instance using SSH
2. On the Compute Cloud@Customer Console **Dashboard**, do one of the following to open the **Create Instance** dialog box:
   - Select **Create Instance** in the top-left corner.
   - In the **Compute** block, select **Instances**. At the top of the instances list on the right side, select **Create Instance**.
   - In the **Compute** block, select **Custom Images**. For the image that you want to use to create the instance, select the Actions menu (⋮), then select the **Create Instance From Image**. 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 box and you can't change it. You don't need to enter any of the information described in "Source Image" in the following step.

3. In the **Create Instance** dialog box, 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.
     - Doesn't 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 can't 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 to view another page of the list.

     To use a platform image that was previously available but is no longer listed, use the CLI to create the instance and specify the OCID of the image.

     The source image can't be changed after the instance is created.
   - **Shape**: If you're using a platform image, select the appropriate shape for the intended use:
     - Generic workloads and applications: Select a flexible shape and configure the number of OCPUs and memory.

       For the OCPU and memory values, select inside each value field to see the minimum and maximum allowed values. The OCPU and memory configuration can be changed after the instance is created.
     - 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 a description of the supported shapes, see Compute Shapes on page 301.
   - **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 of gigabytes up to 16384 GB (16 TB) or use the increment and decrement arrows. You can't enter a value smaller than the default.

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 OSs or for custom images, follow the instructions for that OS.

- **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 on page 381.

  If you specify the high performance option and a high performance pool exists but the specified image doesn't 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.

  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 won't incur this image copy delay.

  You receive an error message from the image copy if the image is larger than 200 gigabytes. Platform images aren't 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 don't 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 on page 193 for instructions.
- **Private IP Address:** (Optional) Specify an available private IP address from the subnet CIDR. By default, a private IP address is automatically assigned.
- **Hostname:** (Optional) Enter a hostname if you're 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 on page 442 and Mounting File Systems on UNIX-based Instances on page 453 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 can't 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 isn't 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. Select Add Another NSG if you want to attach to another NSG.
  c. To remove an NSG from the list, select the trash can to the right of that NSG. To remove the last NSG or all NSGs, clear the **Enable Network Security Groups** box.

  To update NSG attachments for this instance later, see Updating a VNIC on page 185.

  See Controlling Traffic with Network Security Groups on page 163 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 on page 324. For more information about upgrading your applications, see Upgrading to IMDS Version 2 Endpoints on page 325.

- **Availability configuration:** (Optional) Specify how to handle this instance during compute node maintenance"
  - **Let** Compute Cloud@Customer **choose the best migration option**

    This option is selected by default to allow the system to choose the best option to handle this instance during compute node maintenance. The best option typically is live migration to a healthy compute node. You can't change this setting. If this instance should not be live migrated, for example live migration isn't 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**

    This option is selected by default to specify that running instances should be automatically restarted after a maintenance operation such as live migration. If this box isn't checked, the instance is recovered in the stopped state.
- **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.

4. Select **Create Instance**.

   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 select **Work Request(s)**.

   If instance launch fails because of resource constraints, try remedies such as the following:

   - Specify a different fault domain, or don't specify any fault domain and allow the system to choose it.
   - Specify a less resource-intensive shape.
   - Stop an instance that you no longer need.
   - Terminate an instance that you no longer need.

   If the status of the work request is Failed, and no reason is listed for the failure, the cause of the failure might be temporary. Wait a short time and then retry the instance create.

Using the CLI

Use the oci compute instance launch command and required parameters to create an instance.

```
oci compute instance launch --availability-domain availability_domain --
compartment-id compartment_OCID --shape shape --subnet-id subnet_OCID --
source-details file://image_info.json [OPTIONS]
```

For a complete list of required and optional parameters, use the following command:

```
oci compute instance launch -h
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Procedure

1. Create or get the following resources and information:
   - The name of the availability domain that you want to use: `oci iam availability-domain list`
   - The OCID of the compartment where you want to create the instance: `oci iam compartment list`
   - The name of the shape for this instance. Select the appropriate shape for the intended use:

- Generic workloads and applications – select a flexible shape (with *Flex* in its name) – for example:

  ```
  VM.PCAStandard.E5.Flex
  ```

  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}'
  ```

- GPU-accelerated workloads:

  ```
  VM.GPU.L40S.1 | VM.GPU.L40S.2 | VM.GPU.L40S.4
  ```

  GPU shapes are not flexible. There are no custom configuration parameters.

See Compute Shapes on page 301.

The 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 Console tab.
- Gather one of the following values to specify the image source – either an image or a boot volume.
  - The OCID of the image used to boot the instance:

    ```
    oci compute image list
    ```

    > **Note:**
    >
    > Don't select an image that has "`-OKE-`" in its display name. The "`-OKE-`" images can only be used withKubernetes Engine (OKE).

  - 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 can't 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
```

For information about `--display-name` and `--hostname-label` values, see descriptions in the Console tab on this page.

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 Console tab.

   For information about `bootVolumeVpusPerGB`, see "High Performance" in the Console tab.

> **Note:**
>
> When you later `list` or `get` this instance, the value of `bootVolumeVpusPerGB` is `null` because this boot volume property isn't stored in the instance object after the instance is created. 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 don't provide a correct value for `firmware`, the instance might not launch. You can't 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 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. Extended metadata can be nested JSON objects.

   The combined size of the metadata and extended metadata can be a maximum of 32,000 bytes.

   SSH keys can alternatively be provided in the file argument of the `--ssh-authorized-keys-file` option. User data can alternatively be provided in the file argument of the `--user-data-file` option. Use the `-h` option for more information.

   In the example in the next step, the authorized keys file contains one or more public SSH keys in the format required by the SSH `authorized_keys` file. Use a newline character to separate multiple keys. SSH public keys can be provided as the value of the `ssh_authorized_keys` key in the `--metadata` option, or in the file argument of the `--ssh-authorized-keys-file` option. Use `-h` for more information.

5. Run the instance launch command.

   Syntax:

   ```
   oci compute instance launch --availability-domain availability_domain_name
    \
   --compartment-id compartment_OCID --shape shape --subnet-id subnet_OCID \
   ```

```
--source-details file://image_info.json
```

Example:

If you're 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 on page 193 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 on page 324.

```
$ oci compute instance launch --availability-domain AD-1 --
compartment-id <compartment_OCID> --display-name ops1 --shape
 VM.PCAStandard.E5.Flex --subnet-id <subnet_OCID> --source-details
 '{"bootVolumeSizeInGBs":100,"bootVolumeVpusPerGB":20,"imageId":"<image_OCID>","sourc
 --assign-public-ip true --ssh-authorized-keys-file ./.ssh/<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.PCAStandard.E5.Flex",
    "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.

Using the API

Use the LaunchInstance operation to create an instance.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Retrieving Instance Metadata from Within the Instance

On Compute Cloud@Customer, 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's 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 information:

- The same information that you see on the details page of an instance in the Console and in the output of the instance `get` 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 can't 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 Guest Operating System Matrix on page 33. Other platform images and most other images don't 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 on page 325.
3. Disable IMDSv1 endpoints.

   Perform one of the following steps, as described in Creating an Instance on page 316.

   - 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, select 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.PCAStandard.E5.Flex",
    "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 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

On Compute Cloud@Customer, you can change instance parameters.

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 on page 383 and Creating and Attaching a Secondary VNIC on page 183. You can also specify block volumes and secondary VNICs when you use an instance configuration to create an instance.

If you didn't 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 on page 193 for instructions.

You can update the display name, fault domain, shape, instance options, availability configuration, and tags of an instance. Using the CLI, you can also update the instance metadata. See the descriptions of these properties in Creating an Instance on page 316.

Avoid entering confidential information in names and tags.

Using the Console

When you update an instance using the Console, you can change the following instance parameters:

- **Instance display name**

  Avoid entering confidential information
- **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 on page 328 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 can't 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 on page 328 for how to prepare for an instance to stop. Stopping and starting an instance can take up to five minutes.

  If you're using a flexible shape, you must specify the number of OCPUs you want and you can specify the total amount of memory you want. See Compute Shapes on page 301.

  If the specified shape and shape configuration can't 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 on page 324.
- **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.
- **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.

1. On the Compute Cloud@Customer Console **Dashboard**, in the **Compute** block, select **Instances**.
2. At the top of the page, select the compartment that contains the instance.
3. For the instance that you want to update, select the Actions menu (⋮), then select **Edit**.
4. In the **Edit *`instance_name`*** dialog box, make the changes.
5. Select **Save Changes**.

   If you changed the fault domain, shape, or shape configuration of a running instance, you must confirm that you understand that the instance must be rebooted. For these changes, the instance is stopped, reconfigured, and restarted.

Using the CLI

Use the oci compute instance update command and required parameters to update the specified instance.

```
oci compute instance update --instance-
id <instance_OCID> <options_with_values_to_update> [OPTIONS]
```

For descriptions of instance properties that you can change using the CLI, enter the following command and scroll to Optional Parameters:

```
$ oci compute instance update -h
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Procedure

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>
```

Using the API

Use theUpdateInstance operation to update the specified instance.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Stopping, Starting, and Resetting an Instance

On Compute Cloud@Customer, you can control the state of an instance using the Console, CLI, and API.

The following note applies to all the stop and reset actions.

> **Important:**
>
> For soft stop and soft reset, any application that's 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's running on the instance is immediately stopped, possibly resulting in data corruption. To avoid stopping the instance while applications are running, manually shut down the instance 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.

You can perform the following actions on an instance:

- **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's Stopped can't be migrated to a different compute node.
- **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 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 on page 326), change the configuration of the instance, or stop, reconfigure, or delete other instances.

  If the start operation fails, and no reason is provided for the failure, the cause of the failure might be temporary. If no reason is provided for the failure, wait a short time and then retry the instance start.

  Starting an instance can take up to five minutes.
- **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.

- **SOFTSTOP**. Gracefully shut down the instance by sending a shutdown command to the OS. After waiting 15 minutes for the OS to shut down, the instance is powered off. If the applications that run on the instance take more than 15 minutes to shut down, they could be improperly stopped, resulting in data corruption. To avoid this, manually shut down the instance using the commands available in the OS before you softstop the instance. See the description of STOP and the important note at the beginning of this section.
- **SOFTRESET**. Gracefully reboot the instance by sending a shutdown command to the OS. 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 Console

1. On the Compute Cloud@Customer Console **Dashboard**, in the **Compute** block, select **Instances**.
2. At the top of the page, select the compartment that contains the instance you want to manage.
3. For the instance that you want to manage, select Actions menu (⋮), then select **Start**, **Stop**, or **Reset**.

   Alternatively, in the instance list, select the name of the instance to display the details page for that instance. Select the **Controls**, then select the **Start**, **Stop**, or **Reset** option.
4. If applicable, enable force stop or force reboot options.

   By default, selecting **Stop** selects Soft Stop and selecting **Reset** selects Soft Reset. To stop or reset the instance immediately, select the **Force** option on the confirmation dialog box.

   - On the **Stop the instance named <instance_name>** dialog box, enable **Force stop the instance by immediately powering off**.
   - On the**Reboot the instance named <instance_name>** dialog box, enable **Force reboot the instance by immediately powering off, then powering back on**.
5. Select **Start Instance**, **Stop Instance**, or **Reboot Instance** on the confirmation dialog box.

   On the instance details page, under **Resources**, select **Work Request(s)** to check the status of the instance stop, start, or reboot.

Using the CLI

Use the oci compute instance action command and required parameters to stop, start, or reset the specified instance.

```
oci compute instance action --instance-id <instance_OCID> --action {START |
 STOP | RESET | SOFTSTOP | SOFTRESET} [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the InstanceAction operation to stop, start, or reset the specified instance.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting an Instance

On Compute Cloud@Customer, you can delete instances using the Console, CLI, and API.

By default, the boot volume of the instance is preserved when you delete (terminate) the instance. You can attach the boot volume to a different instance as a data volume, or use it to create a new instance.

You have the option to permanently delete the attached boot volume when you delete the instance. If you preserve the attached boot volume and then later decide to permanently delete the volume, see Deleting a Boot Volume on page 402.

Using the Console

1. On the Compute Cloud@Customer Console **Dashboard**, in the **Compute** block, select **Instances**.
2. At the top of the page, select the compartment that contains the instance you want to delete.
3. Select the name of the instance you want to delete.

4. On the instance details page, select **Controls** (upper right corner), then select **Terminate**.
5. On confirmation dialog box, do one of the following actions:
   - To keep the boot volume: Leave the slider in the default position.
   - To permanently delete the boot volume: Move the slider to the right.
6. Select **Confirm**.
7. Under **Resources**, select **Work Request(s)** to check the status of the termination.

   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 CLI

Use the oci compute instance terminate command and required parameters to delete the specified instance.

```
oci compute instance terminate --instance-id <instance_OCID> [OPTIONS]
```

To permanently delete the attached boot volume when the instance is terminated, specify `--preserve-boot-volume false` on the command line.

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.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the TerminateInstance operation to delete the specified instance.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Configuring Instances for Calling Services

A Compute Cloud@Customer compute instance can be configured to enable applications running on the instance to call services and manage resources similar to the way Compute Cloud@Customer 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 on page 330.
- Ensure the instance is included in a dynamic group (configured in your tenancy) that grants permissions to access the required resources. See Managing Dynamic Groups.

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

Configuring Instance Firewalls to Allow Calling Services

This section 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

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

2. 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 the 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 boot time.

   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

On Compute Cloud@Customer, by default, endpoints (such as `iaas` and `identity`) offer a certificate that's signed by a CA that's specific to that Compute Cloud@Customer. By default, OSs don't trust certificates that are signed by a CA that's specific to this Compute Cloud@Customer. If the OS doesn't trust the certificates that are offered, attempts to use the OCI SDK or CLI fail with a `CERTIFICATE_VERIFY_FAILED` error.

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

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

Option 1: Bring Your Own Certificate (BYOC)

On Compute Cloud@Customer, you can provide your own Certificate Authority (CA) certificates which enables you to use your CA trust chain. To use your own certificates, open a Support Request, and request to use your own certificates. See Creating a Support Request. To access support, sign in to the Oracle Cloud Console as described in Sign In to the OCI Console.

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

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

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

This method copies the Compute Cloud@Customer-specific CA bundle to the instance, but doesn't 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 Compute Cloud@Customer.

   ```
   curl --insecure -sS -o external_ca.crt --noproxy "*" https://
   iaas.ccc_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 many 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 Compute Cloud@Customer 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.

> **Important:**
>
> 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 Compute Cloud@Customer.

   ```
   curl --insecure -sS -o external_ca.crt --noproxy "*" https://
   iaas.ccc_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's 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 runs by cloud-init inside the instance during init, saving the effort of performing these steps manually on many instances.

Working with Instance Configurations

On Compute Cloud@Customer, an instance configuration contains settings that are used to create compute instances. Instance configurations enable you to consistently create instances with the same configuration without reentering configuration values. You can use an instance configuration to create a single instance or to create an instance pool.

Creating an Instance Configuration

On Compute Cloud@Customer, 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

On Compute Cloud@Customer you can 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 doesn't include any information from the boot volume of the template instance. For example, installed applications, binaries, and files that are on the template instance aren't 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 a Custom Image from an Instance on page 285.
  2. Use the custom image to create a new instance. See Creating an Instance on page 316.
  3. Use the instance that you created in the preceding step as the template to create the instance configuration.
- Instances with replicated boot volumes can't be used as template instances for an instance configuration. For example, instances that are included in a disaster recovery configuration can't be used as template instances for an instance configuration.
- The instance configuration doesn't 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 on page 334.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. At the top of the page, select the compartment that contains the instance that you want to use to create the new instance configuration.
3. Select the name of the instance that you want to use to create the new instance configuration.
4. On the instance details page, select **Controls** (upper right corner), then select **Create Instance Configuration**.
5. In the **Create Instance Configuration** dialog box, 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 one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
6. Select **Create Instance Configuration**.

Using the CLI

Use the oci compute-management instance-configuration create-from-instance command and required parameters to create an instance configuration by using the configuration information from an existing compute instance.

```
oci compute-management instance-configuration create-from-instance --
compartment-id <compartment_OCID> --instance-id <instance_OCID> --display-
name <IC_name>
```

The specified compartment is where this instance configuration is created.

The specified display name is the name of the instance configuration. If you don't 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.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateInstanceConfiguration operation to create an instance configuration using an instance.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Creating an Instance Configuration by Entering Configuration Values

On Compute Cloud@Customer, you can create an instance configuration by entering values for individual instance configuration settings.

> **Note:**
>
> If you specify a boot volume size that's larger than the default, you need to extend the volume to take advantage of the larger size. See Resizing Volumes on page 403.

Avoid entering confidential information in names and tags.

Using Marketplace Images

On Compute Cloud@Customer, the first time you create an instance using a Marketplace image, you must use the Compute Cloud@Customer Console so that you can accept the user agreement. After that, you can use the Console, CLI, and API to create instances with a Marketplace image.

To maintain the integrity of Marketplace images, there are restrictions and permissible actions. For more information, see Marketplace Images on page 292.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instance Configurations**.
2. Select **Create Instance Configuration**.
3. In the **Create Instance Configuration** dialog box, 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 to be created.
   - **Fault Domain**: (Optional) You can select a fault domain. By default, the system automatically selects the best fault domain for instances. If you specify a fault domain, and the requested fault domain can't accommodate the instance, instance launch fails. See more information about fault domains in Creating an Instance on page 316.
   - **Source Image:** Select an image or boot volume.
     a. Select the **Source Type:**

- **Platform Image**: Select to create an instance based on one of the Oracle Linux and Oracle Solaris images that are available in every compartment. See Compute Cloud@Customer Platform Images on page 284.
- **Custom Image**: Selectable if a custom image was created. You might need to select the compartment where the custom image is located. See Managing Custom Images on page 285.
- **Boot Volume**: Select to create the instance based on an existing boot volume. You might need to select the compartment where the boot volume is located.
- **Marketplace Image**: Selectable if Marketplace images are available on Compute Cloud@Customer. See Marketplace Images on page 292.

b. 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 OS, 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 Console to create the instance and specify the OCID of the image.

If you selected a **Marketplace image**, select a Marketplace image. If this is the first time you've created an instance with this Marketplace image, an **Agreement Panel** is displayed. Before the instance is created, you must select **Accept Agreement** and confirm the agreement.

- **Shape:** If you're using a platform image, select the **VM.PCAStandard.E5.Flex** shape, and configure the number of **OCPUs** and **memory**.

  For the **OCPU** and **memory** values, select inside each value field to see the minimum and maximum allowed values. The OCPU and memory configuration can be changed after the instance is created.

  For a description of the supported shape, see Compute Shapes on page 301.
- **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 can't 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 OSs or for custom images, follow the instructions for that OS.
  - **Boot volume performance (VPUs)**: Use the increment and decrement arrows to toggle between balanced performance (10 VPUs/GB) and high performance (20 VPUs/GB).
- **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 don't 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 on page 193 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 requested information requested, see Creating and Attaching a Secondary VNIC on page 183.
- **Private IP Address:** (Optional) Specify an available private IP address from the subnet CIDR. By default, a private IP address is automatically assigned.

  > **Note:**
  >
  > Because the private IP address must be unique for each instance, don't specify a private IP address if you're 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're using DNS within the cloud network. The hostname must be unique across all VNICs in the subnet.

  > **Note:**
  >
  > Don't 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 on page 442 and Mounting File Systems on UNIX-based Instances on page 453 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 can't provide this SSH key after the instance is created.

- **Network Security Group:** (Optional) By default, instances aren't 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. Select **Add Network Security Group** to attach to another NSG.
  c. To remove an NSG from the list, select 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 on page 163 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 on page 324. For more information about upgrading your applications, see Upgrading to IMDS Version 2 Endpoints on page 325.
- **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 isn't checked, the instance is recovered in the stopped state.
- **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.

4. Select **Create Instance Configuration**.

Using the CLI

Use the oci compute-management instance-configuration create command and required parameters to create an instance configuration by entering configuration values.

```
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 is 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 don't 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.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Procedure

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 name of the availability domain for instances that use this instance configuration.
   - 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 supplies input to the configuration create command.

   The configuration file is a JSON file of property/value pairs.

   - Use the following command to generate 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 don't need all the data that's output by this command. Copy only 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 are placed in only that fault domain.

     If a fault domain that you specify doesn't have enough resources, instances might fail to launch:

     - When you create a single instance (Using an Instance Configuration to Create an Instance on page 341), and you specify a fault domain in the instance configuration, only that specified fault domain is used to create the instance. Resource constraints might 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 on page 350 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 Create an Instance on page 341) 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 on page 193 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 can't 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 Create an Instance on page 341. If you don't provide a value for the `displayName` property, the default name of instances is `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 on page 350.

- 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 on page 324.

Specify a flexible shape – with *Flex* in its name, for example `VM.PCAStandard.E5.Flex` –, and also specify the shape configuration. You must provide a value for `ocpus`. The `memoryInGBs` property is optional; the default value in gigabytes is 16 times the number of `ocpus`.

For information about `bootVolumeSizeInGBs`, see "Boot volume size" in Creating an Instance Configuration from an Instance on page 333 CLI tab.

For information about `bootVolumeVpusPerGB`, see "High Performance" in Creating an Instance Configuration from an Instance on page 333 CLI tab. When instances are launched, the value of `bootVolumeVpusPerGB` is `null` because this boot volume property isn't 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` can't 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 is 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 don't 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.

Using the API

Use the CreateInstanceConfiguration operation to create an instance configuration entering configuration values.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Updating an Instance Configuration

On Compute Cloud@Customer, you can change the instance configuration name, compartment, and tags.

> **Note:**
>
> To change other configuration details such as the subnet, or image, create a new instance configuration. See Creating an Instance Configuration on page 333.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instance Configurations**.

2. At the top of the page, select the compartment that contains the instance configuration that you want to update.
3. For the instance configuration that you want to update, select the Actions menu (⋮), and select **Edit**.
4. In the **Update Instance Configuration** dialog box, make the changes.
5. Select **Update Instance Configuration**.

Using the CLI

Use the oci compute-management instance-configuration update command and required parameters to change the instance configuration name or tags.

```
oci compute-management instance-configuration update --instance-
configuration-id <compartment_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateInstanceConfiguration operation to change the instance configuration name or tags.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Moving an Instance Configuration to a Different Compartment

On Compute Cloud@Customer, 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 aren't 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.

Avoid entering confidential information in names and tags.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci compute-management instance-configuration change-compartment command and required parameters move an instance configuration to a different compartment.

```
oci compute-management instance-configuration change-compartment --
compartment-id destination_compartment_OCID --instance-configuration-
id instance_configuration_OCID
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

This task is not available in the API.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting an Instance Configuration

On Compute Cloud@Customer, you can delete an instance configuration as long as the instance configuration isn't being used by an instance pools.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instance Configurations**.

2. At the top of the page, select the compartment that contains the instance configuration.
3. Select the name of the instance configuration that you want to delete.
4. On the instance configuration details page, select **Delete**.
5. Select **Confirm**.

Using the CLI

Use the oci compute-management instance-configuration delete command and required parameters to delete an instance configuration.

```
oci compute-management instance-configuration delete [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteInstanceConfiguration operation to delete an instance configuration.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Using an Instance Configuration to Create an Instance

On Compute Cloud@Customer, you can use 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 on page 316.

The name of the instance is determined as follows:

- 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 aren't required to be unique.
- If the instance configuration doesn't 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.

Avoid entering confidential information in names.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci compute-management instance-configuration launch-compute-instance command and required parameters to change the instance configuration name or tags.

```
oci compute-management instance-configuration launch-compute-instance
 [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the GetInstanceConfiguration operation to change the instance configuration name or tags.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Connecting to a Compute Instance

On Compute Cloud@Customer, you can connect to a running instance by using a Secure Shell (SSH) or Remote Desktop connection the same way you connect to instances in Oracle Cloud Console (OCI).

For more information, see these sections:

- To create an instance console connection, see Creating an Instance Console Connection on page 342 on Compute Cloud@Customer.
- For connection prerequisites, connection instructions, and troubleshooting suggestions, see Connecting to an Instance.
- For information about SSH key pairs, see Managing Key Pairs on Linux Instances.

Creating an Instance Console Connection

On Compute Cloud@Customer, 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.

Console Connection Prerequisites

Ensure that you have the following resources on the system that you plan to use to connect to the instance console.

- The console connection needs access to port 1443. Previously, port 443 was used for VM console access. For more information, see Port Matrix
- SSH key pair

  If you don't 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 don't need to provide a passphrase; using a passphrase makes it more difficult to automate connecting.
- 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.
  - Git for Windows

    Git for Windows includes OpenSSH.
  - Windows Subsystem for Linux (WSL)

    WSL includes OpenSSH.
- VNC viewer to connect to the VNC console
- Ensure that you belong to a group that has the following permissions.

```
Allow group group_name to manage instance-console-connection in tenancy
Allow group group_name to read instance in tenancy
```

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. At the top of the page, select the compartment that contains the instance.
3. Select the name of the instance where you want to create a console connection.
4. On the instance details page, under **Resources**, select **Console Connection**.

5. If a console connection doesn't already exist, select **Create Console Connection**.
6. Provide the public key portion of your SSH key.

   In the **Create Console Connection** dialog box, do one of the following to enter your public SSH key:

   - Select the key file(s).
     - Select 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. Select **Create Console Connection** in the dialog.

   When the console connection is created and is available, the state changes to **Active**.

What's Next

Continue to Connecting to the VNC Console or Making a Local Connection to the Serial Console.

Using the CLI

Use the oci compute instance-console-connection create command and required parameters to create a new serial console connection to the specified instance. Once the serial console connection is created and is available, you connect to the serial console using an SSH client.

```
oci compute instance-console-connection create --instance-id <instance_OCID>
 --ssh-public-key-file public_SSH_key_path [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Procedure

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 compartment_OCID --
   instance-id instance_OCID
   ```
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.

4. When you're finished using this instance console connection, use the following command to delete the connection.

```
$ oci compute instance-console-connection delete --instance-console-
connection-id instance_console_connection_OCID
```

What's Next

Continue to Connecting to the VNC Console or Making a Local Connection to the Serial Console.

Using the API

This task can't be performed using the API.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Backing Up and Restoring an Instance

On Compute Cloud@Customer, supports backing up and restoring instances. The instance backup is created in an Object Storage bucket. From there, you can copy it to another server in your data center for safekeeping. When needed, you can import the backup into any Compute Cloud@Customer Object Storage bucket, and use it to create instances.

Use Cases

- Back up instances and any attached block volumes.
- Store the backups on another server for safekeeping.
- Restore a faulty instance and any attached block volumes.
- Use the backup to create matching instances.
- Use the backup and restore feature to migrate instances to another tenancy, or to another Compute Cloud@Customer.

The restored instance has some of the same characteristics as the source instance. For example:

- The type and version of the OS.
- The OS configuration matches the OS configuration of the source instance. This includes things like OS user accounts, installed applications, and so on.
- The type of storage, high-performance or balanced-performance, matches the source instance.
- All the software on the block volumes is available after you attached the block volumes to the restored instance.

Some aspects of the restored instance differ from the source instance, such as:

- The restored instance and associated components, like boot and block volumes have unique OCIDs that don't match the source instance.
- The source instance user account SSH keys aren't included in the restored instance.
- While creating the restored instance, you can configure the instance in a different compartment, with a different name, shape, subnet, and all the other attributes that you configure during the launch.

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 on page 480 • Creating a Bucket on page 479
2.	Create the instance backup.	Creating an Instance Backup on page 345

No.	Task	Links
3.	Transfer the backup object from Object Storage to another system in your data center.	Transferring an Instance Backup to Another System on page 347

Task Map – Restoring an Instance From a Backup

The following tasks assume that you are restoring an instance from a backup that's on another system in your data center. If the backup is already on the Compute Cloud@Customer 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 on page 480 • Creating a Bucket on page 479
2.	Upload the backup to the bucket.	Transferring an Instance Backup from Another System to Compute Cloud@Customer on page 348
3.	Identify the backup OCID.	Listing Instance Backups on page 346
4.	Import the backup from the bucket into the Compute Cloud@Customer.	Importing an Instance Backup on page 348
5.	Finish restoring the instance by creating an instance using the imported instance backup.	Finishing the Instance Restore on page 349

Creating an Instance Backup

On Compute Cloud@Customer, you can use the Console and API to back up an instance.

The instance can be running or stopped. The boot volume and any block volumes must be attached.

> **Caution:**
>
> During the backup process, don't 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 on page 479.
- Quiesce instance activities such as running applications so that the backup is created at a known state.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. At the top of the page, select the compartment that contains the instance.
3. Perform **one** of the following actions:
   - Select the instance Actions menu (⋮), and select **Export**.
   - Select the name of the instance you plan to back up. Then select **Controls** (upper right) and select **Export**.
4. In the dialog box, select these items:

- If needed, change the compartment to the compartment where the bucket is located by selecting **(Change)**.
- Select the bucket.
- Select **Create Export**.

5. To see the status of an instance backup, under **Resources**, select **Instance Exports**.

Using the CLI

This task can't be performed using the CLI.

Using the API

The `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>"
}
```

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Listing Instance Backups

On Compute Cloud@Customer, you can list instance backups using the Console and API.

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 Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select 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 CLI

This task can't be performed using the CLI.

Using the 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.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Transferring an Instance Backup

On Compute Cloud@Customer, you can transfer instance backups to and from another system in your data center.

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 on page 484.

Using the CLI

1. Gather the information that you need to run the command.
   - Namespace name (see Obtaining the Object Storage Namespace on page 479)
   - Bucket name (`oci os bucket list`), see Listing Buckets on page 480
   - Object name (`oci os object list`), see Viewing Objects in a Bucket on page 484
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 path for the file being downloaded, such as `C:\workspace\backups\ocid1.instance.uniqueID` or `/home/downloads/backups/ocid1.instance.uniqueID`.

   Example:

   ```
   oci os object get  \
   --namespace-name mytenant  \
   --bucket-name my-backup-bucket  \
   --name ocid1.instance.........uniqueID  \
   --file /home/downloads/backups/ocid1.instance.........uniqueID
   ```

```
Downloading object  [########################-----------]   68%  00:00:05
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Transferring an Instance Backup from Another System to Compute Cloud@Customer

Using the CLI

1. Gather the information that you need to run the command.
   - Namespace name. See Obtaining the Object Storage Namespace on page 479.
   - Bucket name (`oci os bucket list`). See Listing Buckets on page 480.
2. Upload the instance backup to an Object Storage bucket in the target Compute Cloud@Customer.

   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"
   ```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Restoring an Instance from an Instance Backup

On Compute Cloud@Customer, 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

On Compute Cloud@Customer, importing an instance backup copies the backup from an Object Storage backup to a location that's internal.

For a particular 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 on page 349.

Prerequisites

- You must have an instance backup in an Object Storage bucket.

  If needed, transfer the backup to Compute Cloud@Customer. See Transferring an Instance Backup from Another System to Compute Cloud@Customer on page 348.

Finishing the Instance Restore

On Compute Cloud@Customer, you can create new instances from instance backups.

Perform the following 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 on page 316). 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 on page 387.

Deleting an Instance Backup

On Compute Cloud@Customer, 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 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.

Instance Pools

On Compute Cloud@Customer, instance pools simplify the management of compute instances. An instance pool defines a set of compute instances that's managed as a group. Managing instances as a group enables you to efficiently provision instances and manage the state of instances.

Creating an instance pool requires an instance configuration and a placement configuration. If you update the instance configuration or placement configuration, existing instances aren't affected. New instances that are added to the pool use the new instance and placement configurations.

> **Caution:**
>
> 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.

You can define an autoscaling configuration and policies that scale the instance pool in (remove instances) or out (add instances) or stop, start, or reboot the instances according to a schedule. Autoscaling a pool enables you to use resources even more effectively by stopping or removing instances when demand is lower and starting or adding instances when demand is higher. You can use instance pool autoscaling along with network load balancing by defining the backend set of servers to be an autoscaled instance pool.

When working with compute instance configurations and compute instance pools, keep the following points in mind:

- You can't delete a compute instance configuration if it's associated with at least one compute instance pool.
- You can use the same compute instance configuration for multiple compute instance pools. However, a compute instance pool can have only one compute instance configuration associated with it.
- If you modify the compute instance configuration for a compute instance pool, existing compute instances that are part of that pool don't change. Any new compute instances that are created after you modify the compute instance configuration will use the new compute instance configuration. New compute instances will not be created unless you have increased the size of the compute instance pool or terminate existing compute instances.
- If you decrease the size of a compute instance pool, the oldest compute instances are deleted first.

Creating an Instance Pool

On Compute Cloud@Customer, you can create an instance pool of instances that are within the same region.

Performing operations such as reset or delete on the pool object performs that operation on all instances that are members of the pool. Performing these operations on an individual instance that's a member of the pool doesn't 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 don't 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 can't 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 can't launch because of resource constraints, those instances remain in the Provisioning state and the pool remains in the Scaling state. After `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 Domain specification in the Console 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 isn't a member of a pool in the same fault domain where the pool instance is failing to launch because of resource constraints.
- Delete an instance that isn't a member of a pool in the same fault domain where the pool instance is failing to launch because of resource constraints.

Prerequisite

Before you can create an instance pool, you need an instance configuration. An instance configuration is a template that defines the settings to use when creating instances. See Working with Instance Configurations on page 333.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instance Configurations**.
2. At the top of the page, select the compartment that contains the instance configuration that you want to use to create the pool.
3. Select the instance configuration that you want to use for the instances in this pool.
4. Under **Resources**, select **Attached Instance Pools**.
5. At the top of the page, select other compartments to list pools in other compartments.

   Select **Create Instance Pool**.
6. In the **Attach Instance Pool to *`instance_configuration_name`*** dialog box, enter the following information:

   - **Name**: Enter a name for the instance pool. The name doesn't need to be unique. This name is used in the names of the created instances. If you don't 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's 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**: Select the **Attach Load Balancers** box to specify load balancing for this pool. For information about load balancing, see Load Balancer as a Service on page 211. 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, select **Add Load Balancer**. To attach a load balancer after the instance pool is created, see Managing Instance Pool Load Balancer Attachments on page 365.
   - **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
7. Select **Create Instance Pool**.

   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're 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 have the new name.

   Select **Work Requests** in the **Resources** box to check the status of the instance pool create.

Using the Console

Use the oci compute-management instance-pool create command and required parameters to create an instance pool.

```
oci compute-management instance-pool create --compartment-
id <compartment_OCID> --instance-configuration-
id <instance-configuration_OCID> --placement-
configurations <placement_configurations.json> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Procedure

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 are created in the compartment 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 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 on page 365.

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 isn't required to be unique. If you don't 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 isn't 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 the API

Use the CreateInstancePool operation to create an instance pool.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Using Schedule-Based Autoscaling

On Compute Cloud@Customer, 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 on page 362.

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 on page 362 and Stopping and Starting Instances in an Instance Pool on page 367, 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 CLI 3.15.1 or newer and Oracle Cloud Infrastructure Python SDK 2.80.0 or newer.

Multiple Schedule Management

On Compute Cloud@Customer, 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 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

On Compute Cloud@Customer, 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 Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Autoscaling Configurations**.
2. Select **Create Autoscaling Configuration**.
3. In the **Create Autoscaling Configuration** dialog box, 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. Clear 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 on page 357.

     To add another policy, select **Add Policy**. You can also add policies after the autoscaling configuration is created, as described in Creating a Schedule-Based Autoscaling Policy on page 357.

     To delete a policy, select the trash can icon for that policy.
   - **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
   - Select **Submit**.

     The details page for the new autoscaling configuration is displayed.

     On the details page, ignore the Cooldown Period value. Cooldown period doesn't apply to schedule-based autoscaling configurations.

     The new autoscaling configuration is enabled by default. To disable the configuration, see Updating an Autoscaling Configuration on page 360.

Using the Console

Use the oci autoscaling configuration create command and required parameters to create an autoscaling configuration.

```
oci autoscaling configuration create --compartment-id compartment_OCID --
from-json file://input_file.json [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Procedure

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 on page 357.

   The optional display name is 1-255 characters, doesn't need to be unique, and can be updated. If you don't 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:**
   >
   > Don't 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 aren't 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.

Using the API

Use the AutoScalingConfiguration operation to create an autoscaling configuration.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Creating a Schedule-Based Autoscaling Policy

On Compute Cloud@Customer, you can create schedule-based autoscaling policies.

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 Quartz cron expression generator to verify your schedule expressions.
- Ensure that policy schedules don't conflict. See the descriptions in Multiple Schedule Management on page 353 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 is not retried.
- Was the policy set to scale out, but not enough resources were available?

  The scale policy sets the pool size, and the pool continues to try to reach that size as resources become available.
- Is the operation specified by the policy still executing or waiting to run?
  - 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 tries to run while a state change operation is already running on the same pool, the second operation fails to run.
  - A limited number of pools can be changing state concurrently. If too many other pools are already changing state, then the pool must wait to begin changing state. The time to change state is longer when more instances are involved because 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 because instances are deleted or created serially. Delete 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 Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Autoscaling Configurations**.

2. At the top of the page, select the compartment that contains the configuration to which you want to add a policy.
3. Select the name of the autoscaling configuration to which you want to add a policy.
4. Under **Resources**, select **Autoscaling Policies**, then select **Create Scheduled Policy**.
5. In the **Create policy** dialog box, 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 scales to at the scheduled time.
     - **Change lifecycle state of all instances:** Select the state that all instances in the pool transition to at the scheduled time.
     - **Enable Schedule:** By default, the **Schedule Enabled** box is selected to enable the policy to run at the next scheduled time. Clear the box to disable this policy.
   - **Execution schedule:** Define the schedule for implementing this autoscaling policy. See Designing Policies on page 357.
6. Select **Submit**.

Using the Console

Use the oci autoscaling policy create command and required parameters to create an autoscaling policy for the specified autoscaling configuration.

```
oci autoscaling policy create  --auto-scaling-configuration-
id autoscaling_configuration_OCID  --from-json
 file://policy_definitions.json --policy-type scheduled [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Procedure

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:**
   >
   > Don't specify values for `capacity` `min` or `max`. These properties don't apply to schedule-based autoscaling configurations.

   The default values for `capacity` `min` and `max` appear in the created autoscaling policy but aren't used for schedule-based autoscaling.

   The display name is 1-255 characters, doesn't need to be unique, and can be updated.

   Use the references in Creating a Schedule-Based Autoscaling Policy on page 357 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.

   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.

Using the API

See the AutoScalingPolicy Reference for API operations that define the criteria that trigger autoscaling actions and the actions to take.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Updating an Autoscaling Configuration

On Compute Cloud@Customer, 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 on page 357 and Updating a Schedule-Based Autoscaling Policy on page 360.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Autoscaling Configurations**.
2. At the top of the page, select the compartment that contains the autoscaling configuration that you want to update.
3. For the autoscaling configuration that you want to update, select the Actions menu (⋮) and select **Edit**.
4. In the **Update Autoscaling Configuration** dialog box, make the changes. Ignore the Cooldown Period value.
5. Select **Submit**.

Using the Console

Use the oci autoscaling configuration update command and required parameters to update certain fields on the specified autoscaling configuration, such as the name, and whether the autoscaling configuration is enabled.

```
oci autoscaling configuration update --auto-scaling-configuration-
id autoscalingConfiguration_OCID  [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

See the AutoScaling API Reference for API operations that enable you to dynamically scale compute resources to meet application requirements.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Updating a Schedule-Based Autoscaling Policy

On Compute Cloud@Customer, you can update a autoscaling policy display name, either the target pool size or the lifecycle action, and the execution schedule. You can also enable or disable the policy.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Autoscaling Configurations**.
2. At the top of the page, select the compartment that contains the configuration for which you want to update a policy.
3. Select the name of the autoscaling configuration for which you want to update a policy.
4. Under **Resources**, select **Autoscaling Policies**.
5. For the policy that you want to update, select the Actions menu (⋮) and then select **Edit**.

6. In the **Update Policy** dialog box, change the policy information.
7. Select **Submit**.

Using the Console

Use the oci autoscaling policy update command and required parameters to update an autoscaling policy in the specified autoscaling configuration.

```
oci autoscaling policy update  --auto-scaling-configuration-
id <autoscaling_config_OCID> --auto-scaling-policy-id <policy_OCID> --
policy-type <policy_type> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

See the AutoScalingPolicy Reference for API operations that define the criteria that trigger autoscaling actions and the actions to take.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting an Autoscaling Configuration

On Compute Cloud@Customer, you can delete an autoscaling configuration.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Autoscaling Configurations**.
2. At the top of the page, select the compartment that contains the configuration that you want to delete.
3. For the autoscaling configuration that you want to delete, select the Actions menu (⋮), then select **Delete**.

Using the Console

Use the oci autoscaling configuration delete command and required parameters to delete an autoscaling configuration.

```
SYNTX [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

See the AutoScaling API Reference for API operations that enable you to dynamically scale compute resources to meet application requirements.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting an Autoscaling Policy

On Compute Cloud@Customer, you can delete an autoscaling policy.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Autoscaling Configurations**.
2. At the top of the page, select the compartment that contains the autoscaling configuration from which you want to delete a policy.
3. Select the name of the autoscaling configuration for which you want to delete a policy.
4. Under **Resources**, select **Autoscaling Policies**.
5. For the policy that you want to delete, select the Actions menu (⋮) and then select **Delete**.

Using the Console

Use the oci autoscaling policy delete command and required parameters to delete an autoscaling policy for the specified autoscaling configuration.

```
oci autoscaling policy delete --auto-scaling-configuration-
id <autoscaling_configuration_OCID>  --auto-scaling-policy-
id <autoscaling_policy_id> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

See the AutoScalingPolicy Reference for API operations that define the criteria that trigger autoscaling actions and the actions to take.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Updating an Instance Pool

On Compute Cloud@Customer, when you update an instance pool, you can change the name of the pool, the size of the pool, the instance configuration that's 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 on page 363 and Detaching an Instance from an Instance Pool on page 364.

To attach load balancers or detach load balancer attachments, see Managing Instance Pool Load Balancer Attachments on page 365.

Configuration changes don't 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 created evenly across the fault domains specified by the instance configuration or the placement configuration.

If you decrease the size of the pool, instances are deleted evenly across the fault domains specified by the instance configuration or the placement configuration. In each fault domain, instances are deleted in creation date order, oldest first.

You can't select which instances to delete when you decrease the size of a pool. If you delete an individual instance that's a member of a pool, as described in Deleting an Instance on page 329, 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 can't 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 on page 350.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instance Pools**.
2. At the top of the page, select the compartment that contains the instance pool that you want to update.
3. For the instance pool that you want to update, select the Actions menu (⋮), then select **Edit**.
4. In the **Update Instance Pool** dialog box, make the changes.
5. Select **Update Instance Pool**.

Using the Console

Use the oci compute-management instance-pool update command and required parameters to update the specified instance pool.

```
oci compute-management instance-pool update --instance-pool-
id instance_pool_OCID options_with_values_to_update[OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateInstancePool operation to update the specified instance pool.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Attaching an Instance to an Instance Pool

On Compute Cloud@Customer, 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's large enough for the expanded pool. The next time the scheduled autoscaling policy runs, the target pool size is reset to the value that's set in the policy; if the policy size is smaller than the current size, instances are 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 isn't 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's in a fault domain that's not included in the pool instance configuration, or is using a VCN and subnet that aren't 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 on page 362.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instance Pools**.
2. At the top of the page, select the compartment that contains the instance pool.
3. Select the name of the instance pool to which you want to attach an instance.
4. On the instance pool details page, under **Resources**, select **Attached Instances**.
5. Select **Attach Instance** on the **Attached Instances** table.
6. In the **Input type** field of the **Attach Instance** dialog box, 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's 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 box.

     If an instance that you think should be attachable isn't shown, 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. Select **Attach**.

   Even if the criteria listed in the dialog are met, the instance could fail to attach for some other reason. Under **Resources**, select **Work Requests**, then select the applicable work request in the list to troubleshoot any problems.

Using the Console

Use the oci compute-management instance-pool-instance attach command and required parameters to attach an instance to an instance pool.

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 <instance_pool_OCID> --instance-id <instance_ocid>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

This operation can't be performed using the API.

Detaching an Instance from an Instance Pool

On Compute Cloud@Customer, 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 Console, 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 Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instance Pools**.
2. At the top of the page, select the compartment that contains the instance pool.
3. Select the name of the instance pool from which you want to detach an instance.
4. On the **Instance Pool** details page, under **Resources**, select **Attached Instances**.
5. For the instance that you want to detach, select the Actions menu (⋮), then select **Detach**.
6. (Optional) To delete the instance and its boot volume, select the button under "Permanently terminate (delete) this instance and its attached boot volume."
7. Select **Confirm**.

   The pool size is decreased.

Using the Console

Use the oci compute-management instance-pool detach-lb command and required parameters to detach an instance from an instance pool.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Procedure

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 don't 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.

Using the API

This operation can't be performed using the API.

Managing Instance Pool Load Balancer Attachments

On Compute Cloud@Customer, you can attach a load balancer to an instance pool or detach a load balancer attachment from an instance pool.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instance Pools**.
2. At the top of the page, select the compartment that contains the instance pool for which you want to manage load balancer attachments.
3. Select the name of the pool that you want to manage.
4. Under **Resources**, select **Load Balancers**.
   - To attach a load balancer, select **Attach Load Balancer**.

     On the **Attach Load Balancer** dialog box, specify the load balancer, backend set, port number, and VNIC as described in Creating an Instance Pool on page 350, and select **Attach Load Balancer**.

- To remove a load balancer attachment, select the Actions menu (⋮) for the load balancer attachment that you want to remove, and select **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 Console

Use the oci compute-management instance-pool attach-lb command and required parameters to attach a load balancer to the instance pool.

```
oci compute-management instance-pool attach-lb --backend-set-
name <backend-set-name> --instance-pool-id <instance-pool_OCID> --load-
balancer-id <load-balancer_OCID> --port <port_for_backend_set> --vnic-
selection <VNIC_display_name> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Procedure

- **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's 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
     ```

- **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.

Using the API

Use the following operations to attach or detach a load balancer to an instance pool.

- AttachLoadBalancer

- DetachLoadBalancer

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Stopping and Starting Instances in an Instance Pool

On Compute Cloud@Customer, 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's a member of the pool doesn't affect any other member instances.

When instances are stopped, 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're stopped, and configuration of stopped pool instances is preserved. Configuration changes such as fault domain changes don't apply to pool instances that are restarted or reset.

When you perform an operation on an instance pool, all the instances in the pool are stopped, started, or rebooted. See Stopping, Starting, and Resetting an Instance on page 328 for how to prepare for an instance to stop. Stopping and starting an instance can take up to five minutes.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instance Pools**.
2. At the top of the page, select the compartment that contains the instance pool that you want to manage.
3. For the instance pool that you want to manage, select the Actions menu (⋮), and select either **Start**, **Stop**, or **Reboot**.

   By default, selecting Stop selects Soft Stop and selecting Reboot selects Soft Reboot. To stop or reboot all instances in a pool immediately, select the **Force** option on the confirmation dialog.
4. Confirm the action.

   Under **Resources**, select **Work Request(s)** to check the status of the instance pool stop, start, or reboot. Select **Attached Instances** to view the status of the instances.

Using the Console

Use the oci compute-management instance-pool commands and required parameters to manage the state of instanced in an instance pool.

```
oci compute-management instance-pool {start | stop | reset | softreset}
 [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use these operations to manage the state of instanced in an instance pool:

- StartInstancePool
- StopInstancePool
- ResetInstancePool

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting an Instance Pool

On Compute Cloud@Customer, 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.

> **Caution:**
>
> 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.

After the instance pool deletion, instances that had been attached to this pool remain visible in the instance list in a terminated state for at least 24 hours, up to 24.5 hours.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instance Pools**.
2. At the top of the page, select the compartment that contains the instance pool that you want to delete.
3. For the instance pool that you want to delete, select the Actions menu (⋮), then select **Delete**.
4. Select **Confirm**.

   All the instances in the pool are deleted. Deleted instances aren't attached and therefore aren't listed in **Attached Instances** under **Resources**.

   To check the status of the instance pool delete, Under **Resources**, select **Work Requests**.

Using the Console

Use the oci compute-management instance-pool terminate command and required parameters to delete an instance pool.

```
oci compute-management instance-pool terminate --instance-pool-id <instance-pool_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the TerminateInstancePool operation to delete an instance pool.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Container Instances

On Compute Cloud@Customer, a Container Instance is a serverless compute service that enables you to quickly and easily run containers without managing any servers. Container Instances run your containers on minimal compute instances that are optimized for container workloads.

A container instance is a minimal instance created from an image that runs only what is needed for containers. A container instance has no boot volume. A single user-specified container is started in the container instance.

Resources are created as needed. A container instance creates a 15 gigabyte ephemeral writable Data Volume. Persistent storage isn't supported for container instances.

Container Instance Shapes

On Compute Cloud@Customer, the Container Instance shapes are flexible shapes: You can specify the number of OCPUs and amount of memory when you select the shape.

Both Container Instances shapes have the same specifications, as shown in the following table. The difference between the two shapes is the following:

- The CI.Standard.x86.Generic shape can run on either Intel or AMD processors.
- The CI.Standard.E5.Flex shape runs on the AMD EPYC 9J14 96-Core processor.

The maximum number of VNICs is 1 because each container instance has just one container.

Shape	OCPUs	Memory (GB)	Maximum VNICs	Maximum Bandwidth (Gbps)
CI.Standard.x86.Generic CI.Standard.E5.Flex	1–64	Default: 3 GB per OCPU 64 GB maximum per OCPU 1024 GB maximum per container instance	1 VNIC	1 to 40 OCPUs: 1 Gbps per OCPU 41 to 64 OCPUs: 40.0 Gbps

Working with Container Instances

Learn how to create a container instance On Compute Cloud@Customer, 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 Compute Cloud@Customer 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 (see Container Instance Shapes on page 369)
- The VCN and subnet

Container instances leverage 15 GB of ephemeral storage that's shared between the container instance and the container. This ephemeral storage is created automatically when the container instance is created. You can't manually create storage and attach it to the container instance.

Using the Console

1. On the Compute Cloud@Customer Console **Dashboard**, select **Containers / Container Instances**.
2. Select **Create Container Instance** above the Container Instances list.
3. In the **Create Container Instance** dialog box, 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.
     - Don't 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 can't 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.

  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 selected by default. Clear the box if you don't 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.
  - **Skip Source Destination Check**: (Optional) Skip source destination check.
- **Container Image URL**: Enter the container image URI. The container image must come from a registry that offers a certificate from a well know certificate authority (CA). Registries that offer self-signed certificates or certificates signed by a private CA are not supported at this time. The container image manifest must be V2 schema 2.
- **Environmental Variables**: You can set the environmental variables used by the container. Container images support environment variables to customize their execution. For example, the official NGINX image supports NGINX_HOST and NGINX_PORT environment variables, so you can set the value to customize the execution to the following variables:
  - NGINX_HOST=foobar.com
  - NGINX_PORT=80
- **Command Arguments**: You can specify the actual command or entrypoint the container runs upon starting, allowing you to start the main application, pass specific flags, or run custom scripts. Setting this value overrides or customizes the default command in the container image.
- **Memory Limit**: You can set the maximum amount of RAM allocated to the shape configuration for the instance.
- **vCPU Limit**: You can set the maximum number of virtual CPUs the shape configuration for the instance can utilize. This bounds the computational resources available.
- **Security**: You can specify the security setting of the container.
  - Use the User ID and Group ID fields to set the user ID (ID) and group ID (GID) to run the entrypoint process of the container.
    - The value for User ID and Group ID must be an integer between 0 and 65535. The default value is 0.
    - The User ID and Group ID values that you specify override values that are set in the container image. When the User ID value is not specified, the entrypoint process of the container runs as root user.
    - You must set the User ID before you set the Group ID.
  - **Configure Linux capabilities**: You can configure Linux capabilities for your container. By default, the container launches with several capabilities that you can choose to drop.

    Both the Add capabilities field and Drop capabilities field support the ALL value, which allows or drops all capabilities. The value ALL refers to the Linux capabilities that are enabled by default.

- If you leave both Add capabilities and Drop capabilities blank, all default capabilities are available for your container.
- If you enter ALL in the Add capabilities field, all default capabilities are available except those that you list in the Drop capabilities field, and ignores ALL in the Drop capabilities field.
- If you enter ALL in the Drop capabilities field, the container contains only the default capabilities that you list in the Add capabilities field.
- For other cases, drops all the capabilities from the default set listed in the Drop capabilities field, then adds the capabilities listed in the Add capabilities field, and finally returns the result as the capability for your container.
- The creation of the container instance fails if you provide any capabilities that are not in the list below.

Capability	**Description**
CAP_AUDIT_WRITE	Writes records to kernel auditing log.
CAP_CHOWN	Makes changes to file UIDs and GIDs.
CAP_DAC_OVERRIDE	Discretionary access control (DAC) that bypasses file read, write, and execute permission checks.
CAP_FOWNER	Bypasses permission checks on operations that normally require the file system UID of the process to match the UID of the file, excludes the operations that are covered by CAP_DAC_OVERRIDE and CAP_DAC_READ_SEARCH.

Capability	Description
CAP_FSETID	• Does not clear set-user-ID and set-group-ID mode bits when a file is modified. • Sets the set-group-ID bit for a file whose GID does not match the file. system or any of the supplementary GIDs of the calling process.
CAP_KILL	Bypasses permission checks for sending signals, which includes use of the ioctl(2) KDSIGACCEPT operation.
CAP_MKNOD	Creates special files using mknod(2).
CAP_NET_BIND_SERVICE	Binds a socket to internet domain privileged ports (port numbers less than 1024).
CAP_NET_RAW	• Uses RAW and PACKET sockets. • Binds to any address for transparent proxying.
CAP_SETFCAP	Sets file capabilities.

Capability	Description
CAP_SETGID	• Makes manipulations of process GIDs and supplementary GID list. • Forges GID when passing socket credentials via UNIX domain sockets. • Writes a group ID mapping in a user namespace.
CAP_SETPCAP	If the file capabilities are not supported, grants or removes any capability in the caller's permitted capability set to or from any other process.
CAP_SETUID	• Makes manipulations of process UIDs. • Forges UID when passing socket credentials via UNIX domain sockets. • Writes a user ID mapping in a user namespace.

Capability	Description
CAP_SYS_CHROOT	Uses chroot(2) to change to a different root directory.

- **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.

4. Select **Create Container Instance**.

   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`*.

Using the CLI

Use the oci container-instances container-instance create command and required parameters to create a container instance.

```
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 [OPTIONS]
```

Example:

```
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"}]'
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Procedure

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.

     ```
     $ 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 Console 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
```

Using the API

Use the CreateContainerInstance operation to create a container instance.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing a Container Instance

Learn how to list all container instances in a compartment and how to view more information about a specified container instance On Compute Cloud@Customer.

Using the Console

1. On the Compute Cloud@Customer Console **Dashboard**, select **Containers / Container Instances**.

2. If the container instance that you want to view isn't listed, use the Compartment drop-down menu above the container instances list to select the correct compartment.
3. 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.
4. On the container instance details page, scroll to the **Resources** section, and select **Containers** to view information about the container instance container.

Using the CLI

Use the oci container-instances container-instance list command and required parameters to list all container instances in a compartment.

```
oci container-instances container-instance list --container-instance-
id <container_instance_OCID> [OPTIONS]
```

Use the oci container-instances container-instance get command and required parameters to get information about a specified container instances.

```
oci container-instances container-instance list --container-instance-
id <container_instance_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the GetContainerInstance operation to get information about a specified container instances.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Updating a Container Instance

On Compute Cloud@Customer you can change the name and tags of a specified container instance.

Using the Console

1. On the Compute Cloud@Customer Console **Dashboard**, select **Containers / Container Instances**.
2. If the container instance that you want to view isn't listed, use the Compartment drop-down menu above the container instances list to select the correct compartment.
3. In the container instances list, for the container instance that you want to update, select the Actions menu (⋮) and select **Edit**.
4. When you're finished making your changes, select **Save Changes**.

Using the CLI

Use the oci container-instances container-instance update command and required parameters to update a container instance.

```
oci container-instances container-instance update --container-instance-
id <container-instance_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateContainerInstance operation to update a container instance.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Moving a Container Instance

On Compute Cloud@Customer, you can move a container instance to a different compartment.

To move a container instance, you must use the CLI or API.

Using the Console

This task can't be performed using the Console.

Using the CLI

Use the oci container-instances container-instance change-compartment command and required parameters to move a container instance to a different compartment.

```
oci container-instances container-instance change-compartment -compartment-
id <container-instance_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ChangeContainerInstanceCompartment operation to move a container instance to a different compartment.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Stopping, Starting, and Restarting a Container Instance

On Compute Cloud@Customer you can stop, start, and restart a container instance.

Using the Console

1. On the Compute Cloud@Customer Console **Dashboard**, select **Containers / Container Instances**.
2. If the container instance that you want to view isn't listed, use the Compartment drop-down menu above the container instances list to select the correct compartment.
3. For the container instance that you want to start, stop, or restart, select the Actions menu (⋮), and select **Start**, **Stop**, or **Restart**.

   Alternatively, in the container instance list, select the name of the container instance to display the details page for that container instance. Select **Controls**, then select **Start**, **Stop**, or **Restart**.
4. Confirm the action.

   To check the status of the action, under **Resources**, select **Work Request(s)**.

Using the CLI

Use the following commands to stop, start, and restart a container instance:

- Use the oci container-instances container-instance stop command and required parameters to stop a container instance.

  ```
  oci container-instances container-instance stop --container-instance-
  id <container-instance_OCID> [OPTIONS]
  ```

- Use the oci container-instances container-instance start command and required parameters to start a container instance.

  ```
  oci container-instances container-instance start --container-instance-
  id <container-instance_OCID> [OPTIONS]
  ```

- Use the oci container-instances container-instance restart command and required parameters to restart a container instance.

```
oci container-instances container-instance restart --container-instance-
id <container-instance_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the following API operations to stop, start, and restart a container instance:

- Use the StopContainerInstance operation to stop a container instance.
- Use the StartContainerInstance operation to start a container instance.
- Use the RestartContainerInstance operation to restart a container instance.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Container Instance

On Compute Cloud@Customer, you can delete 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.

> **Important:**
>
> When the containers are deleted by the service, any data on the ephemeral storage is lost.

Using the Console

1. On the Compute Cloud@Customer Console **Dashboard**, select **Containers / Container Instances**.
2. If the container instance that you want to view isn't listed, use the Compartment drop-down menu above the container instances list to select the correct compartment.
3. For the container instance that you want to delete, select the Actions menu (⋮), and select **Delete**.

   Alternatively, in the container instance list, select the name of the container instance to display the details page. Select **Controls**, then select **Delete**.
4. Confirm the deletion.

   To check the status of the container instance deletion, under **Resources**, select **Work Request(s)**. Deleted container instances temporarily remain in the list of container instances with the state **Deleted**.

Using the CLI

Use the oci container-instances container-instance delete command and required parameters to delete a container instance.

```
oci container-instances container-instance delete --container-instance-
id <container-instance_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteContainerInstance operation to delete a container instance.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Working with Containers in Container Instances

On Compute Cloud@Customer, you can 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

On Compute Cloud@Customer, you can view containers.

Using the Console

1. On the Compute Cloud@Customer Console **Dashboard**, select **Containers / Container Instances**.
2. If the container instance that you want to view isn't listed, use the Compartment drop-down menu above the container instances list to select the correct compartment.
3. In the container instances list, select the name of the container instance for which you want to view the container.
4. Under **Resources**, select **Containers**.
5. Select the name of the container in the list to show more information about the container, including log data.

Using the CLI

Use the oci container-instances container list command and required parameters to list containers.

```
oci container-instances container list --compartment-id <compartment_OCID>
 [OPTIONS]
```

Use the oci container-instances container get command and required parameters to get the details for a specified container.

```
oci container-instances container get --container-id <container_OCID>
 [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the GetContainer operation to get the details for a specified container.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Updating a Container Instance Container

On Compute Cloud@Customer, you can change the name and tags of a specified container instance container.

Using the Console

1. On the Compute Cloud@Customer Console **Dashboard**, select **Containers / Container Instances**.
2. If the container instance that you want to view isn't listed, use the Compartment drop-down menu above the container instances list to select the correct compartment.
3. In the container instances list, select the name of the container instance for which you want to update the container.
4. Under **Resources**, select **Containers**.
5. For the container in the list, select the Actions menu (⋮) and select **Edit**.

Using the CLI

Use the oci container-instances container update command and required parameters to update a container.

```
oci container-instances container update --container-
id <container_OCID>[OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateContainer operation to update a container.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Block Volume Storage

On Compute Cloud@Customer, Block Volumes provide high-performance network storage capacity that supports a broad range of I/O intensive workloads.

Block Volume Overview

A block volume is a detachable block storage device that enables you to dynamically expand the storage capacity of your compute instances, provide durable and persistent data storage that can be migrated across compute instances, and host large databases.

The Block Volume service enables you to group multiple volumes in a volume group. Volume groups simplify the process to create backups and clones.

You can create, attach, connect, and move volumes, and change volume performance to meet your storage, performance, and application requirements.

After you attach and connect a volume to a compute instance, you can use the volume like a regular hard drive. You can also disconnect a volume and attach it to another compute instance without the loss of data.

The default size for block volumes is 1024 GB (1TB). Block volume size range is 50 GB to 32768 GB (32 TB).

Types of Block Volumes

- **Boot volume**: A detachable boot volume device that contains the image that's used to boot a Compute instance.
- **Block volume**: A detachable block storage device that lets you to dynamically expand the storage capacity of an instance.

You can create, attach, connect, and move volumes, and change volume performance to meet your storage, performance, and application requirements.

After you attach and connect a volume to a compute instance, you can use the volume like a regular hard drive. You can also disconnect a volume and attach it to another compute instance without the loss of data.

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.

> **Note:**
>
> The disk space consumed by a LUN is larger than the thick provisioned volume size, because overhead and metadata also occupy space within the storage pool.

Required Components

These components are required to create a volume and attach it to a compute instance:

- **Compute Instance**: A virtual machine (VM) running in the Compute Cloud@Customer.
- **Volume attachment**: A paravirtualized attachment that's available for compute instances.
- **Volume**: A block volume or boot volume.

Volume Access Types

When you attach a block volume, you can specify one of the following access types:

- **Read/write:** This is the default option for volume attachments. With this option, an instance can read and write data to the volume.
- **Read/write, shareable:** With this option, you can attach a volume to more than one instance at a time and those instances can read and write data to the volume. To prevent data corruption from uncontrolled read/write operations with multiple instance volume attachments you must install and configure a cluster-aware solution before you can use the volume.
- **Read-only:** With this option, an instance can only read data on the volume. It can't update data on the volume. Specify this option to safeguard data against accidental or malicious modifications.

The access type for boot volumes is always read/write.

Block Volume Performance Options

Elastic Performance

When you create block storage, you can optionally enable High performance. By default, the volume is created with balanced performance. The volume performance setting can't be changed after the volume is created.

- **High performance**: The High performance option is recommended for workloads with higher I/O requirements.
- **Balanced performance**: The Balanced performance option is the default for new and existing block and boot volumes. This performance option provides a good balance between performance and cost savings for most workloads, including workloads that perform random I/O such as boot volumes.

Performance Limitations and Considerations

The following performance results are for unformatted data volumes.

- Throughput performance on compute instances depends on the network bandwidth that's available to the compute instance, and further limited by that bandwidth for the volume.
- If Microsoft Defender Advanced Threat Protection (Microsoft Defender ATP) is enabled in an instance, it has a significant negative impact on disk I/O performance. However, consider the security implications of disabling Microsoft Defender ATP.
- Block volume performance is per volume, so when a block volume is attached to multiple compute instances, the performance is shared across all the attached compute instances.

Block Volume Scenarios

Adding Storage Capacity

A common use of a block volume is to add storage capacity to an instance. After you create an instance and set up a cloud network, you can create a block storage volume. Then, you attach the volume to a compute instance using a volume attachment. The volume can then be mounted and used by your compute instance.

Moving a Volume to Another Compute Instance

A block volume can be detached from a compute instance and moved to a different compute instance without the loss of data. This data persistence lets you to migrate data between compute instances and ensures that data is safely stored, even when it's not connected to a compute instance. Any data remains intact until you delete the volume.

To move the volume to another compute instance, unmount the drive from the initial compute instance, detach the block volume, then attach the volume to another compute instance. From there, you connect and mount the drive from that instance's guest OS to have access to the data.

Scaling a Compute Instance

When you delete a compute instance, you can keep the associated boot volume and use it to create a new compute instance with a different compute instance type or shape. This capability lets you to easily scale up or scale down the number of cores for a compute instance.

Volume Backups and Clones

Backup and Restore

Backup and restore operations are supported on boot volumes, data volumes, and volume groups. All backups are full backups, not incremental.

To back up a boot volume, block volume, or volume group, use one of the following methods:

- **Clone**: Cloning volumes enables you to create a copy without performing the backup and restore operations. For more information about clones, see the "Clones" section and "Differences Between Volume Clones and Backups" table below. For general information about cloning boot volumes, see Cloning a Boot Volume on page 401.
- **Manual backup**: These backups are performed one time as soon as you create the backup. These backups are retained indefinitely, or until a policy-based (scheduled) backup is created for this volume.
- **Scheduled backups**: These backups are performed periodically according to a schedule defined in a backup policy. The policy schedule specifies the frequency (period) and time of the backups, and how long to retain the backups. You can create your own backup policies, or you can use Oracle provided backup policies.

Oracle Provided Backup Policies

Oracle defined backup policies cannot be used to back up a volume group. These policies can be used to back up individual block volumes and boot volumes.

Oracle defined policies cannot be changed. If you need a different backup time or retention time, for example, create a user defined backup policy as described in Managing Backup Policies on page 426.

The following backup policies are provided by Oracle and available in every compartment. A resource can have only one backup policy assigned, but a backup policy can have multiple schedules. The Oracle Bronze policy has one schedule, the Silver policy has two schedules, and the Gold policy has three schedules. All schedule times are your regional data center time. All backup types are full backups.

Bronze

1. Monthly backups that run at 00:00 on the first day of the month and are retained for twelve months.

Silver

1. Weekly backups that run at 00:00 every Monday and are retained for four weeks.
2. Monthly backups that run at 00:00 on the first day of the month and are retained for twelve months.

Gold

1. Daily backups that run at 00:00 and are retained for seven days.
2. Weekly backups that run at 00:00 every Monday and are retained for four weeks.
3. Monthly backups that run at 00:00 on the first day of the month and are retained for twelve months.

The Block Volume service will not run more than one scheduled backup of a particular resource in one day. When schedules conflict, for example, daily, weekly, and monthly backups are scheduled to run at the same time, the backup with the longest period runs.

Volume Group Back Up and Restore

You can perform most of the same backup operations and tasks with volume groups that you can perform with individual block volumes.

Volume group backups enable you to manage the backup settings for several volumes in one place. This feature simplifies the process to create time-consistent backups of running enterprise applications that span multiple storage volumes across multiple compute instances.

You can restore a volume group backup to a volume group, or you can restore individual volumes in the volume group from volume backups.

Clones

You can create a clone from a volume using the Block Volume service. Cloning enables you to make a copy of an existing block volume without needing to go through the backup and restore process.

The clone operation occurs immediately, and you can attach and use the cloned volume as a regular volume when the state changes to Available. At this point, the volume data is being copied in the background, and can take up to thirty minutes depending on the size of the volume.

There's a single point-in-time reference for a source volume while it's being cloned. If the source volume is attached when a clone is created, you need to wait for the first clone operation to complete from the source volume before creating additional clones. If the source volume is detached, you can create up to 10 clones from the same source volume simultaneously.

You can only create a clone for a volume within the same tenant. You can create a clone of a volume in a different compartment from the source volume compartment if you have the required access permissions.

Differences Between Volume Clones and Backups

Consider the following criteria when you decide whether to create a backup or a clone of a volume.

Comparison	Volume Backup	Volume Clone
Description	Creates a point-in-time backup of data on a volume. You can restore multiple new volumes from the backup later in the future.	Creates an immediately usable copy of a block volume without having to go through the backup and restore process.
Use Case	Retain a backup of the data in a volume, so that you can duplicate an environment later or preserve the data for future use. Meet compliance and regulatory requirements, because the data in a backup remains unchanged over time. Support business continuity requirements. Reduce the risk of outages or data mutation over time.	Creates an immediately usable copy of a block volume without having to go through the backup and restore process.
Storage Location	Block Volume	Block Volume
Retention Policy	Policy-based backups expire, manual backups don't expire.	No expiration
Volume Groups	Supported. You can back up a volume group.	Supported. You can clone a volume group.

Creating and Attaching Block Volumes

You can create and attach a block volume to an instance to expand the available storage on the instance. The topics in this section describe how to administer the Block Volume Storage service for Compute Cloud@Customer.

To create and attach block volumes to an instance, you perform a series of tasks.

Task Flow

No.	Task	Links
1.	Create a block volume.	Creating a Block Volume on page 384

No.	Task	Links
2.	Attach the block volume to one or more instances.	Attaching a Volume on page 387 or Attaching a Volume to Multiple Instances on page 388
3.	Identify the added block volume and perform administrative tasks.	Find Your Volume in the Instance on page 389
4.	Configure the volume to automatically mount when the instance is rebooted.	Find Your Volume in the Instance on page 389

Creating a Block Volume

On Compute Cloud@Customer, block volumes are detachable block storage devices that you can use to dynamically expand the storage capacity of an instance.

During the block volume creation, you can optionally use free-form tags to change the following block volume parameter:

Volume Block size

The default value is 8192 bytes. To change the value, specify a value in bytes for the `PCA_blocksize` free-form tag. Supported values are a power of 2 between 512 bytes and 1 megabyte, specified as a string and fully expanded. We recommend setting the value to at least 8192 bytes.

The block size can't be changed after the volume has been created.

See Using Free-Form Tags for Extended Functionality on page 120.

(Optional) To create a block volume with nondefault values for the following attributes, first configure the required tags as described in Using Free-Form Tags for Extended Functionality on page 120. Then use the following information when you create the block volume.

- **Synchronous Write Bias**

  You can configure the write cache flash devices for a share or LUN ("Logzilla"). The value can be changed later by updating the block volume.

  To use a nondefault value, during block volume creation, assign this defined tag:

  - **Tag namespace:** OraclePCA
  - **Tag name:** logBias
  - **Value** (select one): LATENCY (default) or THROUGHPUT
- **Secondary Cache**

  You can configure the use of the read cache flash devices for a share or LUN ("Readzilla"). The value can be changed later by updating the block volume.

  To use a nondefault value, during block volume creation, assign this defined tag:

  - **Tag namespace:** OraclePCA
  - **Tag name:** secondaryCache
  - **Value** (select one): ALL (default), METADATA, or NONE

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Block Volumes**.
2. Select **Create Block Volume**.
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, select the Enable High Performance button. You can only select high performance volumes if such volumes are available. For more information, see Block Volume Performance Options on page 381.

  This selection can't 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, and any user defined policies. For information about Oracle defined policies (`bronze`, `silver`, and `gold`) see Oracle Provided Backup Policies on page 427.

  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 on page 426. You can also back up this volume manually as described in Creating a Manual Boot or Block Volume Backup on page 423.
- **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.

  To use nondefault values for the block size, write bias, or secondary cache, specify the appropriate tags to set the values for these attributes. See Adding Tags at Resource Creation (IAM in OCI) on page 118.

4. Select **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 on page 387.

Using the CLI

Use the oci bv volume create command and required parameters to create a volume.

```
oci bv volume create --availability-domain <availability_domain_name> --
compartment-id <compartment_OCID>  [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

To use nondefault values for the block size, write bias, or secondary cache, specify the appropriate tags to set the values for these attributes. See Adding Tags at Resource Creation (IAM in OCI) on page 118.

Procedure

1. Gather the information that you need to run the command:
   - Availability domain name (`oci iam availability-domain list`)
   - Compartment OCID (`oci iam compartment list`)
2. Run the volume create command.

   This procedure doesn't show all available parameters for this command. For information about additional parameters, run the command with the `--help` option.

   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. You can only select high performance volumes if such volumes are available. For more information, see Block Volume Performance Options on page 381.

VPUs per GB can't be changed after the volume is created.

You can't auto-tune volumes.

Example:

```
$ oci bv volume create --availability-domain AD-1 --compartment-
id compartment_OCID --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.

When the volume is in the AVAILABLE state, you can attach the volume to an instance. See Attaching a Volume on page 387.

Using the API

Use the CreateVolume operation to create a block volume.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Attaching a Volume

You attach a volume to a Compute Cloud@Customer instance to expand the available storage on the instance.

You can attach volumes to more than one instance at a time, see Attaching a Volume to Multiple Instances on page 388.

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 Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. Select the compartment where the instance resides.
3. In the Instances list, select the instance that you want to attach a volume to.
4. Under **Resources**, select **Attached Block Volumes**.
5. In the **Attached Block Volumes** panel, select **Attach Block Volume**.
6. Enter the following information:
   a. Select the compartment where the block volume resides.
   b. Select a Block Volume.
   c. Select one of the following access modes:
      - **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.
7. Select **Attach to Instance**.

Using the CLI

Use the **oci compute boot-volume-attachment attach** command and required parameters to attach a boot volume to an instance.

```
oci compute volume-attachment attach --instance-id instance_OCID \
--volume-id volume_OCID --type paravirtualizedS]
```

> **Note:**
>
> The `--is-shareable` option is required to attach a shareable volume. The default value of this option is `false`.
>
> ```
> --is-shareable true
> ```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

This task is not available in the API.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Attaching a Volume to Multiple Instances

The Compute Cloud@Customer 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.

> **Important:**
>
> If you're attaching a volume that was detached, the volume might be associated with a different device name and the instance OS might not recognize the volume.

Limits and Considerations

- The Block Volume service doesn't provide coordination for concurrent write operations to block volumes attached to multiple instances, so if you configure the block volume as read/write and shareable, you must deploy a cluster aware system or solution on top of the shared storage.
- After you attach a block volume to an instance as read-only, it can only be attached to other instances as read-only. 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.
- If the block volume is already attached to an instance as read/write nonshareable you can't attach it to another instance until you detach it from the first instance. You can then reattach it to both the first and second instances as read/write shareable.
- You can't delete a block volume until it has been detached from all instances it was attached to.
- You can attach a block volume as read/write shareable or read-only shareable up to a maximum of eight instances.
- Block volumes attached as read-only are configured as shareable by default.
- Performance characteristics are per volume, so when a block volume is attached to multiple instances the performance is shared across all the attached instances.

Configuring Multiple Instance Volume Attachments with Read/Write Access

The Block Volume service doesn't 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 Console, CLI, or API.

   See Attaching a Volume on page 387.
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. 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 Console, CLI, or API.

   See Attaching a Volume on page 387.

2. Attach the block volume to additional instances as read-only using the Console, CLI, or API.

   See Attaching a Volume on page 387.

Find Your Volume in the Instance

In Compute Cloud@Customer, 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 and administer it in the OS.

For UNIX images, 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) on page 391.

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 more administrative information, refer to the documentation for the version of the OS that's 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. See Connecting to an Instance.
2. List the disk devices.

   Important:

   On UNIX OSs, the order in which volumes are attached is nondeterministic, 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 nonroot volume, there's 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 doesn't 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's mounted on `/boot` (the boot volume).
- `/dev/sdb` isn't listed because this block volume was just attached and hasn't had a file system created and isn't 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) on page 391.

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 Compute Cloud@Customer, for 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 page 389.

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 list, edit, move, clone, detach, and delete block volumes On Compute Cloud@Customer.

Listing Block Volumes and Block Volume Details

On Compute Cloud@Customer, you can list all block volumes in a specific compartment, and detailed information about a single volume.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Block Volumes**.
2. At the top of the page, select the compartment that contains the volume you want to see.
3. To view block volume details, select 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 select 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 CLI

Use these commands to list volumes:

- List all volumes in a compartment:

  Use the oci bv boot-volume list command.

```
oci bv volume list --compartment <compartment_OICD>  [OPTIONS]
```

- List the details for a single volume:

  oci bv boot-volume get command.

```
oci bv volume get --compartment <compartment_OICD>  [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use theListVolumes operation to list all volumes in a compartment.

Use the GetVolume operation to list the details for a single volume.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Listing Block Volume Attachments

On Compute Cloud@Customer, you can list all Block Volume volume attachments in a specific compartment, and detailed information for a single volume attachment.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. At the top of the page, select the compartment that contains the instance that has the attached block volume.
3. Select the instance name to display the details.
4. Under **Resources**, 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, select the block volume name.

Using the CLI

Use these commands to list volume attachments:

- Use xx to list all volume attachments in a compartment:

```
oci compute volume-attachment list --compartment-id <compartment_OCID>
 [OPTIONS]
```

- Use xx to list the details of a single volume attachment:

```
oci compute volume-attachment list --volume-id <volume_OCID>  [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use these APIs:

- List all volume attachments in a compartment: ListVolumeAttachments
- List the details for a single volume attachment: GetVolumeAttachment

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Updating a Block Volume

On Compute Cloud@Customer, you can change settings for block volumes while the volumes are online, without any downtime. You can change the display name, increase the size, and change tags.

To increase the volume size, see Resizing Volumes on page 403.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Block Volumes**.
2. At the top of the page, select the compartment that contains the volume that you want to edit.
3. For the volume that you want to edit, select the Actions menu (⋮), then select **Edit**.
4. In the Edit Block Volume dialog box, make 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 32 TB. You can't decrease the size. Before you change the size, see Resizing Volumes on page 403 to understand the implications of this change.
   - **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.

     To use nondefault values for the block size, write bias, or secondary cache, specify the appropriate tags to set the values for these attributes. See Adding Tags at Resource Creation (IAM in OCI) on page 118.
5. Select **Save Changes**.

Using the CLI

Use the oci bv volume update command and required parameters to update a volume.

```
oci bv volume update --volume-id <volume_OCID> --display-name <new-display-
name>  [OPTIONS]
```

To use nondefault values for the block size, write bias, or secondary cache, specify the appropriate tags to set the values for these attributes. See Adding Tags at Resource Creation (IAM in OCI) on page 118.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateVolume operation to update a volume.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Moving a Volume to a Different Compartment

On Compute Cloud@Customer, 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 can't move a block volume or boot volume from a security zone to a standard compartment.
- You can't 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 Console

This task isn't available in the Console.

Using the CLI

Use the oci bv boot-volume change-compartment:

```
oci bv volume change-compartment --volume-id <volume_OCID> --compartment-
id <destination_compartment_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ChangeVolumeCompartment operation to move a volume to another compartment.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Cloning a Block Volume

On Compute Cloud@Customer, you can create a clone from a volume using the Block Volume service. Cloning enables you to make a copy of an existing block volume without needing to go through the backup and restore process.

A cloned volume is a point-in-time direct disk-to-disk deep copy of the source volume. All the data that's in the source volume is copied to the clone volume. Any subsequent changes to the data on the source volume aren't copied to the clone.

By default, the clone is the same size as the source volume unless you 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, see Volume Backups and Clones on page 382.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Block Volumes**.
2. At the top of the page, select the compartment that contains the block volume you want to clone.
3. For the volume you want to clone, select the Actions menu (⋮), then select **Create Clone**.
4. In the dialog box, 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 keep the size the same, or increase the size in 1 GB increments up to 32768 (32 TB). You can't 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.
   - **Backup Policy:**(Optional) Select a backup policy from the drop-down list. You might need to change the compartment.

     Oracle defined policies are listed, and any user defined policies. For information about Oracle defined policies (`bronze`, `silver`, and `gold`) see Oracle Provided Backup Policies on page 427.

     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 on page 426. You can also back up this volume manually as described in Creating a Manual Boot or Block Volume Backup on page 423.
   - **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
5. Select **Create Clone**.

Using the CLI

Use the oci bv volume create command and required parameters to clone a volume.

```
oci bv volume create --availability-domain <availability_domain_name> --
compartment-id <compartment_OCID> --source-volume-id <source_volume_OCID> --
display-name <display_name>  [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateVolume operation to clone a volume.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Detaching a Block Volume

On Compute Cloud@Customer, 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.

Prerequisites

- 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.

Using the Console

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 Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
3. At the top of the page, select the compartment that contains the instance with the volume that you want to detach.
4. Select the name of the instance that has the volume attached.
5. Under Resources, select **Attached Block Volumes**.
6. Next to the volume you want to detach, select the Actions menu ( ), then select **Detach**.
7. Confirm when prompted.

Using the CLI

Use the oci compute volume-attachment detach command and required parameters to detach a block volume from an instance.

```
oci compute volume-attachment detach --volume-attachment-id <volume-attachment_OCID>  [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DetachVolume operation to detach a block volume from an instance.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Block Volume

On Compute Cloud@Customer, you can delete a volume that's no longer needed.

> **Caution:**
>
> You can't 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 on page 422.

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 Console

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 Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Block Volumes**.
3. At the top of the page, select the compartment that contains the block volume that you plan to delete.
4. For the volume you plan to delete, select the Actions menu (⋮), then select **Terminate**.
5. Confirm the termination.

   The block volume enters the TERMINATED state, and remains in that state for about 24 hours until it's removed.

Using the CLI

Use the oci bv volume delete command and required parameters to delete a block volume.

```
oci bv volume delete --volume-id <volume_OCID>  [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteVolume operation to delete a volume.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Managing Boot Volumes

On Compute Cloud@Customer, when you launch an instance, a new boot volume for the instance is created in the same compartment and attached to the instance.

You can accept the image's default boot volume size, or you can specify a larger size, up to 16 TB. If you specify a size smaller than the default boot volume size, your request is ignored and the boot volume is the default size.

The boot volume is associated with the 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.

You can list, detach, reattach, back up, clone, and delete boot volumes.

For Linux-based images, the custom boot volume size must be at least the image's default boot volume size or 50 GB, whichever is larger.

For Microsoft Windows-based images, the custom boot volume size must be at least the image's default boot volume size or 256 GB, whichever is larger. The minimum size requirement for Microsoft Windows images is to ensure that enough space is available for patches and updates.

If you specify a custom boot volume size, extend the volume to take advantage of the larger size.

Listing Boot Volumes

On conkeyref="cnvVariables/cnv-prodname-short"/>, you can list all the boot volumes in a compartment.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Boot Volumes**.
2. At the top of the page, select the compartment that contains the boot volume that you want to see.

   A list of boot volumes is displayed.
3. To view details about a boot volume, select the boot volume name.

Using the CLI

Use the oci bv boot-volume list command and required parameters to list the boot volumes in a compartment.

```
oci bv boot-volume list --availability-domain <availability_domain_name> --
compartment-id <compartment_OCID>  [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListBootVolumes operation to list the boot volumes in a compartment.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Listing Boot Volume Attachments

On Compute Cloud@Customer, you can list the boot volume that is attached to an instance.

Using the API

Use the ListBootVolumeAttachments operation to list the boot volume attached to an instance.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. At the top of the page, select the compartment that contains the instance with boot volume attachment you want to see.
3. Select the name of the instance for which you want to view the boot volume attachment.
4. Under **Resources**, select **Attached Boot Volumes**.

   The boot volume attachments are displayed.
5. To view the details about an attachment, select the boot volume attachment name.

Using the CLI

Use the oci compute boot-volume-attachment list command and required parameters to list the boot volume attached to an instance.

```
oci compute boot-volume-attachment list --availability-
domain <availability_domain_name> --compartment-id <compartment_OCID>
 [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListBootVolumeAttachments operation to list the boot volume attached to an instance.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Detaching a Boot Volume

On Compute Cloud@Customer, 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.

Prerequisite:

The instance must be in the Stopped state to detach a boot volume. You must reattach a boot volume before you can start the instance. See Stopping, Starting, and Resetting an Instance on page 328.

When you stop the instance, any application that's running on the instance is 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.

Using the Console

1. Stop the instance:
   a. Shut down the instance from the instance OS.
   b. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
   c. At the top of the page, select the compartment that contains the instance you plan to stop.
   d. Select the name of the instance.
   e. On the instance details page, select **Controls** (upper right corner), then select **Stop**.

   Wait for the status to change from **Stopping** to **Stopped**. Stopping an instance can take up to five minutes.
2. Under **Resources**, select **Boot Volumes**.
3. Select the boot volume Actions menu (⋮), then select **Detach**.
4. Confirm when prompted.

Using the CLI

Use the oci compute boot-volume-attachment detach command and required parameters to detach a boot volume from an instance.

```
oci compute boot-volume-attachment detach --boot-volume-attachment-
id <boot_volume-attachment_OCID>  [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DetachBootVolume operation to detach a boot volume from an instance.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Reattaching a Boot Volume

On Compute Cloud@Customer, you can reattach a boot volume to an instance.

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. For that information, see Detaching a Boot Volume on page 400 and Attaching a Volume on page 387. This procedure describes how to reattach a boot volume to an instance.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. At the top of the page, select the compartment that contains the instance for which you want to reattach a boot volume.
3. Select the instance name.
4. Under **Resources**, select **Boot Volumes**.

5. Select the boot volume Actions menu (⋮), then select **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 on page 328.

Using the CLI

Use the oci compute boot-volume-attachment attach command and required parameters to reattach a boot volume to an instance.

```
oci compute boot-volume-attachment attach --boot-volume-id
  <boot_volume_OCID> --instance-id <instance_OCID>  [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the AttachBootVolume operation to reattach a boot volume to an instance.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Cloning a Boot Volume

On Compute Cloud@Customer, you can create a clone from a boot volume using the Block Volume service. Cloning enables you to make a copy of an existing boot volume without needing to go through the backup and restore process.

Any subsequent changes to the data on the source boot volume are not copied to the boot volume clone. The clone is the same size as the source boot volume unless you specify a larger volume size when you create the clone.

The clone operation occurs immediately and you can use the cloned boot volume when the state changes to available.

There is a single point-in-time reference for a source boot volume while it is being cloned. If you clone a boot volume while the associated compute instance is running, you need to wait for the first clone operation to complete before creating more clones. You also need to wait for any backup operations to complete.

You can only create a clone for a boot volume within the same tenant. You can create a clone of a boot volume in a different compartment from the source volume compartment if you have the required access permissions.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Boot Volumes**.
2. At the top of the page, select the compartment that contains the boot volume that you want to clone.
3. For the volume you want to clone, select the Actions menu (⋮), then select **Create Clone**.
4. In the dialog box, 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.
   - **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
5. Select **Create Clone**.

   The new boot volume is ready to use when it reaches the Available state. For example, you can select the Actions menu and then select **Create Instance** to use this new boot volume to create a new instance.

Using the CLI

Use the oci bv boot-volume create command and required parameters to clone a boot volume.

```
oci bv boot-volume create --source-boot-volume-id OCID_of_volume_to_clone
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

This task is not available in the API.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Boot Volume

On Compute Cloud@Customer, you can delete a boot volume that's detached from an instance.

> **Caution:**
>
> Any data on a volume is permanently deleted when the volume is deleted. You can't restart the associated instance.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Compute**, then select **Instances**.
2. At the top of the page, select the compartment that contains the instance with the boot volume you want to delete.
3. For the instance that has the boot volume you want to delete, select the Actions menu (⋮), then select **Stop**.
4. When prompted, select **Force stop the instance by immediately powering off**, then select **Force Stop Instance**.
5. Wait for the instance to reach the **Stopped** state.
6. Select the name of the instance that has the boot volume you want to delete.
7. Under **Resources**, select **Boot Volumes**.
8. For the boot volume that you want to delete, select the Actions menu (⋮), then select **Detach**.

   Confirm when prompted.
9. Select the name of the boot volume.
10. Select the **Controls** button (upper right), then select **Terminate**.
11. Confirm the termination.

    The block volume enters the TERMINATED state, and remains in that state for about 24 hours until it's removed.

Using the CLI

Use the command and required parameters to delete a boot volume.

```
oci bv boot-volume delete --boot-volume-id <boot_volume_OCID>  [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteBootVolume operation to delete a boot volume.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Resizing Volumes

On Compute Cloud@Customer, 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 can't decrease the size of a volume.

Online Block Volume Resizing

This topic describes how to resize a block volume while it is online.

> **Caution:**
>
> Before you resize a boot or block volume, create a full backup of the volume. See Backing Up Block Volumes on page 422.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Block Volumes**.
2. At the top of the page, select the compartment that contains the block volume that you want to resize.
3. For the volume you plan to resize, select the Actions menu (⋮), then select **Edit**.
4. Change the size:
   - **Size (in GBs):** You can increase the size in 1 GB increments up to 16384 (16 TB). You can't decrease the size.
5. Select **Save Changes**.
6. Perform disk administrative tasks. See Completing the Online Resizing Operation on page 404.

Using the CLI

Use the oci bv volume update command and required parameters to resize the volume.

```
oci bv volume update --volume-id <volume_OCID> --size-in-gbs <size_in_GBS>
 [OPTIONS]
```

<size_in_GBS> is the size of the block volume. You can increment the size. You can't decrease the size. The value must be between 50 GB and 32 TB and specified in 1 GB increments.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

After resizing, perform disk administrative tasks. See Completing the Online Resizing Operation on page 404.

Using the API

Use the UpdateVolumeDetails operation to resize a volume.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Online Boot Volume Resizing

This topic describes how to resize a boot volume while it's online.

> **Important:**
>
> Before you resize a boot or block volume, create a full backup of the volume. See Backing Up Block Volumes on page 422.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Boot Volumes**.
2. At the top of the page, select the compartment that contains the boot volume.
3. For the volume you plan to resize, select the Actions menu (⋮), then select **Edit**.
4. Change the size:
   - **Size (in GBs):** You can keep the size the same, or increase the size in 1 GB increments up to 16384 (16 TB). You can't decrease the size.
5. Select **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 CLI

Use the oci bv boot-volume update command and required parameters to resize a boot volume.

```
oci bv boot-volume update --boot-volume-id <volume_OCID> --size-in-
gbs <size_in_GBSs  [OPTIONS]
```

For *size _in_GBs* specify the size of the boot volume. You can increase the size in 1 GB increments up to 16384 (16 TB). You can't decrease the size.

> **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.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

After resizing, perform disk administrative tasks. See Completing the Online Resizing Operation on page 404.

Using the API

Use the UpdateBootVolume operation to resize a boot volume.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Completing the Online Resizing Operation

After resizing the online block volume or boot volume, you need to perform the following disk administrative tasks:

1. Rescan the disk.

   For details, consult the OS documentation for the OS type and version running in the instance. Also see Rescanning the Disk for a Block Volume or Boot Volume
2. Extend the partition.

   For details, consult the OS documentation for the OS type and version running in the instance. Also see Extending the Partition for a Block Volume and Extending the Partition for a Boot Volume.

Offline Block Volume Resizing

On Compute Cloud@Customer, for 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 based and Microsoft Windows instances. Be aware of the following cautions.

> **Caution:**
>
> Before you resize a boot or block volume, create a full backup of the volume. See Backing Up Block Volumes on page 422.

> **Caution:**
>
> 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.

Prerequisite

- Block Volumes: Detach the block volume. See Detaching a Block Volume on page 397.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Block Volumes**.
2. At the top of the page, select the compartment that contains the block volume you want to resize.
3. For the volume you plan to resize, select the Actions menu (⋮), then select **Edit**.
4. Change the size:
   - **Size (in GBs):** You can increase the size in 1 GB increments up to 32768 (32 TB). You can't decrease the size.
5. Select **Save Changes**.
6. Complete the resizing operation.

   See Completing the Offline Resizing Operation for Block Volumes on page 406

Using the CLI

Use the oci bv boot-volume update command and required parameters to resize the block volume.

```
oci bv volume update --volume-id <volume_OCID> --size-in-gbs <size_in_GBS>
 [OPTIONS]
```

<size_in_GBs> is the size of the block volume. You can increase the size in 1 GB increments up to 32768. You can't decrease the size.

Next, complete the resizing operation.

See Completing the Offline Resizing Operation for Block Volumes on page 406

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateVolume operation to resize a block volume.

Next, complete the resizing operation.

See Completing the Offline Resizing Operation for Block Volumes on page 406

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Completing the Offline Resizing Operation for Block Volumes

After resizing the offline block volume or boot volume, you need to perform the following administrative tasks.

1. Reattach the volume.

   See Attaching a Volume on page 387.
2. Rescan the disk.

   For details, consult the OS documentation for the OS type and version running in the instance. Also see Rescanning the Disk for a Block Volume or Boot Volume
3. Extend the partition.

   For details, consult the OS documentation for the OS type and version running in the instance. Also see Extending the Partition for a Block Volume and Extending the Partition for a Boot Volume.

Offline Boot Volume Resizing

On Compute Cloud@Customer, for offline boot volume resizing, you must stop the instance and 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 don't 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 based and Microsoft Windows instances. Be aware of the following cautions.

> **Caution:**
>
> Before you resize a boot or block volume, create a full backup of the volume. See Backing Up Block Volumes on page 422.

> **Caution:**
>
> When you reattach a volume to an instance after resizing, if you aren't using consistent device paths, or if the instance doesn't support consistent device paths, device order and path might change. If you're using a tool such as Logical Volume Manager (LVM), you might need to fix the device mappings.

Prerequisites

- Stop the instance that has the boot volume. See Stopping, Starting, and Resetting an Instance on page 328
- Detach the boot volume. See Detaching a Boot Volume on page 400.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Boot Volumes**.
2. At the top of the page, select the compartment that contains the boot volume you want to resize.
3. For the volume you plan to resize, select the Actions menu (⋮), then select **Edit**.
4. Change the size:
   - **Size (in GBs):** You can keep the size the same, or increment the size. You can't decrease the size. The value must be between 50 GB and 32 TB and specified in 1 GB increments.
5. Select **Save Changes**.
6. Complete the resizing operation.

   See Completing the Offline Resizing Operation for Boot Volumes on page 407

Using the CLI

Use the oci bv boot-volume update command and required parameters to resize a boot volume.

```
oci bv boot-volume update --boot-volume-id <volume_OCID> --size-in-
gbs <size_in_GBs>  [OPTIONS]
```

The *size_in_GBs* value is the size of the boot volume. You can increment the size. You can't decrease the size. The value must be between 50 GB and 32 TB and specified in 1 GB increments.

Next, complete the resizing operation.

See Completing the Offline Resizing Operation for Boot Volumes on page 407.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateBootVolume operation to resize a boot volume.

Next, complete the resizing operation.

See Completing the Offline Resizing Operation for Boot Volumes on page 407

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Completing the Offline Resizing Operation for Boot Volumes

After resizing the online block volume or boot volume, you need to perform the following administrative tasks.

1. Attach the boot volume to a second instance as a data volume.

   See Attaching a Volume on page 387.
2. Extend the partition and grow the file system.

   For details, consult the OS documentation for the OS type and version running in the instance.
3. Detach the data volume.

   See Detaching a Block Volume on page 397.
4. Reattach the boot volume.

   See Reattaching a Boot Volume on page 400.
5. Restart the instance.

   See Stopping, Starting, and Resetting an Instance on page 328.

Managing Volume Groups

On Compute Cloud@Customer, 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. Ensure 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

On Compute Cloud@Customer, you can view volume groups using the Console, CLI, and API.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Volume Groups**.
2. At the top of the page, select the compartment that contains the volume group you want to see.
3. In the **Volume Groups** list, select the volume group you want to see.
4. To view the block volumes in the volume group, under **Resources**, select **Volumes**.

Using the CLI

Use the oci bv volume-group list command and required parameters to list volume groups.

```
oci bv volume-group list --compartment-id <compartment_OCID>  [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListVolumeGroups operation to list volume groups.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Creating a Volume Group

On Compute Cloud@Customer, you can create volume groups that include block and boot volumes using the Console, CLI, and API.

Prerequisite

Ensure that the volumes you plan to group are created. See Creating a Block Volume on page 384.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Volume Groups**.
2. Select **Create Volume Group**.
3. Provide the following information:
   - **Name**: A user-friendly name or description for the volume group. Avoid entering confidential information.
   - **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. Select **+ 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.

4. **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
5. Select **Create Volume Group**.

Using the CLI

Use the oci bv volume-group create command and required parameters to create a volume group.

```
oci bv volume-group create --availability-domain <availability_domain_name>
 --compartment-id <compartment_OCID> --source-details <json_string> or
 file://<path_to_JSON_file>  [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateVolumeGroup operation to create a volume group:

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Adding Volumes to a Group

On Compute Cloud@Customer, you can add block and boot volumes to an existing volume group using the Console, CLI, and API.

> **Note:**
>
> You can't 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 Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Volume Groups**.
2. At the top of the page, select the compartment that contains the volume group.
3. In the **Volume Groups** list, select the volume group you want to add the volume to.
4. Under **Resources**, select **Volumes**.
5. Select **Add Volumes**.
6. 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. Select **+ Add Volume** to add another volume.
7. Select **Update Volume Group**.

Using the CLI

Use the oci bv volume-group update command and required parameters to update a volume group.

```
oci bv volume-group update --volume-group-id <volume_group_OCID> --volume-
ids <volume_OCIDs_JSON> [OPTIONS]
```

For details about the JSON format, run this command:

```
oci bv volume-group update --generate-param-json-input volume-ids
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateVolumeGroup operation to update a volume group.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Removing Volumes from a Group

On Compute Cloud@Customer, you can remove block and boot volumes from an existing volume group using the Console, CLI, and API.

When you remove the last volume in a volume group, the volume group is deleted.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Volume Groups**.
2. At the top of the page, select the compartment that contains the volume group that has the volume you want to remove.
3. In the **Volume Groups** list, select the volume group that contains the volume you plan to remove.
4. Under **Resources**, select **Volumes**.
5. For the volume that you want to remove, select the Actions menu (⋮), then select **Remove**.
6. Confirm the removal.

Using the CLI

Use the oci bv volume-group update command and required parameters to update a volume group.

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 on page 409.

```
oci bv volume-group update --volume-group-id <volume_group_OCID> --volume-
ids <volume_OCIDs_JSON> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateVolumeGroup operation to update a volume group:

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Cloning a Volume Group

On Compute Cloud@Customer, you can create a clone of an existing volume group.

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's 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, see Volume Backups and Clones on page 382.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Volume Groups**.
2. At the top of the page, select the compartment that contains the volume group that you want to clone.
3. Select the name of the Volume Group you plan to clone.
4. Under **Resources**, select **Volume Group Clones**.
5. Select **Create Volume Group Clone**.
6. Enter the following information:
   - **Volume Group Clone Name:** Enter a descriptive name for the clone. Avoid entering confidential information.
   - **Create in Compartment:** Select the compartment where the clone will be created.

7. Select **Create Volume Group Clone**.

Using the CLI

This task is not available in the CLI.

Using the API

Use the CreateVolume operation and specify VolumeSourceFromVolumeDetails for CreateVolumeDetails.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Volume Group

On Compute Cloud@Customer, you can delete a volume group using the Console, CLI, and API.

When you delete a volume group the individual volumes in the group are not deleted, only the volume group is deleted.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Volume Groups**.
2. At the top of the page, select the compartment that contains the volume group that you want to delete.
3. In the **Volume Groups** list, select the volume group you want to delete.
4. On the details page for the volume group, select **Terminate**.
5. Confirm the termination.

Using the CLI

Use the oci bv volume-group delete command and required parameters to delete a volume group.

```
oci bv volume-group delete --volume-group-id <volume_group_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteVolumeGroup operation to delete a volume group.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Replicating Volumes

On Compute Cloud@Customer, you can asynchronously replicate a boot volume, block volume, or volume group to a different infrastructure and synchronize the data at regular intervals. A replica is not a backup. A backup is a point-in-time copy that enables you to return to a previous version of a volume or volume group. A replica is always the current version of the data. Although a replica is different from a backup, a replica can be used for recovery. A replica can also be used as the source to create a new volume.

> **Note:**
>
> **For Compute Cloud@Customer only.** Before you can configure storage replication, you must submit a request for Oracle Support to configure infrastructure peering. See Creating a Support Request. To access support, sign in to the Oracle Cloud Console as described in Sign In to the OCI Console.

Each volume or volume group can have only one replica. To create a replica on a different infrastructure, see Updating Replication Configuration for a Volume on page 416.

You can't resize a volume that has replication enabled. When you resize a volume that has replication enabled, you must disable replication, which deletes the replica. After the volume is resized, you can re-enable replication for the volume. See Enabling Replication When You Update a Volume on page 414.

Replication cannot be enabled for a volume that is in a DR configuration.

Listing Volume Replicas

On Compute Cloud@Customer, you can list all volume replicas in a specified compartment, and show more information about a particular replica.

The Last Sync column on the replicas list shows the time of the last data synchronization from the source volume to the replica. The Actions menu enables you to activate the replica. See Activating a Boot or Block Volume Replica on page 416.

The replica details shows the following:

- Availability Domain
- Size of the replica
- Source volume OCID
- Source infrastructure
- Time of the last data synchronization from the source volume to the replica
- Total data transferred
- Name of the volume group replica that this replica is in, if applicable
- Activated volumes

The activated volumes is the list of new volumes that have been created from this replica.

The total data transferred is the amount of data, in gigabytes, that has been transferred during the replication process for the volume. This includes all data from the time the volume replication was enabled to the current time.

Using the Console

Use the following procedure to list all boot or block replicas in the selected compartment and view details of a selected replica.

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select either Block Volume Replicas or Boot Volume Replicas.
2. Select the name of a replica to view details about that replica.

Using the CLI

Use the oci bv block-volume-replica list and oci bv boot-volume-replica list commands to list all block or boot volume replicas in the specified compartment:

```
oci bv block-volume-replica list --availability-domain AD-1 --compartment-
id compartment_OCID
oci bv boot-volume-replica list --availability-domain AD-1 --compartment-
id compartment_OCID
```

Use the oci bv block-volume-replica get and oci bv boot-volume-replica getcommands to get details about the specified replica.

```
oci bv block-volume-replica get --block-volume-replica-id block-volume-
replica-OCID
oci bv boot-volume-replica get --boot-volume-replica-id boot-volume-replica-
OCID
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListBlockVolumeReplicas operation to list all block volume replicas in the specified compartment. Use the GetBlockVolumeReplica operation to get details about the specified block volume replica.

Use the ListBootVolumeReplicas operation to list all boot volume replicas in the specified compartment. Use the GetBootVolumeReplica operation to get details about the specified boot volume replica.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Enabling Replication for a Boot or Block Volume

On Compute Cloud@Customer, you can enable replication for a boot volume or a block volume either when you create a new volume or when you update a volume.

> **Note:**
>
> Before you can configure storage replication, you must submit a request for Oracle Support to configure infrastructure peering. See Creating a Support Request. To access support, sign in to the Oracle Cloud Console as described in Sign In to the OCI Console.

The replica that you specify when you enable replication has the following restrictions:

- Must be on a different infrastructure than the infrastructure that the source volume is on.

  When you specify the replica, you specify an Availability Domain in the form *`infrastructure_ID`*`:AD-1`. You can get the *`infrastructure_ID`* from the OCID of a resource on the target infrastructure. The *`infrastructure_ID`* is the fifth field in the OCID, where fields are delimited by period characters. In the following example, `ilewlw4qpza` is the *`infrastructure_ID`*:

  `ocid1.cccvolume.oc1.phx.ilewlw4qpza.`***`unique_ID`***
- Must be in the same pool (capacity or performance vpusPerGB) on the target infrastructure that the source volume is in on the source infrastructure. Attempting to create a replica in a different pool will fail.

The initial synchronization of the data from the source to the replica can take hours, depending on the amount of data.

After the initial synchronization, the data is synchronized every ten minutes.

Enabling Replication When You Create a Volume

On Compute Cloud@Customer, you can create a replica at the same time that you create the volume.

When the new volume is in the Available state, then the boot or block volume replica creation starts and the replica is in the Provisioning state. When the replica is successfully created, its state changes to Available.

Using the Console

Use the following procedure to enable replication for a new block volume.

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select Block Volumes.
2. Select the Create Block Volume button.
3. In the Create Block Volume dialog, in addition to the other information required to create the block volume, provide the following information to enable volume replication:
   - In the Replication section, select the On button.
   - Select an Availability Domain from the drop-down menu.

     Availability Domains are in the form *`infrastructure_ID`*`:AD-1`, where *`infrastructure_ID`* identifies the target infrastructure.
   - Enter a name for the replica.

     The display name can be a maximum of 255 characters, does not need to be unique, and is changeable.
   - (Optional) Select the Encryption button to enable encryption using Oracle-managed keys.

   See Creating a Block Volume on page 384 to learn more about creating a volume.
4. Select the Create Block Volume button.

   The details page of the new block volume is displayed. Scroll to the Resources section and select Block Volume Replicas. The list shows the Availability Domain of the replica.

Use the following procedure to enable replication for a new boot volume. To create a new boot volume, clone an existing boot volume.

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select Boot Volumes.
2. For the boot volume that you want to clone, select the Actions menu and select Create Clone.
3. In the Create Clone dialog, in addition to the other information required to create the boot volume, provide the information described in Step 3 of the preceding procedure to enable volume replication.

   See Cloning a Boot Volume on page 401 to learn more about cloning a boot volume.
4. Select the Create Clone button.

   The details page of the new boot volume is displayed. Scroll to the Resources section and select Boot Volume Replicas.

Using the CLI

Use the `--block-volume-replicas` option with the oci bv volume create command to configure replication for a new block volume. Although the value of the `--block-volume-replicas` option is an array, do not specify more than one replica. A volume can have only one replica. See Creating a Block Volume on page 384 or use the `-h` option to learn more about creating a volume.

```
oci bv volume create --block-volume-replicas
 '[{"availabilityDomain":"serial_number:AD-1","displayName":"Sample_Replica"}]'
```

Use the `--boot-volume-replicas` option with the oci bv boot-volume create command to configure replication for a new boot volume. Although the value of the `--boot-volume-replicas` option is an array, do not specify more than one replica. A volume can have only one replica. See Cloning a Boot Volume on page 401 or use the `-h` option to learn more about cloning a boot volume.

```
oci bv boot-volume create --source-boot-volume-
id OCID_of_volume_to_clone --boot-volume-replicas
 '[{"availabilityDomain":"serial_number:AD-1","displayName":"Sample_Replica"}]'
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateVolume operation to create a block volume, specifying blockVolumeReplicas in CreateVolumeDetails.

Use the CreateBootVolume operation to create a boot volume by cloning an existing boot volume. In CreateBootVolumeDetails, specify the boot volume to clone in sourceDetails, and specify replication configuration in bootVolumeReplicas.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Enabling Replication When You Update a Volume

To enable replication for an existing volume On Compute Cloud@Customer, configure a replica when you update the volume the same way you configure a replica when you create a volume.

If a replica is already configured for the volume, you cannot configure a second replica. Each volume can have only one replica. To update the replica configuration, see Updating Replication Configuration for a Volume on page 416.

Using the Console

Use the following procedure to enable replication for an existing boot or block volume.

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select either Block Volumes or Boot Volumes.
2. Check whether replication is already configured for the volume for which you want to enable replication.

   If the value in the Replicas column is Off, continue with the next step in this procedure.

   If the value in the Replicas column is On, then replication is already configured for this volume. You cannot change the replication configuration by editing the volume. See Updating Replication Configuration for a Volume on page 416.

3. Select the Actions menu and select Edit.
4. In the Edit Volume dialog, scroll to the Replication section.

   Provide the following information:

   - Select the On button.
   - Select an Availability Domain from the drop-down menu.

     Availability Domains are in the form *infrastructure_ID*:AD-1, where *infrastructure_ID* identifies the target infrastructure.
   - Enter a name for the replica.

     The display name can be a maximum of 255 characters, does not need to be unique, and is changeable.
   - (Optional) Select the Encryption button to enable encryption using Oracle-managed keys.
5. Select the Save Changes button.

   On the details page of the volume, the replica is listed in the Resources section.

Using the CLI

First, check whether the volume for which you want to configure replication already has a replica. You can configure replication only for a volume that does not already have replication configured.

Use the oci bv volume get command to confirm that replication is not configured for this volume. Use the `--block-volume-replicas` option with the oci bv volume update command to configure replication for the specified block volume. Although the value of the `--block-volume-replicas` option is an array, do not specify more than one replica. A volume can have only one replica.

```
oci bv volume get --volume-id volume_to_update_OCID
oci bv volume update --volume-id volume_to_update_OCID --block-volume-
replicas
 '[{"availabilityDomain":"serial_number:AD-1","displayName":"Sample_Replica"}]'
```

Use the oci bv boot-volume get command to confirm that replication is not configured for this volume. Use the `--boot-volume-replicas` option with the oci bv boot-volume update command to configure replication for the specified boot volume. Although the value of the `--boot-volume-replicas` option is an array, do not specify more than one replica. A volume can have only one replica.

```
oci bv boot-volume get --boot-volume-id volume_to_update_OCID
oci bv boot-volume update --boot-volume-id volume_to_update_OCID --boot-
volume-replicas
 '[{"availabilityDomain":"serial_number:AD-1","displayName":"Sample_Replica"}]'
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

First, check whether the volume for which you want to configure replication already has a replica. You can configure replication only for a volume that does not already have replication configured.

Use the GetVolume operation to confirm that replication is not configured for this volume. Use the UpdateVolume operation to update a block volume, specifying blockVolumeReplicas in UpdateVolumeDetails.

Use the GetBootVolume operation to confirm that replication is not configured for this volume.
Use the UpdateBootVolume operation to update a boot volume, specifying bootVolumeReplicas in UpdateBootVolumeDetails.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Updating Replication Configuration for a Volume

On Compute Cloud@Customer, you cannot change the availability domain of a volume replica by updating the replica or the volume. If you want to change the availability domain of a replica, you must delete the replica and then re-enable replication for that volume, specifying the new replication configuration values.

To change the availability domain in the replication configuration for a boot or block volume:

1. Disable replication on the volume where you want to change the replication configuration. See Disabling Replication for a Boot or Block Volume on page 417. Disabling replication deletes the existing replica.
2. Re-enable replication on the volume, specifying the new configuration values. See Enabling Replication When You Update a Volume on page 414.

Activating a Boot or Block Volume Replica

On Compute Cloud@Customer, activating a boot or block volume replica means creating a new volume using that boot or block volume replica as the source.

> **Note:**
>
> If the volume belongs to a volume group, do not activate the volume replica. Instead, activate the volume group replica. See Activating a Volume Group Replica on page 421.

The activation process creates a new volume by cloning the replica. The replica is still serving as the replication target for a different volume. During this cloning process, the volume replica is in the Activating state, and synchronization to the replica is paused. When the new volume is in the Available state, the volume replica state changes to the Available state, and synchronization to the replica is resumed.

You cannot change the vpusPerGB when you create a new volume by cloning a volume replica. The value of vpusPerGB in the new volume is the same as the value in the replica. The new volume is in the same performance pool as the replica.

Before activating a block volume replica, the LUN's iSCSI group is set to default values and the LUN name is changed to a new unique name.

A boot volume replica references the same image as the replica source, which is on a different infrastructure. If you try to launch an instance on the target infrastructure using the activated boot volume replica, the instance launch will fail because the image cannot be reached. Check the following to ensure you are not trying to use a boot volume that references an image on a different infrastructure:

- The OCID of the source for the boot volume is a boot volume replica. For example, you will see the following in the CLI:

  ```
  "source-details": {
    "id": "ocid1.cccbootvolumereplica.oc1.unique_ID",
    "type": "bootVolumeReplica"
  }
  ```
- In the image details section in the console, only the image OCID is shown. The operating system and version are not shown.

Using the Console

Use the following procedure to activate a boot or block volume replica.

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select either Block Volume Replicas or Boot Volume Replicas.
2. On the Block Volume Replicas list, for the block volume replica that you want to activate, select the Actions menu and select Activate Replica. If you instead select the name of a block volume replica to display its details page, select the Activate button on the details page to display the Activate Replica dialog.

   On the Boot Volume Replicas list, for the boot volume replica that you want to activate, select the Actions menu and select Create Clone. If you instead select the name of a boot volume replica to display its details page, select the Activate button on the details page to display the Create Clone dialog.

3. In the Activate Replica or Create Clone dialog, configure the new block or boot volume the same way you would if you were using create instead of activate, including configuring volume replication if needed.

Using the CLI

Use the oci bv volume create command or the oci bv boot-volume create command to activate a block or boot volume replica, specifying the replica as the source for the new volume.

```
oci bv volume create --source-volume-replica-id boot-volume-replica-OCID --
compartment-id $cidcompartment_OCID --availability-domain AD-1
oci bv boot-volume create --source-volume-replica-id boot-volume-replica-
OCID --compartment-id compartment_OCID --availability-domain AD-1
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateVolume operation to activate a block volume replica. Specify the OCID of the block volume replica to activate in VolumeSourceFromBlockVolumeReplicaDetails.

Use the CreateBootVolume operation to activate a boot volume replica. Specify the OCID of the boot volume replica to activate in BootVolumeSourceFromBootVolumeReplicaDetails.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Disabling Replication for a Boot or Block Volume

On Compute Cloud@Customer, you must disable replication for a volume to change the replication configuration, resize the volume, or delete the volume.

> **Note:**
>
> Disabling replication deletes the replica.

You can later re-enable replication for the volume.

To disable replication for a volume, update the volume as shown in Enabling Replication When You Update a Volume on page 414.

- When using the console, select the Off button in the Replication section of the Edit dialog. Select Confirm to acknowledge that the volume replica will be deleted.
- When using the CLI, set the value of `--block-volume-replicas` or `--boot-volume-replicas` to empty: `'[]'`
- When using the API, set the value of `blockVolumeReplicas` or `bootVolumeReplicas` to null.

Recovering a Source Volume

This topic describes how to recover a source volume that has become inaccessible or corrupted On Compute Cloud@Customer.

If you only need temporary access to the volume, or if the source infrastructure is not available, you can stop after Step 1 and use the activated replica.

If the following resources are available and you want to reconfigure replication, then you can continue with Step 2 and configure replication at the same time that you activate the replica:

- The source infrastructure
- A volume on that infrastructure that is in the same pool (capacity or performance) as the original source volume

Alternatively, you can configure replication of this new source volume (the activated replica) later by updating the volume.

1. Activate the boot or block volume replica as described in Activating a Boot or Block Volume Replica on page 416.

2. In the activating operation, configure replication.

   In *infrastructure_ID*:AD-1, use the *infrastructure_ID* for the infrastructure that the inaccessible source volume is on. A copy of the volume that was the replica is now the source volume on the infrastructure where the original replica is, and the new replica will be created on the infrastructure where the original source volume is.

   > **Note:**
   >
   > If these volumes are boot volumes, the new synchronization will take more time than the original. This is because the activated boot volume replica that is now the source volume must transmit the image data in every synchronization, even though the image data has not changed.

Listing Volume Group Replicas

On Compute Cloud@Customer, you can list all volume group replicas in a specified compartment, and show more information about a particular replica.

The Last Sync column on the replicas list shows the time of the last data synchronization from the source volume group to the replica. The Actions menu enables you to activate the replica. See Activating a Volume Group Replica on page 421.

The replica details shows the following:

- Total size of the replica
- Source volume group OCID
- Number of volume replicas in the volume group replica
- Time of the last data synchronization from the source volume to the replica
- Activated volume groups

The activated volume groups is the list of new volume groups that have been created from this volume group replica.

Using the Console

Use the following procedure to list all volume group replicas in the selected compartment and view details of a selected replica.

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select Volume Group Replicas.
2. Select the name of a replica to view details about that replica.

   The Resources section of the volume group replica details page shows block volume replicas, boot volume replicas, and activated volume groups.

Using the CLI

Use the oci bv volume-group-replica list command to list all volume group replicas in the specified compartment:

```
oci bv volume-group-replica list --availability-domain AD-1 --compartment-
id compartment_OCID
```

Use the oci bv volume-group-replica get command to get details about the specified replica.

```
oci bv volume-group-replica get --volume-group-replica-id volume-group-
replica-OCID
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListVolumeGroupReplicas operation to list all volume group replicas in the specified compartment.

Use the GetVolumeGroupReplica operation to get details about the specified volume group replica.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Enabling Replication for a Volume Group

On Compute Cloud@Customer, volume group replicas themselves don't contain the full source data from the volumes in the group. Instead, the volume group replica contains references to individual replicas for each of the volumes in the group. This approach provides a seamless replication experience for volumes whether they remain in the volume group, are removed from the volume group, or if replication is disabled for the volume group itself. You can enable replication when you create a volume group or when you update a volume group.

When you enable replication for a volume group, all volumes in the group are included in the volume group replica in the destination infrastructure. Volumes are no longer replicated individually, regardless of whether they were configured for replication before they were included in a volume group with replication enabled. You can no longer update the replication configuration for an individual volume. Replication can only be configured at the volume group level.

When you enable replication for a volume group, if the group contains volumes that are already configured for replication, the replication destination infrastructure of those volumes must match the replication destination infrastructure of the volume group. If these infrastructures do not match, you will receive a conflicting volumes error message while configuring replication for the volume group. To successfully configure replication for the volume group, remove volumes from the group that have a different replication destination infrastructure.

When you add one or more volumes that are not configured for replication to a volume group that is configured for replication, the regular volume group replication synchronization might degrade while the initial synchronization process runs for the volumes that were added to the group. Recall that initial synchronization of a volume to its replica can take a long time, depending on factors such as the size of the data on the volume. To prevent a negative effect on the regular synchronization of the volume group to the volume group replica, enable replication for the volume before you add the volume to the volume group. Add the volume to the volume group after the initial synchronization process is complete for the volume. The initial synchronization process is complete for the volume when the status of the volume replica changes from Provisioning to Available.

When you specify the volume group replica, you specify an Availability Domain in the form `infrastructure_ID`:AD-1. You can get the `infrastructure_ID` from the OCID of a resource on the target infrastructure. The `infrastructure_ID` is the fifth field in the OCID, where fields are delimited by period characters. In the following example, ilewlw4qpza is the `infrastructure_ID`:

`ocid1.cccvolume.oc1.phx.ilewlw4qpza.unique_ID`

Enabling Replication When You Create a Volume Group

On Compute Cloud@Customer, you can create a replica at the same time that you create the volume group.

When the new volume group is in the Available state, then the volume group replica creation starts and the replica is in the Provisioning state. When the replica is successfully created, its state changes to Available.

Using the Console

Use the following procedure to enable replication for a new volume group.

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select Volume Groups.
2. Select the Create Volume Group button.
3. In the Create Volume Group dialog, in addition to the other information required to create the volume group, such as selecting volumes for the group, provide the following information to enable volume replication:
   - In the Replication section, select the On button.
   - Select an Availability Domain from the drop-down menu.

     Availability Domains are in the form `infrastructure_ID`:AD-1, where `infrastructure_ID` identifies the target infrastructure.

     If any of the volumes you add to the group are configured for replication, the destination infrastructure configured for them must match the destination infrastructure you configure for the volume group.
   - Enter a name for the replica.

The display name can be a maximum of 255 characters, does not need to be unique, and is changeable.

- (Optional) Select the Encryption button to enable encryption using Oracle-managed keys.

See Creating a Volume Group on page 408 to learn more about creating a volume group.

4. Select the Create Volume Group button.

   The details page of the new volume group is displayed. The general information section shows the total size of the volume group source.

   Scroll to the Resources section and select Volume Group Replicas. The list shows the Availability Domain of the replica.

Using the CLI

Use the `--volume-group-replicas` option with the oci bv volume-group create command to configure replication for a new volume group. Although the value of the `--volume-group-replicas` option is an array, do not specify more than one replica. A volume can have only one replica. See Creating a Block Volume on page 384 or use the `-h` option to learn more about creating a volume.

```
oci bv volume-group create --availability-domain AD-1 --compartment-
id compartment_OCID --source-details volume_OCIDs --volume-group-replicas
 '[{"availabilityDomain":"serial_number:AD-1","displayName":"Sample_Replica"}]'
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateVolumeGroup operation to create a volume group, specifying volumeGroupReplicas in CreateVolumeGroupDetails.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Enabling Replication When You Update a Volume Group

To enable replication for an existing volume group On Compute Cloud@Customer, configure a replica when you update the volume group the same way you configure a replica when you create a volume group.

If a replica is already configured for the volume group, you cannot configure a second replica. Each volume group can have only one replica. To update the replica configuration, see Updating Replication Configuration for a Volume Group on page 421.

Using the Console

Use the following procedure to enable replication for an existing volume group.

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select Volume Groups.
2. Check whether replication is already configured for the volume group for which you want to enable replication.

   If the value in the Replicas column is Off, continue with the next step in this procedure.

   If the value in the Replicas column is On, then replication is already configured for this volume group. You cannot change the replication configuration by editing the volume. See Updating Replication Configuration for a Volume Group on page 421.
3. Select the Actions menu and select Edit.
4. In the Edit Volume Group dialog, scroll to the Replication section.

   Provide the following information:

   - Select the On button.
   - Select an Availability Domain from the drop-down menu.

     Availability Domains are in the form *`infrastructure_ID`*`:AD-1`, where *`infrastructure_ID`* identifies the target infrastructure.
   - Enter a name for the replica.

The display name can be a maximum of 255 characters, does not need to be unique, and is changeable.

- (Optional) Select the Encryption button to enable encryption using Oracle-managed keys.

5. Select the Save Changes button.

   On the details page of the volume group, Volume Group Replicas are listed in the Resources section. The list shows the Availability Domain of the replica.

Using the CLI

First, check whether the volume group for which you want to configure replication already has a replica. You can configure replication only for a volume group that does not already have replication configured.

Use the oci bv volume-group get command to confirm that replication is not configured for this volume. Use the `--volume-group-replicas` option with the oci bv volume-group update command to configure replication for the specified volume group. Although the value of the `--volume-group-replicas` option is an array, do not specify more than one replica. A volume group can have only one replica.

```
oci bv volume-group get --volume-group-id volume_group_to_update_OCID
oci bv volume-group update --volume-group-id volume_group_to_update_OCID --
volume-group-replicas
 '[{"availabilityDomain":"serial_number:AD-1","displayName":"Sample_Replica"}]'
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

First, check whether the volume group for which you want to configure replication already has a replica. You can configure replication only for a volume group that does not already have replication configured.

Use the GetVolumeGroup operation to confirm that replication is not configured for this volume. Use the UpdateVolumeGroup operation to update a volume group, specifying volumeGroupReplicas in UpdateVolumeGroupDetails.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Updating Replication Configuration for a Volume Group

On Compute Cloud@Customer, you cannot change the availability domain of a volume group replica by updating the replica or the volume group. If you want to change the availability domain of a replica, you must delete the replica and then re-enable replication for that volume group, specifying the new replication configuration values.

To change the availability domain in the replication configuration for a volume group:

1. Disable replication on the volume group where you want to change the replication configuration. See Disabling Replication for a Volume Group on page 422. Disabling replication deletes the existing replica.
2. Re-enable replication on the volume group, specifying the new configuration values. See Enabling Replication When You Update a Volume Group on page 420.

Activating a Volume Group Replica

On Compute Cloud@Customer, activating a volume group replica means creating a new volume group using that volume group replica as the source.

> **Note:**
>
> Do not activate individual volume replicas that belong to a volume group replica. Instead, activate the volume group replica.

While it is possible to activate individual volume replicas separately from the volume group replica, to ensure that all the activated volumes have the same data, activate the volume group replica instead. Activating the volume group replica ensures that all replicas are activated from the same coordinated synchronization point.

Using the Console

Use the following procedure to activate a volume group replica.

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select Volume Group Replicas.
2. On the Volume Group Replicas list, for the volume group replica that you want to activate, select the Actions menu and select Activate Replica. If you instead select the name of a volume group replica to display its details page, select the Activate button on the details page to display the Activate Replica dialog.
3. In the Activate Volume Group Replica dialog, configure the new volume group the same way you would if you were using create instead of activate, including configuring volume replication if needed.
4. Select the Create Volume Group button in the dialog.
5. Select the Confirm button in the Confirm Activation dialog to acknowledge that the initial synchronization of the volumes might take a long time.

Using the CLI

Use the oci bv volume-group create command to activate a volume group replica, specifying the replica as the source for the new volume.

```
oci bv volume-group create --availability-domain AD-1 --compartment-
id compartment_OCID --source-details '{"type": "volumeGroupReplicaId",
 "volumeGroupReplicaId": "volume_group_replica_OCID"}'
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateVolumeGroup operation to activate a volume group replica. Specify the OCID of the volume group replica to activate in VolumeGroupSourceFromVolumeGroupReplicaDetails in CreateVolumeGroupDetails.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Disabling Replication for a Volume Group

To disable replication for an existing volume group On Compute Cloud@Customer, update the volume group and set the value of the replicas parameter to null.

> **Note:**
>
> Disabling replication deletes the replica.

When you disable replication for a volume group, the volumes in the volume group continue to replicate, but as separate volume replicas, not as part of a volume group replica. After replication for the volume group is disabled, you can disable replication for individual volumes.

To disable replication for a volume group, update the volume group as shown in Enabling Replication When You Update a Volume Group on page 420.

- When using the console, select the Off button in the Replication section of the Edit dialog. Select Confirm to acknowledge that the volume group replica will be deleted.
- When using the CLI, set the value of `--volume-group-replicas` to empty: `'[]'`
- When using the API, set the value of `volumeGroupReplicas` to null.

Backing Up Block Volumes

On Compute Cloud@Customer, the backup feature for the Block Volume service enables you make a point-in-time snapshot of the data on a block or boot volume. These backups can then be restored to new volumes any time.

On Compute Cloud@Customer, all backups are full backups.

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 on page 423.
- **Policy-Based Backups:**These are automated scheduled backups defined by the backup policy assigned to the volume. See Managing Backup Policies on page 426.

For information to help you decide whether to create a backup or a clone of a boot volume, see Volume Backups and Clones on page 382.

Viewing Volume Backups

On Compute Cloud@Customer, you can view volume backups using the Console, CLI, and API.

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.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **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're viewing the correct compartment, which is displayed at the top of the page.
3. To view the details of a backup, select the backup name.

Using the CLI

Use the oci bv backup list command and required parameters to view the volume backups.

```
oci bv backup list --compartment-id <Compartment_OCID> [OPTIONS]
```

Use the oci bv backup get command and required parameters to view the details of a volume backup.

```
oci bv volume-group-backup list --compartment-id <Compartment_OCID>
```

When there are many volume backups, for example hundreds of backups, it can take a long time to list. The `backup list` commands support pagination of results. Use the `--limit` option to specify the number of backups to list. To show the next page of results, use the value of the `opc-next-page` property with the `--page` option. Specify the `--limit` option again for each new page command. Otherwise, all remaining resources are 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.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Viewing Volume Backups Using the API

Use the ListVolumeBackups operation to view the volume backups.

Use the GetVolumeBackup command and required parameters to view the details of a volume backup.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Creating a Manual Boot or Block Volume Backup

On Compute Cloud@Customer, you can create a block or boot volume backup manually using the Console, CLI, and API.

To create scheduled backups, see Managing Backup Policies on page 426

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select either **Block Volumes** or **Boot Volumes**.
2. At the top of the page, select the compartment that contains the volume that you want to back up.
3. For the volume you plan to back up, select the Actions menu (⋮), then select **Create Manual Backup**.
4. Enter the following 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 one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
5. For a block volume backup, select **Create Manual Backup**. For a boot volume backup, select **Create**.

Using the CLI

Use the oci bv backup create command and required parameters to create a manual backup.

```
oci bv backup create --volume-id <block_volume_OCID> --display-
name <Backup_Name> --type FULL|INCREMENTAL [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateVolumeBackup operation to create a manual backup.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Creating a Manual Backup of a Volume Group

On Compute Cloud@Customer, you can create a volume group backup manually using the Console, CLI, and API.

A volume group backup provides coordinated point-in-time-consistent backups of all the volumes in a volume group automatically. You can perform most of the same backup operations and tasks with volume groups that you can perform with individual block volumes and boot volumes.

You can restore a volume group backup to a volume group, or you can restore individual volumes in the volume group from volume backups. With volume group backups, you can manage the backup settings for several volumes in one place, consistently. This simplifies the process to create time-consistent backups of running enterprise applications that span multiple storage volumes across multiple instances.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Volume Groups**.
2. At the top of the page, select the compartment that contains the volume group that you want to back up.
3. For the volume group you want to back up, select the Actions menu (⋮), then select **Create Volume Group Backup**.
4. Enter a descriptive name for the backup. Avoid entering confidential information.
5. Select **Create Volume Group Backup**.

Using the CLI

Use the oci bv volume-group-backup create command and required parameters to back up a volume group.

```
oci bv volume-group-backup create --volume-group-id <volume_group_OCID> --
compartment-id <compartment_OCID> --display-name <display_name> --type FULL
 [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateVolumeGroupBackup operation to back up a volume group.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Restoring a Backup to a New Volume

On Compute Cloud@Customer, you can restore a volume backup to a new volume using the Console, CLI, and API.

You can restore a volume from any of your incremental or full volume backups. Both backup types enable you to restore the full volume contents to the point-in-time snapshot of the volume when the backup was taken. You don't need to keep the initial full backup or subsequent incremental backups in the backup chain and restore them in sequence, you only need to keep the backups taken for the times you care about.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Block Volume Backups**.
2. At the top of the page, select the compartment that contains the block volume backup that you want to restore.
3. For the block volume backup that you want to restore, select the Actions menu (⋮), then select **Restore Block Volume**.
4. Enter 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, select the Enable High Performance button.

     This selection can't 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 on page 426.
   - **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
5. Select **Create Block Volume**.

   The volume is ready to attach after its icon no longer lists it as PROVISIONING in the volume list.

Using the CLI

Use the oci bv volume create command and required parameters to restore a backup to a new volume.

```
oci bv volume create --availability-domain  <availability_domain_name> --
volume-backuup-id <source_volume_backup_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateVolume operation to restore a backup to a new volume. The API has an optional `volumeBackupId` parameter that you can use to define the backup from which the data should be restored on the newly created volume.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Restoring a Volume Group from a Volume Group Backup

On Compute Cloud@Customer, you can restore a volume group from a volume group backup using the Console, CLI, and API.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Volume Group Backups**.
2. At the top of the page, select the compartment that contains the volume group backup that you want to restore.
3. For the volume group backup that you want to restore, select the Actions menu (⋮), then select **Restore Volume Group**.
4. Enter 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 on page 426.
   - **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
5. Select **Create Volume Group**.

Using the CLI

Use the oci bv volume-group create command and required parameters to create a new volume group from a volume group backup.

```
oci bv volume-group create --availability-domain <availability_domain_name>
 --compartment-id  <compartment_OCID>--source-details <json_string> or
 file://<path_to_JSON_file> [OPTIONS]
```

Using the API

Use the CreateVolumeGroupBackup operation to create a new volume group from a volume group backup.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Managing Backup Policies

On Compute Cloud@Customer, you can perform volume backups and volume group backups automatically using a schedule, and retain them based on the retention setting in the backup policy.

The Block Volume service enables you to perform volume backups and volume group backups automatically according to a schedule that's defined in a backup policy. The backup policy also specifies how long to retain the backup.

You can use one of the backup policies 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 on page 427. You can't modify policies that are provided by Oracle. You can't use an Oracle defined backup policy to back up a volume group.

On Compute Cloud@Customer, all backups are full backups.

A particular resource can't 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 on page 422.

Oracle Provided Backup Policies

Oracle defined backup policies can be used to back up individual block volumes and boot volumes. These policies can't be used to back up a volume group.

Oracle defined policies can't be changed. If you need a different backup time or retention time, for example, create a user defined backup policy as described in Creating a Backup Policy on page 427.

The following backup policies are provided by Oracle and available in every compartment. A resource can have only one backup policy assigned, but a backup policy can have multiple schedules. The Oracle Bronze policy has one schedule, the Silver policy has two schedules, and the Gold policy has three schedules. All schedule times are your regional data center time. All backup types are full backups.

Bronze

1. Monthly backups that run at 00:00 on the first day of the month and are retained for twelve months.

Silver

1. Weekly backups that run at 00:00 every Monday and are retained for four weeks.
2. Monthly backups that run at 00:00 on the first day of the month and are retained for twelve months.

Gold

1. Daily backups that run at 00:00 and are retained for seven days.
2. Weekly backups that run at 00:00 every Monday and are retained for four weeks.
3. Monthly backups that run at 00:00 on the first day of the month and are retained for twelve months.

The Block Volume service will not run more than one scheduled backup of a particular resource in one day. When schedules conflict, for example, daily, weekly, and monthly backups are scheduled to run at the same time, the backup with the longest period runs.

Creating a Backup Policy

On Compute Cloud@Customer, 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 on page 431.

Using the Console

When you create a backup policy by using the Console, creating the backup policy schedule is a separate step. A policy can't be assigned to a resource until the policy schedule is defined.

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Backup Policies**.
2. Select **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 one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.

4. Select **Create Backup Policy**.

   A policy can't be assigned to a resource until at least one schedule is defined.
5. Select the name of the new backup policy to go to the details page for the policy.
6. On the policy details page, under Resources, select **Add Schedule**.
7. 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. Select **Add Schedule**.

   The policy can now be assigned to one or more resources. See Assigning a Backup Policy to a Volume or Volume Group on page 428.

   A policy can have more than one schedule. To add another schedule to this policy, select **Add Schedule** again. See the schedule notes in Managing Backup Policies on page 426.

Using the CLI

Use the oci bv volume-backup-policy create command and required parameters to create a new user defined backup policy.

```
oci bv volume-backup-policy create --compartment-id <compartment_OCID>
 --display-name <discriptive_name> --schedule <json_string> or
 file://<path_to_JSON_file> [OPTIONS]
```

Once the policy is created, it can be applied to one or more volumes and volume groups. See Assigning a Backup Policy to a Volume or Volume Group on page 428.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateVolumeBackupPolicy operation to create a new user defined backup policy.

Once the policy is created, it can be applied to one or more volumes and volume groups. See Assigning a Backup Policy to a Volume or Volume Group on page 428.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Assigning a Backup Policy to a Volume or Volume Group

On Compute Cloud@Customer, after creating a backup policy, you assign the policy to one or more volumes or volume groups.

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 Console or specify the `--backup-policy-id` option in the 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 on page 429.

> **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 Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Block Volumes**, **Boot Volumes**, or **Volume Groups**.
2. At the top of the page, select the compartment that contains the volume or volume group to which you want to assign a backup policy.
3. For the volume or volume group to which you want to assign a backup policy, select the Actions menu (⋮), then select **Assign Backup Policy**.
4. In the **Assign Backup Policy dialog**, select a backup policy from the drop-down list.

   You can't assign an Oracle defined policy to a volume group.
5. Select **Assign Backup Policy**.

Using the CLI

Use the oci bv volume-backup-policy-assignment create command and required parameters to assign a volume backup policy to the specified volume.

```
oci bv volume-backup-policy-assignment create --asset-id <volume_OCID> --
policy-id <backup-policyABC> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateVolumeBackupPolicy operation to assign a volume backup policy to the specified volume.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Removing a Backup Policy Assignment

On Compute Cloud@Customer, you can 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 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-aaa442ac4348",
  "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
$
```

Accessing the Backups

On Compute Cloud@Customer, after a manual or automatic backup takes place, you can use the topics in this section to locate the backup.

> **Note:**
>
> For scheduled or policy-based backups, there might be a delay of up to 5 minutes for backups to show up in the list. This delay applies to viewing backups in the Console, CLI, and API.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select one of the following items:
   - **Block Volume Backups**
   - **Volume Group Backups**

   The manual and automatic backups are displayed.
2. To see the details of a backup, select the backup name.

Using the OCI CLI

Use the oci bv backup list command and required parameters to list the volume backups in the specified compartment.

```
oci bv backup list --compartment-id <compartment_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListVolumeBackups operation to list the volume backups in the specified compartment.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing Backup Policies

On Compute Cloud@Customer, after creating a backup policy, you can use the Console, CLI, or API to view the policy.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **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 top of the page.
3. To see details about a policy, select the policy name. Scroll to the **Resources** section to see the schedules.

Using the CLI

Use the oci bv volume-backup-policy list command and required parameters to list all the volume backup policies available in the specified compartment.

```
oci bv volume-backup-policy list --compartment-id <compartment_OCID>
 [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListVolumeBackupPolicies operation to list all the volume backup policies available in the specified compartment.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Editing a Backup Policy Schedule

On Compute Cloud@Customer, you cannot edit an Oracle defined backup policy. You can only edit user defined backup policies.

Avoid entering confidential information.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Backup Policies**.
2. At the top of the page, select the compartment that contains the backup policy.
3. Select 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, select the Actions menu (⋮), then select **Edit**.
6. Make your changes. See Creating a Backup Policy on page 427 for details.
7. Select **Update Schedule**.

Using the CLI

Use the oci bv volume-backup-policy update command and required parameters to update a user defined backup policy.

> **Caution:**
>
> Running this command replaces all schedule items with the items you specify in the --schedule JSON. Include all the schedule parameters, not just the schedule parameter changes.

```
oci bv volume-backup-policy update --policy-id <backup_policy_OCID> --
schedule <json_string> or file://<path_to_JSON_file> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateVolumeBackupPolicy operation to update a user defined backup policy.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Backup Policy Schedule

On Compute Cloud@Customer, you cannot change an Oracle defined backup policy. You can only change or delete the schedules of user defined backup policies.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Backup Policies**.
2. At the top of the page, select the compartment that contains the backup policy that has the schedule you want to delete.
3. Select the name of the backup policy that has the schedule you want to delete.
4. On the **Backup Policy** page, scroll to the **Resources** section.
5. For the schedule you want to delete in the **Schedules** list, select the Actions menu (⋮), then select **Delete**.
6. Confirm the deletion.

Using the CLI

Use the oci bv volume-backup-policy delete command and required parameters to delete a backup policy.

```
oci bv volume-backup-policy delete \
--policy-id ocid1.volumebackuppolicy.unique_ID --force
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

This task is not available in the API.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Backup Policy

On Compute Cloud@Customer, you cannot delete an Oracle defined backup policy. You can only delete user defined backup policies.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Block Storage**, then select **Backup Policies**.
2. At the top of the page, select the compartment that contains the backup policy.
3. For the policy you want to delete, select the Actions menu (⋮), then select **Delete**.
4. Confirm the deletion.

Using the CLI

Use the oci bv volume-backup-policy delete command and required parameters to delete a user defined backup policy.

```
oci bv volume-backup-policy delete --policy-id <policy_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteVolumeBackupPolicy operation to delete a user defined backup policy.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

File Storage

On Compute Cloud@Customer, the File Storage service provides a durable, scalable, secure network file system. You can connect to a File Storage service file system from any Compute Cloud@Customer compute instance in your Virtual Cloud Network (VCN).

File Storage Overview

The File Storage service supports these protocols:

- Network File System version 4.1 (NFSv4.1)
- Network File System version 4.0 (NFSv4)
- Network File System version 3.0 (NFSv3)
- Server Message Block (SMBv2 - SMBv3.1) – Requires Active Directory

File Storage Connectivity

You can connect to a File Storage service file system from any instance in your Virtual Cloud Network (VCN).

Suitable Workloads

The File Storage service is designed to meet the needs of applications and users that need an enterprise file system across a wide range of use cases, including the following:

- **General Purpose File Storage:** Access to a pool of file systems to manage growth of structured and unstructured data.
- **Big Data and Analytics:** Run analytic workloads and use shared file systems to store persistent data.
- **Lift and Shift of Enterprise Applications:** Migrate existing applications that need NFS storage, such as Oracle Cloud Infrastructure E-Business Suite and PeopleSoft.
- **Databases and Transactional Applications:** Run test and development workloads with Oracle Cloud Infrastructure, MySQL, or other databases.
- **Backups, Business Continuity, and Disaster Recovery:** Host a secondary copy of relevant file systems from on premises to the cloud for backup and disaster recovery purposes.
- **Portable OS Interface (POSIX)-compliant file system**
- **MicroServices and Docker:** Deliver stateful persistence for containers. Easily scale as your container-based environments grow.

Data Protection with 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 data protection, you can use a tool that supports NFS to copy your data to a different file system, object storage, or remote location.

For best performance, use the parallel tar (`partar`) and parallel copy (`parcp`) tools provided in the File Storage Parallel File Toolkit for this purpose. These tools work best with parallel workloads and requests. The Parallel File Toolkit is available for Oracle Linux, Red Hat Enterprise Linux, and CentOS. You can use `rsync` or regular `tar` for other OS types. See Installing the Parallel File Tools for more information.

File Storage Objects

Mount Target

A mount target is an NFS endpoint in a subnet of your choice. The mount target provides the IP address that's used in the mount command when connecting NFS clients to a file system. File systems are exported (made available) through mount targets.

For an instance to mount a file system, the instance's Virtual Cloud Network (VCN) must have a mount target. A VCN can only have one mount target.

You can reuse the same mount target to make many file systems available. To reuse the same mount target for multiple file systems, create an export in the mount target for each file system.

Export

Exports control how NFS clients access file systems when they connect to a mount target. File systems are exported (made available) through mount targets. Each mount target maintains an export set which contains one or many exports. A file system must have at least one export in one mount target for compute instances to mount the file system.

Export Set

An export set is a collection of one or more exports that control what file systems the mount target exports and how those file systems are found using the NFS mount protocol. Each mount target has an export set. Each file system associated with the mount target has at least one export in the export set.

Export Path

The export path uniquely identifies the file system within the mount target. The export path is used by a compute instance to mount (logically attach to) the file system. For more information, see File Storage Paths on page 435.

Export Options

NFS export options are a set of parameters within the export that specify the level of access granted to NFS clients when they connect to a mount target. An NFS export options entry within an export defines access for a single IP address or CIDR block range.

File System

In Compute Cloud@Customer, file system refers to a file system that's accessed by one or more clients over the network. 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 that has mounted (has access to) the file system.

Data is added to a file system from the client that has mounted (has access to) the file system. A file system can have a quota of up to 8 petabytes.

The total number of file systems is limited to 100 per tenancy.

Virtual Cloud Network (VCN)

A private network that you set up in Compute Cloud@Customer, with firewall rules and specific types of communication gateways that you can choose to use. A VCN covers a single, contiguous IPv4 CIDR block of your choice.

Subnet

Subnets are subdivisions you define in a VCN (for example, 10.0.0.0/24 and 10.0.1.0/24). Subnets contain virtual network interface cards (VNICs), which attach to compute instances. A subnet consists of a contiguous range of IP addresses that don't overlap with other subnets in the VCN.

Security Rules

Security rules are virtual firewall rules for your VCN. Your VCN comes with a default security list, and you can add more. These security lists provide ingress and egress rules that specify the types of traffic allowed in and out of the compute instances. You can choose whether a particular rule is stateful or stateless. Security list rules must be set up so that clients can connect to file system mount targets.

Another method for applying security rules is to set them up in a network security group (NSG), and then add the mount target to the NSG. Unlike security list rules that apply to all VNICs in the subnet, NSGs apply only to resource VNICs you add to the NSG.

Snapshots

Snapshots provide a consistent, point-in-time view of your file system, and you can take as many snapshots as you need. Each snapshot reflects only data that changed from the previous snapshot.

File Storage Paths

The File Storage service uses these kinds of paths:

- **Export Paths** are part of the information contained in an export that makes a file system available through a mount target.

  The export path is automatically generated when you create an export, and it uniquely identifies the file system within the mount target.

  Note – When you create an export from the CLI, you must specify a `--path` <*path*> argument. The path you specify is recorded but not used for mounting file systems. Compute Cloud@Customer auto-generates a path that's used to mount the file system.

  Export path syntax:

  ```
  /export/<file-system-OCID-unique-string>
  ```

  where:

  - /export/ – Is the beginning of the export path.
  - <*file-system-OCID-unique-string*> – Is the unique character string portion of the file system's OCID.

  For example, a file system with this OCID . . .

  ```
  ocid1.filesystem.oc1.pca.d0v812zdp48onybubehhx1c67i4p3mjfth5avt3z2rkn50uqpbce3fhsa8nm
  ```

  . . . has an export path that looks like this:

  ```
  /export/d0v812zdp48onybubehhx1c67i4p3mjfth5avt3z2rkn50uqpbce3fhsa8nm
  ```

  The export path is used by a file system client to mount (logically attach to) the file system. This path is unrelated to any path within the file system or the client instance. It exists solely as a way to distinguish one file system from another within a single mount target.

  Example of an export path in a client's mount command:

  ```
  sudo mount -t nfs  \
   -o nfsvers=4.0 192.0.2.0:/export/
  d0v812zdp48onybubehhx1c67i4p3mjfth5avt3z2rkn50uqpbce3fhsa8nm /mnt/fs
  ```

  In this mount command example, `192.0.2.0` is the mount target IP address.`/export/d0v812zdp48onybubehhx1c67i4p3mjfth5avt3z2rkn50uqpbce3fhsa8nm` is the unique export path that was specified when the file system was associated with a mount target during creation.

  Export paths can't be edited after the export is created.

- **Mount Point Paths** are paths within a client instance to a locally accessible directory to which the remote file system is mounted.

  In this mount command example, `/mnt/fs` is the path to the directory on the client instance on which the external file system is mounted.

  ```
  sudo mount -t nfs  \
   -o nfsvers=4.0 192.0.2.0:/export/
  d0v812zdp48onybubehhx1c67i4p3mjfth5avt3z2rkn50uqpbce3fhsa8nm /mnt/fs
  ```

- **File System Paths** are paths to directories within the file system, and contain the contents of the file system. When the file system is mounted, you can create any directory structure within it.

VCN Security Rules for File Storage

On Compute Cloud@Customer, 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:

- Open Network Computing Remote Procedure Call (ONC RPC) rpcbind utility protocol
- Network File System (NFS) protocol
- Network File System (MOUNT) protocol
- Network Lock Manager (NLM) protocol

Ways to Enable Security Rules for File Storage

The Networking service offers two virtual firewall features that both use security rules to control traffic at the packet level. The two features are:

- **Security lists:** The original virtual firewall feature from the Networking service. When you create a VCN, a default security list is also created. Add the required rules to the security list for the subnet that contains the mount target.
- **Network security groups (NSGs):** A subsequent feature designed for application components that have different security postures. Create an NSG that contains the required rules, and then add the mount target to the NSG. Each mount target can belong to up to five (5) NSGs.

> **Important:**
>
> You can use security lists alone, network security groups alone, or both together. It depends on your particular security needs.
>
> If you choose to use both security lists and network security groups, the set of rules that applies to a given mount target VNIC is the combination of these items:
>
> - The security rules in the security lists associated with the VNIC subnet
> - The security rules in all NSGs that the VNIC is in
>
> It doesn't matter which method you use to apply security rules to the mount target VNIC, as long as the ports for protocols necessary for File Storage are correctly configured in the rules applied.

For additional conceptual information, see Virtual Firewall on page 147.

For instructions on how to create security rules and NSGs for the File Storage service, see Controlling Access to File Storage on page 450.

For general instructions for creating security lists and NSGs, see Controlling Traffic with Security Lists on page 158.

File Storage Network Ports

On Compute Cloud@Customer, when configuring the File Storage service, you might need to know the ports that are used for a service or protocol.

This table lists the port numbers used for networking services that support File Storage.

When you configure security lists and network security groups for the File Storage service, use the port numbers for the particular service you are using. For example, for NFS configure ports 2048–2050 for TCP and UDP.

Service	Protocol	Port
RPC	TCP, UDP	111
NFS	TCP, UDP	2048, 2049, 2050
lockd	TCP, UDP	4045
mountd	TCP, UDP	20048
SMB	TCP, UDP	445
LDAP	TCP, UDP	389
Kerberos	TCP, UDP	88
DNS	TCP, UDP	53

Deleting File System Resources Overview

On Compute Cloud@Customer, you can't delete a file system resource that has dependencies.

File Systems

You can delete a file system if it's not a parent file system. If it's a parent file system, all descendant snapshots and clones must first be deleted.

Snapshots

A parent snapshot can't be deleted. A snapshot that isn't a parent can be deleted.

Clones

A parent clone can't be deleted. A clone that isn't a parent can be deleted.

Export Options for File Storage

On Compute Cloud@Customer, NFS export options enable you to create more granular access control than is possible using security list rules to limit VCN access. You can use NFS export options to specify access levels for IP addresses or CIDR blocks connecting to file systems through exports in a mount target. Access can be restricted so that each client’s file system is inaccessible and invisible to the other, providing better security controls in multitenant environments.

Using NFS export option access controls, you can limit clients' ability to connect to the file system and view or write data. For example, if you want to allow clients to consume but not update resources in your file system, you can set access to read-only. You can also reduce client root access to your file systems and map specified User IDs (UIDs) and Group IDs (GIDs) to a single anonymous UID/GID of your choice.

Export Options

Exports control how NFS clients access file systems when they connect to a mount target. File systems are exported (made available) through mount targets.

NFS export options are a set of parameters within the export that specify the level of access granted to NFS clients when they connect to a mount target. An NFS export options entry within an export defines access for a single IP address or CIDR block range. You can have up to 100 options per file system.

Each separate client IP address or CIDR block you want to define access for needs a separate export options entry in the export. For example, if you want to set options for NFS client IP addresses 10.0.0.6, 10.0.0.08, and 10.0.0.10, you need to create three separate entries, one for each IP address.

When there is more than one export using the same file system and same mount target, the export options that are applied to an instance are the options with a source that most closely matches the instance IP address. The smallest (most specific) match takes precedence across all the exports. Therefore, you can determine which export options are applied to an instance by looking at source value of all the exports.

For example, consider the following two export options entries specifying access for an export:

Entry 1: Source: 10.0.0.8/32, Access: Read/Write

Entry 2: Source: 10.0.0.0/16, Access: Read-only

In this case, clients who connect to the export from IP address 10.0.0.8 have read/write access. When there are multiple export options, the most specific match is applied.

> **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 Doc ID 2823994.1.

> **Important:**
>
> File systems can be associated with one or more exports, contained within one or more mount targets.
>
> If the client source IP address does not match any entry on the list for a single export, then that export is not visible to the client. However, the file system could be accessed through other exports on the same or other mount targets. To completely deny client access to a file system, be sure that the client source IP address or CIDR block is not included in any export for any mount target associated with the file system.

For information about how to configure export options for various file sharing scenarios, see NFS Access Control Scenarios on page 439.

For more information about configuring export options, refer to the section titled Setting NFS Export Options on page 451.

NFS Export Option Defaults

When you create a file system and export, the NFS export options for that file system are set to the following defaults, which allow full access for all NFS client source connections. These defaults must be changed if you want to restrict access:

Note – When using the CLI to set the export options, if you set the options with an empty array (no options specified), the export is not accessible to any clients.

Export Option in the Console	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 Console	Export Option in the CLI	Default Value	Description
Ports:	`require-privileged-source-port`	Any	Always set to: • Console: 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. These are the 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.

NFS Access Control Scenarios

On Compute Cloud@Customer, Learn different ways to control NFS access by reviewing some scenarios.

- Scenario A: Control Host Based Access on page 439: Provides a managed environment for two clients. The clients share a mount target, but each has their own file system, and can't access each others data.
- Scenario B: Limit the Ability to Write Data on page 440: Provides data to clients for consumption, but doesn't allow them to update the data.
- Scenario C: Improve File System Security on page 441: Increases security by limiting the root user's privileges when connecting to a file system.

Scenario A: Control Host Based Access

On Compute Cloud@Customer, provide a managed hosted environment for two clients. The clients share a mount target, but each has their own file system, and can't access each other's data.

For example:

- Client A is assigned to CIDR block 10.0.0.0/24, and requires read/write access to file system A, but not file system B.
- Client B is assigned to CIDR block 10.1.1.0/24, and requires read/write access to file system B, but not file system A.
- Client C is assigned to CIDR block 10.2.2.0/24, and has no access of any kind to file system A or file system B.
- Both file systems A and B are associated with a single mount target, MT1. Each file system has an export contained in the export set of MT1.

Because Client A and Client B access the mount target from different CIDR blocks, you can set the client options for both file system exports to allow access to only a single CIDR block. Client C is denied access by not including its IP address or CIDR block in the NFS export options for any export of either file system.

Console Example

Set 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 aren't included in this CIDR block, and can't access the file system.

> **Note:**
>
> To learn how to access the NFS export options in the Console see Setting NFS Export Options on page 451.

Source	Ports	Access	Squash	Squash UID/GID
10.0.0.0/24	Any	Read/Write	None	(not used)

Set the export options for file system B to allow read/write access only to Client B, who is assigned to CIDR block 10.1.1.0/24. Client A and Client C aren't included in this CIDR block, and can't access the file system.

Source	Ports	Access	Squash	Squash UID/GID
10.1.1.0/24	Any	Read/Write	None	(not used)

CLI Example

> **Note:**
>
> To learn how to access the NFS export options in the CLI, see Setting NFS Export Options on page 451.

Set 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 aren't included in this CIDR block, and can't 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"}]'
```

Set the export options for file system B to allow `Read_Write` access only to Client B, who is assigned to CIDR block 10.1.1.0/24. Client A and Client C are not included in this CIDR block, and can't access the file system.

```
oci fs export update --export-id <File_system_B_export_ID> --export-options
 \
'[{"source":"10.1.1.0/24 ","require-privileged-source-
port":"false","access":"READ_WRITE","identity-squash":"NONE","anonymous-
uid":"65534","anonymous-gid":"65534"}]'
```

Scenario B: Limit the Ability to Write Data

On Compute Cloud@Customer, provide data to customers for consumption, but don't allow them to update the data.

For example, you'd like to publish a set of resources in file system A for an application to consume, but not change. The application connects from IP address 10.0.0.8.

Console Example

Set the source IP address 10.0.0.8 to read-only in the export for file system A.

> **Note:**
>
> To learn how to access the NFS export options in the Console see Setting NFS Export Options on page 451.

Source	Ports	Access	Squash	Squash UID/GID
10.0.0.8	Any	Read-only	None	(not used)

CLI Example

> **Note:**
>
> To learn how to access the NFS export options in the CLI, see Setting NFS Export Options on page 451.

Set the source IP address 10.0.0.8 to `READ_ONLY` in the export for file system A.

```
oci fs export update --export-id <File_System_A_export_OCID> --export-
options \
'[{"source":"10.0.0.8","require-privileged-source-
port":"false","access":"READ_ONLY","identitysquash":"NONE","anonymousuid":"65534","anony
```

Scenario C: Improve File System Security

On Compute Cloud@Customer, to increase security, you can limit the root user's privileges when connecting to File System A. Use Identity Squash to remap root users to UID/GID 65534.

In UNIX-like systems, this UID/GID combination is reserved for '*nobody*', a user with no system privileges.

Console Example

Set the source IP address 10.0.0.8 to read-only in the export for file system A.

> **Note:**
>
> To learn how to access the NFS export options in the Console see Setting NFS Export Options on page 451.

Source	Ports	Access	Squash	Squash UID/GID
0.0.0.0/0	Any	Read/Write	Root	65534

CLI Example

> **Note:**
>
> To learn how to access the NFS export options in the CLI, see Setting NFS Export Options on page 451.

```
oci fs export update --export-id <File_System_A_export_OCID> --export-
options   \
'[{"source":"0.0.0.0/0","require-privileged-source-
port":"false","access":"READ_WRITE","identitysquash":"ROOT","anonymousuid":"65534","anon
```

Creating a File System, Mount Target, and Export

On Compute Cloud@Customer, you can use this task flow 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 on page 442
2.	Create the file system.	Creating a File System on page 445
3.	Create a file system export in the mount target.	Creating an Export for a File System on page 448
4.	Enable Security Rules for File Storage.	Controlling Access to File Storage on page 450
5.	Change NFS export options to control access to the file system.	Setting NFS Export Options on page 451

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 on page 453.

Creating a Mount Target

On Compute Cloud@Customer, 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's used in the mount command when connecting NFS clients to a file system.

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 *Backing store pool* in Creating a File System on page 445. Two mount targets in a VCN are counted as one with regard to resource limits.

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:**
>
> Don't use /30 or smaller subnets for mount target creation because they might not have enough 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 on page 140
- (Required for Mounting File Systems Across Compute Cloud@Customer Infrastructures on page 449) A Dynamic Routing Gateway (DRG) with a route rule in the VCN. See Connecting to the On-Premises Network through a Dynamic Routing Gateway (DRG) on page 174.
- (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 on page 450.

  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.
- To ensure that the mount target is created using high performance storage, first configure the OraclePCA tag namespace and backing store pool (`poolName`) attribute in your OCI tenancy. See Using Free-Form Tags for Extended Functionality on page 120. Then when you create the mount target, assign the OraclePCA tag namesapce and `poolName` attribute to the mount target. 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.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **File Storage**, then select **Mount Targets**.
2. 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, select 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.
   - Select the **Select Tag(s)** tab. The mount target must be in the same backing store pool that's 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 no **OraclePCA.poolName** tag, then the mount target is in the default pool. For more information about these tags, see Using Free-Form Tags for Extended Functionality on page 120.

   If the mount target meets your needs, skip this procedure and go to Creating a File System on page 445.
3. Select **Create Mount Target**.
4. 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 where you want to create 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, seeControlling Access to File Storage on page 450 and Configuring VCN Security Rules for File Storage on page 450.

- (Optional) Add one or more tags to this resource. For more information about tagging resources, see Resource Tags.

  By default, the mount target is created on the default storage pool. 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**. The poolName property can be set only when the mount target is created. You can't set or change this property value after the mount target is created.

5. Select **Create Mount Target**.

   Next, create a file system. See Creating a File System on page 445.

Using the CLI

Use the oci fs mount-target create command and required parameters to create a new mount target in the specified compartment and subnet.

```
oci fs mount-target create --availability-domain <availability_domain_name>
 --compartment-id <compartment_OCID>--subnet-id <subnet_OCID> --display-
name <name_to_assign_to_mount-target> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Procedure

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.

   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. The `poolName` property can be set only when the mount target is created. You can't 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 on page 445.

Using the API

Use the CreateMountTarget operation to create a new mount target in the specified compartment and subnet.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Creating a File System

On Compute Cloud@Customer, you can create a shared file system using the File Storage service.

(Optional) To create a file system with specific values for the following attributes, first configure the attributes as described in Using Free-Form Tags for Extended Functionality on page 120. Then use the following information when you create the file system.

- **Quota**

  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 changed with the file system `update` command.

  To apply a quota, during file system creation, assign this defined tag:

  - **Tag namespace:** `OraclePCA`
  - **Tag name:** `quota`
  - **Value:** *<quota value in gigabytes>*
- **Database record size**

  The default database record size is `131072` bytes. You can specify one of the following values (in bytes): `512`, `1024`, `2048`, `4096`, `8192`, `16384`, `32768`, `65536`, `131072`, `262144`, `524288`, `1048576`. The record size can be set only when the file system is created. You can't set or change this property value later.
- To use a nondefault value, during file system creation, assign this defined tag:
  - **Tag namespace:** `OraclePCA`
  - **Tag name:** `databaseRecordSize`
  - **Value** (select one): `512`, `1024`, `2048`, `4096`, `8192`, `16384`, `32768`, `65536`, `131072`, `262144`, `524288`, `1048576`
- **Backing store pool**

By default, the backing store of a file system is the default pool of the attached ZFS Storage Appliance, specified as **PCA_POOL**. You can specify `PCA_POOL_HIGH` to indicate that you want to use a high performance pool for the backing store. Before you specify `PCA_POOL_HIGH`, verify that a high performance pool is available. This property can be set only when the file system is created. You can't set or change this property value later.

To use a nondefault value, during file system creation, assign this defined tag:

- **Tag namespace:** `OraclePCA`
- **Tag name:** `databaseRecordSize`
- **Value** (select one): `PCA_POOL` (default) or `PCA_POOL_HIGH`

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **File Storage**, then select **File Systems**.
2. 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.
3. Select **Create File System**.
4. In the **Create File System** dialog box, 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 one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.

     If the OraclePCA tag namespace and OraclePCA tag key definitions for file systems are configured in your OCI tenancy (see Using Free-Form Tags for Extended Functionality on page 120), you can specify the **OraclePCA** Tag Namespace, select a key and a value. Your file system is created with the applicable attribute. The OraclePCA tag namespace can't be added later.
5. Select **Create File System**.

   The file system is created.

   Next, create an export for the file system. See Creating an Export for a File System on page 448.

Using the CLI

Use the oci fs file-system create command and required parameters to create a new file system in the specified compartment and availability domain.

```
oci fs file-system create --availability-domain <availability_domain_name>
  --compartment-id <compartment_id> --display-name <fs_display_name>
 [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Procedure

1. Gather the information that you need to run the command:
   - Availability Domain Name (`oci iam availability-domain list`)
   - Compartment OCID (`oci iam compartment list`)
   - File System Name: The display name you want assigned to this file system
2. To use nondefault values for the quota, database record size, or backend store pool, specify the appropriate tags to set the values for these attributes. See Adding Tags at Resource Creation (IAM in OCI) on page 118.

   See the following step for an example of setting these values.
3. Run the create file system command.

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 on page 448.

Using the API

Use the CreateFileSystem operation to create a new file system in the specified compartment and availability domain.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Creating an Export for a File System

On Compute Cloud@Customer, 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 Console

1. In the Compute Cloud@Customer Console navigation menu, select **File Storage**, then select **File Systems**.
2. In the left panel, select **File Systems**.
3. At the top of the page, select the compartment that contains the file system for which you plan to create an export.
4. Select the name of the file system.
5. Select **Create Export**.
6. Enter the required 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 10.0.0.0/29 must be added first.
7. Select **Create Export**.

   The file system export is created and the export details page is displayed.
8. In the export details page, copy the export path. The export path is used to mount the file system on an instance.
9. In the lower panel, 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 to restrict access.
10. Consider your next action:
    - Mount the file system from an NFS client. See Mounting File Systems on UNIX-based Instances on page 453.
    - Configure NFS options to secure the exported file system. See Setting NFS Export Options on page 451.

Using the CLI

Use the oci fs export create command and required parameters to create a new export in the specified export set, path, and file system.

```
oci fs export create --export-set-id <export_set_OCID> --file-system-
id <file_system_OCID>
--path "</pathname>" [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateExport operation to create a new export in the specified export set, path, and file system.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Mounting File Systems Across Compute Cloud@Customer Infrastructures

You can create a file system on one Compute Cloud@Customer infrastructure, and mount the file system from an instance that's on another infrastructure. To achieve this scenario, you must configure certain network parameters on each infrastructure.

Restriction

The Compute Cloud@Customer infrastructure hosting the file system and the instance on the remote infrastructure that mounts the file system can't have overlapping VCN CIDR blocks.

On the Infrastructure Hosting the File System

1. Configure these network parameters:
   a. Create a Dynamic Routing Gateway (DRG).

      See Creating a Dynamic Routing Gateway on page 174.
   b. For the VCN subnet that will be used by the mount target, attach the VCN to the DRG.

      See Attaching VCNs to a Dynamic Routing Gateway on page 175.
   c. For the VCN subnet, add a route rule with the DRG as the target, and assign a destination CIDR that matches the remote infrastructure VCN CIDR.

      For example, if the remote 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 Compute Cloud@Customer racks. This configuration isn't required for file system mounts within the same rack.

      See Working with Route Tables on page 155.
2. Create a mount target. See Creating a Mount Target on page 442.
3. Create a file system. See Creating a File System on page 445.
4. Create an export that exports to both the source and remote VCN CIDR.

   The export CIDR must be big enough to cover both VCN subnet CIDRs. For example, if the host VCN CIDR is 10.10.0.0/16, and the remote 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 Compute Cloud@Customer hosting the file system.

   See Creating an Export for a File System on page 448.

On the Remote Infrastructure

1. Configure these network parameters:
   a. Create a Dynamic Routing Gateway (DRG).

      See Creating a Dynamic Routing Gateway on page 174.
   b. For the VCN subnet, attach the VCN to the DRG.

      See Attaching VCNs to a Dynamic Routing Gateway on page 175.
   c. For the VCN subnet, add a route rule with the DRG as the target, and assign a destination CIDR that matches the host VCN CIDR.

For example, if the host 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 Compute Cloud@Customer racks. not required for file system mounts within the same Compute Cloud@Customer rack.

See Working with Route Tables on page 155.

2. Log in to the instance and mount the file system.

   See:

   - Mounting File Systems on UNIX-based Instances on page 453
   - Mounting File Systems on Microsoft Windows Instances on page 457

Controlling Access to File Storage

On Compute Cloud@Customer, 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

Configuring VCN Security Rules for File Storage

On Compute Cloud@Customer, 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, see File Storage Network Ports on page 437.

For more information about managing VCNs and subnets, see Managing VCNs and Subnets on page 140.

Adding File Storage to a Network Security Group

On Compute Cloud@Customer, adding File Storage to a network security group is accomplished by performing several procedures. This section lists the procedures.

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 on page 163
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 on page 451

No.	Description	Links to Procedures
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 on page 185

Adding a Mount Target to a Network Security Group

On Compute Cloud@Customer, 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 Console

1. Ensure that an NSG with ingress and egress rules has been configured. See Configuring VCN Security Rules for File Storage on page 450.
2. Ensure that a mount target is created. See Creating a Mount Target on page 442.
3. In the Compute Cloud@Customer Console navigation menu, select **File Storage**, then select **Mount Targets**.
4. At the top of the page, select the compartment that contains the mount target.
5. Select the mount target name to see the details page.
6. Select **Edit**.
7. Enable Network Security Groups.
8. Select the NSG from the list.
9. Select **Save Changes**.

Using the CLI

Use the oci fs mount-target update command and required parameters to update the specified mount target's information.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

```
oci fs mount-target update --mount-target-id <mount_target_OCID>  --nsg-ids
 '["<nsg1_OCID>","i"]' [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateMountTarget operation to update the specified mount target's information.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Setting NFS Export Options

On Compute Cloud@Customer, when you create a file system and export, the NFS export options for that file system are set to the defaults. 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 Console	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.
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, see Export Options for File Storage on page 437.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **File Storage**, then select **File Systems**.
2. At the top of the page, select the compartment that contains the file system.
3. Select the file system name.
4. Under **Resources**, select **Exports**.
5. Select the export's export path.

   The NFS Export Options are displayed.
6. Select **Edit Options**.
7. In the **NFS Export Options** dialog box, configure the NFS options.
8. Select **Update Options**.

Using the CLI

Use the oci fs export update command and required parameters to update the export information.

```
oci fs export update --export-id <export_id> --export-options <file:// json_file or json_string> [OPTIONS]
```

Note – The `require-privileged-source-port` option can only be set to `false`.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateExport operation to update the export information.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Mounting File Systems on UNIX-based Instances

On Compute Cloud@Customer, 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 on page 442.
- The mount target must have correctly configured security rules or be assigned to an NSG. See Configuring VCN Security Rules for File Storage on page 450.

> **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 haven't 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 OS documentation.

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 Console

1. In the Compute Cloud@Customer Console navigation menu, select **File Storage**, then select **Mount Target**.
2. At the top of the page, select the compartment that contains the mount target.
3. Select the mount target name to see the details page.

   The private IP address is displayed.

Using the CLI

Use the oci fs mount-target get command and required parameters to get mount target information.

```
oci fs mount-target get --mount-target-id <mount_target_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the GetMountTarget operation to get mount target information.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

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 an 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 on page 453.
   - Replace <*fs-export-path*> with the export path that was generated when the export was created. See Creating an Export for a File System on page 448.
   - 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 on page 453.
   - Replace **<*fs-export-path*>** with the export path that was generated when the export was created.

     See Creating an Export for a File System on page 448.
   - 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 an Instance.

2. Create a mount point, if one hasn't 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 on page 453 and Creating an Export for a File System on page 448.
   - **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 on page 454. 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

On Compute Cloud@Customer, 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.

Prerequisite

Using the SMB protocol requires that the Microsoft Windows instances and Compute Cloud@Customer belong to the same Active Directory domain. Before performing any tasks in this section, ensure the following requirements are set up:

- Installed and configured Microsoft Active Directory Federation Services for your organization.
- Set up groups in Active Directory that will map to groups in Compute Cloud@Customer.
- Created users in Active Directory who will sign in to the Console.

> **Note:**
>
> Consider naming Active Directory groups that you intend to map to Compute Cloud@Customer groups with a common prefix to make it easy to apply a filter rule, for example, `CCC_Administrators`, `CCC_NetworkAdmins`, `CCC_InstanceLaunchers`.

- Open an Oracle support request to request an Active Directory configuration in the Compute Cloud@Customer infrastructure. See Creating a Support Request. To access support, sign in to the Oracle Cloud Console as described in Sign In to the OCI Console.

Mounting a File system On a 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 on page 442.
- The mount target must have correctly configured security rules or be assigned to an NSG. See Configuring VCN Security Rules for File Storage on page 450.
- You must know the mount target's IP address. See Obtaining the Mount Target IP Address on page 453.
- 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

1. Log into your Microsoft Windows instance.

   See Connecting to an Instance.
2. Open Microsoft Windows PowerShell and run as Administrator:
   a. Go to Start and open PowerShell.
   b. In PowerShell, type the following to run as Administrator:

   ```
   Start-Process powershell -Verb runAs
   ```

   c. In the User Account Control window, select 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 on page 453)
   - `fs-export-path` with the file system export path (see Creating a File System, Mount Target, and Export on page 442)
   - `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 the Microsoft Windows File Explorer

1. Log into your Microsoft Windows instance.

   See Connecting to an Instance.
2. Open Microsoft Windows PowerShell and run as Administrator:
   a. Go to Start and open PowerShell.
   b. In PowerShell, type the following to run as Administrator:

   ```
   Start-Process powershell -Verb runAs
   ```

   c. In the User Account Control window, select 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. Select Windows Search.
   b. Enter `regedit` in the Search field and press Enter.
   c. Select Yes to allow changes to your device.
   d. Select `HKEY_LOCAL_MACHINE`. Then, browse to: `Software\Microsoft\ClientForNFS\CurrentVersion\Default`.
6. Add a new DWORD32 registry entry for `AnonymousGid`:
   a. Select 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 select 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 on page 453)
- `fs-export-path` with the file system export path (see Creating a File System, Mount Target, and Export on page 442)

Line:

```
\\10.x.x.x\fs-export-path
```

13. Select Finish.

Mounting a File System on a Windows Instance Using SMB

On Compute Cloud@Customer, 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. The SMB process is described here.

> **Note:**
>
> These instructions apply only to file systems created On Compute Cloud@Customer.

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 on page 442.
- The mount target must have correctly configured security rules or be assigned to an NSG. See Configuring VCN Security Rules for File Storage on page 450.
- You must know the mount target's IP address. See Obtaining the Mount Target IP Address on page 453.
- You must be able to log into the Microsoft Windows OS on the instance with superuser or administrator privileges.

Specific Prerequisites for SMB Support

Using the SMB protocol requires that the Microsoft Windows instances and Compute Cloud@Customer belong to the same Active Directory domain.

This procedure assumes that the AD service is already configured in your data center.

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 Compute Cloud@Customer to your AD service, Oracle must add the AD domain name to the Oracle Cloud Infrastructure Active Directory Domain configuration. To request this administrative action, submit a support request. See Creating a Support Request. To access support, sign in to the Oracle Cloud Console as described in Sign In to the OCI Console.

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 Windows Instance Using NFS on page 457.
2. Relax the file system permissions:
   a. Open File Explorer, select the mapped drive and right-select 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. Select 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.
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 on page 448).

     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. Select Finish.
8. When prompted, provide the user name and password of the AD domain user used for mapping the network drive.
9. Select 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

On Compute Cloud@Customer, 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's 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 Doc ID 2823994.1.

For instructions to create a mount target, see Creating a File System, Mount Target, and Export on page 442.

This section provides instructions for administering mount targets.

Listing Mount Targets and Viewing Details

On Compute Cloud@Customer, you can list File System mount targets.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **File Storage**, then select **Mount Targets**.
2. At the top of the page, select the compartment that contains the mount target.

   The mount targets are displayed.
3. To see the mount target details, select the mount target name.

Using the CLI

Listing Mount Targets

Use the oci fs mount-target list command and required parameters to list the mount target resources in the specified compartment.

```
oci fs mount-target list --availability-domain <availability_domain_name>
 --compartment-id <compartment_id> [OPTIONS]
```

Listing Details About a Mount Target

Use the oci fs mount-target get command and required parameters to get the specified mount target's information.

```
oci fs mount-target get --mount-target-id <mount-target_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListMountTargets operation to list the mount target resources in the specified compartment.

Use the GetMountTarget operation to get the specified mount target's information.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Updating a Mount Target

On Compute Cloud@Customer, when you update a mount target, you can set or change the Network Security Groups and display name. You can't change the backing store pool type assignment (default or high performance).

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **File Storage**, then select **Mount Targets**.
2. At the top of the page, select the compartment that contains the mount target.
3. Select the Actions menu (⋮) for the mount target, and select **Edit**.
4. Make changes.
5. Select **Save**.

Using the CLI

Use the oci fs mount-target update command and required parameters to change the mount target.

```
oci fs mount-target update --mount-target-id <mount_target_OCID> <parameters
 to change> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateMountTarget command and required parameters to update a mount target.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Listing Exports

On Compute Cloud@Customer, you can list File System exports.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **File Storage**, then select **Mount Targets**.
2. At the top of the page, select the compartment that contains the mount targets.
3. Select the mount target name.

   The exports are displayed at the bottom of the page.
4. To see the export details, select the export name.

Using the CLI

Use the oci fs export list command and required parameters to list exports for the specified compartment.

```
oci fs export list --compartment-id <compartment_id> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListExports operation to list exports for the specified compartment.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Listing Export Sets

On Compute Cloud@Customer, you can list File System export sets using the CLI and API.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci fs export-set list command and required parameters to list the export set resources in the specified compartment.

```
oci fs export-set list --availability-domain <availability_domain_name> --
compartment-id <compartment_id> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Procedure

1. Get the compartment where you want to list export sets (`oci iam compartment list`)
2. Run the command.

   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"
    }
  ]
}
```

Using the API

Use the ListExportSets operation to list the export set resources in the specified compartment.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting an Export

On Compute Cloud@Customer, 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 Console

1. In the Compute Cloud@Customer Console navigation menu, select **File Storage**, then select **File Systems**.
2. At the top of the page, select the compartment that contains the file system with the export you plan to delete.
3. Select the name of a file system that uses the export you plan to delete.
4. Select the Actions menu (⋮) for the export, then select **Delete**.
5. Confirm the deletion.

Deleting an Export Using the CLI

Use the oci fs export delete command and required parameters to delete the specified export.

```
oci fs export delete --export-id <export_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Deleting an Export Using the API

Use the DeleteExport operation to delete the specified export.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Moving a Mount Target to a Different Compartment

On Compute Cloud@Customer, you can move a Mount Target to a different compartment.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci fs mount-target change-compartment command and required parameters to move a mount target to a different compartment.

```
oci fs mount-target change-compartment --mount-target-id <mount_target_OCID>
 --compartment-id <destination_compartment_OCID>
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

This task is not available in the API.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Mount Target

On Compute Cloud@Customer, deleting a mount target deletes all the exports that are associated with the mount target.

> **Caution:**
>
> Deleting the mount target also deletes all of its exports of associated file systems. File systems are no longer available through the deleted mount target.
>
> **Deleting a mount target has no effect on file system data or file system snapshots.**

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **File Storage**, then select **Mount Targets**.
2. At the top of the page, select the compartment that contains the mount target that you want to delete.
3. Select the Actions menu (⋮) for the mount target you plan to delete, then select **Delete**.
4. Confirm the deletion.

Using the CLI

Use the oci fs mount-target delete command and required parameters to delete the specified mount target.

```
ci fs mount-target delete --mount-target-id <mount_target_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteMountTarget command and required parameters to delete the specified mount target.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Managing File Systems

On Compute Cloud@Customer, a file system in the File Storage service represents a network file system that's 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're created. For instructions to create a file system, see Creating a File System, Mount Target, and Export on page 442.

Listing File Systems

On Compute Cloud@Customer, file systems are associated with a single compartment. When you select a compartment, the Console displays all file systems in the compartment. You can also see exports and snapshots associated with each file system.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **File Storage**, then select **File Systems**.
2. At the top of the page, select the compartment that contains the file system.

The file systems for the compartment are listed.

3. To see file system details, select the name of the file system.

Using the CLI

Use the oci fs file-system list command and required parameters to list file systems.

```
oci fs file-system list --availability-domain <availability_domain_name> --
compartment-id <compartment_OCID> [OPTIONS]
```

Use the oci fs file-system get command and required parameters to view file system details.

```
oci fs file-system get --file-system-id <file-system_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListFileSystems operation to list file systems.

Use the GetFileSystem operation to get more information about the specified file system.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Updating a File System

On Compute Cloud@Customer, you can change some file system parameters.

You can change the file system name and quota. You can't 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 can't 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 using the Compute Cloud@Customer Console or `metered-bytes` in the file system `get` output using the 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 the Compute Cloud@Customer Console or in `defined-tags` in the file system `get` output in the CLI. If the quota is less than metered bytes, then the quota isn't enforced and you should set a higher quota.

Avoid entering confidential information.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **File Storage**, then select **File Systems**.
2. At the top of the page, select the compartment that contains the file system.
3. For the file system you want to update, select the Actions menu (⋮), then select **Edit**.
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 on page 445.
5. Select **Save Changes**.

Using the CLI

Use the oci fs file-system update command and required parameters to update the specified file system's information. You can use this operation to rename a file system.

```
oci fs file-system update --file-system-id <file_system_OCID> --display-
name <new_file-system_name> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Procedure

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 on page 445.
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}}'
   ```

Using the API

Use the UpdateFileSystem operation to update the specified file system's information. You can use this operation to rename a file system.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Moving a File System to a Different Compartment

On Compute Cloud@Customer, you can move a file system to another compartment using the CLI or API.

The file system is moved immediately. Moving a file system doesn't affect mounted instances.

For information about moving resources between compartments, see Moving Resources to a Different Compartment.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci fs file-system change-compartment command and required parameters to move a file system and its associated snapshots into a different compartment within the same tenancy.

```
oci fs file-system change-compartment --file-system-id <file-system_OCID> --
compartment-id <destination_compartment_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ChangeFileSystemCompartment operation to move a file system and its associated snapshots into a different compartment within the same tenancy.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a File System

On Compute Cloud@Customer, you can delete a file system that doesn't have any dependencies.

> **Caution:**
>
> You can't undo this operation. Any data in a file system is permanently deleted with the file system. You can't recover a deleted file system or its snapshots.

A file system that has an export can't be deleted. To delete the export, see Deleting an Export on page 464.

You can't 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 can't delete the source file system.

For more information, see Deleting File System Resources Overview on page 437.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **File Storage**, then select **File Systems**.
2. At the top of the page, select the compartment that contains the file system.
3. Select the Actions menu (⋮) for the file system and select **Delete**.
4. Confirm the deletion.

Using the CLI

Use the oci fs file-system delete command and required parameters to delete the specified file system. Before you delete the file system, verify that no remaining export resources still reference it.

```
oci fs file-system delete --file-system-id<file-system_OCID>[OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteFileSystem operation to delete the specified file system. Before you delete the file system, verify that no remaining export resources still reference it.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Managing File System Snapshots

On Compute Cloud@Customer, the File Storage service supports snapshots for file system data protection.

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.

The snapshot is accessible at `.zfs/snapshot/`*name*.

Listing Snapshots

On Compute Cloud@Customer, you can list snapshots and view snapshot details.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **File Storage**, then select **File Systems**.
2. At the top of the page, select the compartment that contains the file system with the snapshot.
3. Select the file system name.
4. Under **Resources**, select **Snapshots**.

   The file system snapshots are listed.
5. To get the details for a specific snapshot, select the snapshot name.

Using the CLI

Use the oci fs snapshot list command and required parameters to list snapshots of the specified file system.

```
oci fs snapshot list --file-system-id <file-system_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListSnapshots operation to list snapshots of the specified file system.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Creating a Snapshot

On Compute Cloud@Customer, 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*.

Avoid entering confidential information.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **File Storage**, then select **File Systems**.
2. At the top of the page, select the compartment that contains the file system for which you want to create a snapshot.
3. Select the file system name.
4. Under **Resources**, select **Snapshots**.
5. Select **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. Select **Create Snapshot**.

   The snapshot is accessible under the root directory of the file system at `.zfs/snapshot/`*name*.

Using the CLI

Use the oci fs snapshot createoci_fs_snapshot_create command and required parameters to create a new snapshot of the specified file system.

```
oci fs snapshot create --file-system-id <file-system_OCID> --
name <snapshot_name> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateSnapshot operation to create a new snapshot of the specified file system.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Accessing a Snapshot on the Mounted File System

On Compute Cloud@Customer, 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*`>`.

From a UNIX OS Client

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)

On Compute Cloud@Customer, 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

On Compute Cloud@Customer, you can delete file system snapshots.

There are dependencies between file systems, snapshots, and clones. Compute Cloud@Customer doesn't allow you to delete any resources for which there is a dependency. For details, see Deleting File System Resources Overview on page 437.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **File Storage**, then select **File Systems**.
2. At the top of the page, select the compartment that contains the file system snapshot that you want to delete.
3. Select the name of the file system where the snapshot resides.
4. Under **Resources**, select **Snapshots**.
5. Select the Actions menu (⋮), then select **Delete**.
6. Confirm the deletion.

Using the CLI

Use the oci fs snapshot delete command and required parameters to delete the specified snapshot.

```
oci fs snapshot delete --snapshot-id <snapshot_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteSnapshot operation to delete the specified snapshot.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Managing File System Clones

On Compute Cloud@Customer, 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 on page 465.

Creating a File System Clone

On Compute Cloud@Customer, you can create a file system clone by cloning the file system snapshot.

Prerequisite

A snapshot of the file system must exist. See Creating a Snapshot on page 469.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **File Storage**, then select **File Systems**.
2. At the top of the page, select the compartment that contains the file system you want to clone.
3. Select the name of the file system that you want to clone.
4. Under **Resources**, select **Snapshots**.

5. For the snapshot that you want to clone, Select the Actions menu (⋮), and select **Clone**.
6. In the **Clone Snapshot from File System** dialog box, provide the following information:
   - **Name**: Enter a name for the clone. Avoid entering confidential information.
   - **Create in Compartment**: Select the compartment where you want to create the clone.
   - **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
7. Select **Create File System**.

   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 selectable, and the name of the **Source Snapshot** is shown and is selectable.

   If you select the name of the parent file system, the **Clones** section of the details page of that parent file system shows **Clone Parent** is true, and **Parent File System** and **Source Snapshot** have no values.

Using the CLI

Use the oci fs file-system create command and required parameters to create a file system clone of a file system snapshot.

```
oci fs file-system create --availability-domain <availability_domain_name>
 --compartment-id <compartment_id> --display-name <fs_clone_display_name> --
source-snapshot-id <fs_snapshot_id> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateFileSystem operation to create a file system clone of a file system snapshot.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a File System Clone

On Compute Cloud@Customer, cloned file systems are managed; you delete a clone the same way you delete a file system. However, dependencies exist between file systems, snapshots, and clones. You can't delete any resources that have a dependency.

See Deleting a File System on page 468.

For details about dependencies, see Deleting File System Resources Overview on page 437.

Replicating File Systems

On Compute Cloud@Customer, you can asynchronously replicate a file system and its snapshots to a different infrastructure at regular intervals.

> **Note:**
>
> **For Compute Cloud@Customer only.** Before you can configure storage replication, you must submit a request for Oracle Support to configure infrastructure peering. See Creating a Support Request. To access support, sign in to the Oracle Cloud Console as described in Sign In to the OCI Console.

Replication Concepts

Source File System

The file system whose data you want to replicate to another infrastructure at regular intervals. Folder and file structure, snapshots, metadata, and permissions settings are replicated to the target file system. File system resource-specific data such as file locks, encryption keys, and tags are not replicated.

Target File System

The destination for replicated data from the source file system. Replicated data in the target file system has the same folder and file structure, snapshots, metadata, and permissions settings as the source file system. Only a file system that has never been exported can be used as a target file system.

Replication Resource

The control component of the replication process. It contains all of the necessary configuration information for the replication. The replication resource is attached to the source file system. The replication resource retrieves the delta of data between two snapshots of the source file system and sends that delta to the associated replication target resource. The replication target resource retrieves the delta and applies it to the target file system. Tags applied to a replication resource are copied to the replication target resource.

Replication Target Resource

The replication target resource is attached to the target file system and receives a replication snapshot from the replication resource and applies the data to the target file system. The replication target is automatically created when you create a replication resource, and any tags applied to the replication resource are copied to the target.

Replication Snapshot

The replication snapshot captures incremental changes in the source file system since the last snapshot. The snapshot data is transmitted from the replication resource to the replication target, which applies it to the target file system. The most recent two replication snapshots are retained.

Replication Interval

The frequency with which the replication operation is performed, in minutes.

Delta Cycle / Delta Status

A delta cycle is a complete replication operation.

Delta status values are the following:

- **Idle:** The replication is not capturing or applying data.
- **Transferring:** The replication is both capturing and committing snapshot data.
- **Failed:** The replication operation failed.

Listing Replications

On Compute Cloud@Customer, you can list replications and get details for a specific replication.

A replication is a resource attached to a source file system that controls transferring data to the replication target. Replication information in the details for the source file system include the following:

- **Targetable**. The Targetable parameter indicates whether this file system can be used as a target file system for replication. This value is always `false` for a source file system. The system sets this value to `true` if the file system is unexported, has no user snapshots, and has not been specified as a target file system in any replication resource. After the file system has been specified as a target in a replication, this value is `false` since a target file system can only be used for one replication. If the file system was specified as a target in a replication but never exported, and the replication resource is deleted, this value can revert to `true`.
- **Replication Target**. This is a link to any replication target associated with this file system. This value is `null` for a source file system since a replication target is associated with the target file system.

Information shown in the details for the replication resource includes the source file system, the replication target OCID, the target file system OCID, the delta status, and the replication interval. For more information, see Replicating File Systems on page 472.

Detailed information about a replication resource also includes the following:

- **Replication Progress**. The current progress of the replication while applying data to the target file system, shown as a percentage of complete.
- **Last Snapshot**. A link to the last successfully captured and applied replication snapshot. The link text is the date and time the snapshot was created.
- **Recovery Point Time**. The date and time of the last successful data replication.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **File Storage**, then select **File Systems**.
2. In the list, select the name of the file system for which you want to see replications.

   On the details page, see the Replication information on the File System Details tab.
3. Scroll to the Resources section and select Replications.

   The list of replication resources is displayed. If no replications are listed, then this file system is not a replication source.
4. In the Replications list, select the name of a replication to view more details about that replication.

Using the CLI

Use the oci fs replication list command to list replication resources in the specified compartment.

```
oci fs replication list --availability-domain "AD-1" --compartment-
id compartment_OCID
```

Use the oci fs replication get command to get more information about the specified replication resource.

```
oci fs replication get --replication-id replication_OCID
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListReplications operation to list replication resources.

Use the GetReplication operation to get more information about the specified replication resource.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Listing Replication Targets

On Compute Cloud@Customer, you can list replication targets and get details for a specific replication target.

Using the Console

This task is not available in the console.

Using the CLI

Use the oci fs replication-target list command to list replication targets in the specified compartment.

```
oci fs replication-target list --availability-domain "AD-1" --compartment-
id compartment_OCID
```

Use the oci fs replication-target get command to get more information about the specified replication target.

```
oci fs replication-target get --replication-target-
id replication_target_OCID
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListReplicationTargets operation to list replication targets.

Use the GetReplicationTarget operation to get more information about the specified replication target.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Creating a Replication

On Compute Cloud@Customer, creating a File Storage replication resource also creates an associated replication target resource. Data is replicated from the source file system to the target file system at the interval you specify.

A source file system can have at most three replication resources, including any replication resources for this source file system that are in state Failed. The replication resources can have different replication intervals. Therefore, the content of the target file systems could be different at a specific point in time.

The target file system is an existing file system that you specify. The target file system has the following restrictions:

- Cannot be a file system that has ever been mounted for export.
- Must be on a different infrastructure than the infrastructure that the source file system is on.
- Must be in the same pool (capacity or performance vpusPerGB) on the target infrastructure that the source file system is in on the source infrastructure. Attempting to apply the replication snapshot to a file system in a different pool will fail.

If the source file system was created from a snapshot, then the data being replicated to the target file system includes the data from the snapshot as a whole. The data from the snapshot is not replicated separately.

Replicated data in the target file system has the same file and folder structure, snapshots, metadata, and permissions settings as the source file system. File system resource-specific data such as file locks, encryption keys, and tags are not replicated.

The replication resource is created immediately. The first replication operation starts as soon as the replication resource is active. The initial replication operation can take hours depending on the amount of data in the source file system.

After the initial replication operation completes, replication operations are performed according to the specified replication interval. The replication interval is the number of minutes between replication snapshots. The default value is 60 minutes. The value of the replication interval must be one of the following: 5, 10, 15, 20, 30, 60, 120, 240, 480, 720, 1440 (one day), 10080 (one week), or 43200 (one month).

Using the Console

Get the OCID for the file system that you plan to use for the target file system. Only a file system that has never been exported can be used as a target file system.

1. In the Compute Cloud@Customer Console navigation menu, select **File Storage**, then select **File Systems**.
2. On the list, select the name of the file system that you want to use as the source file system to replicate.
3. On the details page for the selected source file system, the Targetable parameter value must be `true`.
4. Scroll to the Resources section, select Replications, and select the Create Replication button.
5. On the Create Replication dialog, enter the following information:
   - **Name**: Enter a meaningful display name for the replication resource. Avoid entering confidential information.
   - **Target File System**: Enter the OCID of the file system that you want to use for the target file system.
   - **Replication interval**: Enter the replication interval in minutes.
   - **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
6. Select the Create Replication button on the dialog.

   The replication resource is created and the initial replication operation begins.

Using the CLI

Use the oci fs replication create command to create a replication resource. Use the `--source-id` option to specify the file system to use as the source file system and use the `--target-id` option to specify the file system to use as the target file system. The replication target resource is created automatically when you create this replication resource. The `--display-name` and `--replication-interval` are optional.

```
oci fs replication create --compartment-id compartment_OCID --display-
name "Replication1" --source-id source_file_system_OCID --target-
id target_file_system_OCID --replication-interval interval_in_minutes
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateReplication operation to create a replication resource.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Updating a Replication

On Compute Cloud@Customer, you can update the replication display name, replication interval, or tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **File Storage**, then select **File Systems**.
2. On the list, select the name of the source file whose replication resource you want to update.
3. On the details page for the source file system, scroll to the Resources section and select Replications.
4. Select the Actions menu, and select Edit.
5. In the Edit dialog, make changes and select the Save button.

Using the CLI

Use the oci fs replication update command to update a replication resource.

```
oci fs replication update --replication-id replication_OCID
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateReplication operation to update a replication resource.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Replication

When you delete a replication resource On Compute Cloud@Customer, you can select the delete mode that you want to use.

> **Note:**
>
> Deleting a replication resource also deletes the replication target resource.

The most recent replication snapshot is retained. User-created snapshots are also retained.

You can select one of the following delete modes:

Replicate and delete

CLI and API: `ONE_MORE_CYCLE`

1. If the replication status is `transferring`, finish the current cycle, start and finish one more cycle, and then delete the replication resource.
2. If the replication status is `idle`, start and finish one more cycle, and then delete the replication resource.

This mode is slower, depending on how much data is being replicated and whether a delta cycle is already in progress when you delete the replication resource.

Apply and delete

CLI and API: `FINISH_CYCLE_IF_APPLYING`

This mode is the default.

1. If the replication status is `transferring`, finish and then delete the replication resource.
2. If the replication status is `idle`, delete the replication resource immediately.

This mode is moderately faster depending on how much data is being replicated.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **File Storage**, then select **File Systems**.
2. On the list, select the name of the source file whose replication resource you want to delete.
3. On the details page for the source file system, scroll to the Resources section and select Replications.
4. Select the Actions menu, and select Delete.
5. On the Delete Replication dialog, select the delete mode that you want to use, and select the Delete button.
6. Read the delete notification and select the Confirm button.

Using the CLI

Use the oci fs replication delete command to delete the specified replication resource and the associated replication target.

```
oci fs replication delete --replication-id replication_OCID --delete-
mode mode
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteReplication operation to delete the specified replication resource and the associated replication target.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Replication Target

When you delete a replication target resource On Compute Cloud@Customer, you delete the specified replication target resource and immediately release the target file system if no replication operation is in progress. Deleting a replication target resource makes the associated target file system exportable.

> **Note:**
>
> You should only delete the replication target if the source file system is no longer available (such as during an outage). If the source file system is still available, delete the replication resource instead.

If any replication operation is in progress, the delete operation is blocked until the replication operation is completed.

Deleting the replication target resource places the replication resource in the Failed state.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **File Storage**, then select **File Systems**.
2. On the list, select the name of the source file whose replication target you want to delete.
3. On the details page for the source file system, find the Replication section on the File System Information tab.
4. Select the Replication Target link.
5. On the details page for the replication target, select the Delete button.
6. On the delete confirmation dialog, select the Confirm button.

Using the CLI

Use the oci fs replication-target delete command to delete the specified replication target.

```
oci fs replication-target delete --replication-target-id replication_OCID
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteReplicationTarget operation to delete the specified replication target.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Recovering a Source File System

This topic describes how to recover a source file system that has become inaccessible or corrupted On Compute Cloud@Customer.

1. Delete the associated replication target. See Deleting a Replication Target on page 477.

   When you delete the replication target, the target file system is no longer a target file system for the damaged or inaccessible source file system. The target file system can now be used as a source file system.
2. On the file system that was the target file system, create a replication resource. See Creating a Replication on page 475. The file system that was the target file system is now the source file system.

You now have a new source file system and a new target file system that is up-to-date with the source file system before the source file system became inaccessible or corrupted, except for any changes that might have occurred since the most recent replication operation.

Object Storage

On Compute Cloud@Customer, the Object Storage service provides reliable and cost-efficient data durability.

The Object Storage service stores unstructured data of any content type, including analytic data and rich content, such as 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 and retrieve data directly from the internet or from within Compute Cloud@Customer.

Object Storage Resources

The Object Storage service uses these components to organize the data stored in object storage:

Objects

Any type of data, regardless of content type, is stored as an object.

An object is composed of the object itself and metadata about the object. Each object is stored in a bucket. The object is processed as a single entity.

Buckets

Buckets are logical containers for storing objects.

Users or systems create buckets as needed. A bucket is associated with a single compartment that has policies that determine what actions a user can perform on a bucket and on all the objects in the bucket. Buckets can't be nested.

You determine the bucket names, but each bucket name must be unique within a namespace.

Namespace

An Object Storage namespace serves as the top-level container for all buckets and objects.

When your tenancy is provisioned, the tenancy is assigned one unique system-generated and immutable Object Storage namespace name. The namespace spans all compartments within the tenant.

Within a namespace, buckets and objects exist in flat hierarchy, but you can simulate a directory structure to help navigate a large set of objects.

Compartment

A compartment is the primary building block used to organize your cloud resources.

When your tenancy is provisioned, a root compartment is created for you. You can then create compartments under your root compartment to organize your resources. You control access by creating policies that specify what actions groups of users can take on the resources in those compartments. An Object Storage bucket can only exist in one compartment.

Obtaining the Object Storage Namespace

On Compute Cloud@Customer, 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 Console

1. In the Compute Cloud@Customer Console, select your user name (upper right corner), then select **Tenancy**.

   The namespace string is listed under Object Storage Settings.

Using the CLI

Run the following command and add the `iaas` endpoint to the command. For example:

```
oci os ns get --endpoint https://iaas.<mysystem>.example.com
{
  "data": "<myobjstor_namespace_name>"
}
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Managing Object Storage Buckets

On Compute Cloud@Customer, 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 can't contain other buckets.

Creating a Bucket

On Compute Cloud@Customer, you can create Object Storage buckets.

When you create a bucket, the bucket doesn't provide public access. To make the bucket publicly available, see Using Preauthenticated Requests on page 495.

Avoid entering confidential information in names and tags.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Object Storage**, then select **Object Storage**.
2. Select **Create Bucket**.
3. Enter the following details:
   - **Name:** Enter a name for the bucket.

     Specify a name that's 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, see Managing Object Versioning on page 492.
   - **Tagging:** (Optional) Add one or more tags to this resource. Tags can also be applied later. For more information about tagging resources, see Adding Tags at Resource Creation (IAM in OCI) on page 118.
4. Select **Create Bucket**.

   The bucket is created immediately and you can start uploading objects. See Uploading an Object on page 485.

Using the CLI

Use the oci os bucket create command and required parameters to create a bucket in the particular namespace with a bucket name and optional user-defined metadata.

```
oci os bucket create --namespace-name <object_storage_namespace> --
compartment-id <compartment_OCID> --name <bucket_name> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateBucket operation to create a bucket in the given namespace with a bucket name and optional user-defined metadata.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Listing Buckets

On Compute Cloud@Customer, you can list buckets.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Object Storage**, then select **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 the compartment at the top of the page).

   The page shows only the resources in that compartment.

Using the CLI

Use the oci os bucket list command and required parameters to get a list of all bucket summary items in a compartment. A bucket summary contains only summary fields for the bucket and does not contain fields like the user-defined metadata.

```
oci os bucket list --namespace-name <object_storage_namespace> --
compartment-id <compartment_OCID> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListBuckets operation to get a list of all bucket summary items in a compartment. A bucket summary contains only summary fields for the bucket and doesn't contain fields such as the user-defined metadata.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Getting Bucket Details

On Compute Cloud@Customer, you can view bucket details.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Object Storage**, then select **Object Storage**.
2. At the top of the page, select the compartment that contains the bucket.
3. Select the bucket name to display the details.
4. Select **View** or **Copy**.

Using the CLI

Use the oci os bucket get command and required parameters to get the current representation of the given bucket in the given Object Storage namespace.

```
oci os bucket get --namespace-name <object_storage_namespace> --bucket-
name <bucket_name> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the GetBucket operation to get the current representation of the given bucket in the given Object Storage namespace.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Moving a Bucket to a Different Compartment

On Compute Cloud@Customer, 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 lower level within the source compartment.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci os bucket update command and required parameters to move a bucket from one compartment to another within the same tenancy. Supply the compartment Id of the compartment that you want to move the bucket to.

```
oci os bucket update --namespace-name <object_storage_namespace> --
compartment-id <target_compartment_id> --bucket-name <bucket_name> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateBucket operation to move a bucket from one compartment to another within the same tenancy. Supply the compartment Id of the compartment that you want to move the bucket to.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Bucket

On Compute Cloud@Customer, you can permanently delete an empty bucket.

> **Caution:**
>
> You can't recover a deleted bucket.

You can't delete a bucket that contains any of the following:

- Any objects
- Previous versions of an object
- A multipart upload in progress
- A preauthenticated 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 on page 492.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci os bucket delete command and required parameters to delete a bucket if the bucket is empty.

```
oci os bucket delete --namespace-name <object_storage_namespace> --bucket-
name <bucket_name>[OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteBucket operation to delete a bucket if the bucket is empty.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Managing Storage Objects

On Compute Cloud@Customer, you can manage objects in a bucket using the Console, CLI, and API.

Object Naming Prefixes and Hierarchies

Within an Object Storage namespace, buckets and objects exist in a flat structure. However, you can simulate a directory structure by adding a prefix string that includes one or more forward slashes (/) to an object name. Doing so lets you list one directory at a time, which is helpful when navigating a large set of objects.

For example:

```
marathon/finish_line.jpg
marathon/participants/p_21.jpg
```

If you add prefixes to object names, you can:

- Use the CLI or API to perform bulk downloads and bulk deletes of all objects at a specified level of the hierarchy.
- Use the Console to display a hierarchical view of your objects in virtual folders. In the previous example, `marathon` would be displayed as a folder containing an object named `finish_line.jpg` and `participants` would be a subfolder of `marathon`, containing an object named `p_21.jpg`. You can bulk upload objects to any level of the hierarchy and perform bulk deletes of all the objects in a bucket or folder.

Bulk operations at a specified level of the hierarchy do not affect objects in any level above.

When naming objects, you can also use prefix strings without a delimiter. No delimiters allow search operations and certain bulk operations in to match on the prefix portion of the object name. For example, in the object names below, the string `gloves_27_` can serve as a prefix for matching purposes when performing bulk operations:

```
gloves_27_dark_green.jpg
gloves_27_light_blue.jpg
```

When you perform bulk uploads, you can add a prefix string to the names of the files you are uploading.

Object Names

Unlike other resources, objects don't have Cloud Identifiers (OCIDs). Instead, you define an object name when you upload an object.

Use the following guidelines when naming an object:

- The maximum length for an object and bucket name is 255 characters.
- Valid characters are letters (upper or lowercase), numbers, and characters other than line feed, carriage return, and NULL.
- Bucket names and object names are case-sensitive.
- Use only Unicode characters for which the UTF-8 encoding doesn't exceed 1024 bytes. Clients are responsible for URL-encoding characters.
- Make the name unique within the bucket.
- Object names can include one or more slash (/) characters in the name. See Object Naming Prefixes and Hierarchies on page 482.
- At the top of the page, select the compartment in names.

Optional Response Headers and Metadata

When you upload objects, you can provide optional response headers and user-defined metadata. Response headers are HTTP headers sent from Object Storage to Object Storage clients when objects are downloaded.

User-defined metadata are name-value pairs stored with an object.

> **Important:**
>
> No validation is performed on the response headers or metadata you provide.

You can specify values for the following response headers:

- **Content-Disposition**

  Defines presentation only information for the object. Specifying values for this header has no effect on Object Storage behavior. Programs that read the object determine what to do based on the value provided. For example, you could use this header to let users download objects with custom file names in a browser. For Example:

  ```
  attachment; filename="fname.ext"
  ```

- **Cache-Control**

  Defines the caching behavior for the object. Specifying values for this header has no effect on Object Storage behavior. Programs that read the object determine what to do based on the value provided. For example, you could use this header to identify objects that require caching restrictions. For Example:

  ```
  no-cache, no-store
  ```

Metadata

You specify user-defined metadata in the form of name-value pairs. User-defined metadata names are stored and returned to Object Storage clients with the mandatory prefix of `opc-meta-`.

Viewing Objects in a Bucket

On Compute Cloud@Customer, you can view objects in a bucket.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Object Storage**, then select **Object Storage**.
2. At the top of the page, select the compartment that contains the bucket that contains your object.

   A list of buckets is displayed.
3. Select the bucket name that contains your object.
4. Under **Resources**, select **Objects**.

   The objects are displayed.

Using the CLI

Use the oci os object list command and required parameters to list objects in a bucket.

```
oci os object list --namespace-name <object_storage_namespace> --bucket-
name <bucket_name> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Viewing Objects Using the API

Use the ListObjects operation to list objects in a bucket.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Creating a Folder or Subfolder

On Compute Cloud@Customer, you can create a folder in a bucket or you can create a subfolder in an existing folder or subfolder using the CLI and API.

Avoid entering confidential information in names and tags.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci os object put command and required parameters to create a folder in a bucket or a subfolder in an existing folder or subfolder.

Syntax:

```
oci os object put --namespace-name<object_storage_namespace>--bucket-
name<bucket_name>--file<file_location>--name<object_name>[OPTIONS]
```

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==" }
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the PutObject operation to create a folder in a bucket or a subfolder in an existing folder or subfolder.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Uploading an Object

On Compute Cloud@Customer, you can upload an object using the CLI and API.

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 on page 485.

> **Note:**
>
> Objects stored in Compute Cloud@Customer Object Storage are compressed. You might notice that the size of an uploaded object is smaller that the size of the original object before it was uploaded.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci os object put command and required parameters to upload an object to a bucket.

```
oci os object put --namespace-name <object_storage_namespace> --bucket-
name <bucket_name> --file <file_location> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the PutObject operation to upload an object to a bucket.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Performing a Multipart Upload

On Compute Cloud@Customer, you can upload objects in multiple parts.

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.

Object Parts

With multipart upload, you split the object you want to upload into individual parts. Individual parts can be as large as 50 GiB. The maximum size for an uploaded object is 10 TiB.

Decide what part number you want to use for each part. Part numbers can range from 1 to 10,000. You do not need to assign contiguous numbers, but Object Storage constructs the object by ordering part numbers in ascending order.

Multipart Upload API

Before you use the multipart upload API, you are responsible for creating the parts to upload. Object Storage provides API operations for the remaining steps.

A multipart upload performed using the API consists of the following steps:

1. Start an upload.
2. Upload object parts.
3. Commit the upload.

The service also provides API operations for listing in-progress multipart uploads, listing the object parts in an in-progress multipart upload, and aborting in-progress multipart uploads initiated through the API.

Multipart Upload CLI

When you perform a multipart upload using the CLI, you don't need to split the object into parts as you are required to do by the API. Instead, you specify the part size of your choice, and Object Storage splits the object into parts and performs the upload of all parts automatically. You can choose to set the maximum number of parts that can be uploaded in parallel. By default, the CLI limits the number of parts that can be uploaded in parallel to three. When using the CLI, you don't have to perform a commit when the upload is complete.

You can also use the CLI to list in-progress multipart uploads, and to cancel multipart uploads initiated through the API.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci os object put command and required parameters to upload an object in multiple parts.

Syntax:

```
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 [OPTIONS]
```

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"
}
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

A multipart upload performed using the API consists of the following steps:

1. Starting an upload.

2. Uploading object parts.
3. Committing the upload.

Before you use the multipart upload API, you're responsible for creating the parts to upload. Object Storage provides API operations for the remaining steps. The service also provides API operations for listing in-progress multipart uploads, listing the object parts in an in-progress multipart upload, and aborting in-progress multipart uploads initiated through the API.

> **Note:**
>
> These multipart upload API requirements do not apply to the CLI.

For more information about using the multipart Upload API, see Using the Multipart Upload API.

Use the PutObject operation to upload an object in multiple parts.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Listing Multipart Uploads in Object Storage

On Compute Cloud@Customer, you can view a list of the in-progress objects in multipart uploads for an Object Storage bucket using the CLI and API.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci os multipart list command and required parameters to list all the in-progress objects in a multipart upload.

Syntax:

```
oci os multipart list --namespace-name <object_storage_namespace> --bucket-
name <bucket_name> [OPTIONS]
```

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"
    }
```

```
...
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListMultipartUploadParts operation to list all the in-progress objects in a multipart upload.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Performing a Bulk Object Upload

On Compute Cloud@Customer, you can perform bulk object uploads to a bucket using the CLI.

Bulk operations at a specific level of the hierarchy do not affect objects in any level above.

Using the CLI

Use the oci os object bulk-upload command and required parameters to bulk upload objects.

Syntax:

```
oci os object bulk-upload --namespace-name <object_storage_namespace> --
bucket-name <bucket_name> --src-dir <source_directory_location> [OPTIONS]
```

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=="
    }
  }
}
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Copying an Object to a Different Bucket

On Compute Cloud@Customer, you can copy an object to a different bucket as long as the target bucket is located in the same Compute Cloud@Customer infrastructure.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci os object copy command and required parameters to copy an object to a different bucket.

```
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> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CopyObject operation to copy an object to a different bucket.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Downloading an Object

On Compute Cloud@Customer, you can download an object using the CLI and API.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci os object get command and required parameters to download an object.

Syntax:

```
oci os object get --namespace-name <object_storage_namespace> --bucket-
name <bucket_name> --name<object_name> --file<file_location> [OPTIONS]
```

<*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
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the GetObject operation to download an object.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Performing a Multipart Download

On Compute Cloud@Customer, you can download an object in multiple parts using CLI and API operations.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci os object get command and required parameters to operation to download an object in multiple parts.

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> [OPTIONS]
```

The byte-range for the download. Multipart object downloading is available using the byte-range request standard defined in RFC 7233, section 2.1

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
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use theGetObject operation to download an object in multiple parts.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Performing a Bulk Download

On Compute Cloud@Customer, you can download objects in bulk.

Using the CLI

Use the oci os object copy command and required parameters to perform a bulk download.

```
oci os object bulk-download
--namespace-name <object_storage_namespace>
--bucket-name <bucket_name>
--download-dir <download_directory_location> [OPTIONS]
```

The 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.

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": []
}
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Deleting an Object

On Compute Cloud@Customer, you can permanently delete an object from a bucket or folder.

> **Caution:**
>
> You can't recover a deleted object unless you have object versioning enabled. See Managing Object Versioning on page 492 for details.

You can't delete an object that has an active retention rule. See Defining Retention Rules on page 501 for details.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci os object delete command and required parameters to delete an object or folder.

```
oci os object delete --namespace-name <object_storage_namespace> --bucket-
name <bucket_name> --object-name <object_name> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteObject operation to delete an object or folder.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Bulk Deleting All Objects in a Bucket

On Compute Cloud@Customer, you can delete all objects in a bucket using the CLI.

> **Caution:**
>
> You can't recover a deleted object unless you have object versioning enabled. See Managing Object Versioning on page 492 for details.

You can't delete an object that has an active retention rule. See Defining Retention Rules on page 501 for details.

Using the CLI

Use the oci os object bulk-delete command and required parameters to delete an object or folder.

To see a list of the files impacted by a bulk delete command without actually deleting the files, use the `--dry-run` option. Perform the command without the `--dry-run` option to bulk delete all objects in the bucket.

```
oci os object bulk-delete --namespace-name <object_storage_namespace> --
bucket-name <bucket_name> --dry-run
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Managing Object Versioning

On Compute Cloud@Customer, object versioning provides data protection against accidental or malicious object update, overwrite, or deletion.

Object versioning is enabled at the bucket level. Versioning directs Object Storage to automatically create an object version each time one of these actions takes place:

- A new object is uploaded.
- An existing object is overwritten.
- 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.

Scope and Constraints

- Versioning can be enabled on a bucket in Object Storage.
- You can rename the latest version of an object, but you can't rename a previous object version. Renaming an object creates a new object.

Interaction Between Versioning and Other Object Storage Features

This section describes some key things you need to know about the interaction between object versioning and other Object Storage features.

Copying Objects

If you copy the latest version of an object to a different bucket, only the object is copied. None of the object's previous versions are copied. You can copy a previous version of an object to another bucket, but that action creates either the latest version of a new object or a new object version in the destination bucket.

Retention Rules

- You can't add retention rules to a bucket that has versioning enabled.
- You can't enable versioning on a bucket with active retention rules.
- You can add retention rules to bucket that has versioning suspended. However, you can't resume versioning with active retention rules.

Enabling Versioning During Bucket Creation

On Compute Cloud@Customer, you can enable object versioning during bucket creation using the CLI and API.

Object versioning provides data protection against accidental or malicious object updates and deletions.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci os bucket create command and required parameters to enable object versioning during bucket creation.

```
oci os bucket create --namespace-name <object_storage_namespace> --
compartment-id <target_compartment_id> --name <bucket_name> --versioning
 enabled [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateBucket operation to enable object versioning during bucket creation.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Enabling or Suspending Versioning

On Compute Cloud@Customer, object versioning provides data protection against accidental or malicious object updates and deletions.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci os bucket update command and required parameters to enable or suspend versioning.

For the `--versioning` option, choose one of these values: `Enabled` or `Suspended`.

```
oci os bucket update --namespace-name <object_storage_namespace> --
compartment-id <target_compartment_id> --bucket-name <bucket_name> --
versioning <enabled | suspended> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateBucket operation to enable or suspend versioning.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing Object Versions and Details

On Compute Cloud@Customer, you can list object versions and details using the CLI and API.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci os object list-object-versions command and required parameters to list the object versions in a bucket.

```
oci os object list-object-versions --namespace-
name <object_storage_namespace> --bucket-name <bucket_name> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListObjectVersion operation to list the object versions in a bucket.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting the Previous Version of an Object

On Compute Cloud@Customer, when versioning is enabled, deleting an object without targeting a specific version creates a delete marker and a previous version of the object that object can be recovered. However, deleting a previous version of an object is a permanent deletion.

No object is physically deleted from a bucket that has versioning enabled until you take explicit action to do so.

When you delete an object without targeting a specific version, the latest object version becomes a previous object version and a special delete marker is created that marks the deletion point. A delete marker contains only minimal metadata. If you delete a folder, a delete marker is created for each object in the folder. You can simply delete the delete marker to make that deleted version become the latest object version.

When you upload an object with the same name as the delete marker, the uploaded object becomes the latest version of the object. The delete marker remains. There can be multiple delete markers for an object and you can recover any of the previous object versions.

Object version deletion is different. When you delete an object version, the version is permanently deleted. Permanent deletion also happens if you explicitly delete the latest version by version ID. All delete operations that target a specific object version ID permanently deletes the data.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci os object delete command and required parameters to delete the previous version of an object.

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> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteObject operation to delete the previous version of an object.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Recovering a Deleted Object Version

On Compute Cloud@Customer, 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.

To perform this task, you need to know which object is marked for deletion. To obtain that information, list the objects in the bucket. See Viewing Object Versions and Details on page 493. 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.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci os object delete command and required parameters to remove the delete marker from an object.

Syntax:

```
oci os object delete --namespace-name <object_storage_namespace>
 --bucket-name <bucket_name> --object-name <object_name> --version-
id <bucket_version_id> [OPTIONS]
```

Example:

```
oci os object delete
--namespace-name examplenamespace  \
--bucket-name  MyBucket
--object-name application.log
--version-id  6ce3eb93-8850-4732-8949-uniqueID
Are you sure you want to delete this resource? [y/N]: y
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteObject operation to remove the delete marker from an object.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Using Preauthenticated Requests

On Compute Cloud@Customer, preauthenticated 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 preauthenticated request, a unique URL is generated. Anyone you provide this URL to can access the Object Storage resources identified in the preauthenticated request, using standard HTTP tools like curl and wget.

> **Important:**
>
> Assess the business requirement for and the security ramifications of preauthenticated access to a bucket or objects.

> A preauthenticated request URL gives anyone who has the URL access to the targets identified in the request. Carefully manage the distribution of the URL.

Required Permissions

To Create a Preauthenticated Request

You need the `PAR_MANAGE` permission to the target bucket or object.

You must also have the appropriate permissions for the access type that you are granting. For example:

- If you are creating a preauthenticated request for uploading objects to a bucket, you need the `OBJECT_CREATE` and `OBJECT_OVERWRITE` permissions.
- If you are creating a preauthenticated request for read/write access to objects in a bucket, you need the `OBJECT_READ`, `OBJECT_CREATE`, and `OBJECT_OVERWRITE` permissions.

> **Important:**
>
> If the creator of a preauthenticated request is deleted or loses the required permissions after they created the request, the request will no longer work.

To Use a Preauthenticated Request

Permissions of the preauthenticated request creator are checked each time you use a preauthenticated request.

The preauthenticated request no longer works when any of the following occurs:

- Permissions of the preauthenticated request creator have changed.
- The user who created the preauthenticated request is deleted.
- A Federated user who created the preauthenticated request has lost the user capabilities that they had when they created the request.
- Preauthenticated request has expired.

Types of Preauthenticated Requests

When creating a preauthenticated request, you have the following options:

- You can specify the name of a bucket that a preauthenticated request user has write access to and can upload one or more objects to.
- You can specify the name of an object that a preauthenticated request user can read from, write to, or read from and write to.

Scope and Constraints

Understand the following scope and constraints regarding preauthenticated requests:

- Users can't list bucket contents.
- You can create an unlimited number of preauthenticated requests.
- There is no time limit to the expiration date that you can set.
- You can't edit a preauthenticated request. If you want to change user access options in response to changing requirements, you must create a new preauthenticated request.
- The target and actions for a preauthenticated request are based on the creator's permissions. The request isn't, however, bound to the creator's account login credentials. If the creator's login credentials change, a preauthenticated request isn't affected.
- You can't delete a bucket that has a preauthenticated request associated with that bucket or with an object in that bucket.

> **Important:**
>
> The unique URL provided by the system when you create a preauthenticated request is the only way a user can access the bucket or object specified as the request target. Copy the URL to durable storage. The URL is displayed only at the time of creation and can't be retrieved later.

Creating a Preauthenticated Request for All Objects in a Bucket

On Compute Cloud@Customer, you can create preauthenticated requests for all objects in a bucket using the CLI and API.

> **Important:**
>
> Immediately after creating the request, 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 can't be retrieved later.**

To construct a URL from the unique `access-uri`, see Constructing the Preauthenticated Request URL on page 499.

> **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 preauthenticated request that lets users list the objects in the bucket.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci os preauth-request create command and required parameters to create a preauthenticated request for all objects in a bucket.

```
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> [OPTIONS]
```

For access type, use one of these 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.

`time-expires` is a required argument and must be an RFC 3339 timestamp. For example: `2017-09-01T00:09:51.000+02:00`.

This example creates a preauthenticated 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"
  }
}
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreatePreauthenticatedRequest operation to create a preauthenticated request for all objects in a bucket.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Creating a Preauthenticated Request for a Specific Object

On Compute Cloud@Customer, you can create a preauthenticated request for a specific object in a bucket using the CLI and API.

> **Important:**
>
> Immediately after creating the request, 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 can't be retrieved later.**

To construct a URL from the unique `access-uri`, see Constructing the Preauthenticated Request URL on page 499.

> **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 preauthenticated request that lets users list the objects in the bucket.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci os preauth-request create command and required parameters to create a preauthenticated request for a specific object in a bucket.

```
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>
 [OPTIONS]
```

For access type, use one of these 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.

`--time-expires` is a required argument and must be an RFC 3339 timestamp. For example: `2017-09-01T00:09:51.000+02:00`.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreatePreauthenticatedRequest operation to create a preauthenticated request for a specific object in a bucket.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Constructing the Preauthenticated Request URL

On Compute Cloud@Customer, after you have a unique `access-uri`, you can construct the access URL that enables users to access preauthenticated objects. The `access-uri` is generated when you create a preauthenticated request.

Construct the URL using this syntax.

```
https://objectstorage.<ccc_fqdn><access-uri>
```

where:

- *<ccc_fqdn>* is the fully qualified domain name of your Compute Cloud@Customer infrastructure.
- *<access-uri>* is the access URI that was obtained from one of these procedures:
  - Creating a Preauthenticated Request for All Objects in a Bucket on page 497
- Creating a Preauthenticated Request for a Specific Object on page 498

Example:

```
https://objectstorage.myccc01.us.example.com/p/
MrxLFkKlFkIlNDhvhcZnrjbUAlsoeah/n/mynamespace/b/my-bucket/o/my-object
```

Listing Preauthenticated Requests

On Compute Cloud@Customer, you can list preauthenticated requests using the unique request URL.

For information about the unique request URL, see Constructing the Preauthenticated Request URL on page 499.

Using curl

- Syntax:

  ```
  $ 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"}]}
```

Listing Objects for Preauthenticated Requests

On Compute Cloud@Customer, using the unique request URL, you can use a tool like curl to list, read, and write data using the preauthenticated 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":"developer-reference.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 Preauthenticated Request

On Compute Cloud@Customer, using the unique request URL, you can use a tool like curl to read and write data using the preauthenticated 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 Preauthenticated Request

On Compute Cloud@Customer, using the unique request URL, you can use a tool like curl to read and write data using the preauthenticated request.

Using curl

- Syntax (entered on a single line):

```
$ curl -X GET <unique-PAR-URL>
```

Example:

```
$ curl -X GET https://objectstorage.us.example.com/p/
tnjDhazP9o6s2KzLyFUxILQzSamEp/n/examplenamespace/b/MyParBucket/o/
developer_reference.pdf
'@data.1''@data.2''@data.3'
```

Deleting a Preauthenticated Request

On Compute Cloud@Customer, you can delete preauthenticated requests.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci os preauth-request delete command and required parameters to delete a preauthenticated request.

```
oci os preauth-request delete --namespace-name <object_storage_namespace> --
bucket-name <bucket_name> --par-id <preauthenticated_request_id> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeletePreauthenticatedRequest operation to delete a preauthenticated request.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Defining Retention Rules

On Compute Cloud@Customer, 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.

Object Storage provides a flexible approach to data retention that supports the following use cases.

- **Regulatory Compliance**

  Your industry might require you to retain a certain class of data for a defined length of time. Your data retention regulations might also require that you lock the retention settings. If you lock the settings, the only change you can make is to increase the retention duration.

  For Object Storage regulatory compliance, you create a time-bound retention rule and specify a duration. Object modification and deletion are prevented for the duration specified. Duration is applied to each object individually, and is based on the object's Last Modified timestamp. Lock the rule as required.
- **Data Governance**

  You might need to protect certain datasets as a part of internal business process requirements. While retaining the data for a defined length of time is necessary, that time period could change.

  Create a time-bound retention rule and specify a duration. Object modification and deletion are prevented for the duration specified. Duration is applied to each object individually, and is based on the object's Last Modified timestamp. To be able to delete the rule and allow changes to the duration as required, do not lock the rule.
- **Legal Hold**

  You might need to preserve certain business data in response to potential or on-going lawsuits. A legal hold does not have a defined retention period and remains in effect until removed.

For Object Storage legal holds, you create an indefinite retention rule. Object modification and deletion are prevented until you delete the rule. You can't lock an indefinite retention rule because the rule has no duration.

It's important to understand retention duration for time-bound rules. Even though you are creating retention rules for a bucket, the duration of a rule is applied to each object in the bucket individually, and is based on the object's Last Modified timestamp. Let's say you have two objects in the bucket, ObjectX and ObjectY. ObjectX was last modified 14 months ago and ObjectY was last modified 3 months ago. You create a retention rule with a duration of 1 year. This rule prevents the modification or deletion of ObjectY for the next 9 months. The rule allows the modification or deletion of ObjectX because the retention rule duration (1 year) is less that the object's Last Modified timestamp (14 months). If ObjectX is overwritten some time in the coming year, modification and deletion would be prevented for the rule duration time remaining.

Locking a retention rule is an irreversible operation. Not even a tenancy administrator can delete a locked rule. There is a mandatory 14-day delay before a rule is locked. This delay lets you thoroughly test, modify, or delete the rule or the rule lock before the rule is permanently locked. A rule is active at the time of creation. The lock only controls whether the rule itself can be modified. After a rule is locked, only increases in the duration are allowed. Object modification is prevented and the rule can only be deleted by deleting the bucket. A bucket must be empty before it can be deleted.

Scope and Constraints

- Retention rules can be applied to a bucket in the Object Storage.
- The actions that you can perform on a bucket with active retention rules are limited. You can't update, overwrite, or delete objects or object metadata until the retention rule is deleted (indefinite rule) or for the duration specified (time-bound rules). The duration for time-bound rules is applied to each object individually, and is based on the object's Last Modified timestamp.
- You can create multiple retention rules for a bucket. Indefinite retention rule is applied before any time-bound rule is considered.
- When a retention rule is locked, the rule can only be deleted by deleting the bucket. A bucket must be empty before it can be deleted.

Interaction Between Retention and Other Object Storage Features

Carefully review the policies and rules that you have in place for the other Object Storage features that you are using. Some of these policies and rules might not make sense with retention rules. This section describes some key things you need to know about the interaction between retention rules and other Object Storage features.

Multipart Uploads

Uncommitted (unfinished or failed) multipart uploads are not protected by retention rules and can be deleted at any time.

Versioning

- You can't add retention rules to a bucket that has versioning enabled.
- You can't enable versioning on a bucket with active retention rules.
- You can add retention rules to bucket that has versioning suspended. However, you can't resume versioning with active retention rules.

Creating a Retention Rule

On Compute Cloud@Customer, you can create Object Storage retention rules.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci os retention-rule create command and required parameters to create a retention rule.

```
oci os retention-rule create --namespace-name <object_storage_namespace> --
bucket-name <bucket_name> --display-name <name_displayed_for_rule> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the CreateRetentionRule operation to create a retention rule.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Viewing Retention Rules and Details

On Compute Cloud@Customer, you can view Object Storage retention rules and details.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci os retention-rule list command and required parameters to list retention rules.

```
oci os retention-rule list --namespace-name <object_storage_namespace> --
bucket-name <bucket_name> [OPTIONS]
```

Use the oci os retention-rule get command and required parameters to see the details for a retention rule.

```
oci os retention-rule get --namespace-name <object_storage_namespace> --
bucket-name <bucket_name>
--retention-rule-id <retention_rule_identifier> [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the ListRetentionRules operation to list retention rules.

Use the GetRetentionRule operation to get the details for a retention rule.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Modifying a Retention Rule

On Compute Cloud@Customer, you can modify Object Storage retention rules.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci os retention-rule update command and required parameters to update a retention rule.

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
  }
}
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateRetentionRule operation to update a retention rule.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Retention Rule

On Compute Cloud@Customer, you can delete Object Storage retention rules.

Using the Console

This task isn't available in the Console.

Using the CLI

Use the oci os retention-rule delete command and required parameters to delete a retention rule.

```
oci os retention-rule delete --namespace-name <object_storage_namespace> --
bucket-name <bucket_name> --retention-rule-id <retention_rule_identifier>
 [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteRetentionRule operation to delete a retention rule.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Troubleshooting Retention Rules

Identify and address common issues that can occur while working with Object Storage retention rules On Compute Cloud@Customer.

Unable to Create a Retention Rule

If creating a retention rule fails, the most likely cause is missing or incomplete IAM permissions. Rule creation requires:

- User permissions that let you access the bucket and manage the objects in those buckets.
- Minimally, BUCKET_UPDATE and RETENTION_RULE_MANAGE permissions.
- Minimally, BUCKET_UPDATE and RETENTION_RULE_MANAGE permissions.

Unable to Lock a Retention Rule

If locking a retention rule fails, the most likely cause is missing or incomplete IAM permissions. Minimally, BUCKET_UPDATE, RETENTION_RULE_MANAGE, and RETENTION_RULE_LOCK permissions are required to lock retention rules.

Unable to Delete a Retention Rule

You can't delete a time-bound retention rule that's locked. When a retention rule is locked, the rule can only be deleted by deleting the bucket. A bucket must be empty before it can be deleted.

If deleting an indefinite retention rule fails, the most likely cause is missing or incomplete IAM permissions. Rule deletion requires:

- User permissions that let you access the bucket and manage the objects in those buckets.
- Minimally, BUCKET_UPDATE and RETENTION_RULE_MANAGE permissions.

Kubernetes Engine (OKE) on Compute Cloud@Customer

The Oracle Kubernetes Engine (OKE) is a scalable, highly available service that can be used to deploy any containerized application to Compute Cloud@Customer.

The Compute Cloud@Customer OKE documentation doesn't cover OKE extensively. It covers OKE networking requirements and OKE administration that are specific to Compute Cloud@Customer.

For more information about Kubernetes in Oracle, see What is Kubernetes?. For more general information about Kubernetes, see the Kubernetes site.

OKE Overview

The OKE service uses Kubernetes, the open source system for automating deployment, scaling, and management of containerized applications across clusters of hosts. Kubernetes groups the containers that make up an application into logical units called pods for easy management.

The OKE service uses *Cluster API Provider* (CAPI) and *Cluster API Provider for Oracle Cloud Infrastructure* (CAPIOCI) to orchestrate the cluster on the Compute Cloud@Customer.

You can access the OKE service to create OKE clusters by using the Compute Cloud@Customer Console, the CLI, and API.

You can access OKE clusters by using the Kubernetes command line (`kubectl`), the Kubernetes Dashboard, and the Kubernetes API.

On Compute Cloud@Customer, OKE service manages all OKE cluster nodes, which are compute instances. An authorized user can perform tasks such as patch the instance.

For information about OKE service limits, see Limits on Resources Provided by Compute Cloud@Customer on page 34.

Supported Versions of Kubernetes

The OKE service uses versions of Kubernetes that are certified as conformant by the Cloud Native Computing Foundation (CNCF). The OKE service is itself ISO-Compliant (ISO-IEC 27001, 27017, 27018).

Supported versions of Kubernetes are 1.30.3, 1.29.9, and 1.28.8.

Best practice is to keep your clusters upgraded so that they're always running versions of Kubernetes that are currently supported by OKE. Viewing a cluster tells you if a newer Kubernetes version is available for that cluster. See Updating Cluster Add-ons on page 613.

Supported Versions of the OCI Terraform Provider

This guide provides example Terraform scripts to configure the network resources. To use these scripts, you must install both Terraform and the Oracle Cloud Infrastructure (OCI) Terraform provider.

In your `provider` block, specify the version of the OCI Terraform provider to install as at least v4.50.0 but no greater than v6.36.0:

```
provider "oci" {
    version          = ">= 4.50.0, <= 6.36.0"
...
}
```

Public IP Address Requirements

At least three available public IP addresses are required to use OKE on Compute Cloud@Customer for the NAT gateway, the control plane load balancer, and the worker load balancer. For more information, see Creating OKE Network Resources on page 517.

The public IP addresses are configured specifically for your environment by Oracle, when Oracle installs the Compute Cloud@Customer infrastructure in your data center. If you think you might not have three available IP addresses, submit a support request. See Creating a Support Request. To access support, sign in to the Oracle Cloud Console as described in Sign In to the OCI Console.

OKE Workflow

Compute Cloud@Customer has specific requirements and a workflow to enable the OKE service and to configure OKE.

First, a tenancy administrator with appropriate authorizations, enables the OKE service by creating a group, dynamic group, policies, and a defined tag in the OCI tenancy. See Tenancy Administrator Tasks on page 506.

Then OKE cluster administrators perform tasks to configure the OKE service on Compute Cloud@Customer to suit their workload needs. See Cluster Administrator Tasks on page 513.

Tenancy Administrator Tasks

Learn about the set of tasks you perform in the OCI tenancy to enable the OKE service on Compute Cloud@Customer.

> **Note:**
>
> At least three available public IP addresses are required to use OKE on Compute Cloud@Customer. See Public IP Address Requirements on page 506.

Tenancy Administrator Tasks

Perform the following one-time administrative tasks to configure OKE for use on Compute Cloud@Customer.

These administrative tasks involve configuring IAM groups, policies, and a tag namespace with key definitions in the OCI tenancy that's associated with Compute Cloud@Customer. If you don't have tenancy administration privileges, work with your tenancy administrator.

> **Note:**
>
> After you complete these tasks, it can take up to 15 minutes for the IAM changes to synchronize on the Compute Cloud@Customer infrastructure.

No.	Task
1.	Create an OKE User Group and Policies on page 507
2.	Create a Cluster Dynamic Group and Policies on page 508
3.	Understand how Certificate Authority Bundles on page 513 are handled for OKE.
4.	Create the OraclePCA-OKE.cluster_id Tag on page 510
5.	Create OraclePCA Tags For OKE on page 513
6.	Understand how Certificate Authority Bundles on page 513 are handled for OKE.

Create an OKE User Group and Policies

In your OCI tenancy that's associated with Compute Cloud@Customer, create a user group and policies that authorize users to use OKE.

You need at least one OKE user group with OKE policies that authorizethe users to use OKE. You need to create separate OKE user groups to authorize different users to use OKE in different compartments.

This procedure assumes the following conditions:

- Your intended OKE administrators already have OCI user accounts.
- You tenancy is configured with Identity Domains. If you're not sure which type of identity service you're using, see Determining the Tenancy Type. If your tenancy isn't using Identity Domains, create a group and policies as described in Managing Groups (without Identity Domains).

1. Create an OKE user group that includes all users who manage OKE resources.

   For instructions, see Creating a Group.
2. Create the following policies for the group.

   See IAM Policies Overview and Creating a Policy.

   Include the `manage cluster-family` authorization in the user group policy. The following is an example policy for an OKE user group. Depending on your organization, for example if you have a separate team who manage network resources, some of the following "manage" authorizations could be "read" or "use" authorizations, or you might need to add authorizations.

```
allow group group-name to read all-resources in tenancy
allow group group-name to manage cluster-family in
 compartment compartment-name
allow group group-name to manage instance-family in
 compartment compartment-name
allow group group-name to manage network-load-balancers in
 compartment compartment-name
allow group group-name to manage virtual-network-family in
 compartment compartment-name
```

What's Next:

Create a Cluster Dynamic Group and Policies on page 508

Create a Cluster Dynamic Group and Policies

In your OCI tenancy that's associated with Compute Cloud@Customer, create a dynamic group and policies to authorize member instances to manage OKE resources.

You can create a cluster dynamic group and policies Using the Oracle Cloud Console on page 508. Or you can create the group and policies using Terraform scripts.

Using the Oracle Cloud Console

Create a Dynamic Group

1. Create a dynamic group.

   For instructions, see Creating a Dynamic Group.
2. Enter the following matching rule to define the group:

   `tag.OraclePCA-OKE.cluster_id.value`

   All cluster nodes that have this tag are members of the dynamic group.
3. Create policies for the dynamic group.

   See IAM Policies Overview and Creating a Policy.

   The following are example policies for the dynamic group. In this example. `oke_dyn_grp` is the name of the dynamic group and `oke` is the name of the compartment where resources are created. Note that all policy statements are for the same compartment. If clusters in this group require access to resources in other compartments, change the policy accordingly.

   ```
   allow dynamic-group oke_dyn_grp to manage file-family in compartment oke
   allow dynamic-group oke_dyn_grp to manage volume-family in compartment oke
   allow dynamic-group oke_dyn_grp to manage load-balancers in compartment
    oke
   allow dynamic-group oke_dyn_grp to manage instance-family in compartment
    oke
   allow dynamic-group oke_dyn_grp to manage virtual-network-family in
    compartment oke
   allow dynamic-group oke_dyn_grp to use tag-namespaces in compartment oke
   ```

   For information about the purpose of the `use tag-namespaces` policy, see Exposing Containerized Applications on page 633.

What's Next?

Create the OraclePCA-OKE.cluster_id Tag on page 510

Using Terraform to Create a Dynamic Group

The following example shows how to use Terraform to create a dynamic group.

variables.tf

```
variable "oci_config_file_profile" {
  type    = string
  default = "DEFAULT"
}

variable "tenancy_ocid" {
    description = "tenancy OCID"
    type        = string
    nullable    = false
}

variable "compartment_name" {
```

```
    description = "compartment name"
    type        = string
    nullable    = false
}

variable "oke_dyn_grp" {
    description = "Dynamic group that needs to be created for instance
 principal"
    default = "oke-dyn-ip-grp"
}

variable "oke_policy_name" {
    description = "Policy set name for dynamic group"
    default = "oke-instance-principal-policy"
}
```

terraform.tfvars

```
# Name of the profile to use from $HOME/.oci/config
oci_config_file_profile = "DEFAULT"

# Tenancy OCID from the oci_config_file_profile profile.
tenancy_ocid = "ocid1.tenancy.UNIQUE_ID"

# Compartment name
compartment_name = "oke"

# Dynamic Group Name
oke_dyn_grp = "oke-dyn-ip-group"

# OKE Dynamic Group Policy Name
oke_policy_name = "oke-dyn-grp-policy"
```

provider.tf

```
provider "oci" {
  config_file_profile = var.oci_config_file_profile
  tenancy_ocid        = var.tenancy_ocid
}
```

main.tf

```
terraform {
  required_providers {
    oci = {
      source  = "oracle/oci"
      version = ">= 4.50.0, <= 6.36.0"
      # If necessary, you can pin a specific version here
      # version = "6.36.0"
    }
  }
  required_version = ">= 1.1"
}
```

oke-dyn-grp.tf

```
resource "oci_identity_dynamic_group" "oke-dynamic-grp" {
    compartment_id = "${var.tenancy_ocid}"
    description    = "PCA OKE worker dynamic group for instance principal"
    matching_rule  = "tag.${oci_identity_tag_namespace.oracle-pca.name}.
${oci_identity_tag.cluster-id.name}.value"
    name           = "${var.oke_dyn_grp}"
```

```
    depends_on = [oci_identity_tag.cluster-id]
}
```

oke-policy.tf

```
resource "oci_identity_policy" "oke-dyn-grp-policy" {
    compartment_id = "${var.tenancy_ocid}"
    description    = "Dynamic group policies for OKE Resources"
    name           = "${var.oke_policy_name}"
    statements     = [
        "allow dynamic-group ${oci_identity_dynamic_group.oke-dynamic-
grp.name} to manage load-balancers in compartment ${var.compartment_name}",
        "allow dynamic-group ${oci_identity_dynamic_group.oke-dynamic-
grp.name} to manage volume-family in compartment ${var.compartment_name}",
        "allow dynamic-group ${oci_identity_dynamic_group.oke-dynamic-
grp.name} to manage file-family in compartment ${var.compartment_name}",
        "allow dynamic-group ${oci_identity_dynamic_group.oke-dynamic-
grp.name} to manage instance-family in compartment ${var.compartment_name}",
        "allow dynamic-group ${oci_identity_dynamic_group.oke-
dynamic-grp.name} to manage virtual-network-family in compartment
 ${var.compartment_name}",
        "allow dynamic-group ${oci_identity_dynamic_group.oke-dynamic-
grp.name} to use tag-namespaces in compartment ${var.compartment_name}"
    ]

    depends_on = [oci_identity_dynamic_group.oke-dynamic-grp]
}
```

oke-tag-ns.tf

Create the OraclePCA-OKE.cluster_id tag, which is also described in Create the OraclePCA-OKE.cluster_id Tag on page 510.

```
resource "oci_identity_tag" "cluster-id" {
    description      = "Default tag key definition"
    name             = "cluster_id"
    tag_namespace_id = "${oci_identity_tag_namespace.oracle-pca.id}"
    depends_on = [oci_identity_tag_namespace.oracle-pca]
}

resource "oci_identity_tag_namespace" "oracle-pca" {
    compartment_id = "${var.tenancy_ocid}"
    description    = "Default Tag namespace for Oracle PCA OKE"
    name           = "OraclePCA-OKE"
}
```

What's Next?

Create the OraclePCA-OKE.cluster_id Tag on page 510

Create the OraclePCA-OKE.cluster_id Tag

The OraclePCA-OKE.cluster_id defined tag consists of an OraclePCA-OKE tag namespace with a cluster_id tag. This defined tag is required to create or update an OKE cluster or node pool. When you create a node pool, or update the node pool to add nodes, this tag is applied to every node to identify instances that need to be members of the dynamic group.

This procedure describes how to create the OraclePCA-OKE tag namespace and the cluster_id tag key. You must create both the tag namespace and the tag key.

> **Important:**
>
> The tag namespace name must be exactly OraclePCA-OKE, and the tag key name must be exactly cluster_id.

Using the Console

1. If you're not already signed in to your tenancy, use the Oracle Cloud Console to sign in to your tenancy.
2. Open the navigation menu and select **Governance & Administration**. Under **Tenancy Management**, select **Tag Namespaces**.

   Ensure that the compartment that's selected is the compartment where you want to create the tag.
3. If OraclePCA-OKE isn't shown in the Tag Namespaces list, select **Create Tag Namespace**. If the tag already exists, go to Step 5 on page 511.
4. Enter the following information:

   - **Namespace Definition Name**: Enter `OraclePCA-OKE`
   - **Description**: Enter a description for the tag namespace. For example, `Required to create or update an OKE cluster or node pool.` Avoid entering confidential information.
   - Select **Create Tag Namespace**.

     The details page for the OraclePCA-OKE tag namespace is shown.
5. On the details page, select **Create Tag Key Definition**.

   Enter the following information:

   - **Tag Key**: Enter `cluster_id`
   - **Description**: Enter a description for the tag key. For example, `Required to create or update an OKE cluster or node pool.` Avoid entering confidential information.
   - In the **Tag Value Type** section:
     - **Static Value**: Ensure that **Static Value** is selected.
6. Select **Create Tag Key Definition**.

What's Next:

Create OraclePCA Tags For OKE on page 513

Using the CLI

1. Use the oci iam tag-namespace create command and required parameters to create the OraclePCA-OKE tag namespace in your OCI tenancy.

   > **Note:**
   >
   > If the OraclePCA-OKE tag has already been created in a different compartment, then it's available to use in any other compartment, and you will receive an error message from this create tag namespace command.

   Example:

```
$ oci iam tag-namespace create --compartment-id <compartment_OCID> --name
 OraclePCA-OKE --description "Required to create or update an OKE cluster
 or node pool."
{
  "data": {
    "compartment-id": "ocid1.compartment.unique_ID",
    "defined-tags": {
      "Oracle-Tags": {
        "CreatedBy": "okeuser",
        "CreatedOn": "2024-06-06T14:37:29.26Z"
      }
    },
```

```
    "description": "Required to create or update an OKE cluster or node
 pool.",
    "freeform-tags": {},
    "id": "ocid1.tag_namespace.unique_ID",
    "is-retired": false,
    "lifecycle-state": "ACTIVE",
    "locks": null,
    "name": "OraclePCA-OKE",
    "time-created": "2024-06-06T14:37:29.357678+00:00"
  },
  "etag": "a86bcf9b-f9a3-4891-b632-37d490161fe5"
}
```

2. Use the oci iam tag create and required parameters to create the cluster_id tag in the OraclePCA-OKE tag namespace.

   Example:

```
$ oci iam tag create --tag-namespace-id <compartment_OCID> --name
 "cluster_id" --description "Required to create or update an OKE cluster
 or node pool."
--validator '{"validatorType": "DEFAULT"}'
{
  "data": {
    "compartment-id": "unique_ID",
    "defined-tags": {
      "Oracle-Tags": {
        "CreatedBy": "okeuser",
        "CreatedOn": "2024-06-06T21:36:51.38Z"
      }
    },
    "description": "Required to create or update an OKE cluster or node
 pool.",
    "freeform-tags": {},
    "id": "ocid1.tag.unique_ID",
    "is-cost-tracking": false,
    "is-retired": false,
    "lifecycle-state": "ACTIVE",
    "name": "cluster_id",
    "tag-namespace-id": "ocid1.tag_namespace.unique_ID",
    "tag-namespace-name": "OraclePCA-OKE",
    "time-created": "2024-06-06T21:36:51.456538+00:00",
    "validator": {
      "validator-type": "DEFAULT"
    }
  },
  "etag": "5bf59d9a-5998-4857-a590-fca2a3386cc2"
}
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

What's Next:

Create OraclePCA Tags For OKE on page 513

Using the API

Use the CreateTagNamespace operation to create the OraclePCA-OKE tag namespace in your OCI tenancy.

Use the CreateTag operation to create the cluster_id tag in the OraclePCA-OKE tag namespace.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

What's Next:

Create OraclePCA Tags For OKE on page 513

Create OraclePCA Tags For OKE

In your OCI tenancy that's associated with Compute Cloud@Customer, create specific defined tags to enable OKE attributes on Oracle Compute Cloud@Customer.

> **Important:**
>
> Don't delete these tag keys. Don't create this tag namespace and these keys unless you plan to use the Compute Cloud@Customer Console to create clusters. If you create this tag namespace and these keys, create them exactly as shown, don't modify them, and don't delete them.

Compute Cloud@Customer uses the OraclePCA tag namespace to set attributes that aren't available as CLI options or API attributes. For OKE, some cluster attributes must be set using OraclePCA tags.

Other attributes that can only be set by using OraclePCA tags, such as some block volume and file system attributes, are documented in Creating OraclePCA Tags (IAM in OCI) on page 120. You might want to set some of these for nodes in a node pool.

What's Next:

To learn how certificate authority bundles are handled, see Certificate Authority Bundles on page 513.

Certificate Authority Bundles

The Certificate Authority (CA) bundle for Compute Cloud@Customer is downloaded and made available to a cluster when the cluster is created. The CA bundle includes the certificate, private and public keys, and other authorization information.

The CA bundle is automatically updated when regular certificate rotation occurs or when Compute Cloud@Customer is upgraded.

A process runs every hour to check the validity of the CA bundle and updates the CA bundle if needed.

When the CA bundle is updated on the infrastructure, it must be updated on the local system that you use to manage the OKE service. For example, the CA bundle authorizes the use of `cluster-api`. This is similar to replacing the CA bundle in your `~/.oci` configuration so that you can run CLI commands.

To obtain the CA bundle for your local system, contact Oracle for support. See Creating a Support Request. To access support, sign in to the Oracle Cloud Console as described in Sign In to the OCI Console.

Cluster Administrator Tasks

Perform a set of tasks to create OKE clusters on Compute Cloud@Customer.

Prerequisites

Perform the following tasks on your local system that you use to administer OKE:

1. Configure the OCI CLI so you can run CLI commands. See Command Line Interface (CLI).

   If you already have the OCI CLI installed, use `oci -v` to check the version. The minimum required version for using OKE is 3.48.0.

   If you work in more than one tenancy, create a profile for each tenancy as described in Creating a Profile in the Oracle Cloud Infrastructure CLI Configuration File.
2. Install the Kubernetes client command line tool, `kubectl`. See Install kubectl.

   If you already have `kubectl` installed, ensure the version is within one minor version of the Kubernetes version that you're using. See Supported Versions of Kubernetes on page 506.

Cluster Administrator Workflow

Perform the tasks in the following table to configure OKE on Compute Cloud@Customer.

No.	Task	Links
1.	**Create Network resources.** Create a VCN, subnets, internet gateway, NAT gateway, route tables, and security lists.	Creating OKE Network Resources on page 517
2.	**Create an OKE cluster.**	Creating an OKE Cluster on page 589
3.	**Create a Kubernetes configuration file for the cluster.**	Creating a Kubernetes Configuration File on page 596
4.	**Create a Kubernetes Dashboard.** The dashboard helps you manage the cluster and manage and troubleshoot applications running in the cluster.	On the Kubernetes site, see Deploy and Access the Kubernetes Dashboard
5.	**Create a worker node pool.**	Creating an OKE Worker Node Pool on page 619
6.	**Configure any registries or repositories that the worker nodes need.**	
7.	**Create a service to expose containerized applications outside the infrastructure.**	Exposing Containerized Applications on page 633
8.	**Create persistent storage for applications to use.**	Adding Storage for Containerized Applications on page 634

OKE Best Practices

Learn about OKE best practices for Compute Cloud@Customer.

Use the best practices described in this topic to get the most from your OKE clusters.

Cluster Management Best Practices

Upgrade Clusters

Keep your clusters upgraded so that they're always running versions of Kubernetes that are listed as currently supported by OKE. Viewing a cluster tells you if a newer Kubernetes version is available for that cluster. See Supported Versions of Kubernetes on page 506 and Updating an OKE Cluster on page 598.

Use Kubernetes labels.

Use Kubernetes labels to organize the many Kubernetes resources (such as services, pods, containers, and networks) that comprise a cluster.

Kubernetes labels are key-value pairs that help you to maintain these resources and keep track of how they interact with each other in a cluster.

Use resource tagging.

Use resource tagging to organize the many resources (such as worker nodes, VCNs, load balancers, and block volumes) used by the Kubernetes clusters you create with OKE.

When a large number of resources is spread across multiple compartments in a tenancy, it can be challenging to track the resources that are used for specific purposes. It can also be challenging to aggregate the resources, report on them, and take bulk actions on them.

Tagging enables you to define keys and values, and associate those tags with resources. You can then use the tags to organize and list resources based on your business needs.

For more information, see Tagging Resources (IAM in OCI) on page 115.

Set resource requests and limits.

- Set resource requests to specify the minimum amount of resources a container can use.
- Set resource limits to specify the maximum amount of resources a container can use.

Sometimes an application fails to deploy on a Kubernetes cluster due to limited availability of resources on that cluster. The failure of the application to deploy can be avoided by correctly setting resource requests and resource limits.

If you do not set resource requests and limits, pods in a cluster can start utilizing more resources than necessary. If a pod starts consuming more CPU or memory on a node, then the Kubernetes scheduler (`kube-scheduler`) might not be able to place new pods on the node, and the node might even crash.

For more information, see Resource Management for Pods and Containers on the `kubernetes.io` site.

Provide dedicated nodes by using taints and tolerations.

Use Kubernetes taints and tolerations to limit resource-intensive applications to specific worker nodes.

Using taints and tolerations enables you to keep node resources available for workloads that require them, and prevents the scheduling of other workloads on the nodes.

For more information, see Taints and Tolerations on the `kubernetes.io` site.

Control pod scheduling by using node selectors and affinity.

Several different methods are available to constrain a pod to run on particular nodes, or to specify a preference for a pod to run on particular nodes. The recommended approaches all use label selectors to specify the node selection.

Often, the `kube-scheduler` makes a reasonable placement when constraints and preferences are not specified. However, there are some circumstances where you might want to control the node on which a pod runs. In these situations, best practice is to control the scheduling of pods on nodes using Kubernetes node selectors, node affinity, and inter-pod affinity.

Using node selectors, node affinity, and inter-pod affinity enables the `kube-scheduler` to logically isolate workloads, such as according to the node's hardware.

Use third-party tools for backup and disaster recovery.

Use third-party tools such as Velero with OKE for backup and disaster recovery.

The combined backup and disaster recovery capabilities of these tools and OKE can provide a reliable, robust, and scalable Kubernetes platform that is production-ready.

Networking Best Practices

Create separate compartments for each team.

If you expect multiple teams to create clusters, create a separate compartment for each team.

Size your VCN appropriately.

Allow for possible future cluster and node pool scaling requirements when sizing the VCN in which you want to create and deploy Kubernetes clusters.

Ensure that the VCN has a CIDR block that is large enough to allocate network addresses to all the resources that a cluster requires: subnets, Kubernetes API endpoint, worker nodes, pods, load balancers.

Select the pod networking CNI plugin that best suits your needs.

Consider pod networking requirements carefully, and then select the pod networking CNI plugin that best suits your needs.

- If applications require the use of base networking requirements (and not the use of IP addresses from the VCN), or require a high density of pods per worker node, best practice is to use the Flannel Overlay CNI plugin. See Creating Flannel Overlay Network Resources on page 520.

- If applications require pods to have an IP address from the VCN CIDR, or require the consistent network performance offered by virtual machines (regardless of the nodes on which the pods are running) with no additional overlay, best practice is to use the OCI VCN-Native Pod Networking CNI plugin. See Creating VCN-Native Pod Network Resources on page 548.

Configure externalTrafficPolicy appropriately when exposing applications.

Carefully consider the most appropriate value for the `externalTrafficPolicy` setting when provisioning a network load balancer for a Kubernetes service of type LoadBalancer.

Avoid overlapping pod and service CIDR blocks with an on-premise CIDR block and when using the Flannel Overlay CNI plugin.

Avoid the situation where the CIDR block used by the Flannel Overlay network to provision pods and services with IP addresses overlaps with a CIDR block used to provision external compute instances with IP addresses.

Kubernetes clusters require a unique IP address for every pod. Therefore, IP address planning is necessary because addresses cannot overlap with the private IP address space used on-premises or in other connected VCNs.

Plan the number of nodes you will need.

Create a plan for the number of nodes in a cluster that takes into account node size, the application profile of pods, and the selected pod networking CNI plugin.

Use separate subnets and security rules.

Use separate subnets and security rules when configuring network resources. The VCN in which you want to create and deploy clusters must have at least two different subnets, and can have more:

- A Kubernetes API endpoint subnet
- A worker nodes subnet
- One regional, or two AD-specific, load balancer subnets (optional)
- A pods subnet (when using the OCI VCN-Native Pod Networking CNI plugin)
- A bastion subnet (optional)

You can choose to combine the subnets, and also to combine security rules. However, this approach makes security harder to manage and is therefore not recommended unless you are using network security groups to control access to clusters.

Security Best Practices

Plan exposure level.

Answer the following questions before implementing a security plan for the clusters you create with OKE:

- How much internet exposure do you want clusters to have?
- How do you plan to expose workloads internally to your VCN, and externally to the internet?
- How do you plan to scale workloads?
- Which types of Oracle services will the cluster consume?

Create private clusters.

If your cluster doesn't require direct access from the internet, create a private cluster. In a private cluster, the Kubernetes API server and worker nodes are assigned only private IP addresses.

Optionally use a NAT gateway for outbound internet access, a Dynamic Routing Gateway (DRG) to enable access from the on-premises network, a Local Peering Gateway (LPG) to allow access from other VCNs.

Place all applications in private subnets.

If the applications running on worker nodes don't require direct access to the internet, both the worker nodes subnet and the worker load balancer subnet should be private.

Restrict cluster traffic using Network Security Groups.

Define security rules in network security groups (NSGs), rather than in security lists, for the VCN in which you want to create and deploy clusters. See Controlling Traffic with Network Security Groups on page 163.

General security best practices.

- Apply security patches regularly.
- Use a combination of Kubernetes network policies and NSGs.
- Use NSGs in conjunction with infrastructure-as-code tools (such as Terraform).
- Rotate secrets and certificates regularly.
- Run all applications as a non-privileged user.
- Treat containers as immutable.

Auditing, logging, and monitoring.

- Check logs regularly.
- Enable audit logging.
- Use Kubernetes cluster-based logging.
- Monitor cluster components.
- Log network traffic metadata and analyze it regularly.
- Use small and secure container images.
- Limit credential exposure.

Storage Best Practices

- Choose the appropriate storage type.
- Create and use storage classes to define application needs.
- Create and use volumes for persistent storage.
- Limit storage resource consumption.
- Secure and back up data.

Upgrade Best Practices

- Use the latest supported version of Kubernetes.
- Set up test and production environments.
- Cordon and drain worker nodes in preparation for maintenance.
- Treat worker nodes as immutable.

Creating OKE Network Resources

Learn about the required network resources for Kubernetes Engine (OKE) on Compute Cloud@Customer.

The resource definitions in the following sections create a working example set of network resources for workload clusters. Use this configuration as a guide when you create these resources. You can change the values of properties such as CIDR blocks and IP addresses. Don't change the values of properties such as the network protocol, the stateful setting, or the private/public setting.

See Workload Cluster Network Ports (Flannel Overlay) on page 522 and Workload Cluster Network Ports (VCN-Native Pod) on page 551 for specific ports that must be open for specific purposes.

You can create network resources for two networking types:

- Creating Flannel Overlay Network Resources on page 520
- Creating VCN-Native Pod Network Resources on page 548

Public and Private Clusters on page 518 summarizes which network resources you need to create a public cluster and which network resources you need to create a private cluster.

Pod Networking

The Kubernetes networking model assumes containers (pods) have unique and routable IP addresses within a cluster. In the Kubernetes networking model, pods use those IP addresses to communicate with other pods on the same node in a cluster or on a different node, with pods on other clusters, with the cluster's control plane nodes, with other services (such as storage services), and with the internet.

By default, pods accept traffic from any source. To enhance cluster security, control access to and from pods using security rules defined as part of network security groups (recommended) or security lists. The security rules apply to all pods in all the worker nodes connected to the pod subnet specified for a node pool. See Controlling Traffic with Network Security Groups on page 163 and Controlling Traffic with Security Lists on page 158.

Public and Private Clusters

On Compute Cloud@Customer, before you create a cluster, decide what kind of network access the cluster requires: whether you need a public cluster or a private cluster. You cannot create both public and private clusters in one VCN.

The key difference between a public cluster and a private cluster is whether you configure public or private subnets for the Kubernetes API endpoint and the worker load balancer.

> **Note:**
>
> The subnets for the worker nodes and control plane nodes are always private.

For the worker nodes and control plane nodes, you can configure route rules that allow access only within the VCN or outside the VCN. This documentation names those route tables "vcn_private" and "nat_private." You can choose either of these private subnet configurations for your worker nodes and control plane nodes whether the cluster is private or the cluster is public.

Public Clusters

A public cluster requires the following network resources:

- A public subnet for the Kubernetes API endpoint. See the instructions for creating a public "control-plane-endpoint" subnet in Creating an OKE Control Plane Subnet (Flannel Overlay) on page 542 and Creating a Control Plane Subnet (VCN-Native Pod) on page 579.
- A public subnet for the worker load balancer. See the instructions for creating a public "service-lb" subnet in Creating a Worker Load Balancer Subnet (Flannel Overlay) on page 540 and Creating a Worker Load Balancer Subnet (VCN-Native Pod) on page 577.
- An internet gateway to connect resources on a public subnet to the internet using public IP addresses.
- A NAT gateway. Use a NAT gateway for outbound internet access. A NAT gateway connects resources on a private subnet to the internet without exposing private IP addresses.
- At least three free public IP addresses. Free public IP addresses are required for the NAT gateway, control plane load balancer, and worker load balancer.

  The worker load balancer requires a free public IP address to expose applications. The worker load balancer might require more free public IP addresses depending on the applications running on the pods.

Private Clusters

If you create multiple OKE VCNs, each CIDR must be unique. CIDRs of different VCNs for private clusters cannot overlap with any other VCN CIDRs or any on-premises CIDR. The IP addresses used must be exclusive to each VCN.

A private cluster has the following network resources:

- A private subnet for the Kubernetes API endpoint. See the instructions for creating a private "control-plane-endpoint" subnet in Creating an OKE Control Plane Subnet (Flannel Overlay) on page 542 and Creating a Control Plane Subnet (VCN-Native Pod) on page 579.

- A private subnet for the worker load balancer. See the instructions for creating a private "service-lb" subnet in Creating an OKE Control Plane Subnet (Flannel Overlay) on page 542 and Creating a Control Plane Load Balancer Subnet (VCN-Native Pod) on page 584.
- A route table with no route rules. This route table allows access only within the VCN.
- (Optional) A Local Peering Gateway (LPG). Use an LPG to allow access from other VCNs. An LPG allows access to the cluster from an instance running on a different VCN. Create an LPG on the OKE VCN, and create an LPG on a second VCN on the Compute Cloud@Customer. Use the LPG connect command to peer the two LPGs. Peered VCNs can be in different tenancies. CIDRs for the peered VCNs cannot overlap. See Connecting VCNs through a Local Peering Gateway on page 173.

  Create a route rule to steer VCN subnet traffic to and from the LPGs, and security rules to allow or deny certain types of traffic. See Creating a VCN (Flannel Overlay) on page 532 or Creating a VCN (VCN-Native Pod Networking) on page 563 for the route table to add to the OKE VCN and similar route table to add to the second VCN. Add the same route rule on the second VCN, specifying the OKE VCN CIDR as the destination.

  Install the OCI SDK and `kubectl` on the instance on the second VCN and connect to the private cluster. See Creating a Kubernetes Configuration File on page 596.
- (Optional) A Dynamic Routing Gateway (DRG). Use a DRG to enable access from the on-premises network. A DRG allows traffic between the OKE VCN and the on-premises network's IP address space. Create the DRG in the OKE VCN compartment, and then attach the OKE VCN to that DRG. See Connecting to the On-Premises Network through a Dynamic Routing Gateway (DRG) on page 174.

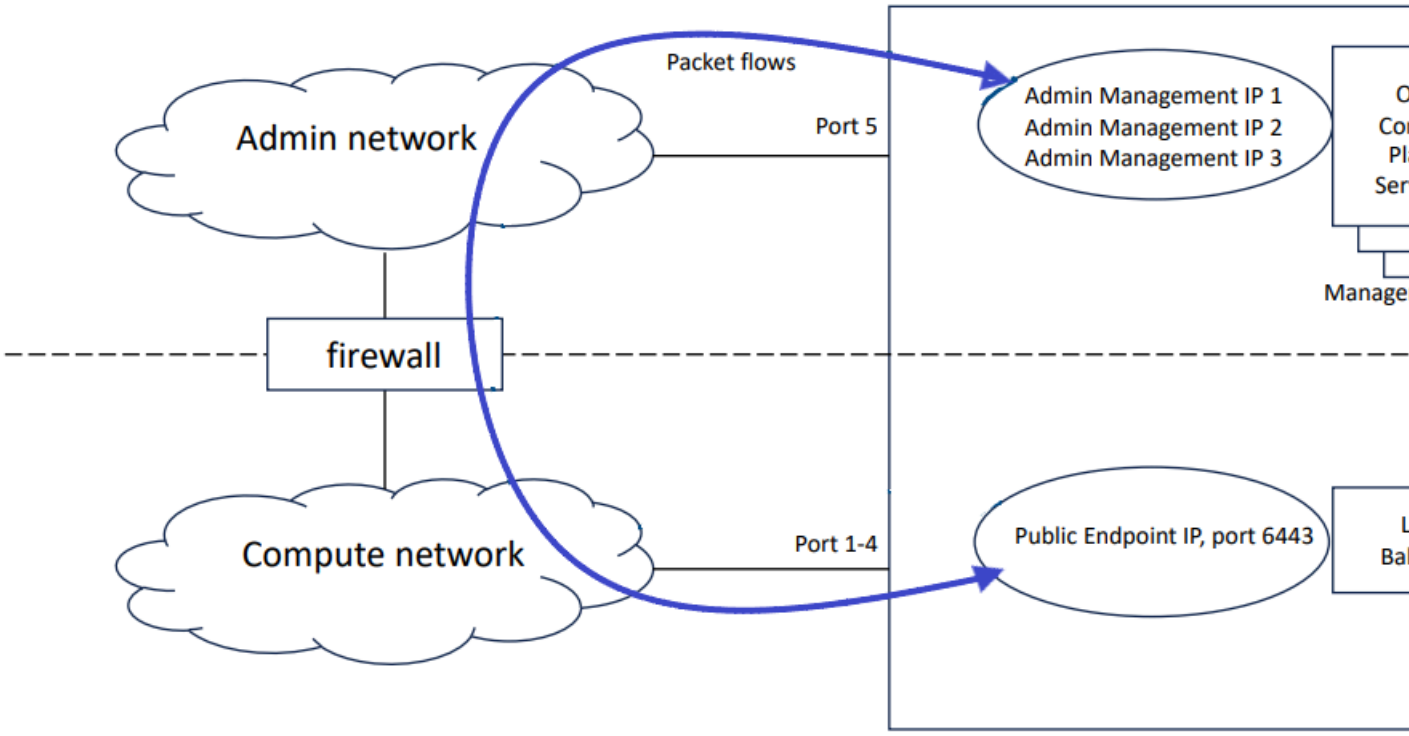
  Create a route rule to steer traffic to the on-premises data center network's IP address space. See Creating a VCN (Flannel Overlay) on page 532 or Creating a VCN (VCN-Native Pod Networking) on page 563 for the route table to add to the OKE VCN.

OKE Cluster Management Across Networks

Learn about OKE cluster management across networks on Compute Cloud@Customer.

Compute Cloud@Customer admin and data networks are configured for your environment by Oracle, when Oracle installs the Compute Cloud@Customer infrastructure in your data center.

The OKE service runs on the management nodes in the admin network, while the OKE clusters are deployed in the data network as shown in the following diagram.



The management interface of an OKE cluster is port 6443 on its load balancer public IP address. This address is assigned from the data center IP range you reserved and configured as public IPs during the Compute Cloud@Customer infrastructure installation.

Because of the network segregation, traffic from the OKE service must exit the infrastructure through the admin network, and reenter through the data network to reach the OKE cluster.

> **Important:**
>
> Your data center network infrastructure must allow traffic in both directions. Without the necessary firewall and routing rules, users can't deploy OKE clusters.

See Workload Cluster Network Ports (Flannel Overlay) on page 522 for ports needed for OKE.

Creating Flannel Overlay Network Resources

On Compute Cloud@Customer, the Flannel Overlay network type encapsulates communication between pods in the Flannel Overlay network.

The Flannel Overlay network is a simple private overlay virtual network that satisfies the requirements of the OKE networking model by attaching IP addresses to containers. The pods in the private overlay network are only accessible from other pods in the same cluster.

The resource definitions in the following sections create a working example set of network resources for workload clusters when you are using Flannel Overlay networking. Use this configuration as a guide when you create these resources. You can change the values of properties such as CIDR blocks and IP addresses. Don't change the values of properties such as the network protocol, the stateful setting, or the private/public setting.

Create all the network resources in the same compartment.

Use the following sections to manually configure flannel overlay network resources:

- Workload Cluster Network CIDR Ranges (Flannel Overlay) on page 521

- Workload Cluster Network Ports (Flannel Overlay) on page 522
- Creating a VCN (Flannel Overlay) on page 532
- Creating a Worker Subnet (Flannel Overlay) on page 536
- Creating a Worker Load Balancer Subnet (Flannel Overlay) on page 540
- Creating an OKE Control Plane Subnet (Flannel Overlay) on page 542
- Creating an OKE Control Plane Subnet (Flannel Overlay) on page 542

To use Terraform scripts to automate the flannel overlay network configuration, see Creating a Control Plane Load Balancer Subnet (VCN-Native Pod) on page 584.

Workload Cluster Network CIDR Ranges (Flannel Overlay)

Review the example CIDR ranges used for OKE network resources for Flannel Overlay networking on Compute Cloud@Customer.

Throughout this documentation, variables are used to represent CIDR ranges for instances in different subnets. The following table lists the CIDR variables and example values for use with Flannel Overlay networking.

> **Note:**
>
> These are examples only. The CIDR ranges you use depend on the number of clusters you have, the number of nodes in each cluster, and the type of networking you are using.

For Flannel Overlay networking, IP addresses are managed by the underlying Container service. Pods are not assigned IP addresses from the IP address pool that is defined in the pod subnet CIDR. This is the reason you do not need a pod subnet when you are using Flannel Overlay networking.

The primary difference between IP address requirements of Flannel Overlay networking and VCN-Native Pod Networking is that VCN-Native Pod Networking requires more IP addresses to be available. The table in Workload Cluster Network CIDR Ranges (VCN-Native Pod) on page 550 shows larger CIDR ranges than the following table for Flannel Overlay CIDR ranges. The CIDR ranges used with Flannel Overlay networking can be much smaller than the CIDR ranges used with VCN-Native Pod Networking.

Variable Name	Description	Example Value
vcn_cidr	VCN CIDR range	172.31.252.0/23
worker_cidr	Worker subnet CIDR	172.31.253.0/24
workerlb_cidr	Worker load balancer subnet CIDR	172.31.252.0/25
kmi_cidr	OKE control plane subnet CIDR	172.31.252.224/28
kmilb_cidr	OKE control plane load balancer subnet CIDR	172.31.252.240/28
kube_client_cidr	CIDR for clients that are allowed to contact the Kubernetes API server	10.0.0.0/8
public_ip_cidr	Public IP CIDR	10.0.0.0/8
kube_internal_cidr	CIDR used by the Kubernetes infrastructure to allocate IP addresses for various internal services and components	253.255.0.0/16

The IP Subnet Calculator on Calculator.net is one tool for finding all available networks for a given IP address and prefix length.

What's Next:

Review the network ports used for Flannel Overlay networking. See Workload Cluster Network Ports (Flannel Overlay) on page 522.

Workload Cluster Network Ports (Flannel Overlay)

Review the ports that are required for OKE for Flannel Overlay networking On Compute Cloud@Customer. These ports must be available to configure workload cluster networking. You might need to open more ports for other purposes.

The following table lists ports that are used by workload clusters when you use Flannel Overlay networking. These ports must be available to configure workload cluster networking. You might need to open more ports for other purposes.

All protocols are TCP. All port states are Stateful. Port 6443 is the port used for the Kubernetes API and is also referred to as *`kubernetes_api_port`*.

See also the table in Access Configuration With Administration Network.

Ports that Must Be Available for Use by Workload Clusters for Flannel Overlay Networking

Source IP Address	Destination IP Address	Port	Description
bastion host: *`vcn_cidr`*	Worker nodes subnet: *`worker_cidr`*	22	Outbound connections from the bastion host to the worker CIDR.
bastion host: *`vcn_cidr`*	Control plane subnet: *`kmi_cidr`*	22	Outbound connections from the bastion host to the control plane nodes.
Worker nodes subnet: *`worker_cidr`*	yum repository	80	Outbound connections from the worker CIDR to external applications.
Worker nodes subnet: *`worker_cidr`*	Secure yum repository	443	Secure outbound traffic from the worker CIDR to external applications.
Worker nodes subnet: *`worker_cidr`*	Container registry	5000	Outbound connections from the worker CIDR to the container registry.
Worker nodes subnet: *`worker_cidr`*	Control plane subnet: *`kmi_cidr`*	6443	Outbound connections from the worker CIDR to the Kubernetes API. This is necessary to allow nodes to join through either a public IP address on one of the nodes or the load balancer public IP address. Port 6443 is called the ***`kubernetes_api_port`***.
Worker nodes subnet: *`worker_cidr`*	Control plane load balancer	6443	Inbound connections from the worker CIDR to the Kubernetes API.
CIDR for clients: *`kube_client_cidr`*	Control plane load balancer	6443	Inbound connections from clients to the Kubernetes API server.
Worker nodes subnet: *`worker_cidr`*	Control plane subnet: *`kmi_cidr`*	6443	Private outbound connections from the worker CIDR to `kubeapi` on the control plane subnet.
`kube_client_cidr`	Worker nodes subnet: *`worker_cidr`*	30000-32767	Inbound traffic for applications from Kubernetes clients.

What's Next:

Depending on how you want to create the network resources, see one of the following sections:

- Use Terraform scripts to automate the creation of the network resources. See Example Terraform Scripts for Network Resources (Flannel Overlay) on page 523.
- Begin to create the network resources individually using the Compute Cloud@Customer Console, CLI, or API. See Example Terraform Scripts (VCN-Native Pod) on page 552.

Example Terraform Scripts for Network Resources (Flannel Overlay)

On Compute Cloud@Customer, you can use Terraform scripts to automate the creation of OKE network resources for Flannel Overlay networking.

The following Terraform scripts create the network resources that are required by Kubernetes Engine (OKE) when you're using Flannel Overlay networking. Subsequent sections in this topic show other ways to define these same network resources.

Most of the values shown in these scripts, such as resource display names and CIDRs, are examples. Some ports must be specified as shown (see Workload Cluster Network Ports (Flannel Overlay) on page 522), and the OKE control plane subnet must be named `control-plane`. See Workload Cluster Network CIDR Ranges (Flannel Overlay) on page 521 for comments about CIDR values.

Terraform Script Examples

- variables.tf on page 523
- terraform.tfvars on page 524
- provider.tf on page 525
- main.tf on page 525
- oke_vcn.tf on page 527
- oke_worker_seclist.tf on page 528
- oke_worker_subnet.tf on page 529
- oke_kmi_seclist.tf on page 529
- oke_kmi_subnet.tf on page 531

variables.tf

This file creates several variables that are used to configure OKE network resources when you're using Flannel Overlay networking. Many of these variables aren't assigned values in this file. One port and five CIDRs are assigned values. The `kubernetes_api_port`, port 6443, is the port used to access the Kubernetes API. See also Workload Cluster Network Ports (Flannel Overlay) on page 522. The five CIDRs that are defined in this file are for the OKE VCN, worker subnet, worker load balancer subnet, control plane subnet, and control plane load balancer subnet.

```
variable "oci_config_file_profile" {
  type    = string
  default = "DEFAULT"
}

variable "tenancy_ocid" {
  description = "tenancy OCID"
  type        = string
  nullable    = false
}

variable "compartment_id" {
  description = "compartment OCID"
  type        = string
  nullable    = false
}

variable "vcn_name" {
  description = "VCN name"
  nullable    = false
}
```

```
variable "kube_client_cidr" {
  description = "CIDR of Kubernetes API clients"
  type        = string
  nullable    = false
}

variable "public_ip_cidr" {
  description = "Public IP CIDR"
  type        = string
  nullable    = false
}

variable "kubernetes_api_port" {
  description = "Port used for Kubernetes API"
  type        = string
  default     = "6443"
}

variable "vcn_cidr" {
  default = "172.31.252.0/23"
}

# Subnet for KMIs where kube-apiserver and other control
# plane applications run, max 9 nodes
variable "kmi_cidr" {
  description = "Kubernetes control plane subnet CIDR"
  default     = "172.31.252.224/28"
}

# Subnet for KMI load balancer
variable "kmilb_cidr" {
  description = "Kubernetes control plane LB subnet CIDR"
  default     = "172.31.252.240/28"
}

# Subnet for worker nodes, max 128 nodes
variable "worker_cidr" {
  description = "Kubernetes worker subnet CIDR"
  default     = "172.31.253.0/24"
}

# Subnet for worker load balancer (for use by CCM)
variable "workerlb_cidr" {
  description = "Kubernetes worker LB subnet CIDR"
  default     = "172.31.252.0/25"
}

# Flag to Enable private endpoint
variable "enable_private_endpoint" {
  description = "Flag to create private control plane endpoint/service-lb"
  type = bool
  default = false
  nullable = false
}
```

terraform.tfvars

This file assigns values to some of the variables that were created in `variables.tf`.

```
# name of the profile to use from $HOME/.oci/config
oci_config_file_profile = "DEFAULT"
```

```
# tenancy ocid from the above profile
tenancy_ocid = "tenancy_OCID"

# compartment in which to build the OKE cluster
compartment_id = "compartment_OCID"

# display-name for the OKE VCN
vcn_name = "oketest"
```

provider.tf

This file is required to use the OCI provider. The file initializes the OCI module using the OCI profile configuration file.

```
provider "oci" {
  config_file_profile = var.oci_config_file_profile
  tenancy_ocid        = var.tenancy_ocid
}
```

main.tf

This file specifies the provider to use (`oracle/oci`), defines several security list rules, and initializes required local variables.

The version of the OCI provider that you use must be at least v4.50.0 but no greater than v6.36.0.

```
terraform {
  required_providers {
    oci = {
      source  = "oracle/oci"
      version = ">= 4.50.0, <= 6.36.0"
      # If necessary, you can pin a specific version here
      #version = "4.71.0"
    }
  }
  required_version = ">= 1.1"
}

locals {
  kube_internal_cidr = "253.255.0.0/16"
  worker_lb_ingress_rules = [
    {
      source   = var.kube_client_cidr
      port_min = 80
      port_max = 80
    },
    {
      source   = var.kube_client_cidr
      port_min = 443
      port_max = 443
    }
  ]
  worker_ingress_rules = [
     {
      source   = var.kube_client_cidr
      port_min = 30000
      port_max = 32767
    },
    {
      source   = var.vcn_cidr
      port_min = 22
      port_max = 22
```

```
    },
    {
      source   = var.workerlb_cidr
      port_min = 30000
      port_max = 32767
    },
    {
      source   = var.workerlb_cidr
      port_min = 10256
      port_max = 10256
    },
    {
      source   = var.kmi_cidr
      port_min = 22
      port_max = 65535
    },
  ]

  kmi_lb_ingress_rules = [
    {
      source   = local.kube_internal_cidr
      port_min = var.kubernetes_api_port
      port_max = var.kubernetes_api_port
    },
    {
      source   = var.kube_client_cidr
      port_min = var.kubernetes_api_port
      port_max = var.kubernetes_api_port
    },
    {
      source   = var.vcn_cidr
      port_min = var.kubernetes_api_port
      port_max = var.kubernetes_api_port
    },
  ]
  kmi_ingress_rules = [
    {
      source   = var.kube_client_cidr
      port_min = var.kubernetes_api_port
      port_max = var.kubernetes_api_port
    },
    {
      source   = var.kmilb_cidr
      port_min = var.kubernetes_api_port
      port_max = var.kubernetes_api_port
    },
    {
      source   = var.worker_cidr
      port_min = 1024
      port_max = 65535
    },
    {
      source   = var.kmi_cidr
      port_min = 1024
      port_max = 65535
    },
  ]
  worker_ingress_udp_rules = [
    {
      source   = var.worker_cidr
      port_min = 8285
      port_max = 8472
    },
    {
```

```
      source   = var.kmi_cidr
      port_min = 8285
      port_max = 8472
    },
  ]
  kmi_ingress_udp_rules = [
    {
      source   = var.worker_cidr
      port_min = 8285
      port_max = 8472
    },
    {
      source   = var.kmi_cidr
      port_min = 8285
      port_max = 8472
    },
  ]
}
```

oke_vcn.tf

This file defines a VCN, NAT gateway, internet gateway, private route table, and public route table. The private route table is the default route table for the VCN.

```
resource "oci_core_vcn" "oke_vcn" {
  cidr_block     = var.vcn_cidr
  dns_label      = var.vcn_name
  compartment_id = var.compartment_id
  display_name   = "${var.vcn_name}-vcn"
}

resource "oci_core_nat_gateway" "vcn_ngs" {
  compartment_id = var.compartment_id
  vcn_id         = oci_core_vcn.oke_vcn.id
  count          = var.enable_private_endpoint ? 0:1

  display_name = "VCN nat g6s"
}

resource "oci_core_internet_gateway" "vcn_igs" {
  compartment_id = var.compartment_id
  vcn_id         = oci_core_vcn.oke_vcn.id
  count          = var.enable_private_endpoint ? 0:1

  display_name = "VCN i6t g6s"
  enabled      = true
}

resource "oci_core_default_route_table" "default_private" {
  manage_default_resource_id = oci_core_vcn.oke_vcn.default_route_table_id
  display_name               = "Default - private"
  count                      = var.enable_private_endpoint ? 1:0
}

resource "oci_core_default_route_table" "private" {
  count          = var.enable_private_endpoint ? 0:1
  manage_default_resource_id = oci_core_vcn.oke_vcn.default_route_table_id
  display_name               = "Default - private"

  route_rules {
    destination       = "0.0.0.0/0"
    destination_type  = "CIDR_BLOCK"
```

```
    network_entity_id = oci_core_nat_gateway.vcn_ngs[0].id
  }
}

resource "oci_core_route_table" "public" {
  count          = var.enable_private_endpoint ? 0:1
  compartment_id = var.compartment_id
  vcn_id         = oci_core_vcn.oke_vcn.id

  display_name = "public"
  route_rules {
    destination       = "0.0.0.0/0"
    destination_type  = "CIDR_BLOCK"
    network_entity_id = oci_core_internet_gateway.vcn_igs[0].id
  }
}
```

oke_worker_seclist.tf

This file defines the security lists for both the worker subnet and the worker load balancer subnet. The rules for these security lists were defined in other Terraform files in this set.

```
resource "oci_core_security_list" "workerlb" {
  display_name   = "${var.vcn_name}-workerlb"
  compartment_id = var.compartment_id
  vcn_id         = oci_core_vcn.oke_vcn.id

  dynamic "ingress_security_rules" {
    iterator = port
    for_each = local.worker_lb_ingress_rules

    content {
      source      = port.value.source
      source_type = "CIDR_BLOCK"
      protocol    = "6"
      tcp_options {
        min = port.value.port_min
        max = port.value.port_max
      }
    }
  }
}

resource "oci_core_security_list" "worker" {
  display_name   = "${var.vcn_name}-worker"
  compartment_id = var.compartment_id
  vcn_id         = oci_core_vcn.oke_vcn.id

  dynamic "ingress_security_rules" {
    iterator = port
    for_each = local.worker_ingress_rules

    content {
      source      = port.value.source
      source_type = "CIDR_BLOCK"
      protocol    = "6"
      tcp_options {
        min = port.value.port_min
        max = port.value.port_max
      }
    }
  }
```

```
  dynamic "ingress_security_rules" {
    iterator = port
    for_each = local.worker_ingress_udp_rules

    content {
      source      = port.value.source
      source_type = "CIDR_BLOCK"
      protocol    = "17"
      udp_options {
        min = port.value.port_min
        max = port.value.port_max
      }
    }
  }
}
```

oke_worker_subnet.tf

This file defines the worker and worker load balancer subnets. The worker load balancer subnet is named `service-lb`.

```
resource "oci_core_subnet" "worker" {
  cidr_block     = var.worker_cidr
  compartment_id = var.compartment_id
  vcn_id         = oci_core_vcn.oke_vcn.id

  display_name   = "worker"
  dns_label      = "worker"
  prohibit_public_ip_on_vnic = true

  security_list_ids = [
    oci_core_default_security_list.oke_vcn.id,
    oci_core_security_list.worker.id
  ]
}

resource "oci_core_subnet" "worker_lb" {
  cidr_block     = var.workerlb_cidr
  compartment_id = var.compartment_id
  vcn_id         = oci_core_vcn.oke_vcn.id

  display_name               = "service-lb"
  dns_label                  = "servicelb"
  prohibit_public_ip_on_vnic = var.enable_private_endpoint
  route_table_id             = var.enable_private_endpoint==false ?
 oci_core_route_table.public[0].id :
 oci_core_vcn.oke_vcn.default_route_table_id

  security_list_ids = [
    oci_core_default_security_list.oke_vcn.id,
    oci_core_security_list.workerlb.id
  ]
}
```

oke_kmi_seclist.tf

This file defines the security lists for the control plane and control plane load balancer subnets. This file also defines updates to make to the default security list for the VCN.

```
resource "oci_core_default_security_list" "oke_vcn" {
```

```
  manage_default_resource_id = oci_core_vcn.oke_vcn.default_security_list_id

  egress_security_rules {
    destination      = "0.0.0.0/0"
    destination_type = "CIDR_BLOCK"
    protocol         = "all"
  }

  dynamic "ingress_security_rules" {
    iterator = icmp_type
    for_each = [3, 8, 11]

    content {
      # ping from VCN; unreachable/TTL from anywhere
      source      = (icmp_type.value == "8" ? var.vcn_cidr : "0.0.0.0/0")
      source_type = "CIDR_BLOCK"
      protocol    = "1"
      icmp_options {
        type = icmp_type.value
      }
    }
  }
}

resource "oci_core_security_list" "kmilb" {
  compartment_id = var.compartment_id
  vcn_id         = oci_core_vcn.oke_vcn.id

  display_name = "${var.vcn_name}-kmilb"

  dynamic "ingress_security_rules" {
    iterator = port
    for_each = local.kmi_lb_ingress_rules

    content {
      source      = port.value.source
      source_type = "CIDR_BLOCK"
      protocol    = "6"
      tcp_options {
        min = port.value.port_min
        max = port.value.port_max
      }
    }
  }

  dynamic "ingress_security_rules" {
    for_each = var.enable_private_endpoint ? [] : [0]

    content {
      source      = var.public_ip_cidr
      source_type = "CIDR_BLOCK"
      protocol    = "6"
      tcp_options {
        min = var.kubernetes_api_port
        max = var.kubernetes_api_port
      }
    }
  }
}

resource "oci_core_security_list" "kmi" {
  compartment_id = var.compartment_id
  vcn_id         = oci_core_vcn.oke_vcn.id
```

```
  display_name = "${var.vcn_name}-kmi"

  dynamic "ingress_security_rules" {
    iterator = port
    for_each = local.kmi_ingress_rules

    content {
      source      = port.value.source
      source_type = "CIDR_BLOCK"
      protocol    = "6"
      tcp_options {
        min = port.value.port_min
        max = port.value.port_max
      }
    }
  }

  dynamic "ingress_security_rules" {
    iterator = port
    for_each = local.kmi_ingress_udp_rules

    content {
      source      = port.value.source
      source_type = "CIDR_BLOCK"
      protocol    = "17"
      udp_options {
        min = port.value.port_min
        max = port.value.port_max
      }
    }
  }
}
```

oke_kmi_subnet.tf

This file defines the control plane and control plane load balancer subnets.

> **Important:**
>
> The name of the `kmi` subnet must be exactly `control-plane`.

```
resource "oci_core_subnet" "kmi" {
  cidr_block                 = var.kmi_cidr
  compartment_id             = var.compartment_id
  display_name               = "control-plane"
  dns_label                  = "kmi"
  vcn_id                     = oci_core_vcn.oke_vcn.id
  prohibit_public_ip_on_vnic = true
  security_list_ids = [
    oci_core_default_security_list.oke_vcn.id,
    oci_core_security_list.kmi.id
  ]
}

resource "oci_core_subnet" "kmi_lb" {
  cidr_block                 = var.kmilb_cidr
  compartment_id             = var.compartment_id
  dns_label                  = "kmilb"
  vcn_id                     = oci_core_vcn.oke_vcn.id
  display_name               = "control-plane-endpoint"
  prohibit_public_ip_on_vnic = var.enable_private_endpoint
```

```
  route_table_id             = var.enable_private_endpoint==false ?
 oci_core_route_table.public[0].id :
 oci_core_default_route_table.default_private[0].id
  security_list_ids = [
    oci_core_default_security_list.oke_vcn.id,
    oci_core_security_list.kmilb.id
  ]
}
```

Creating a VCN (Flannel Overlay)

Learn how to create a VCN for Flannel Overlay networking On Compute Cloud@Customer.

Create the following resources in the order listed:

1. Create the VCN on page 532
2. Create a route table with the following route rules:
   - Public clusters:
     - Internet gateway and a route table with a route rule that references that internet gateway.
     - NAT gateway and a route table with a route rule that references that NAT gateway.
   - Private clusters:
     - Route table with no route rules.
     - (Optional) Dynamic Routing Gateway (DRG) and a route table with a route rule that references that DRG. See Private Clusters on page 518.
     - (Optional) Local Peering Gateway (LPG) and a route table with a route rule that references that LPG. See Private Clusters on page 518.
3. Modify the VCN Default Security List on page 535

Resource names and CIDR blocks are example values.

Create the VCN

To create the VCN, follow the instructions in Creating a VCN on page 141, and use the parameters listed in this section. For Terraform input, see Example Terraform Scripts for Network Resources (Flannel Overlay) on page 523.

> **Note:**
>
> Subnets are created later, and described in later sections.

For this example, use the following input to create the VCN. The VCN covers one contiguous CIDR block. The CIDR block can't be changed after the VCN is created.

Console property	CLI property
• Name: oketest-vcn • CIDR Block: *vcn_cidr* • DNS Label: oketest This label must be unique across all VCNs in the tenancy.	• `--display-name`: `oketest-vcn` • `--cidr-blocks`: `'["`*vcn_cidr*`"]'` • `--dns-label`: `oketest` This label must be unique across all VCNs in the tenancy.

Note the OCID of the new VCN for use later. In the examples in this guide, this VCN OCID is `ocid1.vcn.`*oke_vcn_id*.

Next Steps

- Public internet access. For traffic on a public subnet that connects to the internet using public IP addresses, create an internet gateway and a route rule that references that internet gateway.

- Private internet access. For traffic on a private subnet that needs to connect to the internet without exposing private IP addresses, create a NAT gateway and a route rule that references that NAT gateway.
- VCN-only access. To restrict communication to only other resources on the same VCN, use the default route table, which has no route rules.
- Instances in another VCN. To enable communication between the cluster and an instance running on a different VCN, create a Local Peering Gateway (LPG) and a route rule that references that LPG.
- On-premises IP address space. To enable communication between the cluster and the on-premises network IP address space, create a Dynamic Routing Gateway (DRG), attach the OKE VCN to that DRG, and create a route rule that references that DRG.

Edit the VCN Private Route Table

Edit the default route table that was created when you created the VCN. Change the name of the route table to vcn_private. This route table does not have any route rules. Do not add any route rules.

Create a NAT Private Route Table

Create a NAT gateway and a route table with a route rule that references the NAT gateway.

NAT Gateway

To create the NAT gateway, use the instructions in Configuring a NAT Gateway on page 169. For Terraform input, see Example Terraform Scripts for Network Resources (Flannel Overlay) on page 523.

Note the name and OCID of the NAT gateway for assignment to the private route rule.

Private Route Rule

Modify the default route table, using the following input to create a private route rule that references the NAT gateway that was created in the preceding step. See Creating a Route Table on page 155.

For this example, use the following input to create the route table with a private route rule that references the NAT gateway that was created in the preceding step.

Console property	CLI property
• Name: nat_private	• `--display-name`: nat_private
Route rule	`--route-rules`
• Target Type: NAT Gateway • NAT Gateway: Name of the NAT gateway that was created in the preceding step • CIDR Block: 0.0.0.0/0 • Description: NAT private route rule	• `networkEntityId`: OCID of the NAT gateway that was created in the preceding step • `destinationType`: `CIDR_BLOCK` • `destination`: `0.0.0.0/0` • `description`: NAT private route rule

Note the name and OCID of this route table for assignment to private subnets.

Create a Local Peering Gateway

Create a Local Peering gateway (LPG) and a route table with a route rule that references the LPG.

Local Peering Gateway

Create the LPG. See Connecting VCNs through a Local Peering Gateway (LPG) on page 172.

Note the name and OCID of the LPG for assignment to the private route rule.

Private Route Rule

Create a route table. See Creating a Route Table on page 155.

For this example, use the following input to create the route table with a private route rule that references the LPG that was created in the preceding step.

Console property	CLI property
• Name: lpg_rt Route rule • Target Type: Local Peering Gateway • Local Peering Gateway: Name of the LPG that was created in the preceding step • CIDR Block: *CIDR_for_the_second_VCN* • Description: LPG private route rule	• `--display-name`: lpg_rt `--route-rules` • `networkEntityId`: OCID of the LPG that was created in the preceding step • `destinationType`: `CIDR_BLOCK` • `destination`: *CIDR_for_the_second_VCN* • `description`: LPG private route rule

Note the name and OCID of this route table for assignment to the "control-plane-endpoint" subnet (Creating an OKE Control Plane Subnet (Flannel Overlay) on page 542).

Add the same route rule on the second VCN (the peered VCN), specifying the OKE VCN CIDR as the destination.

Create a Dynamic Routing Gateway

Create a Dynamic Routing gateway (DRG) and a route table with a route rule that references the DRG.

Dynamic Routing Gateway

To create the DRG and attach the OKE VCN to that DRG See Connecting to the On-Premises Network through a Dynamic Routing Gateway (DRG) on page 174. Create the DRG in the OKE VCN compartment, and then attach the OKE VCN to that DRG.

Note the name and OCID of the DRG for assignment to the private route rule.

Private Route Rule

Create a route table. See Creating a Route Table on page 155.

For this example, use the following input to create the route table with a private route rule that references the DRG that was created in the preceding step.

Compute Cloud@Customer Console property	CLI property
• Name: drg_rt Route rule • Target Type: Dynamic Routing Gateway • Dynamic Routing: Name of the DRG that was created in the preceding step • CIDR Block: 0.0.0.0/0 • Description: DRG private route rule	• `--display-name`: drg_rt `--route-rules` • `networkEntityId`: OCID of the DRG that was created in the preceding step • `destinationType`: `CIDR_BLOCK` • `destination`: `0.0.0.0/0` • `description`: DRG private route rule

Note the name and OCID of this route table for assignment to the "control-plane-endpoint" subnet (Creating an OKE Control Plane Subnet (Flannel Overlay) on page 542).

Create a Public Route Table

Create an Internet gateway and a route table with a route rule that references the Internet gateway. This enables internet access for OKE nodes.

Create an Internet Gateway

To create the internet gateway, use the instructions in Configuring an Internet Gateway on page 170. For Terraform input, see Example Terraform Scripts for Network Resources (Flannel Overlay) on page 523.

Note the name and OCID of the internet gateway for assignment to the public route rule.

Create a Public Route Rule

Create a public route rule for the internet gateway you just created. To create a route table, use the instructions in Creating a Route Table on page 155. For Terraform input, see Example Terraform Scripts for Network Resources (Flannel Overlay) on page 523.

For this example, use the following input to create the route table with a public route rule that references the internet gateway that was created in the preceding step.

Compute Cloud@Customer Console property	**CLI property**
• Name: public Route rule • Target Type: Internet Gateway • Internet Gateway: Name of the internet gateway that was created in the preceding step • CIDR Block: 0.0.0.0/0 • Description: OKE public route rule	• `--vcn-id`: `ocid1.vcn.`*`oke_vcn_id`* • `--display-name`: public `--route-rules` • `networkEntityId`: OCID of the internet gateway that was created in the preceding step • `destinationType`: `CIDR_BLOCK` • `destination`: `0.0.0.0/0` • `description`: OKE public route rule

Modify the VCN Default Security List

To modify a security list, see Updating a Security List on page 162. For Terraform input, see Example Terraform Scripts for Network Resources (Flannel Overlay) on page 523.

Delete all the default rules, then create the rules shown in the following table.

Compute Cloud@Customer Console property	**CLI property**
• Name: Default	`--security-list-id`: `ocid1.securitylist.`*`default_securitylist_id`*
One egress security rule: • Stateless: clear the box • Egress CIDR: 0.0.0.0/0 • IP Protocol: All protocols • Description: "Allow all outgoing traffic."	**One egress security rule:** `--egress-security-rules` • `isStateless`: `false` • `destination`: `0.0.0.0/0` • `destinationType`: `CIDR_BLOCK` • `protocol`: `all` • `description`: "Allow all outgoing traffic."
Three ingress security rules:	**Three ingress security rules:** `--ingress-security-rules`

Compute Cloud@Customer Console property	**CLI property**

Ingress Rule 1
• Stateless: clear the box
• Ingress CIDR: *vcn_cidr*
• IP Protocol: ICMP
  • Parameter Type: 8: Echo
• Description: "Allow ping from VCN."
	Ingress Rule 1
	• `isStateless`: `false`
	• `source`: *vcn_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `1`
	• `icmpOptions`
	  • `type`: `8`
	• `description`: "Allow ping from VCN."

Ingress Rule 2
• Stateless: clear the box
• Ingress CIDR: 0.0.0.0/0
• IP Protocol: ICMP
  • Parameter Type: 3: Destination Unreachable
• Description: "Blocks incoming requests from any source."
	Ingress Rule 2
	• `isStateless`: `false`
	• `source`: `0.0.0.0/0`
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `1`
	• `icmpOptions`
	  • `type`: `3`
	• `description`: "Blocks incoming requests from any source."

Ingress Rule 3
• Stateless: clear the box
• Ingress CIDR: 0.0.0.0/0
• IP Protocol: ICMP
  • Parameter Type: 11: Time Exceeded
• Description: "Time exceeded."
	Ingress Rule 3
	• `isStateless`: `false`
	• `source`: `0.0.0.0/0`
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `1`
	• `icmpOptions`
	  • `type`: `11`
	• `description`: "Time exceeded."

Note the name and OCID of this default security list for assignment to subnets.

What's Next:

Creating a Worker Subnet (Flannel Overlay) on page 536

Creating a Worker Subnet (Flannel Overlay)

Learn how to create a worker subnet for Flannel Overlay networking On Compute Cloud@Customer.

Create the following resources in the order listed:

1. Create a Worker Security List on page 536.
2. Create the Worker Subnet on page 539.

Create a Worker Security List

To create a security list, use the instructions in Creating a Security List on page 159. For Terraform input, see Example Terraform Scripts for Network Resources (Flannel Overlay) on page 523.

This security list defines traffic that's allowed to contact worker nodes directly.

For this example, use the following input for the worker subnet security list.

Compute Cloud@Customer Console	CLI property
• Name: worker-seclist	• `--vcn-id`: `ocid1.vcn.`*`oke_vcn_id`* • `--display-name`: `worker-seclist`
Seven ingress security rules:	**Seven ingress security rules:** `--ingress-security-rules`

Ingress Rule 1
• Stateless: clear the box
• Ingress CIDR: *`vcn_cidr`*
• IP Protocol: TCP
  • Destination Port Range: 22
• Description: "Allow intra-VCN `ssh`."
	Ingress Rule 1
	• `isStateless`: `false`
	• `source`: *`vcn_cidr`*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `22`
	  • `min`: `22`
	• `description`: "Allow intra-VCN `ssh`."

Ingress Rule 2
• Stateless: clear the check box
• Ingress CIDR: *`kube_client_cidr`*
• IP Protocol: TCP
  • Destination Port Range: 30000-32767
• Description: "Allow clients to contact the node port range."
	Ingress Rule 2
	• `isStateless`: `false`
	• `source`: *`kube_client_cidr`*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `32767`
	  • `min`: `30000`
	• `description`: "Allow clients to contact the node port range."

Ingress Rule 3
• Stateless: clear the box
• Ingress CIDR: *`workerlb_cidr`*
• IP Protocol: TCP
  • Destination Port Range: 30000-32767
• Description: "Allow the worker load balancer to contact the worker nodes."
	Ingress Rule 3
	• `isStateless`: `false`
	• `source`: *`workerlb_cidr`*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `32767`
	  • `min`: `30000`
	• `description`: "Allow the worker load balancer to contact the worker nodes."

Compute Cloud@Customer Console	CLI property

Ingress Rule 4
• Stateless: clear the box
• Ingress CIDR: *workerlb_cidr*
• IP Protocol: TCP
  • Destination Port Range: 10256
• Description: "Allow the worker load balancer to contact the worker nodes."
	Ingress Rule 4
	• `isStateless`: `false`
	• `source`: *workerlb_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `10256`
	  • `min`: `10256`
	• `description`: "Allow the worker load balancer to contact the worker nodes."

Ingress Rule 5
• Stateless: clear the box
• Ingress CIDR: *kmi_cidr*
• IP Protocol: TCP
  • Destination Port Range: 22-65535
• Description: "Allow the control plane to contact the worker nodes."
	Ingress Rule 5
	• `isStateless`: `false`
	• `source`: *kmi_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `65535`
	  • `min`: `22`
	• `description`: "Allow the control plane to contact the worker nodes."

Ingress Rule 6
• Stateless: clear the box
• Ingress CIDR: *worker_cidr*
• IP Protocol: UDP
  • Destination Port Range: 8285-8472
• Description: "Allow flannel traffic."
	Ingress Rule 6
	• `isStateless`: `false`
	• `source`: *worker_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `17`
	• `udpOptions`
	`destinationPortRange`
	  • `max`: `8472`
	  • `min`: `8285`
	• `description`: "Allow flannel traffic."

Compute Cloud@Customer Console	CLI property

Ingress Rule 7
• Stateless: clear the box
• Ingress CIDR: *kmi_cidr*
• IP Protocol: UDP
  • Destination Port Range: 8285-8472
• Description: "Allow flannel traffic."
	Ingress Rule 7
	• `isStateless`: `false`
	• `source`: *kmi_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `17`
	• `udpOptions`
	`destinationPortRange`
	  • `max`: `8472`
	  • `min`: `8285`
	• `description`: "Allow flannel traffic."

Create the Worker Subnet

To create a subnet, use the instructions in Creating a Subnet on page 142. For Terraform input, see Example Terraform Scripts for Network Resources (Flannel Overlay) on page 523.

For this example, use the following input for the worker subnet security list. Use the OCID of the VCN that was created in Example Terraform Scripts for Network Resources (Flannel Overlay) on page 523. Create the worker subnet in the same compartment where you created the VCN.

Create either a NAT private worker subnet or a VCN private worker subnet. Create a NAT private worker subnet to communicate outside the VCN.

Create a NAT Private Worker Subnet

Compute Cloud@Customer Console property	CLI property

• Name: worker
• CIDR Block: *worker_cidr*
• Route Table: Select "nat_private" from the list
• Private Subnet: check the box
• DNS Hostnames:
  Use DNS Hostnames in this Subnet: check the box
  • DNS Label: worker
• Security Lists: Select "worker-seclist" and "Default Security List for oketest-vcn" from the list
	• `--vcn-id`: `ocid1.vcn.`*oke_vcn_id*
	• `--display-name`: `worker`
	• `--cidr-block`: *worker_cidr*
	• `--dns-label`: `worker`
	• `--prohibit-public-ip-on-vnic`: `true`
	• `--route-table-id`: OCID of the "nat_private" route table
	• `--security-list-ids`: OCIDs of the "worker-seclist" security list and the "Default Security List for oketest-vcn" security list

The difference in the following private subnet is the VCN private route table is used instead of the NAT private route table.

Create a VCN Private Worker Subnet

Compute Cloud@Customer Console property	CLI property
• Name: worker • CIDR Block: *worker_cidr* • Route Table: Select "vcn_private" from the list • Private Subnet: check the box • DNS Hostnames: Use DNS Hostnames in this Subnet: check the box • DNS Label: worker • Security Lists: Select "worker-seclist" and "Default Security List for oketest-vcn" from the list	• `--vcn-id`: `ocid1.vcn.`*oke_vcn_id* • `--display-name`: `worker` • `--cidr-block`: *worker_cidr* • `--dns-label`: `worker` • `--prohibit-public-ip-on-vnic`: `true` • `--route-table-id`: OCID of the "vcn_private" route table • `--security-list-ids`: OCIDs of the "worker-seclist" security list and the "Default Security List for oketest-vcn" security list

What's Next:

Creating a Worker Load Balancer Subnet (Flannel Overlay) on page 540

Creating a Worker Load Balancer Subnet (Flannel Overlay)

Learn how to create a worker load balancer subnet for Flannel Overlay networking On Compute Cloud@Customer.

Create the following resources in the order listed:

1. Create a Worker Load Balancer Security List on page 540
2. Create the Worker Load Balancer Subnet on page 541

Create a Worker Load Balancer Security List

Create a security list and a worker load balancer subnet. To create a security list, use the instructions in Creating a Security List on page 159. For Terraform input, see Example Terraform Scripts for Network Resources (Flannel Overlay) on page 523.

This security list defines traffic, such as applications, that's allowed to contact the worker load balancer.

For this example, use the following input for the worker load balancer subnet security list. These sources and destinations are examples; adjust these for your applications.

> **Note:**
>
> When you create an external load balancer for your containerized applications (see Exposing Containerized Applications on page 633), remember to add that load balancer service front-end port to this security list.

Compute Cloud@Customer Console property	CLI property
• Name: workerlb-seclist	• `--vcn-id`: `ocid1.vcn.`*oke_vcn_id* • `--display-name`: `workerlb-seclist`
Two ingress security rules:	**Two ingress security rules:** `--ingress-security-rules`

Compute Cloud@Customer Console property	CLI property

Ingress Rule 1
• Stateless: clear the box
• Ingress CIDR: *kube_client_cidr*
• IP Protocol: TCP
  • Destination Port Range: 80
• Description: "Allow inbound traffic for applications."
	Ingress Rule 1
	• `isStateless`: `false`
	• `source`: *kube_client_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `80`
	  • `min`: `80`
	• `description`: "Allow inbound traffic for applications."

Ingress Rule 2
• Stateless: clear the box
• Ingress CIDR: *kube_client_cidr*
• IP Protocol: TCP
  • Destination Port Range: 443
• Description: "Allow inbound traffic for applications."
	Ingress Rule 2
	• `isStateless`: `false`
	• `source`: *kube_client_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `443`
	  • `min`: `443`
	• `description`: "Allow inbound traffic for applications."

Create the Worker Load Balancer Subnet

To create a subnet, use the instructions in Creating a Subnet on page 142 For Terraform input, see Example Terraform Scripts for Network Resources (Flannel Overlay) on page 523.

For this example, use the following input to create the worker load balancer subnet. Use the OCID of the VCN that was created in Creating a VCN (Flannel Overlay) on page 532. Create the worker load balancer subnet in the same compartment where you created the VCN.

Create either a private or a public worker load balancer subnet. Create a public worker load balancer subnet to use with a public cluster. Create a private worker load balancer subnet to expose applications in a private cluster.

Create a Public Worker Load Balancer Subnet

Compute Cloud@Customer Console property	CLI property
• Name: service-lb • CIDR Block: *workerlb_cidr* • Route Table: Select "public" from the list • Public Subnet: check the box • DNS Hostnames: Use DNS Hostnames in this Subnet: check the box • DNS Label: servicelb • Security Lists: Select "workerlb-seclist" and "Default Security List for oketest-vcn" from the list	• `--vcn-id`: `ocid1.vcn.`*oke_vcn_id* • `--display-name`: `service-lb` • `--cidr-block`: *workerlb_cidr* • `--dns-label`: `servicelb` • `--prohibit-public-ip-on-vnic`: `false` • `--route-table-id`: OCID of the "public" route table • `--security-list-ids`: OCIDs of the "workerlb-seclist" security list and the "Default Security List for oketest-vcn" security list

The difference in the following private subnet is the VCN private route table is used instead of the public route table.

Create a Private Worker Load Balancer Subnet

Compute Cloud@Customer Console property	CLI property
• Name: service-lb • CIDR Block: *workerlb_cidr* • Route Table: Select "vcn_private" from the list • Private Subnet: check the box • DNS Hostnames: Use DNS Hostnames in this Subnet: check the box • DNS Label: servicelb • Security Lists: Select "workerlb-seclist" and "Default Security List for oketest-vcn" from the list	• `--vcn-id`: `ocid1.vcn.`*oke_vcn_id* • `--display-name`: `service-lb` • `--cidr-block`: *workerlb_cidr* • `--dns-label`: `servicelb` • `--prohibit-public-ip-on-vnic`: `true` • `--route-table-id`: OCID of the "vcn_private" route table • `--security-list-ids`: OCIDs of the "workerlb-seclist" security list and the "Default Security List for oketest-vcn" security list

What's Next:

Creating an OKE Control Plane Subnet (Flannel Overlay) on page 542

Creating an OKE Control Plane Subnet (Flannel Overlay)

Learn how to create a control plane subnet for Flannel Overlay networking On Compute Cloud@Customer.

Create the following resources in the order listed:

1. Create a Control Plane Security List on page 542
2. Create the Control Plane Subnet on page 544

Create a Control Plane Security List

To create a security list, use the instructions in Creating a Security List on page 159. For Terraform input, see Example Terraform Scripts for Network Resources (Flannel Overlay) on page 523.

For this example, use the following input for the control plane subnet security list. Use this configuration as a guide when you create these resources. You can change the values of properties such as CIDR blocks and IP addresses. You should not change the values of properties such as the network protocol, the stateful setting, or the private/public setting.

Compute Cloud@Customer Console property	**CLI property**
• Name: kmi-seclist	• `--vcn-id`: `ocid1.vcn.`*`oke_vcn_id`* • `--display-name`: `kmi-seclist`
Six ingress security rules:	**Six ingress security rules:** `--ingress-security-rules`

Ingress Rule 1
• Stateless: clear the box
• Ingress CIDR: *`kube_client_cidr`*
• IP Protocol: TCP
  • Destination Port Range: *`kubernetes_api_port`*
• Description: "Allow inbound connections to the Kubernetes API server."
	Ingress Rule 1
	• `isStateless`: `false`
	• `source`: *`kube_client_cidr`*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: *`kubernetes_api_port`*
	  • `min`: *`kubernetes_api_port`*
	• `description`: "Allow inbound connections to the Kubernetes API server."

Ingress Rule 2
• Stateless: clear the box
• Ingress CIDR: *`kmilb_cidr`*
• IP Protocol: TCP
  • Destination Port Range: *`kubernetes_api_port`*
• Description: "Allow inbound connections from the control plane load balancer."
	Ingress Rule 2
	• `isStateless`: `false`
	• `source`: *`kmilb_cidr`*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: *`kubernetes_api_port`*
	  • `min`: *`kubernetes_api_port`*
	• `description`: "Allow inbound connections from the control plane load balancer."

Ingress Rule 3
• Stateless: clear the box
• Ingress CIDR: *`worker_cidr`*
• IP Protocol: TCP
  • Destination Port Range: 1024-65535
• Description: "Allow inbound connections from worker nodes to the control plane."
	Ingress Rule 3
	• `isStateless`: `false`
	• `source`: *`worker_cidr`*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `65535`
	  • `min`: `1024`
	• `description`: "Allow inbound connections from worker nodes to the control plane."

Compute Cloud@Customer Console property	CLI property

Ingress Rule 4
• Stateless: clear the box
• Ingress CIDR: *kmi_cidr*
• IP Protocol: TCP
  • Destination Port Range: 1024-65535
• Description: "Allow inbound connections within the control plane."
	Ingress Rule 4
	• `isStateless`: `false`
	• `source`: *kmi_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `65535`
	  • `min`: `1024`
	• `description`: "Allow inbound connections within the control plane."

Ingress Rule 5
• Stateless: clear the box
• Ingress CIDR: *worker_cidr*
• IP Protocol: UDP
  • Destination Port Range: 8285-8472
• Description: "Allow flannel traffic."
	Ingress Rule 5
	• `isStateless`: `false`
	• `source`: *worker_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `17`
	• `udpOptions`
	`destinationPortRange`
	  • `max`: `8472`
	  • `min`: `8285`
	• `description`: "Allow flannel traffic."

Ingress Rule 6
• Stateless: clear the box
• Ingress CIDR: *kmi_cidr*
• IP Protocol: UDP
  • Destination Port Range: 8285-8472
• Description: "Allow flannel traffic."
	Ingress Rule 6
	• `isStateless`: `false`
	• `source`: *kmi_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `17`
	• `udpOptions`
	`destinationPortRange`
	  • `max`: `8472`
	  • `min`: `8285`
	• `description`: "Allow flannel traffic."

Create the Control Plane Subnet

To create a subnet, use the instructions in Creating a Subnet on page 142. For Terraform input, see Example Terraform Scripts for Network Resources (Flannel Overlay) on page 523.

Use the following input to create the control plane subnet. Use the OCID of the VCN that was created in Creating a VCN (Flannel Overlay) on page 532. Create the control plane subnet in the same compartment where you created the VCN.

Create either a NAT private control plane subnet or a VCN private control plane subnet. Create a NAT private control plane subnet to communicate outside the VCN.

> **Important:**
>
> The name of this subnet must be exactly `control-plane`.

Create a NAT Private Control Plane Subnet

Compute Cloud@Customer Console property	**CLI property**
• Name: control-plane • CIDR Block: *kmi_cidr* • Route Table: Select "nat_private" from the list • Private Subnet: check the box • DNS Hostnames: Use DNS Hostnames in this Subnet: check the box • DNS Label: kmi • Security Lists: Select "kmi-seclist" and "Default Security List for oketest-vcn" from the list	• `--vcn-id`: `ocid1.vcn.`*oke_vcn_id* • `--display-name`: `control-plane` • `--cidr-block`: *kmi_cidr* • `--dns-label`: `kmi` • `--prohibit-public-ip-on-vnic`: `true` • `--route-table-id`: OCID of the "nat_private" route table • `--security-list-ids`: OCIDs of the "kmi-seclist" security list and the "Default Security List for oketest-vcn" security list

Create a VCN Private Control Plane Subnet

Compute Cloud@Customer Console property	**CLI property**
• Name: control-plane • CIDR Block: *kmi_cidr* • Route Table: Select "vcn_private" from the list • Private Subnet: check the box • DNS Hostnames: Use DNS Hostnames in this Subnet: check the box • DNS Label: kmi • Security Lists: Select "kmi-seclist" and "Default Security List for oketest-vcn" from the list	• `--vcn-id`: `ocid1.vcn.`*oke_vcn_id* • `--display-name`: `control-plane` • `--cidr-block`: *kmi_cidr* • `--dns-label`: `kmi` • `--prohibit-public-ip-on-vnic`: `true` • `--route-table-id`: OCID of the "vcn_private" route table • `--security-list-ids`: OCIDs of the "kmi-seclist" security list and the "Default Security List for oketest-vcn" security list

What's Next:

Creating an OKE Control Plane Subnet (Flannel Overlay) on page 542

Creating an OKE Control Plane Load Balancer Subnet (Flannel Overlay)

Learn how to create a control plane load balancer subnet for Flannel Overlay networking on Compute Cloud@Customer.

Create the following resources in the order listed:

1. Create a Control Plane Load Balancer Security List on page 545.
2. Create the Control Plane Load Balancer Subnet on page 547.

Create a Control Plane Load Balancer Security List

To create a security list, use the instructions in Creating a Security List on page 159. For Terraform input, see Example Terraform Scripts for Network Resources (Flannel Overlay) on page 523.

The control plane load balancer accepts traffic on port 6443, which is also called *kubernetes_api_port* in these instructions. Adjust this security list to only accept connections from where you expect the network to run. Port 6443 must accept connections from the cluster control plane instances and worker instances.

For this example, use the following input for the control plane load balancer subnet security list.

Compute Cloud@Customer Console property	**CLI property**
• Name: kmilb-seclist	• `--vcn-id`: `ocid1.vcn.`*`oke_vcn_id`* • `--display-name`: kmilb-seclist
Four ingress security rules:	**Four ingress security rules:** `--ingress-security-rules`

Ingress Rule 1:
• Stateless: clear the box
• Ingress CIDR:
`kube_internal_cidr`
This value is required. Do not change this CIDR value.
• IP Protocol: TCP
  • Destination Port Range: *`kubernetes_api_port`*
• Description: "Allow inbound connections to the control plane load balancer."
	Ingress Rule 1:
	• `isStateless`: `false`
	• `source`:
	`kube_internal_cidr`
	This value is required. Do not change this CIDR value.
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: *`kubernetes_api_port`*
	  • `min`: *`kubernetes_api_port`*
	• `description`: "Allow inbound connections to the control plane load balancer."

Ingress Rule 2:
• Stateless: clear the box
• Ingress CIDR: *`kube_client_cidr`*
• IP Protocol: TCP
  • Destination Port Range: *`kubernetes_api_port`*
• Description: "Allow inbound connections to the control plane load balancer."
	Ingress Rule 2:
	• `isStateless`: `false`
	• `source`: *`kube_client_cidr`*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: *`kubernetes_api_port`*
	  • `min`: *`kubernetes_api_port`*
	• `description`: "Allow inbound connections to the control plane load balancer."

Ingress Rule 3:
• Stateless: clear the box
• Ingress CIDR: *`vcn_cidr`*
• IP Protocol: TCP
  • Destination Port Range: *`kubernetes_api_port`*
• Description: "Allow inbound connections to the control plane load balancer."
	Ingress Rule 3:
	• `isStateless`: `false`
	• `source`: *`vcn_cidr`*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: *`kubernetes_api_port`*
	  • `min`: *`kubernetes_api_port`*
	• `description`: "Allow inbound connections to the control plane load balancer."

Compute Cloud@Customer Console property	**CLI property**

Ingress Rule 4: Public Endpoint
• Stateless: clear the box
• Ingress CIDR: *public_ip_cidr*
• IP Protocol: TCP
  • Destination Port Range: *kubernetes_api_port*
• Description: "Used to access the control plane endpoint from the public CIDR. If you do not know what your public IP CIDR is, open a Support Request. See Creating a Support Request. To access support, sign in to the Oracle Cloud Console as described in Sign In to the OCI Console."
	Ingress Rule 4: Public Endpoint
	• `isStateless`: `false`
	• `source`: *public_ip_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: *kubernetes_api_port*
	  • `min`: *kubernetes_api_port*
	• `description`: "Used to access the control plane endpoint from the public CIDR. If you do not know what your public IP CIDR is, open a Support Request. See Creating a Support Request. To access support, sign in to the Oracle Cloud Console as described in Sign In to the OCI Console."

Create the Control Plane Load Balancer Subnet

To create a subnet, use the instructions in Creating a Subnet on page 142. For Terraform input, see Example Terraform Scripts for Network Resources (Flannel Overlay) on page 523.

For this example, use the following input to create the control plane load balancer subnet. Use the OCID of the VCN that was created in Creating a VCN (Flannel Overlay) on page 532. Create the control plane load balancer subnet in the same compartment where you created the VCN.

Create either a private or a public control plane load balancer subnet. Create a public control plane load balancer subnet to use with a public cluster. Create a private control plane load balancer subnet to use with a private cluster.

See Private Clusters on page 518 for information about using Local Peering Gateways to connect a private cluster to other instances on the Compute Cloud@Customer and using Dynamic Routing Gateways to connect a private cluster to the on-premises IP address space. To create a private control plane load balancer subnet, specify one of the following route tables (see Creating a VCN (Flannel Overlay) on page 532):

- vcn_private
- lpg_rt
- drg_rt

Create a Public Control Plane Load Balancer Subnet

Compute Cloud@Customer Console property	CLI property
• Name: control-plane-endpoint • CIDR Block: *kmilb_cidr* • Route Table: Select "public" from the list • Public Subnet: check the box • DNS Hostnames: Use DNS Hostnames in this Subnet: check the box • DNS Label: kmilb • Security Lists: Select "kmilb-seclist" and "Default Security List for oketest-vcn" from the list	• `--vcn-id`: `ocid1.vcn.`*oke_vcn_id* • `--display-name`: `control-plane-endpoint` • `--cidr-block`: *kmilb_cidr* • `--dns-label`: `kmilb` • `--prohibit-public-ip-on-vnic`: `false` • `--route-table-id`: OCID of the "public" route table • `--security-list-ids`: OCIDs of the "kmilb-seclist" security list and the "Default Security List for oketest-vcn" security list

The difference in the following private subnet is the VCN private route table is used instead of the public route table. Depending on your needs, you could specify the LPG route table or the DRG route table instead.

Create a Private Control Plane Load Balancer Subnet

Compute Cloud@Customer Console property	CLI property
• Name: control-plane-endpoint • CIDR Block: *kmilb_cidr* • Route Table: Select "vcn_private" from the list • Private Subnet: check the box • DNS Hostnames: Use DNS Hostnames in this Subnet: check the box • DNS Label: kmilb • Security Lists: Select "kmilb-seclist" and "Default Security List for oketest-vcn" from the list	• `--vcn-id`: `ocid1.vcn.`*oke_vcn_id* • `--display-name`: `control-plane-endpoint` • `--cidr-block`: *kmilb_cidr* • `--dns-label`: `kmilb` • `--prohibit-public-ip-on-vnic`: `true` • `--route-table-id`: OCID of the "vcn_private" route table • `--security-list-ids`: OCIDs of the "kmilb-seclist" security list and the "Default Security List for oketest-vcn" security list

What's Next:

Creating an OKE Cluster on page 589

Creating VCN-Native Pod Network Resources

On Compute Cloud@Customer, VCN-Native Pod Networking enables you to directly manage the traffic from pods. VCN-Native Pod Networking offers more flexibility and control over the traffic and allows you to use different security rules.

With VCN-Native Pod Networking, pod IP addresses come directly from the VCN CIDR block and not from a network overlay such as Flannel Overlay. VCN-Native Pod Networking connects nodes in a Kubernetes cluster to pod subnets in the OKE VCN. Pod IP addresses within the OKE VCN are directly routable from other VCNs that are connected (peered) to the OKE VCN, and from on-premises networks.

When you create a cluster that uses VCN-Native Pod Networking, the VCN that you specify must have a subnet named `pod`. You must provide a subnet named `pod` so that the system can find that subnet. The pod subnet has security rules that enable pods on control plane nodes to communicate directly with pods on worker nodes and with other pods and other resources. See Creating a Pod Subnet (VCN-Native Pod) on page 568. If you select VCN-Native Pod Networking and do not have a subnet named `pod`, the cluster creation will fail.

When you create a node pool for a cluster that is using VCN-Native Pod Networking, the pod subnet that you specify (Pod Communication > Pod Communication Subnet or `--pod-subnet-ids`) serves the function of a pod subnet for pods on worker nodes. That pod subnet should have security rules that enable pods on worker nodes to communicate directly with other pods on worker nodes and control plane nodes. You can optionally specify the worker node subnet as the pod subnet. The CIDR of the pod subnet that you specify must be larger than /25. The pod subnet should be larger than the worker node subnet.

In general, when you use VCN-Native Pod Networking, security rules can enable pods to communicate directly with other pods on the same node or on other nodes in the cluster, with other clusters, with other services, and with the internet.

Node Shapes and Number of Pods

When using the OCI VCN-Native Pod Networking CNI plugin, each pod needs a private IP address. By default, 31 IP addresses are assigned to a VNIC for use by pods running on the worker node.

You can specify the maximum number of pods that you want to run on a worker node. The default maximum is 31 pods per worker node. You can specify up to 110.

A node shape, and therefore a worker node, has a minimum of two VNICs. The first VNIC is connected to the worker subnet. The second VNIC is connected to the pod subnet. Therefore a worker node can support at least 31 pods. If you want more than 31 pods on a single worker node, specify a shape for the node pool that supports three or more VNICs: one VNIC to connect to the worker node subnet, and at least two VNICs to connect to the pod subnet.

A VM.PCAStandard1.4 standard node shape can have a maximum of four VNICs, and the worker node can support up to 93 pods. A VM.PCAStandard.E5.Flex node shape with five OCPUs can have a maximum of five VNICs, and the worker node can support up to 110 pods. A node cannot have more than 110 pods.

The following formula summarizes the maximum number of pods supported per node:

```
MIN( (Number of VNICs - 1) * 31 ), 110)
```

For information about all node shapes, see Compute Shapes on page 301.

VCN-Native Pod Networking Resources

The resource definitions in the following sections in this topic create a working example set of network resources for workload clusters when you are using VCN-Native Pod Networking. Use this configuration as a guide when you create these resources. You can change the values of properties such as CIDR blocks and IP addresses. Don't change the values of properties such as the network protocol, the stateful setting, or the private/public setting.

See Workload Cluster Network Ports (VCN-Native Pod) on page 551 for specific ports that must be open for specific purposes.

Create the following network resources. To use Terraform, see Example Terraform Scripts (VCN-Native Pod) on page 552.

> **Note:**
>
> Create all network resources in the same compartment.

- VCN. See Creating a VCN (VCN-Native Pod Networking) on page 563.
- Internet Gateway
- NAT Gateway
- Dynamic Routing Gateway
- Local Peering Gateway
- Route rules
- Security lists
- The following five subnets:
  - Pod. See Creating a Pod Subnet (VCN-Native Pod) on page 568.

- Worker. See Creating a Worker Subnet (VCN-Native Pod) on page 571.
- Worker load balancer. See Creating a Worker Load Balancer Subnet (VCN-Native Pod) on page 577.
- Control plane. See Creating a Control Plane Subnet (VCN-Native Pod) on page 579.
- Control plane load balancer. See Creating a Control Plane Load Balancer Subnet (VCN-Native Pod) on page 584.

Workload Cluster Network CIDR Ranges (VCN-Native Pod)

Review the example CIDR ranges used for OKE network resources for VCN-Native Pod Networking on Compute Cloud@Customer.

Throughout this documentation, variables are used to represent CIDR ranges for instances in different subnets. The following table lists the CIDR variables and example values for use with VCN-Native Pod Networking.

> **Note:**
>
> These are examples only. The CIDR ranges you use depend on the number of clusters you have, the number of nodes in each cluster, the shape you select for the worker nodes, and the type of networking you're using.

For VCN-Native Pod Networking, every pod gets an IP address assigned from the IP address pool that's defined in the pod subnet CIDR. The shape you specify for the node pool determines the maximum number of VNICs (pods) for each worker node, as described in Node Shapes and Number of Pods on page 549.

The primary difference between IP address requirements of VCN-Native Pod Networking and Flannel Overlay networking is that VCN-Native Pod Networking requires more IP addresses to be available. The table in Workload Cluster Network CIDR Ranges (Flannel Overlay) on page 521 shows smaller CIDR ranges than the following table for VCN-Native Pod Networking CIDR ranges.

Example CIDR Values to Use with VCN-Native Pod Networking

Variable Name	Description	Example Value
vcn_cidr	VCN CIDR range This Is a small VCN with 8192 IP's for creating OKE infrastructure.	172.31.0.0/19
worker_cidr	Worker subnet CIDR	172.31.8.0/21
workerlb_cidr	Worker load balancer subnet CIDR	172.31.0.0/23
kmi_cidr	OKE control plane subnet CIDR	172.31.4.0/22
kmilb_cidr	OKE control plane load balancer subnet CIDR	172.31.2.0/23
pod_cidr	Pod subnet CIDR	172.31.16.0/20
kube_client_cidr	CIDR for clients that are allowed to contact the Kubernetes API server	10.0.0.0/8
public_ip_cidr	Public IP CIDR	10.0.0.0/8
kube_internal_cidr	CIDR used by the Kubernetes infrastructure to allocate IP addresses for various internal services and components	253.255.0.0/16

The IP Subnet Calculator on Calculator.net is one tool for finding all available networks for a given IP address and prefix length.

Workload Cluster Network Ports (VCN-Native Pod)

Review the ports that are required for OKE for VCN-Native Pod Networking On Compute Cloud@Customer. These ports must be available to configure workload cluster networking. You might need to open more ports for other purposes.

All protocols are TCP. All port states are Stateful. Port 6443 is the port used for Kubernetes API and is also known as ***`kubernetes_api_port`*** in this guide.

See also the tables in Port Matrix.

Ports that Must Be Available for Use by Workload Clusters for VCN-Native Pod Networking

Source IP Address	Destination IP Address	Port	Description
bastion host: *`vcn_cidr`*	Worker nodes subnet: *`worker_cidr`*	22	Outbound connections from the bastion host to the worker CIDR.
bastion host: *`vcn_cidr`*	Control plane subnet: *`kmi_cidr`*	22	Outbound connections from the bastion host to the control plane nodes.
Worker nodes subnet: *`worker_cidr`*	yum repository	80	Outbound connections from the worker CIDR to external applications.
Worker nodes subnet: *`worker_cidr`*	Control plane subnet: *`kmi_cidr`*	6443	Outbound connections from the worker CIDR to the Kubernetes API. This is necessary to allow nodes to join through either a public IP address on one of the nodes or the load balancer public IP address. Port 6443 is called the ***`kubernetes_api_port`***.
Worker nodes subnet: *`worker_cidr`*	Control plane load balancer	6443	Inbound connections from the worker CIDR to the Kubernetes API.
CIDR for clients: *`kube_client_cidr`*	Control plane load balancer	6443	Inbound connections from clients to the Kubernetes API server.
Worker nodes subnet: *`worker_cidr`*	Control plane subnet: *`kmi_cidr`*	6443	Private outbound connections from the worker CIDR to `kubeapi` on the control plane subnet.
`kube_client_cidr`	Worker nodes subnet: *`worker_cidr`*	30000-32767	Inbound traffic for applications from Kubernetes clients.
`kmi_cidr`	*`worker_cidr`*, *`pod_cidr`*	10250	Kubernetes API endpoint to worker node communication.
`kmi_cidr`	*`worker_cidr`*, *`pod_cidr`*	10256	Allow load balancer or network load balancer to communicate with `kube-proxy` on worker nodes or pod subnet.
`pod_cidr`	*`kmilb_cidr`*	12250	Pod to Kubernetes API endpoint communication.
`kmi_cidr`	*`kmi_cidr`*	2379-2381	Communication between the `etcd` server and metrics services.
`kmi_cidr`	*`kmi_cidr`*	10257-10260	Inbound connection for Kubernetes components.

Example Terraform Scripts (VCN-Native Pod)

On Compute Cloud@Customer, you can use Terraform scripts to automate the creation of OKE network resources for VCN-Native Pod Networking.

Most of the values shown in these scripts, such as resource display names and CIDRs, are examples. Some ports must be specified as shown (see Workload Cluster Network Ports (VCN-Native Pod) on page 551), the OKE pod subnet must be named `pod`, and the OKE control plane subnet must be named `control-plane`. See Workload Cluster Network CIDR Ranges (VCN-Native Pod) on page 550 for comments about CIDR values.

- variables.tf on page 552
- terraform.tfvars on page 553
- provider.tf on page 554
- main.tf on page 554
- oke_vcn.tf on page 557
- oke_pod_seclist.tf on page 558
- oke_pod_subnet.tf on page 559
- oke_worker_seclist.tf on page 559
- oke_worker_subnet.tf on page 560
- oke_kmi_seclist.tf on page 561
- oke_kmi_subnet.tf on page 562

variables.tf

This file creates several variables that are used to configure OKE network resources when you're using VCN-Native Pod Networking. Many of these variables aren't assigned values in this file. One port and five CIDRs are assigned values. The `kubernetes_api_port`, port 6443, is the port used to access the Kubernetes API. See also Workload Cluster Network Ports (VCN-Native Pod) on page 551. The six CIDRs that are defined in this file are for the OKE VCN, pod subnet, worker subnet, worker load balancer subnet, control plane subnet, and control plane load balancer subnet.

```
variable "oci_config_file_profile" {
  type    = string
  default = "DEFAULT"
}

variable "tenancy_ocid" {
  description = "tenancy OCID"
  type        = string
  nullable    = false
}

variable "compartment_id" {
  description = "compartment OCID"
  type        = string
  nullable    = false
}

variable "vcn_name" {
  description = "VCN name"
  nullable    = false
}

variable "kube_client_cidr" {
  description = "CIDR of Kubernetes API clients"
  type        = string
  nullable    = false
}

variable "kubernetes_api_port" {
  description = "Port used for Kubernetes API"
```

```
  type        = string
  default     = "6443"
}

# IP network addressing
variable "vcn_cidr" {
  default = "172.31.0.0/19"
}

# Subnet for KMIs where kube-apiserver and other control
# plane applications run, max 9 nodes
variable "kmi_cidr" {
  description = "Kubernetes control plane subnet CIDR"
  default     = "172.31.4.0/22"
}

# Subnet for KMI load balancer
variable "kmilb_cidr" {
  description = "Kubernetes control plane LB subnet CIDR"
  default     = "172.31.2.0/23"
}

#Subnet CIDR configured for VCN public IP for NAT in Network
variable "public_ip_cidr" {
  description = "Public IP CIDR"
  type        = string
  nullable    = false
}

# Subnet for worker nodes, max 128 nodes
variable "worker_cidr" {
  description = "Kubernetes worker subnet CIDR"
  default     = "172.31.8.0/21"
}

# Subnet for worker load balancer (for use by CCM)
variable "workerlb_cidr" {
  description = "Kubernetes worker LB subnet CIDR"
  default     = "172.31.0.0/23"
}

# Subnet for pod communication
variable "pod_cidr" {
  description = "Kubernetes pod communication subnet CIDR"
  default     = "172.31.16.0/20"
}

# Flag to Enable private endpoint
variable "enable_private_endpoint" {
  description = "Flag to create private control plane endpoint/service-lb"
  type = bool
  default = false
  nullable = false
}
```

terraform.tfvars

This file assigns values to some of the variables that were created in `variables.tf`.

```
# name of the profile to use from $HOME/.oci/config
oci_config_file_profile = "DEFAULT"

# tenancy ocid from the above profile
```

```
tenancy_ocid = "tenancy_OCID"

# compartment in which to build the OKE cluster
compartment_id = "compartment_OCID"

# display-name for the OKE VCN
vcn_name = "oketest"
```

provider.tf

This file is required in order to use the OCI provider. The file initializes the OCI module using the OCI profile configuration file.

```
provider "oci" {
  config_file_profile = var.oci_config_file_profile
  tenancy_ocid        = var.tenancy_ocid
}
```

main.tf

This file specifies the provider to use (`oracle/oci`), defines several security list rules, and initializes required local variables.

The version of the OCI provider that you use must be at least v4.50.0 but no greater than v6.36.0.

```
terraform {
  required_providers {
    oci = {
      source  = "oracle/oci"
      version = ">= 4.50.0, <= 6.36.0"
      # If necessary, you can pin a specific version here
      #version = "4.71.0"
    }
  }
  required_version = ">= 1.1"
}

locals {
  kube_internal_cidr      = "253.255.0.0/16"
  worker_lb_ingress_rules = [
    {
      source   = var.kube_client_cidr
      port_min = 80
      port_max = 80
    },
    {
      source   = var.kube_client_cidr
      port_min = 443
      port_max = 443
    }
  ]
  worker_ingress_rules = [
    {
      source   = var.kube_client_cidr
      port_min = 30000
      port_max = 32767
    },
    {
      source   = var.kmi_cidr
      port_min = 22
      port_max = 22
    },
    {
```

```
      source   = var.worker_cidr
      port_min = 22
      port_max = 22
    },
    {
      source   = var.worker_cidr
      port_min = 10250
      port_max = 10250
    },
    {
      source   = var.worker_cidr
      port_min = 10256
      port_max = 10256
    },
    {
      source   = var.worker_cidr
      port_min = 30000
      port_max = 32767
    },
    {
      source   = var.workerlb_cidr
      port_min = 10256
      port_max = 10256
    },
    {
      source   = var.workerlb_cidr
      port_min = 30000
      port_max = 32767
    },
    {
      source   = var.kmi_cidr
      port_min = 10250
      port_max = 10250
    },
    {
      source   = var.kmi_cidr
      port_min = 10256
      port_max = 10256
    },
    {
      source   = var.pod_cidr
      port_min = 30000
      port_max = 32767
    },
  ]
  kmi_lb_ingress_rules = [
    {
      source   = local.kube_internal_cidr
      port_min = var.kubernetes_api_port
      port_max = var.kubernetes_api_port
    },
    {
      source   = var.kube_client_cidr
      port_min = var.kubernetes_api_port
      port_max = var.kubernetes_api_port
    },
    {
      source   = var.kmi_cidr
      port_min = var.kubernetes_api_port
      port_max = var.kubernetes_api_port
    },
    {
      source   = var.worker_cidr
      port_min = var.kubernetes_api_port
```

```
      port_max = var.kubernetes_api_port
    },
    {
      source   = var.worker_cidr
      port_min = 12250
      port_max = 12250
    },
    {
      source   = var.pod_cidr
      port_min = 12250
      port_max = 12250
    },
  ]
  kmi_ingress_rules = [
    {
      source   = var.kube_client_cidr
      port_min = var.kubernetes_api_port
      port_max = var.kubernetes_api_port
    },
    {
      source   = var.kmilb_cidr
      port_min = var.kubernetes_api_port
      port_max = var.kubernetes_api_port
    },
    {
      source   = var.kmilb_cidr
      port_min = 12250
      port_max = 12250
    },
    {
      source   = var.worker_cidr
      port_min = var.kubernetes_api_port
      port_max = var.kubernetes_api_port
    },
    {
      source   = var.worker_cidr
      port_min = 12250
      port_max = 12250
    },
    {
      source   = var.kmi_cidr
      port_min = var.kubernetes_api_port
      port_max = var.kubernetes_api_port
    },
    {
      source   = var.kmi_cidr
      port_min = 2379
      port_max = 2381
    },
    {
      source   = var.kmi_cidr
      port_min = 10250
      port_max = 10250
    },
    {
      source   = var.kmi_cidr
      port_min = 10257
      port_max = 10260
    },
    {
      source   = var.pod_cidr
      port_min = var.kubernetes_api_port
      port_max = var.kubernetes_api_port
    },
```

```
    {
      source   = var.pod_cidr
      port_min = 12250
      port_max = 12250
    },
  ]
  pod_ingress_rules = [
    {
      source   = var.vcn_cidr
      port_min = 22
      port_max = 22
    },
    {
      source   = var.workerlb_cidr
      port_min = 10256
      port_max = 10256
    },
    {
      source   = var.worker_cidr
      port_min = 10250
      port_max = 10250
    },
    {
      source   = var.worker_cidr
      port_min = 10256
      port_max = 10256
    },
    {
      source   = var.worker_cidr
      port_min = 80
      port_max = 80
    },
  ]
}
```

oke_vcn.tf

This file defines a VCN, NAT gateway, internet gateway, private route table, and public route table. The private route table is the default route table for the VCN.

```
resource "oci_core_vcn" "oke_vcn" {
  cidr_block     = var.vcn_cidr
  dns_label      = var.vcn_name
  compartment_id = var.compartment_id
  display_name   = "${var.vcn_name}-vcn"
}

resource "oci_core_nat_gateway" "vcn_ngs" {
  compartment_id = var.compartment_id
  vcn_id         = oci_core_vcn.oke_vcn.id
  count          = var.enable_private_endpoint ? 0:1

  display_name = "VCN nat g6s"
}

resource "oci_core_internet_gateway" "vcn_igs" {
  compartment_id = var.compartment_id
  vcn_id         = oci_core_vcn.oke_vcn.id
  count          = var.enable_private_endpoint ? 0:1

  display_name = "VCN i6t g6s"
  enabled      = true
```

```
}

resource "oci_core_default_route_table" "default_private" {
  manage_default_resource_id = oci_core_vcn.oke_vcn.default_route_table_id
  display_name   = "Default - private"
  count          = var.enable_private_endpoint ? 1:0
}

resource "oci_core_default_route_table" "private" {
  count          = var.enable_private_endpoint ? 0:1
  manage_default_resource_id = oci_core_vcn.oke_vcn.default_route_table_id
  display_name   = "Default - private"

  route_rules {
    destination       = "0.0.0.0/0"
    destination_type  = "CIDR_BLOCK"
    network_entity_id = oci_core_nat_gateway.vcn_ngs[0].id
  }
}

resource "oci_core_route_table" "public" {
  count          = var.enable_private_endpoint ? 0:1
  compartment_id = var.compartment_id
  vcn_id         = oci_core_vcn.oke_vcn.id

  display_name = "public"
  route_rules {
    destination       = "0.0.0.0/0"
    destination_type  = "CIDR_BLOCK"
    network_entity_id = oci_core_internet_gateway.vcn_igs[0].id
  }
}
```

oke_pod_seclist.tf

This file defines the security list for the pod subnet. The rules for this security list were defined in other Terraform files in this set.

```
resource "oci_core_security_list" "pod" {
  compartment_id = var.compartment_id
  vcn_id         = oci_core_vcn.oke_vcn.id

  display_name = "${var.vcn_name}-pod"

  dynamic "ingress_security_rules" {
    iterator = port
    for_each = local.pod_ingress_rules

    content {
      source      = port.value.source
      source_type = "CIDR_BLOCK"
      protocol    = "6"
      tcp_options {
        min = port.value.port_min
        max = port.value.port_max
      }
    }
  }

   dynamic "ingress_security_rules" {
    iterator = icmp_type
    for_each = [0, 8]
```

```
    content {
      # ping from VCN; unreachable/TTL from anywhere
      source      = var.kmi_cidr
      source_type = "CIDR_BLOCK"
      protocol    = "1"
      icmp_options {
        type = icmp_type.value
      }
    }
  }

  dynamic "ingress_security_rules" {
    for_each = var.pod_cidr != null ? [var.pod_cidr] : []

    content {
      source      = ingress_security_rules.value
      source_type = "CIDR_BLOCK"
      protocol    = "all"
    }
  }
}
```

oke_pod_subnet.tf

This file defines the pod subnet.

> **Important:**
>
> The name of the `pod` subnet must be exactly `pod`.

```
resource "oci_core_subnet" "pod" {
  cidr_block     = var.pod_cidr
  compartment_id = var.compartment_id
  vcn_id         = oci_core_vcn.oke_vcn.id

  display_name               = "pod"
  dns_label                  = "pod"
  prohibit_public_ip_on_vnic = true

  security_list_ids = [
    oci_core_default_security_list.oke_vcn.id,
    oci_core_security_list.pod.id
  ]
}
```

oke_worker_seclist.tf

This file defines the security lists for both the worker subnet and the worker load balancer subnet. The rules for these security lists were defined in other Terraform files in this set.

```
resource "oci_core_security_list" "workerlb" {
  display_name   = "${var.vcn_name}-workerlb"
  compartment_id = var.compartment_id
  vcn_id         = oci_core_vcn.oke_vcn.id

  dynamic "ingress_security_rules" {
    iterator = port
    for_each = local.worker_lb_ingress_rules

    content {
      source      = port.value.source
      source_type = "CIDR_BLOCK"
      protocol    = "6"
```

```
      tcp_options {
        min = port.value.port_min
        max = port.value.port_max
      }
    }
  }
}

resource "oci_core_security_list" "worker" {
  display_name   = "${var.vcn_name}-worker"
  compartment_id = var.compartment_id
  vcn_id         = oci_core_vcn.oke_vcn.id

  dynamic "ingress_security_rules" {
    iterator = port
    for_each = local.worker_ingress_rules

    content {
      source      = port.value.source
      source_type = "CIDR_BLOCK"
      protocol    = "6"
      tcp_options {
        min = port.value.port_min
        max = port.value.port_max
      }
    }
  }

  dynamic "ingress_security_rules" {
    iterator = icmp_type
    for_each = [0, 8]

    content {
      # ping from VCN; unreachable/TTL from anywhere
      source      = var.kmi_cidr
      source_type = "CIDR_BLOCK"
      protocol    = "1"
      icmp_options {
        type = icmp_type.value
      }
    }
  }
}
```

oke_worker_subnet.tf

This file defines the worker and worker load balancer subnets. The worker load balancer subnet is named `service-lb`.

```
resource "oci_core_subnet" "worker" {
  cidr_block     = var.worker_cidr
  compartment_id = var.compartment_id
  vcn_id         = oci_core_vcn.oke_vcn.id

  display_name               = "worker"
  dns_label                  = "worker"
  prohibit_public_ip_on_vnic = true

  security_list_ids = [
    oci_core_default_security_list.oke_vcn.id,
    oci_core_security_list.worker.id
  ]
```

```
}

resource "oci_core_subnet" "worker_lb" {
  cidr_block     = var.workerlb_cidr
  compartment_id = var.compartment_id
  vcn_id         = oci_core_vcn.oke_vcn.id

  display_name               = "service-lb"
  dns_label                  = "servicelb"
  prohibit_public_ip_on_vnic = var.enable_private_endpoint
  route_table_id             = var.enable_private_endpoint==false ?
 oci_core_route_table.public[0].id :
 oci_core_vcn.oke_vcn.default_route_table_id

  security_list_ids = [
    oci_core_default_security_list.oke_vcn.id,
    oci_core_security_list.workerlb.id
  ]
}
```

oke_kmi_seclist.tf

This file defines the security lists for the control plane and control plane load balancer subnets. This file also defines updates to make to the default security list for the VCN.

```
resource "oci_core_default_security_list" "oke_vcn" {
  manage_default_resource_id = oci_core_vcn.oke_vcn.default_security_list_id

  egress_security_rules {
    destination      = "0.0.0.0/0"
    destination_type = "CIDR_BLOCK"
    protocol         = "all"
  }

  dynamic "ingress_security_rules" {
    iterator = icmp_type
    for_each = [3, 8, 11]

    content {
      # ping from VCN; unreachable/TTL from anywhere
      source      = (icmp_type.value == "8" ? var.vcn_cidr : "0.0.0.0/0")
      source_type = "CIDR_BLOCK"
      protocol    = "1"
      icmp_options {
        type = icmp_type.value
      }
    }
  }
}

resource "oci_core_security_list" "kmilb" {
  compartment_id = var.compartment_id
  vcn_id         = oci_core_vcn.oke_vcn.id

  display_name = "${var.vcn_name}-kmilb"

  dynamic "ingress_security_rules" {
    iterator = port
    for_each = local.kmi_lb_ingress_rules

    content {
      source      = port.value.source
```

```
      source_type = "CIDR_BLOCK"
      protocol    = "6"
      tcp_options {
        min = port.value.port_min
        max = port.value.port_max
      }
    }
  }

  dynamic "ingress_security_rules" {
    for_each = var.enable_private_endpoint ? [1] : []
    content {
      source   = var.kmilb_cidr
      source_type = "CIDR_BLOCK"
      protocol = "6"
      tcp_options {
        min = var.kubernetes_api_port
        max = var.kubernetes_api_port
      }
    }
  }

  dynamic "ingress_security_rules" {
    for_each = var.enable_private_endpoint ? [] : [0]
    content {
      source   = var.public_ip_cidr
      source_type = "CIDR_BLOCK"
      protocol = "6"
      tcp_options {
        min = var.kubernetes_api_port
        max = var.kubernetes_api_port
      }
    }
  }

  depends_on = []
}

resource "oci_core_security_list" "kmi" {
  compartment_id = var.compartment_id
  vcn_id         = oci_core_vcn.oke_vcn.id

  display_name = "${var.vcn_name}-kmi"

  dynamic "ingress_security_rules" {
    iterator = port
    for_each = local.kmi_ingress_rules

    content {
      source      = port.value.source
      source_type = "CIDR_BLOCK"
      protocol    = "6"
      tcp_options {
        min = port.value.port_min
        max = port.value.port_max
      }
    }
  }
}
```

oke_kmi_subnet.tf

This file defines the control plane and control plane load balancer subnets.

> **Important:**
>
> The name of the `kmi` subnet must be exactly `control-plane`.

```
resource "oci_core_subnet" "kmi" {
  cidr_block                 = var.kmi_cidr
  compartment_id             = var.compartment_id
  display_name               = "control-plane"
  dns_label                  = "kmi"
  vcn_id                     = oci_core_vcn.oke_vcn.id
  prohibit_public_ip_on_vnic = true
  security_list_ids = [
    oci_core_default_security_list.oke_vcn.id,
    oci_core_security_list.kmi.id
  ]
}

resource "oci_core_subnet" "kmi_lb" {
  cidr_block                 = var.kmilb_cidr
  compartment_id             = var.compartment_id
  dns_label                  = "kmilb"
  vcn_id                     = oci_core_vcn.oke_vcn.id
  display_name               = "control-plane-endpoint"
  prohibit_public_ip_on_vnic = var.enable_private_endpoint
  route_table_id             = var.enable_private_endpoint==false ?
 oci_core_route_table.public[0].id :
 oci_core_default_route_table.default_private[0].id
  security_list_ids = [
    oci_core_default_security_list.oke_vcn.id,
    oci_core_security_list.kmilb.id
  ]
}
```

Creating a VCN (VCN-Native Pod Networking)

Learn how to create a VCN for VCN-Native Pod Networking On Compute Cloud@Customer.

Create the following resources in the order listed:

1. VCN
2. Route rules
   - Public clusters:
     - Internet gateway and a route table with a route rule that references that internet gateway.
     - NAT gateway and a route table with a route rule that references that NAT gateway.
   - Private clusters:
     - Route table with no route rules.
     - (Optional) Dynamic Routing Gateway (DRG), attach the OKE VCN to that DRG, and create a route table with a route rule that references that DRG. See Private Clusters on page 518.
     - (Optional) Local Peering Gateway (LPG) and a route table with a route rule that references that LPG. See Private Clusters on page 518.
3. Security list. Modify the VCN default security list.

Resource names and CIDR blocks are example values.

VCN

To create the VCN, use the instructions inCreating a VCN on page 141. For Terraform input, see Example Terraform Scripts (VCN-Native Pod) on page 552.

For this example, use the following input to create the VCN. The VCN covers one contiguous CIDR block. The CIDR block can't be changed after the VCN is created.

Console property	CLI property
• Name: oketest-vcn • CIDR Block: *vcn_cidr* • DNS Label: oketest This label must be unique across all VCNs in the tenancy.	• `--display-name`: `oketest-vcn` • `--cidr-blocks`: `'["vcn_cidr"]'` • `--dns-label`: `oketest` This label must be unique across all VCNs in the tenancy.

Note the OCID of the new VCN. In the examples in this guide, this VCN OCID is `ocid1.vcn.oke_vcn_id`.

Next Steps

- Public internet access. For traffic on a public subnet that connects to the internet using public IP addresses, create an internet gateway and a route rule that references that internet gateway.
- Private internet access. For traffic on a private subnet that needs to connect to the internet without exposing private IP addresses, create a NAT gateway and a route rule that references that NAT gateway.
- VCN-only access. To restrict communication to only other resources on the same VCN, use the default route table, which has no route rules.
- Instances in another VCN. To enable communication between the cluster and an instance running on a different VCN, create a Local Peering Gateway (LPG) and a route rule that references that LPG.
- Data center IP address space. To enable communication between the cluster and the on-premises network IP address space, create a Dynamic Routing Gateway (DRG) and a route rule that references that DRG.

VCN Private Route Table

Edit the default route table that was created when you created the VCN. Change the name of the route table to vcn_private. This route table does not have any route rules. Do not add any route rules.

NAT Private Route Table

Create a NAT gateway and a route table with a route rule that references the NAT gateway.

NAT Gateway

To create the NAT gateway, use the instructions in Enabling Public Connections through a NAT Gateway on page 168. For Terraform input, see Example Terraform Scripts (VCN-Native Pod) on page 552.

Note the name and OCID of the NAT gateway for assignment to the private route rule.

Private Route Rule

Create a route table. See Creating a Route Table on page 155. For Terraform input, see Example Terraform Scripts for Network Resources (Flannel Overlay) on page 523.

For this example, use the following input to create the route table with a private route rule that references the NAT gateway that was created in the preceding step.

Console property	CLI property
• Name: nat_private	• `--display-name`: nat_private
Route rule	`--route-rules`
• Target Type: NAT Gateway • NAT Gateway: Name of the NAT gateway that was created in the preceding step • CIDR Block: 0.0.0.0/0 • Description: NAT private route rule	• `networkEntityId`: OCID of the NAT gateway that was created in the preceding step • `destinationType`: `CIDR_BLOCK` • `destination`: `0.0.0.0/0` • `description`: NAT private route rule

Note the name and OCID of this route table for assignment to private subnets.

Local Peering Gateway

Create a Local Peering gateway (LPG) and a route table with a route rule that references the LPG.

Local Peering Gateway

Create the LPG. See Connecting VCNs through a Local Peering Gateway (LPG) on page 172.

Note the name and OCID of the LPG for assignment to the private route rule.

Private Route Rule

Create a route table. See Creating a Route Table on page 155.

For this example, use the following input to create the route table with a private route rule that references the LPG that was created in the preceding step.

Console property	CLI property
• Name: lpg_rt	• `--display-name`: lpg_rt
Route rule	`--route-rules`
• Target Type: Local Peering Gateway • Local Peering Gateway: Name of the LPG that was created in the preceding step • CIDR Block: *`CIDR_for_the_second_VCN`* • Description: LPG private route rule	• `networkEntityId`: OCID of the LPG that was created in the preceding step • `destinationType`: `CIDR_BLOCK` • `destination`: *`CIDR_for_the_second_VCN`* • `description`: LPG private route rule

Note the name and OCID of this route table for assignment to the "control-plane-endpoint" subnet (Creating a Control Plane Load Balancer Subnet (VCN-Native Pod) on page 584).

Add the same route rule on the second VCN (the peered VCN), specifying the OKE VCN CIDR as the destination.

Dynamic Routing Gateway

Create a Dynamic Routing gateway (DRG) and a route table with a route rule that references the DRG.

Dynamic Routing Gateway

Create the DRG and attach the OKE VCN to that DRG. See Connecting to the On-Premises Network through a Dynamic Routing Gateway (DRG) on page 174. Create the DRG in the OKE VCN compartment, and then attach the OKE VCN to that DRG.

Note the name and OCID of the DRG for assignment to the private route rule.

Private Route Rule

Create a route table. See Creating a Route Table on page 155.

For this example, use the following input to create the route table with a private route rule that references the DRG that was created in the preceding step.

Console property	CLI property
• Name: drg_rt Route rule • Target Type: Dynamic Routing Gateway • Dynamic Routing: Name of the DRG that was created in the preceding step • CIDR Block: 0.0.0.0/0 • Description: DRG private route rule	• `--display-name`: drg_rt `--route-rules` • `networkEntityId`: OCID of the DRG that was created in the preceding step • `destinationType`: `CIDR_BLOCK` • `destination`: `0.0.0.0/0` • `description`: DRG private route rule

Note the name and OCID of this route table for assignment to the "control-plane-endpoint" subnet (Creating a Control Plane Load Balancer Subnet (VCN-Native Pod) on page 584).

Public Route Table

Create an Internet gateway and a route table with a route rule that references the Internet gateway.

Internet Gateway

To create the internet gateway, use the instructions in Configuring an Internet Gateway on page 170. For Terraform input, see Example Terraform Scripts (VCN-Native Pod) on page 552.

Note the name and OCID of the internet gateway for assignment to the public route rule.

Public Route Rule

To create a route table, use the instructions in Creating a Route Table on page 155. For Terraform input, see Example Terraform Scripts (VCN-Native Pod) on page 552.

For this example, use the following input to create the route table with a public route rule that references the internet gateway that was created in the preceding step.

Console property	CLI property
• Name: public Route rule • Target Type: Internet Gateway • Internet Gateway: Name of the internet gateway that was created in the preceding step • CIDR Block: 0.0.0.0/0 • Description: OKE public route rule	• `--vcn-id`: `ocid1.vcn.`***`oke_vcn_id`*** • `--display-name`: public `--route-rules` • `networkEntityId`: OCID of the internet gateway that was created in the preceding step • `destinationType`: `CIDR_BLOCK` • `destination`: `0.0.0.0/0` • `description`: OKE public route rule

Note the name and OCID of this route table for assignment to public subnets.

VCN Default Security List

Modify the default security list, using the input shown in the following table. Delete all of the default rules and create the rules shown in the following table.

To modify a security list, use the instructions in Updating a Security List on page 162. For Terraform input, see Example Terraform Scripts (VCN-Native Pod) on page 552.

Console property	CLI property
• Name: Default	`--security-list-id`: `ocid1.securitylist.`***`default_securitylist_id`***
One egress security rule: • Stateless: clear the box • Egress CIDR: 0.0.0.0/0 • IP Protocol: All protocols • Description: "Allow all outgoing traffic."	**One egress security rule:** `--egress-security-rules` • `isStateless`: `false` • `destination`: `0.0.0.0/0` • `destinationType`: `CIDR_BLOCK` • `protocol`: `all` • `description`: "Allow all outgoing traffic."
Three ingress security rules:	**Three ingress security rules:** `--ingress-security-rules`

Ingress Rule 1
• Stateless: clear the box
• Ingress CIDR: *`vcn_cidr`*
• IP Protocol: ICMP
  • Parameter Type: 8: Echo
• Description: "Allow ping from VCN."
	Ingress Rule 1
	• `isStateless`: `false`
	• `source`: ***`vcn_cidr`***
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `1`
	• `icmpOptions`
	  • `type`: `8`
	• `description`: "Allow ping from VCN."

Ingress Rule 2
• Stateless: clear the box
• Ingress CIDR: 0.0.0.0/0
• IP Protocol: ICMP
  • Parameter Type: 3: Destination Unreachable
• Description: "Blocks incoming requests from any source."
	Ingress Rule 2
	• `isStateless`: `false`
	• `source`: `0.0.0.0/0`
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `1`
	• `icmpOptions`
	  • `type`: `3`
	• `description`: "Blocks incoming requests from any source."

Ingress Rule 3
• Stateless: clear the box
• Ingress CIDR: 0.0.0.0/0
• IP Protocol: ICMP
  • Parameter Type: 11: Time Exceeded
• Description: "Time exceeded."
	Ingress Rule 3
	• `isStateless`: `false`
	• `source`: `0.0.0.0/0`
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `1`
	• `icmpOptions`
	  • `type`: `11`
	• `description`: "Time exceeded."

Note the name and OCID of this default security list for assignment to subnets.

Creating a Pod Subnet (VCN-Native Pod)

Learn how to create a pod subnet for VCN-Native Pod Networking On Compute Cloud@Customer.

The instructions in this topic create a pod subnet named "pod" in the VCN that provides the private IP addresses for pods running on the control plane nodes. The number of IP addresses in this subnet should be equal to or greater than the number of IP addresses in the control plane subnet. The pod subnet must be a private subnet.

The pod subnet supports communication between pods and direct access to individual pods using private pod IP addresses. The pod subnet must be private. The pod subnet enables pods to communicate with other pods on the same worker node, with pods on other worker nodes, with OCI services (through a service gateway) and with the internet (through a NAT gateway).

Create the following resources in the order listed:

1. Pod security list
2. Pod subnet

Create a Pod Security List

Create a security list. See Creating a Security List on page 159. For Terraform input, see Example Terraform Scripts (VCN-Native Pod) on page 552.

The security rules shown in the following table define traffic that is allowed to contact pods directly. Use these security rules as part of network security groups (NSGs) or in security lists. We recommend using NSGs.

The security rules apply to all pods in all the worker nodes connected to the pod subnet specified for a node pool.

Route incoming requests to pods based on routing policies specified by routing rules and route tables. See the route tables defined in Creating a VCN (VCN-Native Pod Networking) on page 563.

For this example, use the following input for the pod subnet security list.

Console property	**CLI property**
• Name: pod-seclist	• `--vcn-id`: `ocid1.vcn.`*`oke_vcn_id`* • `--display-name`: `pod-seclist`
One egress security rule: • Stateless: clear the box • Egress CIDR: 0.0.0.0/0 • IP Protocol: All protocols • Description: "Allow all outgoing traffic."	**One egress security rule:** `--egress-security-rules` • `isStateless`: `false` • `destination`: `0.0.0.0/0` • `destinationType`: `CIDR_BLOCK` • `protocol`: `all` • `description`: "Allow all outgoing traffic."
Eight ingress security rules:	**Eight ingress security rules:** `--ingress-security-rules`

Console property	CLI property

Ingress Rule 1
• Stateless: clear the box
• Ingress CIDR: *vcn_cidr*
• IP Protocol: TCP
  • Destination Port Range: 22
• Description: "Allow SSH connection to the pod subnet from all subnets in the VCN."
	Ingress Rule 1
	• `isStateless`: `false`
	• `source`: *vcn_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `22`
	  • `min`: `22`
	• `description`: "Allow SSH connection to the pod subnet from all subnets in the VCN."

Ingress Rule 2
• Stateless: clear the box
• Ingress CIDR: *workerlb_cidr*
• IP Protocol: TCP
  • Destination Port Range: 10256
• Description: "Allow the worker load balancer to contact the pods."
	Ingress Rule 2
	• `isStateless`: `false`
	• `source`: *workerlb_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `10256`
	  • `min`: `10256`
	• `description`: "Allow the worker load balancer to contact the pods."

Ingress Rule 3
• Stateless: clear the box
• Ingress CIDR: *worker_cidr*
• IP Protocol: TCP
  • Destination Port Range: 10250
• Description: "Allow Kubernetes API endpoint to pod (via worker node) communication."
	Ingress Rule 3
	• `isStateless`: `false`
	• `source`: *worker_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `10250`
	  • `min`: `10250`
	• `description`: "Allow Kubernetes API endpoint to pod (via worker node) communication."

Console property	CLI property

Ingress Rule 4
• Stateless: clear the box
• Ingress CIDR: *worker_cidr*
• IP Protocol: TCP
  • Destination Port Range: 10256
• Description: "Allow Load Balancer or Network Load Balancer to communicate with the kube-proxy pod (via the worker subnet)."
	Ingress Rule 4
	• isStateless: false
	• source: *worker_cidr*
	• sourceType: CIDR_BLOCK
	• protocol: 6
	• tcpOptions
	  destinationPortRange
	  • max: 10256
	  • min: 10256
	• description: "Allow Load Balancer or Network Load Balancer to communicate with the kube-proxy pod (via the worker subnet)."

Ingress Rule 5
• Stateless: clear the box
• Ingress CIDR: *worker_cidr*
• IP Protocol: TCP
  • Destination Port Range: 80
• Description: "Allow the worker node to contact the pods."
This ingress is optional. This port is open for an end user application. This rule could be different based on what applications are deployed.
	Ingress Rule 5
	• isStateless: false
	• source: *worker_cidr*
	• sourceType: CIDR_BLOCK
	• protocol: 6
	• tcpOptions
	  destinationPortRange
	  • max: 80
	  • min: 80
	• description: "Allow the worker node to contact the pods."
	This ingress is optional. This port is open for an end user application. This rule could be different based on what applications are deployed.

Ingress Rule 6
• Stateless: clear the box
• Ingress CIDR: *kmi_cidr*
• IP Protocol: ICMP
  • Parameter Type: 8: Echo
• Description: "Test the reachability of a network pod from *kmi_cidr* by sending a request."
	Ingress Rule 6
	• isStateless: false
	• source: *kmi_cidr*
	• sourceType: CIDR_BLOCK
	• protocol: 1
	• icmpOptions
	  • type: 8
	• description: "Test the reachability of a network pod from *kmi_cidr* by sending a request."

Console property	CLI property

Ingress Rule 7
• Stateless: clear the box
• Ingress CIDR: *kmi_cidr*
• IP Protocol: ICMP
  • Parameter Type: 0: Echo Reply
• Description: "If the destination pod is reachable from *kmi_cidr*, respond with an ICMP Echo Reply."
	Ingress Rule 7
	• `isStateless`: `false`
	• `source`: *kmi_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `1`
	• `icmpOptions`
	  • `type`: `0`
	• `description`: "If the destination pod is reachable from *kmi_cidr*, respond with an ICMP Echo Reply."

Ingress Rule 8 • Stateless: clear the box • Ingress CIDR: *pod_cidr* • IP Protocol: All protocols • Description: "Allow the pod CIDR to communicate with itself."	**Ingress Rule 8** • `isStateless`: `false` • `source`: *pod_cidr* • `sourceType`: `CIDR_BLOCK` • `protocol`: `all` • `description`: "Allow the pod CIDR to communicate with itself."

Create the Pod Subnet

Create a subnet. See Creating a Subnet on page 142. For Terraform input, see Example Terraform Scripts (VCN-Native Pod) on page 552.

For this example, use the following input to create the pod subnet. Use the OCID of the VCN that was created in Creating a VCN (VCN-Native Pod Networking) on page 563. Create the pod subnet in the same compartment where you created the VCN.

> **Important:**
>
> The name of this subnet must be exactly "pod".

Console property	CLI property

• Name: pod
• CIDR Block: *pod_cidr*
• Route Table: Select "nat_private" from the list
• Private Subnet: check the box
• DNS Hostnames:
  Use DNS Hostnames in this Subnet: check the box
  • DNS Label: pod
• Security Lists: Select "pod-seclist" and "Default Security List for oketest-vcn" from the list
	• `--vcn-id`: `ocid1.vcn.`*oke_vcn_id*
	• `--display-name`: `pod`
	• `--cidr-block`: *pod_cidr*
	• `--dns-label`: `pod`
	• `--prohibit-public-ip-on-vnic`: `true`
	• `--route-table-id`: OCID of the "nat_private" route table
	• `--security-list-ids`: OCIDs of the "pod-seclist" security list and the "Default Security List for oketest-vcn" security list

Creating a Worker Subnet (VCN-Native Pod)

Learn how to create a worker subnet for VCN-Native Pod Networking On Compute Cloud@Customer.

Create the following resources in the order listed:

1. Worker security list

2. Worker subnet

Create a Worker Security List

Create a security list. See Creating a Security List on page 159. For Terraform input, see Example Terraform Scripts (VCN-Native Pod) on page 552.

This security list defines traffic that is allowed to contact worker nodes directly.

For this example, use the following input for the worker subnet security list.

Console property	CLI property
• Name: worker-seclist	• `--vcn-id`: `ocid1.vcn.`*`oke_vcn_id`* • `--display-name`: `worker-seclist`
One egress security rule: • Stateless: clear the box • Egress CIDR: 0.0.0.0/0 • IP Protocol: All protocols • Description: "Allow all outgoing traffic."	**One egress security rule:** `--egress-security-rules` • `isStateless`: `false` • `destination`: `0.0.0.0/0` • `destinationType`: `CIDR_BLOCK` • `protocol`: `all` • `description`: "Allow all outgoing traffic."
Thirteen ingress security rules:	**Thirteen ingress security rules:** `--ingress-security-rules`

Ingress Rule 1
• Stateless: clear the box
• Ingress CIDR: *`kube_client_cidr`*
• IP Protocol: TCP
  • Destination Port Range: 30000-32767
• Description: "Allow worker nodes to receive connections through the pod subnet."
	Ingress Rule 1
	• `isStateless`: `false`
	• `source`: *`kube_client_cidr`*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `32767`
	  • `min`: `30000`
	• `description`: "Allow worker nodes to receive connections through the pod subnet."

Ingress Rule 2
• Stateless: clear the box
• Ingress CIDR: *`kmi_cidr`*
• IP Protocol: TCP
  • Destination Port Range: 22
• Description: "Allow SSH connection from the control plane subnet."
	Ingress Rule 2
	• `isStateless`: `false`
	• `source`: *`kmi_cidr`*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `22`
	  • `min`: `22`
	• `description`: "Allow SSH connection from the control plane subnet."

Console property	CLI property

Ingress Rule 3
• Stateless: clear the box
• Ingress CIDR: *worker_cidr*
• IP Protocol: TCP
  • Destination Port Range: 22
• Description: "Allow SSH connection from the worker subnet."
	Ingress Rule 3
	• `isStateless`: `false`
	• `source`: *worker_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `22`
	  • `min`: `22`
	• `description`: "Allow SSH connection from the worker subnet."

Ingress Rule 4
• Stateless: clear the box
• Ingress CIDR: *worker_cidr*
• IP Protocol: TCP
  • Destination Port Range: 10250
• Description: "Allow Kubernetes API endpoint to worker node communication."
	Ingress Rule 4
	• `isStateless`: `false`
	• `source`: *worker_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `10250`
	  • `min`: `10250`
	• `description`: "Allow Kubernetes API endpoint to worker node communication."

Ingress Rule 5
• Stateless: clear the box
• Ingress CIDR: *worker_cidr*
• IP Protocol: TCP
  • Destination Port Range: 10256
• Description: "Allow Load Balancer or Network Load Balancer to communicate with `kube-proxy` on worker nodes."
	Ingress Rule 5
	• `isStateless`: `false`
	• `source`: *worker_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `10256`
	  • `min`: `10256`
	• `description`: "Allow Load Balancer or Network Load Balancer to communicate with `kube-proxy` on worker nodes."

Console property	CLI property

Ingress Rule 6
• Stateless: clear the box
• Ingress CIDR: *worker_cidr*
• IP Protocol: TCP
  • Destination Port Range: 30000-32767
• Description: "Allow traffic to worker nodes."
	Ingress Rule 6
	• `isStateless`: `false`
	• `source`: *worker_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `32767`
	  • `min`: `30000`
	• `description`: "Allow traffic to worker nodes."

Ingress Rule 7
• Stateless: clear the box
• Ingress CIDR: *workerlb_cidr*
• IP Protocol: TCP
  • Destination Port Range: 10256
• Description: "Allow Load Balancer or Network Load Balancer to communicate with `kube-proxy` on worker nodes."
	Ingress Rule 7
	• `isStateless`: `false`
	• `source`: *workerlb_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `10256`
	  • `min`: `10256`
	• `description`: "Allow Load Balancer or Network Load Balancer to communicate with `kube-proxy` on worker nodes."

Ingress Rule 8
• Stateless: clear the box
• Ingress CIDR: *workerlb_cidr*
• IP Protocol: TCP
  • Destination Port Range: 30000-32767
• Description: "Allow worker nodes to receive connections through Network Load Balancer."
	Ingress Rule 8
	• `isStateless`: `false`
	• `source`: *workerlb_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `32767`
	  • `min`: `30000`
	• `description`: "Allow worker nodes to receive connections through Network Load Balancer."

Console property	CLI property

Ingress Rule 9
• Stateless: clear the box
• Ingress CIDR: *kmi_cidr*
• IP Protocol: TCP
  • Destination Port Range: 10250
• Description: "Allow Kubernetes API endpoint to worker node communication."
	Ingress Rule 9
	• `isStateless`: `false`
	• `source`: *kmi_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `10250`
	  • `min`: `10250`
	• `description`: "Allow Kubernetes API endpoint to worker node communication."

Ingress Rule 10
• Stateless: clear the box
• Ingress CIDR: *kmi_cidr*
• IP Protocol: TCP
  • Destination Port Range: 10256
• Description: "Allow Load Balancer or Network Load Balancer to communicate with `kube-proxy` on worker nodes."
	Ingress Rule 10
	• `isStateless`: `false`
	• `source`: *kmi_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `10256`
	  • `min`: `10256`
	• `description`: "Allow Load Balancer or Network Load Balancer to communicate with `kube-proxy` on worker nodes."

Ingress Rule 11
• Stateless: clear the box
• Ingress CIDR: *pod_cidr*
• IP Protocol: TCP
  • Destination Port Range: 30000-32767
• Description: "Allow worker nodes to receive connections through the pod subnet."
	Ingress Rule 11
	• `isStateless`: `false`
	• `source`: *pod_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `32767`
	  • `min`: `30000`
	• `description`: "Allow worker nodes to receive connections through the pod subnet."

Console property	CLI property

Ingress Rule 12
• Stateless: clear the box
• Ingress CIDR: *kmi_cidr*
• IP Protocol: ICMP
  • Parameter Type: 8: Echo
• Description: "Test the reachability of a network pod from *kmi_cidr* by sending a request."
	Ingress Rule 12
	• `isStateless`: `false`
	• `source`: *kmi_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `1`
	• `icmpOptions`
	  • `type`: `8`
	• `description`: "Test the reachability of a network pod from *kmi_cidr* by sending a request."

Ingress Rule 13
• Stateless: clear the box
• Ingress CIDR: *kmi_cidr*
• IP Protocol: ICMP
  • Parameter Type: 0: Echo Reply
• Description: "If the destination pod is reachable from *kmi_cidr*, respond with an ICMP Echo Reply."
	Ingress Rule 13
	• `isStateless`: `false`
	• `source`: *kmi_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `1`
	• `icmpOptions`
	  • `type`: `0`
	• `description`: "If the destination pod is reachable from *kmi_cidr*, respond with an ICMP Echo Reply."

Create the Worker Subnet

Create a subnet. See Creating a Subnet on page 142. For Terraform input, see Example Terraform Scripts (VCN-Native Pod) on page 552.

For this example, use the following input to create the worker subnet. Use the OCID of the VCN that was created in Creating a VCN (VCN-Native Pod Networking) on page 563. Create the worker subnet in the same compartment where you created the VCN.

Create either a NAT private worker subnet or a VCN private worker subnet. Create a NAT private worker subnet to communicate outside the VCN.

Create a NAT Private Worker Subnet

Console property	CLI property

• Name: worker
• CIDR Block: *worker_cidr*
• Route Table: Select "nat_private" from the list
• Private Subnet: check the box
• DNS Hostnames:
  Use DNS Hostnames in this Subnet: check the box
  • DNS Label: worker
• Security Lists: Select "worker-seclist" and "Default Security List for oketest-vcn" from the list
	• `--vcn-id`: `ocid1.vcn.`*oke_vcn_id*
	• `--display-name`: `worker`
	• `--cidr-block`: *worker_cidr*
	• `--dns-label`: `worker`
	• `--prohibit-public-ip-on-vnic`: `true`
	• `--route-table-id`: OCID of the "nat_private" route table
	• `--security-list-ids`: OCIDs of the "worker-seclist" security list and the "Default Security List for oketest-vcn" security list

The difference in the following private subnet is the VCN private route table is used instead of the NAT private route table.

Create a VCN Private Worker Subnet

Console property	CLI property
• Name: worker • CIDR Block: *worker_cidr* • Route Table: Select "vcn_private" from the list • Private Subnet: check the box • DNS Hostnames: Use DNS Hostnames in this Subnet: check the box • DNS Label: worker • Security Lists: Select "worker-seclist" and "Default Security List for oketest-vcn" from the list	• `--vcn-id`: `ocid1.vcn.`*oke_vcn_id* • `--display-name`: `worker` • `--cidr-block`: *worker_cidr* • `--dns-label`: `worker` • `--prohibit-public-ip-on-vnic`: `true` • `--route-table-id`: OCID of the "vcn_private" route table • `--security-list-ids`: OCIDs of the "worker-seclist" security list and the "Default Security List for oketest-vcn" security list

Creating a Worker Load Balancer Subnet (VCN-Native Pod)

Learn how to create a worker load balancer subnet for VCN-Native Pod Networking On Compute Cloud@Customer.

Create the following resources in the order listed:

1. Worker load balancer security list
2. Worker load balancer subnet

Create a Worker Load Balancer Security List

Create a security list. See Creating a Security List on page 159. For Terraform input, see Example Terraform Scripts (VCN-Native Pod) on page 552.

This security list defines traffic, such as applications, that is allowed to contact the worker load balancer.

For this example, use the following input for the worker load balancer subnet security list. These sources and destinations are examples; adjust these for your applications.

> **Note:**
>
> When you create an external load balancer for your containerized applications (see Exposing Containerized Applications on page 633), remember to add that load balancer service front-end port to this security list.

Console property	CLI property
• Name: workerlb-seclist	• `--vcn-id`: `ocid1.vcn.`*oke_vcn_id* • `--display-name`: `workerlb-seclist`
One egress security rule: • Stateless: clear the box • Egress CIDR: 0.0.0.0/0 • IP Protocol: All protocols • Description: "Allow all outgoing traffic."	**One egress security rule:** `--egress-security-rules` • `isStateless`: `false` • `destination`: `0.0.0.0/0` • `destinationType`: `CIDR_BLOCK` • `protocol`: `all` • `description`: "Allow all outgoing traffic."
Two ingress security rules:	**Two ingress security rules:** `--ingress-security-rules`

Console property	CLI property

Ingress Rule 1
• Stateless: clear the box
• Ingress CIDR: *kube_client_cidr*
• IP Protocol: TCP
  • Destination Port Range: 80
• Description: "Allow inbound traffic for applications."
	Ingress Rule 1
	• `isStateless`: `false`
	• `source`: *kube_client_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `80`
	  • `min`: `80`
	• `description`: "Allow inbound traffic for applications."

Ingress Rule 2
• Stateless: clear the box
• Ingress CIDR: *kube_client_cidr*
• IP Protocol: TCP
  • Destination Port Range: 443
• Description: "Allow inbound traffic for applications."
	Ingress Rule 2
	• `isStateless`: `false`
	• `source`: *kube_client_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `443`
	  • `min`: `443`
	• `description`: "Allow inbound traffic for applications."

Create the Worker Load Balancer Subnet

Create a subnet. See Creating a Subnet on page 142. For Terraform input, see Example Terraform Scripts (VCN-Native Pod) on page 552.

For this example, use the following input to create the worker load balancer subnet. Use the OCID of the VCN that was created in Creating a VCN (VCN-Native Pod Networking) on page 563. Create the worker load balancer subnet in the same compartment where you created the VCN.

Create either a private or a public worker load balancer subnet. Create a public worker load balancer subnet to use with a public cluster. Create a private worker load balancer subnet to expose applications in a private cluster.

Create a Public Worker Load Balancer Subnet

Console property	CLI property
• Name: service-lb • CIDR Block: *workerlb_cidr* • Route Table: Select "public" from the list • Public Subnet: check the box • DNS Hostnames: Use DNS Hostnames in this Subnet: check the box • DNS Label: servicelb • Security Lists: Select "workerlb-seclist" and "Default Security List for oketest-vcn" from the list	• `--vcn-id`: `ocid1.vcn.`*oke_vcn_id* • `--display-name`: `service-lb` • `--cidr-block`: *workerlb_cidr* • `--dns-label`: `servicelb` • `--prohibit-public-ip-on-vnic`: `false` • `--route-table-id`: OCID of the "public" route table • `--security-list-ids`: OCIDs of the "workerlb-seclist" security list and the "Default Security List for oketest-vcn" security list

The difference in the following private subnet is the VCN private route table is used instead of the public route table.

Create a VCN Private Worker Load Balancer Subnet

Console property	CLI property
• Name: service-lb • CIDR Block: *workerlb_cidr* • Route Table: Select "vcn_private" from the list • Private Subnet: check the box • DNS Hostnames: Use DNS Hostnames in this Subnet: check the box • DNS Label: servicelb • Security Lists: Select "workerlb-seclist" and "Default Security List for oketest-vcn" from the list	• `--vcn-id`: `ocid1.vcn.`*oke_vcn_id* • `--display-name`: `service-lb` • `--cidr-block`: *workerlb_cidr* • `--dns-label`: `servicelb` • `--prohibit-public-ip-on-vnic`: `true` • `--route-table-id`: OCID of the "vcn_private" route table • `--security-list-ids`: OCIDs of the "workerlb-seclist" security list and the "Default Security List for oketest-vcn" security list

Creating a Control Plane Subnet (VCN-Native Pod)

Learn how to create a control plane subnet for VCN-Native Pod Networking On Compute Cloud@Customer.

Create the following resources in the order listed:

1. Control plane security list
2. Control plane subnet

Create a Control Plane Security List

Create a security list. See Creating a Security List on page 159. For Terraform input, see Example Terraform Scripts (VCN-Native Pod) on page 552.

For this example, use the following input for the control plane subnet security list. The `kubernetes_api_port` is the port used to access the Kubernetes API: port 6443. See also Workload Cluster Network Ports (VCN-Native Pod) on page 551.

Console property	CLI property
• Name: kmi-seclist	• `--vcn-id`: `ocid1.vcn.`*oke_vcn_id* • `--display-name`: `kmi-seclist`

Console property	CLI property
One egress security rule: • Stateless: clear the box • Egress CIDR: 0.0.0.0/0 • IP Protocol: All protocols • Description: "Allow all outgoing traffic."	**One egress security rule:** `--egress-security-rules` • `isStateless`: `false` • `destination`: `0.0.0.0/0` • `destinationType`: `CIDR_BLOCK` • `protocol`: `all` • `description`: "Allow all outgoing traffic."
Eleven ingress security rules:	**Eleven ingress security rules:** `--ingress-security-rules`

Ingress Rule 1
• Stateless: clear the box
• Ingress CIDR: *`kube_client_cidr`*
• IP Protocol: TCP
  • Destination Port Range: *`kubernetes_api_port`*
• Description: "Allow clients to communicate with Kubernetes API."
	Ingress Rule 1
	• `isStateless`: `false`
	• `source`: *`kube_client_cidr`*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	  `destinationPortRange`
	  • `max`: *`kubernetes_api_port`*
	  • `min`: *`kubernetes_api_port`*
	• `description`: "Allow clients to communicate with Kubernetes API."

Ingress Rule 2
• Stateless: clear the box
• Ingress CIDR: *`kmilb_cidr`*
• IP Protocol: TCP
  • Destination Port Range: *`kubernetes_api_port`*
• Description: "Allow the load balancer to communicate with Kubernetes control plane APIs."
	Ingress Rule 2
	• `isStateless`: `false`
	• `source`: *`kmilb_cidr`*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	  `destinationPortRange`
	  • `max`: *`kubernetes_api_port`*
	  • `min`: *`kubernetes_api_port`*
	• `description`: "Allow the load balancer to communicate with Kubernetes control plane APIs."

Console property	CLI property

Ingress Rule 3
• Stateless: clear the box
• Ingress CIDR: *kmilb_cidr*
• IP Protocol: TCP
  • Destination Port Range: 12250
• Description: "Allow Kubernetes worker to Kubernetes API endpoint communication via the control plane load balancer."
	Ingress Rule 3
	• isStateless: false
	• source: *kmilb_cidr*
	• sourceType: CIDR_BLOCK
	• protocol: 6
	• tcpOptions
	  destinationPortRange
	  • max: 12250
	  • min: 12250
	• description: "Allow Kubernetes worker to Kubernetes API endpoint communication via the control plane load balancer."

Ingress Rule 4
• Stateless: clear the box
• Ingress CIDR: *worker_cidr*
• IP Protocol: TCP
  • Destination Port Range: *kubernetes_api_port*
• Description: "Allow worker nodes to access the Kubernetes API."
	Ingress Rule 4
	• isStateless: false
	• source: *worker_cidr*
	• sourceType: CIDR_BLOCK
	• protocol: 6
	• tcpOptions
	  destinationPortRange
	  • max: *kubernetes_api_port*
	  • min: *kubernetes_api_port*
	• description: "Allow worker nodes to access the Kubernetes API."

Ingress Rule 5
• Stateless: clear the box
• Ingress CIDR: *worker_cidr*
• IP Protocol: TCP
  • Destination Port Range: 12250
• Description: "Allow Kubernetes worker to Kubernetes API endpoint communication."
	Ingress Rule 5
	• isStateless: false
	• source: *worker_cidr*
	• sourceType: CIDR_BLOCK
	• protocol: 6
	• tcpOptions
	  destinationPortRange
	  • max: 12250
	  • min: 12250
	• description: "Allow Kubernetes worker to Kubernetes API endpoint communication."

Console property	CLI property

Ingress Rule 6
• Stateless: clear the box
• Ingress CIDR: *kmi_cidr*
• IP Protocol: TCP
  • Destination Port Range: *kubernetes_api_port*
• Description: "Allow the control plane to reach itself."
	Ingress Rule 6
	• `isStateless`: `false`
	• `source`: *kmi_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: *kubernetes_api_port*
	  • `min`: *kubernetes_api_port*
	• `description`: "Allow the control plane to reach itself."

Ingress Rule 7
• Stateless: clear the box
• Ingress CIDR: *kmi_cidr*
• IP Protocol: TCP
  • Destination Port Range: 2379-2381
• Description: "Allow the control plane to reach `etcd` services and metrics. Ports 2379 and 2380 are used by Kubernetes to communicate with the `etcd` server. Port 2381 is used by Kubernetes to collect metrics from `etcd`."
	Ingress Rule 7
	• `isStateless`: `false`
	• `source`: *kmi_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `2381`
	  • `min`: `2379`
	• `description`: "Allow the control plane to reach `etcd` services and metrics. Ports 2379 and 2380 are used by Kubernetes to communicate with the `etcd` server. Port 2381 is used by Kubernetes to collect metrics from `etcd`."

Ingress Rule 8
• Stateless: clear the box
• Ingress CIDR: *kmi_cidr*
• IP Protocol: TCP
  • Destination Port Range: 10250
• Description: "Allow Kubernetes API endpoint to control plane node communication."
	Ingress Rule 8
	• `isStateless`: `false`
	• `source`: *kmi_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `10250`
	  • `min`: `10250`
	• `description`: "Allow Kubernetes API endpoint to control plane node communication."

Console property	CLI property

Ingress Rule 9
• Stateless: clear the box
• Ingress CIDR: *kmi_cidr*
• IP Protocol: TCP
  • Destination Port Range: 10257-10260
• Description: "Allow inbound connection for Kubernetes components."
	Ingress Rule 9
	• `isStateless`: `false`
	• `source`: *kmi_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `10260`
	  • `min`: `10257`
	• `description`: "Allow inbound connection for Kubernetes components."

Ingress Rule 10
• Stateless: clear the box
• Ingress CIDR: *pod_cidr*
• IP Protocol: TCP
  • Destination Port Range: *kubernetes_api_port*
• Description: "Allow pods to communicate with Kubernetes APIs."
	Ingress Rule 10
	• `isStateless`: `false`
	• `source`: *pod_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: *kubernetes_api_port*
	  • `min`: *kubernetes_api_port*
	• `description`: "Allow pods to communicate with Kubernetes APIs."

Ingress Rule 11
• Stateless: clear the box
• Ingress CIDR: *pod_cidr*
• IP Protocol: TCP
  • Destination Port Range: 12250
• Description: "Allow Kubernetes pods to Kubernetes API endpoint communication."
	Ingress Rule 11
	• `isStateless`: `false`
	• `source`: *pod_cidr*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `12250`
	  • `min`: `12250`
	• `description`: "Allow Kubernetes pods to Kubernetes API endpoint communication."

Create the Control Plane Subnet

Create a subnet. See Creating a Subnet on page 142. For Terraform input, see Example Terraform Scripts (VCN-Native Pod) on page 552.

Use the following input to create the control plane subnet. Use the OCID of the VCN that was created in Creating a VCN (VCN-Native Pod Networking) on page 563. Create the control plane subnet in the same compartment where you created the VCN.

Create either a NAT private control plane subnet or a VCN private control plane subnet. Create a NAT private control plane subnet to communicate outside the VCN.

> **Important:**
>
> The name of this subnet must be exactly "control-plane".

Create a Data Center Private Control Plane Subnet

Console property	CLI property
• Name: control-plane • CIDR Block: *kmi_cidr* • Route Table: Select "nat_private" from the list • Private Subnet: check the box • DNS Hostnames: Use DNS Hostnames in this Subnet: check the box • DNS Label: kmi • Security Lists: Select "kmi-seclist" and "Default Security List for oketest-vcn" from the list	• `--vcn-id`: `ocid1.vcn.`*oke_vcn_id* • `--display-name`: `control-plane` • `--cidr-block`: *kmi_cidr* • `--dns-label`: `kmi` • `--prohibit-public-ip-on-vnic`: `true` • `--route-table-id`: OCID of the "nat_private" route table • `--security-list-ids`: OCIDs of the "kmi-seclist" security list and the "Default Security List for oketest-vcn" security list

The difference in the following private subnet is the VCN private route table is used instead of the NAT private route table.

Create a VCN Private Control Plane Subnet

Console property	CLI property
• Name: control-plane • CIDR Block: *kmi_cidr* • Route Table: Select "vcn_private" from the list • Private Subnet: check the box • DNS Hostnames: Use DNS Hostnames in this Subnet: check the box • DNS Label: kmi • Security Lists: Select "kmi-seclist" and "Default Security List for oketest-vcn" from the list	• `--vcn-id`: `ocid1.vcn.`*oke_vcn_id* • `--display-name`: `control-plane` • `--cidr-block`: *kmi_cidr* • `--dns-label`: `kmi` • `--prohibit-public-ip-on-vnic`: `true` • `--route-table-id`: OCID of the "vcn_private" route table • `--security-list-ids`: OCIDs of the "kmi-seclist" security list and the "Default Security List for oketest-vcn" security list

Creating a Control Plane Load Balancer Subnet (VCN-Native Pod)

Learn how to create a control plane load balancer subnet for VCN-Native Pod Networking on Compute Cloud@Customer.

Create the following resources in the order listed:

1. Control plane load balancer security list
2. Control plane load balancer subnet

Create a Control Plane Load Balancer Security List

Create a security list. See Creating a Security List on page 159. For Terraform input, see Example Terraform Scripts (VCN-Native Pod) on page 552.

The control plane load balancer accepts traffic on port 6443, which is also called ***kubernetes_api_port*** in this guide. Adjust this security list to only accept connections from where you expect the network to run. Port 6443 must accept connections from the cluster control plane instances and worker instances.

For this example, use the following input for the control plane load balancer subnet security list.

Console property	CLI property
• Name: kmilb-seclist	• `--vcn-id`: `ocid1.vcn.`*`oke_vcn_id`* • `--display-name`: kmilb-seclist
One egress security rule: • Stateless: clear the box • Egress CIDR: 0.0.0.0/0 • IP Protocol: All protocols • Description: "Allow all outgoing traffic."	**One egress security rule:** `--egress-security-rules` • `isStateless`: `false` • `destination`: `0.0.0.0/0` • `destinationType`: `CIDR_BLOCK` • `protocol`: `all` • `description`: "Allow all outgoing traffic."
Eight ingress security rules:	**Eight ingress security rules:** `--ingress-security-rules`

Ingress Rule 1:
• Stateless: clear the box
• Ingress CIDR: *`kube_internal_cidr`*
This value is required. Do not change this CIDR value.
• IP Protocol: TCP
  • Destination Port Range: *`kubernetes_api_port`*
• Description: "Allow a Kubernetes container to communicate with Kubernetes APIs."
	Ingress Rule 1:
	• `isStateless`: `false`
	• `source`: *`kube_internal_cidr`*
	This value is required. Do not change this CIDR value.
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: *`kubernetes_api_port`*
	  • `min`: *`kubernetes_api_port`*
	• `description`: "Allow a Kubernetes container to communicate with Kubernetes APIs."

Ingress Rule 2:
• Stateless: clear the box
• Ingress CIDR: *`kube_client_cidr`*
• IP Protocol: TCP
  • Destination Port Range: *`kubernetes_api_port`*
• Description: "Allow clients to connect with the Kubernetes cluster."
	Ingress Rule 2:
	• `isStateless`: `false`
	• `source`: *`kube_client_cidr`*
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: *`kubernetes_api_port`*
	  • `min`: *`kubernetes_api_port`*
	• `description`: "Allow clients to connect with the Kubernetes cluster."

Console property	CLI property

Ingress Rule 3:
• Stateless: clear the box
• Ingress CIDR: *kmi_cidr*
• IP Protocol: TCP
  • Destination Port Range: *kubernetes_api_port*
• Description: "Allow the control plane to reach itself via the load balancer."
	Ingress Rule 3:
	• isStateless: false
	• source: *kmi_cidr*
	• sourceType: CIDR_BLOCK
	• protocol: 6
	• tcpOptions
	  destinationPortRange
	  • max: *kubernetes_api_port*
	  • min: *kubernetes_api_port*
	• description: "Allow the control plane to reach itself via the load balancer."

Ingress Rule 4:
• Stateless: clear the box
• Ingress CIDR: *worker_cidr*
• IP Protocol: TCP
  • Destination Port Range: *kubernetes_api_port*
• Description: "Allow worker nodes to connect with the cluster via the control plane load balancer."
	Ingress Rule 4:
	• isStateless: false
	• source: *worker_cidr*
	• sourceType: CIDR_BLOCK
	• protocol: 6
	• tcpOptions
	  destinationPortRange
	  • max: *kubernetes_api_port*
	  • min: *kubernetes_api_port*
	• description: "Allow worker nodes to connect with the cluster via the control plane load balancer."

Ingress Rule 5:
• Stateless: clear the box
• Ingress CIDR: *worker_cidr*
• IP Protocol: TCP
  • Destination Port Range: 12250
• Description: "Allow Kubernetes worker to Kubernetes API endpoint communication via the load balancer."
	Ingress Rule 5:
	• isStateless: false
	• source: *worker_cidr*
	• sourceType: CIDR_BLOCK
	• protocol: 6
	• tcpOptions
	  destinationPortRange
	  • max: 12250
	  • min: 12250
	• description: "Allow Kubernetes worker to Kubernetes API endpoint communication via the load balancer."

Console property	CLI property

Ingress Rule 6:
• Stateless: clear the box
• Ingress CIDR: *pod_cidr*
• IP Protocol: TCP
  • Destination Port Range: 12250
• Description: "Allow Kubernetes pods to Kubernetes API endpoint communication via the load balancer."
	Ingress Rule 6:
	• isStateless: false
	• source: *pod_cidr*
	• sourceType: CIDR_BLOCK
	• protocol: 6
	• tcpOptions
	destinationPortRange
	  • max: 12250
	  • min: 12250
	• description: "Allow Kubernetes pods to Kubernetes API endpoint communication via the load balancer."

Ingress Rule 7: Private Endpoint
• Stateless: clear the box
• Ingress CIDR: *kmilb_cidr*
• IP Protocol: TCP
  • Destination Port Range: *kubernetes_api_port*
• Description: "Used to create a private control plane endpoint."
	Ingress Rule 7: Private Endpoint
	• isStateless: false
	• source: *kmilb_cidr*
	• sourceType: CIDR_BLOCK
	• protocol: 6
	• tcpOptions
	destinationPortRange
	  • max: *kubernetes_api_port*
	  • min: *kubernetes_api_port*
	• description: "Used to create a private control plane endpoint."

Ingress Rule 8: Public Endpoint
• Stateless: clear the box
• Ingress CIDR: *public_ip_cidr*
• IP Protocol: TCP
  • Destination Port Range: *kubernetes_api_port*
• Description: "Used to access the control plane endpoint from the public CIDR. If you do not know what your public IP CIDR is, open a Support Request. See Creating a Support Request. To access support, sign in to the Oracle Cloud Console as described in Sign In to the OCI Console."
	Ingress Rule 8: Public Endpoint
	• isStateless: false
	• source: *public_ip_cidr*
	• sourceType: CIDR_BLOCK
	• protocol: 6
	• tcpOptions
	destinationPortRange
	  • max: *kubernetes_api_port*
	  • min: *kubernetes_api_port*
	• description: "Used to access the control plane endpoint from the public CIDR. If you do not know what your public IP CIDR is, open a Support Request. See Creating a Support Request. To access support, sign in to the Oracle Cloud Console as described in Sign In to the OCI Console."

Create the Control Plane Load Balancer Subnet

Create a subnet. See Creating a Subnet on page 142. For Terraform input, see Example Terraform Scripts (VCN-Native Pod) on page 552.

For this example, use the following input to create the control plane load balancer subnet. Use the OCID of the VCN that was created in Creating a VCN (VCN-Native Pod Networking) on page 563. Create the control plane load balancer subnet in the same compartment where you created the VCN.

Create either a private or a public control plane load balancer subnet. Create a public control plane load balancer subnet to use with a public cluster. Create a private control plane load balancer subnet to use with a private cluster.

See Private Clusters on page 518 for information about using Local Peering Gateways to connect a private cluster to other instances on the Compute Cloud@Customer and using Dynamic Routing Gateways to connect a private cluster to the on-premises IP address space. To create a private control plane load balancer subnet, specify one of the following route tables:

- vcn_private
- lpg_rt
- drg_rt

Create a Public Control Plane Load Balancer Subnet

Console property	CLI property
• Name: control-plane-endpoint • CIDR Block: *kmilb_cidr* • Route Table: Select "public" from the list • Public Subnet: check the box • DNS Hostnames: Use DNS Hostnames in this Subnet: check the box • DNS Label: kmilb • Security Lists: Select "kmilb-seclist" and "Default Security List for oketest-vcn" from the list	• `--vcn-id`: `ocid1.vcn.`*oke_vcn_id* • `--display-name`: `control-plane-endpoint` • `--cidr-block`: *kmilb_cidr* • `--dns-label`: `kmilb` • `--prohibit-public-ip-on-vnic`: `false` • `--route-table-id`: OCID of the "public" route table • `--security-list-ids`: OCIDs of the "kmilb-seclist" security list and the "Default Security List for oketest-vcn" security list

The difference in the following private subnet is the VCN private route table is used instead of the public route table. Depending on your needs, you could specify the LPG route table or the DRG route table instead.

Create a Private Control Plane Load Balancer Subnet

Console property	CLI property
• Name: control-plane-endpoint • CIDR Block: *kmilb_cidr* • Route Table: Select "vcn_private" from the list • Private Subnet: check the box • DNS Hostnames: Use DNS Hostnames in this Subnet: check the box • DNS Label: kmilb • Security Lists: Select "kmilb-seclist" and "Default Security List for oketest-vcn" from the list	• `--vcn-id`: `ocid1.vcn.`*oke_vcn_id* • `--display-name`: `control-plane-endpoint` • `--cidr-block`: *kmilb_cidr* • `--dns-label`: `kmilb` • `--prohibit-public-ip-on-vnic`: `true` • `--route-table-id`: OCID of the "vcn_private" route table • `--security-list-ids`: OCIDs of the "kmilb-seclist" security list and the "Default Security List for oketest-vcn" security list

Creating and Managing OKE Clusters

On Compute Cloud@Customer, you can create and manage OKE clusters.

You can create either a public cluster or a private cluster. See Public and Private Clusters on page 518 for the resources required for each.

> **Note:**
>
> You can't create both public and private clusters in one VCN.

A cluster includes cluster management nodes. Creating an OKE Cluster on page 589 describes how to recognize those management nodes in a list of instances.

Other topics in this section describe how to create a Kubernetes configuration file. You need a Kubernetes configuration file for each OKE cluster that you work with. The Kubernetes configuration file enables you to access OKE clusters using the `kubectl` command and the Kubernetes Dashboard.

The following topics describe how to manage Kubernetes clusters:

- Creating an OKE Cluster on page 589
- Creating a Kubernetes Configuration File on page 596
- Updating an OKE Cluster on page 598
- Deleting an OKE Cluster on page 599

Creating an OKE Cluster

Learn how to create OKE Clusters on Compute Cloud@Customer.

If you create a public cluster, the Network Load Balancer and public IP address are created and assigned as part of cluster creation.

> **Important:**
>
> Before you can create a cluster, the following conditions must be met:
>
> - The OraclePCA-OKE.cluster_id defined tag must exist in the tenancy. See Create the OraclePCA-OKE.cluster_id Tag on page 510.
>
> The OraclePCA-OKE.cluster_id defined tag is required to create or update an OKE cluster or node pool. This tag also is used to identify instances that need to be in a dynamic group.
> - All fault domains must be healthy.
> - Each fault domain must have at least one healthy compute instance.
> - Sufficient resources must be available to create a cluster.
> - Ensure that no infrastructure upgrade is scheduled during the cluster create.

To create a node pool at the same time that you create the cluster, you must use the Console.

To specify tags to be applied to all load balancers created by Kubernetes services, you must use the CLI.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Containers**, then select **Kubernetes Clusters**.
2. On the clusters list page, select **Create Cluster**.
3. In the **Create Cluster** dialog box, provide the following information:
   - **Name**: Enter a name for the new cluster. Avoid entering confidential information.
   - **Compartment**: Select the compartment in which to create the new cluster.
   - **Kubernetes Version**: Select the version of Kubernetes to run on the control plane nodes. Accept the default version or select a different version.

If the Kubernetes version that you want to use isn't listed, use the CLI or the API to create the cluster and specify the Kubernetes version.

- **Tagging**: Add the following free-form tags for the cluster resource.

  > **Note:**
  >
  > Don't specify values for the OraclePCA-OKE.cluster_id defined tag or for the ClusterResourceIdentifier free-form tag. These tag values are system-generated and only applied to nodes (instances), not to the cluster resource.

  Use OraclePCA defined tags to provide the following information for control plane nodes. If these tags aren't listed in the Compute Cloud@Customer Console Tagging menus, you must create them. See Create OraclePCA Tags For OKE on page 513.

  > **Note:**
  >
  > None of these values - SSH key, number of nodes, node shape, or node shape configuration - can be set or changed after the cluster is created. If you set these tags when you update the cluster, the new values are ignored.

  - **Your public SSH key**.

    Specify `sshkey` for the tag key (`OraclePCA.sshkey`). Paste your public SSH key into the Value field.

    > **Important:**
    >
    > You can't add an SSH key after the cluster is created.

  - **Number of nodes**.

    By default, the number of nodes in the control plane is 3. You can specify 1, 3, or 5 nodes. To specify the number of control plane nodes, specify `cp_node_count` for the tag key (`OraclePCA.cpNodeCount`), and select `1`, `3`, or `5` in the **Value** field.

  - **Node shape**.

    Specify a shape such as `VM.PCAStandard.E5.Flex`. For descriptions of available shapes, see Compute Shapes on page 301.

  - **Node shape configuration**.

    You can change the default shape configuration.

    To provide shape configuration information, specify `cp_node_shape_config` for the tag key (`OraclePCA.cpNodeShapeConfig`). You must specify the number of OCPUs (`ocpus`) you want. You can optionally specify the total amount of memory you want (`memoryInGBs`). The default value for gigabytes of memory is 16 times the number you specify for OCPUs.

    > **Note:**
    >
    > If the cluster will have 1-10 worker nodes, specify at least 16 GB memory. If the cluster will have 11-128 worker nodes, specify at least 2 OCPUs and 32 GB memory. You can't change the number of OCPUs or amount of memory when you update the cluster.

    In the **Value** field for the tag, enter the node shape configuration value as shown in the following examples.

    In the following example, the default amount of memory is be configured:

    ```
    {"ocpus":1}
    ```

In the following example, the amount of memory is specified:

```
{"ocpus":2, "memoryInGBs":48}
```

> **Note:**
>
> If you use Terraform to specify a complex value (a value that's a key/value pair), then you must escape the double quotation marks in the value as shown in the following example:
>
> ```
> "OraclePCA.cpNodeShapeConfig"="{\"ocpus\":2,
> \"memoryInGBs\":48}"
> ```

- **Add-ons**: This section shows a tile for each add-on that's available for this cluster. In the Create Cluster dialog box, all add-ons are Disabled. See Deploying the WebLogic Kubernetes Operator Add-on on page 604.

4. Select **Next**.
5. On the **Network** page in the **Create Cluster** dialog box, provide the following information:
   - **Network Type**. Specifies how pods running on nodes in the cluster communicate with each other, with the cluster's control plane nodes, with pods on other clusters, with other services (such as storage services), and with the internet.

     The **Flannel Overlay** network type encapsulates communication between pods in the Flannel Overlay network. The Flannel Overlay network is a simple private overlay virtual network that satisfies the requirements of the OKE networking model by attaching IP addresses to containers. The pods in the private overlay network are only accessible from other pods in the same cluster. For more description, see Creating Flannel Overlay Network Resources on page 520.
   - **VCN-Native Pod Networking** connects nodes in a Kubernetes cluster to pod subnets in the OKE VCN. As a result, pod IP addresses within the OKE VCN are directly routable from other VCNs that are connected (peered) to the OKE VCN, and from on-premises networks. For more description, see Creating VCN-Native Pod Network Resources on page 548.

     > **Note:**
     >
     > If you specify VCN-Native Pod Networking, then the VCN you specify must have a subnet named `pod`. See Creating VCN-Native Pod Network Resources on page 548.
   - **VCN**. Select the VCN that has the configuration of the "oke_vcn" VCN described in Creating a VCN (Flannel Overlay) on page 532 or Creating a VCN (VCN-Native Pod Networking) on page 563.
   - **Kubernetes Service LB Subnet**. The subnet that's configured to host the load balancer in a Kubernetes cluster. To create a public cluster, create and specify here the public version of the "service-lb" subnet described in Creating a Worker Load Balancer Subnet (Flannel Overlay) on page 540 or Creating a Worker Load Balancer Subnet (VCN-Native Pod) on page 577. To create a private cluster, create and specify here the private version of the "service-lb" subnet.
   - **Kubernetes API Endpoint Subnet**. The regional subnet in which to place the cluster endpoint. To create a public cluster, create and specify here the public version of the "control-plane-endpoint" subnet described in Creating an OKE Control Plane Subnet (Flannel Overlay) on page 542 or Creating a Control Plane Subnet (VCN-Native Pod) on page 579. To create a private cluster, create and specify here the private version of the "control-plane-endpoint" subnet.
   - **Kubernetes Service CIDR Block**. (Optional) The default value is 10.96.0.0/16.
   - **Pods CIDR Block**. (Optional) The default value is 10.244.0.0/16.
   - **Network Security Group**. If you check the box to enable network security groups, select **Add Network Security Group** and select an NSG from the drop-down list. You might need to change the compartment to find the NSG you want.
6. Select **Next**.
7. On the **Node Pool** page, select the Add Node Pool button to optionally add a node pool as part of creating this cluster. See Creating an OKE Worker Node Pool on page 619 to add node pools after the cluster is created.

If you select the Add Node Pool button, enter the following information in the Add Node Pool section:

- **Name**: The name of the new node pool. Avoid using confidential information.
- **Compartment**: The compartment in which to create the new node pool.
- **Node Count**: Enter the number of nodes you want in this node pool. The default is 0. The maximum number is 128 per cluster, which can be distributed across multiple node pools.
- See Creating an OKE Worker Node Pool on page 619 for information about Network Security Groups, Placement Configuration, Source Image, Shape, and Pod Communication.

8. Review your entries.

   If you created a node pool, you have the opportunity to edit or delete the node pool in this review.

9. Select **Submit**.

   The details page for the cluster is displayed. Scroll to the Resources section and select Work Requests to see the progress of the cluster creation. If you created a node pool, the NODEPOOL_CREATE work request might still be In Progress for a time after the cluster is Active and the CLUSTER_CREATE work request is Succeeded.

   Under **Resources**, select **Work Requests** to see the progress of the cluster creation.

   The cluster details page doesn't list OraclePCA tags on the Tags tab (and you can't filter a list of clusters by the values of OraclePCA tags). To review the settings of the OraclePCA tags, use the CLI.

   The cluster details page doesn't list the cluster control plane nodes. To view the control plane nodes, view the list of instances in the compartment where you created this cluster. Names of control plane nodes are in the following format:

   ```
   oke-ID1-control-plane-ID2
   ```

   - *ID1* - The first 32 characters after the ***ccc_name*** is the cluster OCID.
   - *ID2* - A unique identifier added when the cluster has more than one control plane node.

   Search for the instances in the list whose names contain the *ID1* string from this cluster OCID.

What's Next:

1. Create a Kubernetes configuration file for the cluster. See Creating a Kubernetes Configuration File on page 596.
2. Deploy a Kubernetes Dashboard to manage the cluster and to manage and troubleshoot applications running in the cluster. On the https://kubernetes.io/ site, see Deploy and Access the Kubernetes Dashboard.
3. Create a node pool for the cluster. See Creating an OKE Worker Node Pool on page 619.
4. Create a backup for the workload cluster. For example, see Backing up an etcd cluster and Restoring up an etcd cluster in Operating etcd clusters for Kubernetes. Use the etcd backup to recover OKE clusters under disaster scenarios such as losing all control plane nodes. An etcd backup contains all OKE states and critical information. An etcd backup does not back up applications or other content on cluster nodes.

Using the CLI

Use the oci ce cluster create command and required parameters to create a new cluster.

```
oci ce cluster create --compartment-id <coompartment_OCID> --kubernetes-
version <version-to-install> --name <cluster_name> --vcn-id <VCN_OCID>
 [OPTIONS]
```

To install a cluster add-on, use the `cluster install-addon` command after you have created the cluster. See Deploying the WebLogic Kubernetes Operator Add-on on page 604.

Procedure

1. Get the information you need to run the command:

   - The OCID of the compartment where you want to create the cluster: `oci iam compartment list`

- The name of the cluster. Avoid entering confidential information.
- The Kubernetes version that you want to use. Use the following command to show a list of available Kubernetes versions:

  ```
  oci ce cluster-options get --cluster-option-id all
  ```

  You might be able to list more Kubernetes versions by using the `compute image list` command and looking in the display name. In the following example, the Kubernetes version in the image is 1.29.9:

  ```
  "display-name": "uln-pca-Oracle-Linux8-OKE-1.29.9-20250325.oci"
  ```

  Another way to specify a version that's not listed is to use the OCID of an older cluster instead of the keyword `all` as the argument of the `--cluster-option-id` option to list the Kubernetes version used for that specified cluster:

  ```
  oci ce cluster-options get --cluster-option-id cluster_OCID
  ```
- OCID of the virtual cloud network (VCN) in which you want to create the cluster. Specify the VCN that has the configuration of the "oke_vcn" VCN described in Creating a VCN (Flannel Overlay) on page 532 or Creating a VCN (VCN-Native Pod Networking) on page 563.
- OCID of the OKE service LB subnet. Specify the subnet that has a configuration such as the "service-lb" subnet described in Creating a Worker Load Balancer Subnet (Flannel Overlay) on page 540 or Creating a Worker Load Balancer Subnet (VCN-Native Pod) on page 577. Specify only one OKE service LB subnet.
- OCID of the Kubernetes API endpoint subnet. Specify the subnet that has configuration such as the "control-plane-endpoint" subnet described in Creating an OKE Control Plane Load Balancer Subnet (Flannel Overlay) on page 545 or Creating a Control Plane Subnet (VCN-Native Pod) on page 579. For a public cluster, follow the instructions to create the public version of the "service-lb" subnet. For a private cluster, create the private version of the "service-lb" subnet. Specify only one OKE Service LB subnet.
- OKE service CIDR block. (Optional) The default value is 10.96.0.0/16.
- Pods CIDR block. (Optional) The default value is 10.244.0.0/16.
- (Optional) The OCID of the Network Security Group to apply to the cluster endpoint. Don't specify more than one NSG. If you specify an NSG, use the following syntax:

  ```
  --endpoint-nsg-ids '["ocid1.networksecuritygroup.unique_ID"]'
  ```
- (Optional) Your public SSH key in RSA format. You can't add or update an SSH key after the cluster is created.
- The network type. (Optional) Specify either `OCI_VCN_IP_NATIVE` or `FLANNEL_OVERLAY` for the value of the `cniType` parameter in the argument for the `--cluster-pod-network-options` option. See the descriptions of Flannel Overlay and VCN-Native Pod Networking in the Console tab. If you don't specify the `--cluster-pod-network-options` option, `FLANNEL_OVERLAY` is used.

  ```
  --cluster-pod-network-options '[{"cniType": "OCI_VCN_IP_NATIVE"}]'
  ```

  > **Note:**
  >
  > If you specify `OCI_VCN_IP_NATIVE`, then the VCN you specify must have a subnet named `pod`.

2. (Optional) Add free-form tags for the cluster resource by using the `--defined-tags` or `--freeform-tags` options.

   > **Note:**
   >
   > Don't specify values for the OraclePCA-OKE.cluster_id defined tag or for the ClusterResourceIdentifier free-form tag. These tag values are system-generated and only applied to nodes (instances), not to the cluster resource.

   Define an argument for the `--defined-tags` option to provide the following information for control plane nodes. Specify OraclePCA as the tag namespace.

> **Note:**
>
> None of these values - SSH key, number of nodes, node shape, or node shape configuration - can be set or changed after the cluster is created. If you set these tags when you update the cluster, the new values are ignored.

- Your public SSH key

  Specify `sshkey` for the tag key, and paste your public SSH key as the value.

  > **Important:**
  >
  > You can't add an SSH key after the cluster is created.

- Number of nodes

  By default, the number of nodes in the control plane is 3. You can specify 1, 3, or 5 nodes. To specify the number of control plane nodes, specify `cpNodeCount` for the tag key, and enter `1`, `3`, or `5` for the value.

- Node shape

  Specify a shape such as `VM.PCAStandard.E5.Flex`. For descriptions of available shapes, see Compute Shapes on page 301.

- Node shape configuration

  Specify the shape configuration.

  The default configuration is 1 OCPU and 10 gigabytes of memory.

  To provide shape configuration information, specify `cpNodeShapeConfig` for the tag key. You must specify the number of OCPUs (`ocpus`) you want and you can specify the total amount of memory you want (`memoryInGBs`). The default value for gigabytes of memory is 16 times the number you specify for OCPUs.

  > **Note:**
  >
  > If the cluster will have 1-10 worker nodes, specify at least 16 GB memory. If the cluster will have 11-128 worker nodes, specify at least 2 OCPUs and 32 GB memory. You can't change the number of OCPUs or amount of memory when you update the cluster.

  Specify defined tags, either inline, or in a file in JSON format, such as the following example file:

```
{
  "OraclePCA": {
    "sshkey": "ssh-rsa <remainder_of_key_text>",
    "cpNodeCount": 1,
    "cpNodeShape": "VM.PCAStandard1.Flex",
    "cpNodeShapeConfig": {
      "ocpus": 2,
      "memoryInGBs": 48
    }
  }
}
```

  Use the following syntax to specify a file of tags. Specify the full path to the `.json` file unless the file is in the same directory where you're running the command.

```
--defined-tags file://cluster_tags.json
```

3. (Optional) You can use the `--service-lb-defined-tags` or `--service-lb-freeform-tags` options to specify tags to be applied to all load balancers created by Kubernetes services. Ensure that the applicable dynamic group includes the `use tag-namespaces` policy. See Exposing Containerized Applications on page 633.
4. Run the create cluster command.

If the `--endpoint-subnet-id` that you specify is a public subnet, then a public endpoint is created, and the `--endpoint-public-ip-enabled` option must be set to `true`.

If the `--endpoint-subnet-id` that you specify is a private subnet, then a private endpoint is created, and the `--endpoint-public-ip-enabled` option must be set to `false`.

Example:

```
$ oci ce cluster create \
--compartment-id ocid1.compartment.unique_ID --kubernetes-version version
 \
--name "Native Cluster" --vcn-id ocid1.vcn.unique_ID \
--cluster-pod-network-options '{"cniType":"OCI_VCN_IP_NATIVE"}' \
--endpoint-subnet-id control-plane-endpoint_subnet_OCID \
--endpoint-public-ip-enabled false \
--service-lb-subnet-ids '["service-lb_subnet_OCID"]' \
--defined-tags '{"OraclePCA":{"sshkey":"ssh-rsa remainder_of_key_text"}}'
```

The output from this `cluster create` command is the same as the output from the `cluster get` command.

Use the `work-request get` command to check the status of the create operation. The work request OCID is in `created-by-work-request-id` in the `metadata` section of the `cluster create` output.

```
$ oci ce work-request get --work-request-id workrequest_OCID
```

To identify the control plane nodes for this cluster, list instances in the compartment where you created the cluster. Names of control plane nodes are in the following format:

```
oke-ID1-control-plane-ID2
```

- ***ID1*** - The first 32 characters after the ***pca_name*** in the cluster OCID.
- ***ID2*** - A unique identifier added when the cluster has more than one control plane node.

Search for the instances in the list whose names contain the ***ID1*** string from this cluster OCID.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

What's Next:

1. Create a Kubernetes configuration file for the cluster. See Creating a Kubernetes Configuration File on page 596.
2. Deploy a Kubernetes Dashboard to manage the cluster and to manage and troubleshoot applications running in the cluster. On the https://kubernetes.io/ site, see Deploy and Access the Kubernetes Dashboard.
3. Create a node pool for the cluster. See Creating an OKE Worker Node Pool on page 619.
4. Create a backup for the workload cluster. For example, see Backing up an etcd cluster and Restoring up an etcd cluster in Operating etcd clusters for Kubernetes. Use the etcd backup to recover OKE clusters under disaster scenarios such as losing all control plane nodes. An etcd backup contains all OKE states and critical information. An etcd backup does not back up applications or other content on cluster nodes.

Using the API

Use the CreateCluster operation to create a new cluster.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

What's Next:

1. Create a Kubernetes configuration file for the cluster. See Creating a Kubernetes Configuration File on page 596.

2. Deploy a Kubernetes Dashboard to manage the cluster and to manage and troubleshoot applications running in the cluster. On the https://kubernetes.io/ site, see Deploy and Access the Kubernetes Dashboard.
3. Create a node pool for the cluster. See Creating an OKE Worker Node Pool on page 619.
4. Create a backup for the workload cluster. For example, see Backing up an etcd cluster and Restoring up an etcd cluster in Operating etcd clusters for Kubernetes. Use the etcd backup to recover OKE clusters under disaster scenarios such as losing all control plane nodes. An etcd backup contains all OKE states and critical information. An etcd backup does not back up applications or other content on cluster nodes.

Creating a Kubernetes Configuration File

On Compute Cloud@Customer, you can set up a Kubernetes configuration file for each OKE cluster that you work with. Your Kubernetes configuration file enables you to access OKE clusters using the `kubectl` command and the Kubernetes Dashboard.

Kubernetes configuration files organize information about clusters, users, namespaces, and authentication mechanisms. You can define contexts to easily switch between clusters and namespaces. The `kubectl` tool uses Kubernetes configuration files to find the information it needs to select a cluster and communicate with the API server of a cluster.

Installing the Kubernetes Command Line Tool

Install and configure the Kubernetes command line tool `kubectl`. The `kubectl` tool enables you to perform operations on OKE clusters such as deploy applications, inspect and manage cluster resources, and view logs.

To install `kubectl`, see Kubernetes Install Tools. The `kubectl` version must be within one minor version of the OKE cluster Kubernetes version. For example, a v1.29 client can communicate with v1.28, v1.29, and v1.30 control planes. See Supported Versions of Kubernetes on page 506S.

For more information, including a complete list of `kubectl` operations, see the https://kubernetes.io/docs/reference/kubectl/ reference page.

Create a Kubernetes Configuration File

Use the CLI to create your Kubernetes configuration file.

> **Tip:**
>
> The Quick Start button on a cluster details page in the Compute Cloud@Customer Console shows how to create a Kubernetes configuration file, and provides the OCID of the cluster.

1. Get the OCID of the cluster: `oci ce cluster list`
2. Run the oci ce cluster create-kubeconfig command as described to create the configuration file:

   The `--cluster-id` option is required.

   The default value of the `--file` option is `~/.kube/config`. If you already have a file at the specified location and you want to replace it, use the `--overwrite` option. To maintain more than one configuration file, select a different file by using the `KUBECONFIG` environment variable or the `--kubeconfig` option.

   The value of the `--kube-endpoint` option must be `PUBLIC_ENDPOINT`.

   If you don't specify the `--profile` option, the current value of your `OCI_CLI_PROFILE` environment variable is used. Best practice is to specify this value.

   If provided, the value of the `--token-version` option must be 2.0.0.

   Example:

   Use the following command to configure a Kubernetes configuration file for the specified cluster using the public endpoint:

```
$ oci ce cluster create-kubeconfig --cluster-id ocid1.cluster.unique_ID \
```

```
--file $HOME/.kube/config --kube-endpoint PUBLIC_ENDPOINT --
profile profile-name
New config written to the Kubeconfig file /home/username/.kube/config
```

A Kubernetes configuration file includes an CLI command that dynamically generates an authentication token and inserts it when you run a `kubectl` command. By default, the CLI command in the Kubernetes configuration file uses your current CLI profile when generating an authentication token. If you have defined multiple profiles in your CLI configuration file, use one of the following methods to specify which profile to use when generating the authentication token. The value of *profile-name* is the name of the profile in your CLI configuration file.

- Ensure that your `OCI_CLI_PROFILE` environment variable is set to the profile for the tenancy where the `ocid1.cluster.unique_ID` resides. This setting is ignored if one of the following methods was used to specify the profile for this cluster in the Kubernetes configuration file.
- Specify the `--profile` option on the `create-kubeconfig` command line as shown in the preceding example command.
- Edit the generated configuration file as shown in the following example.

```
user:
  exec:
    apiVersion: client.authentication.k8s.io/v1beta1
    args:
    - ce
    - cluster
    - generate-token
    - --cluster-id
    - cluster ocid
    - --profile
    - profile-name
    command: oci
    env: []
```

Use the following command to set your `KUBECONFIG` environment variable to the Kubernetes configuration file that you created or updated in the preceding command:

```
$ export KUBECONFIG=$HOME/.kube/config
```

The following command shows the content of your new YAML configuration file:

```
$ kubectl config view
```

If you run the command again with a different cluster OCID, the new information is merged with the existing information. The following message is displayed:

```
Existing Kubeconfig file found at /home/username/.kube/config and new
 config merged into it
```

Verify Your Cluster Access

Before you run `kubectl` commands, enure that your `OCI_CLI_PROFILE` environment variable is set to the name of the profile that is defined in your OCI configuration file:

```
$ export OCI_CLI_PROFILE=profile-name
```

Run the following command to confirm that you can access your cluster:

```
$ kubectl cluster-info
```

Every Kubernetes namespace contains at least one ServiceAccount: the default ServiceAccount for that namespace, which is named `default`. If you don't specify a ServiceAccount when you create a Pod, the OKE service automatically assigns the ServiceAccount named `default` in that namespace.

An application running inside a Pod can access the Kubernetes API using automatically mounted service account credentials.

Create a Kubernetes Dashboard

The dashboard helps you manage the cluster and manage and troubleshoot applications running in the cluster.

On the Kubernetes site, see Deploy and Access the Kubernetes Dashboard.

What's Next:

Creating an OKE Worker Node Pool on page 619

Updating an OKE Cluster

On Compute Cloud@Customer, when you update an OKE cluster, you can change the cluster name, Kubernetes version, and tags.

Best practice is to keep your clusters upgraded so that they're always running versions of Kubernetes that are currently supported by OKE. See the instructions in the following procedures to determine whether a newer supported version of Kubernetes is available.

> **Note:**
>
> If you set or modify any of the following tags, the new values are ignored: SSH key (OraclePCA.sshkey), number of nodes (OraclePCA.cpNodeCount), node shape (OraclePCA.cpNodeShape), or node shape configuration (OraclePCA.cpNodeShapeConfig). These values can be set only when you create the cluster.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Containers**, then select **Kubernetes Clusters (OKE)**.
2. In the clusters list, if a Kubernetes version update is available, an exclamation point icon is displayed next to the Kubernetes Version number. To upgrade to a newer version, select the Actions menu (⋮) and select **Upgrade Available**. Select a new version from the drop-down menu.

   Alternatively, on the cluster details page, select **Upgrade Available** next to the Kubernetes Version number, or select **Upgrade** at the top of the page.
3. Select the name of the cluster that you want to update.
4. At the top of the cluster details page, select **Edit**.

   Don't specify values for the OraclePCA-OKE.cluster_id defined tag or for the ClusterResourceIdentifier free-form tag. These tag values are system-generated and only applied to nodes (instances), not to the cluster resource.
5. When you're finished making changes, select **Save Changes**.

Using the CLI

Use the oci ce cluster update command and required parameters to update the details of a cluster.

```
oci ce cluster update --cluster-id cluster_OCID [OPTIONS]
```

1. Get the OCID of the cluster that you want to update: `oci ce cluster list`
2. Check whether a newer version of Kubernetes is available.

   Run the get cluster command: `oci ce cluster get`

If the value of `available-kubernetes-upgrades` isn't the empty set, specify one of the listed versions as the `--kubernetes-version` in the update cluster command.

3. Run the update cluster command.

   If you specify the `--defined-tags` or `--freeform-tags` options, don't specify values for the OraclePCA-OKE.cluster_id defined tag or for the ClusterResourceIdentifier free-form tag. These tag values are system-generated and only applied to nodes (instances), not to the cluster resource.

   Example:

   ```
   $ oci ce cluster update --cluster-id ocid1.cluster.unique_ID \
   --kubernetes-version newer_kubernetes_version --name new_cluster_name
   ```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateCluster operation to update the details of a cluster.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting an OKE Cluster

Learn how to delete OKE clusters On Compute Cloud@Customer.

On Private Cloud Appliance, deleting an OKE cluster deletes the cluster control plane nodes, worker nodes, and node pools. Other cluster resources such as VCNs, internet gateways, NAT gateways, route tables, security lists, load balancers, and block volumes aren't deleted when you delete the cluster. Those resources must be deleted separately.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Containers**, then select **Kubernetes Clusters (OKE)**.
2. For the cluster that you want to delete, select the Actions menu (⋮), and select **Delete**.
3. Confirm that you want to delete the cluster.

   Enter the cluster name, and select **Delete**.

Using the CLI

Use the oci ce cluster delete command and required parameters to delete a cluster.

```
oci ce cluster delete --cluster-id cluster_OCID [OPTIONS]
```

Procedure

1. Get the OCID of the cluster that you want to delete: `oci ce cluster list`
2. Run the delete cluster command.

   Example:

   ```
   $ oci ce cluster delete --cluster-id ocid1.cluster.unique_ID --force
   ```

   The `--force` option performs the deletion without prompting for confirmation.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteCluster operation to delete a cluster.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Managing OKE Cluster Add-ons

On Compute Cloud@Customer, cluster add-ons are components that you can choose to deploy on a Kubernetes cluster. Cluster add-ons extend core Kubernetes functionality and improve cluster manageability and performance. This section describes how to install and manage the following supported add-ons.

- The WebLogic Kubernetes Operator add-on, which supports running WebLogic Server and Fusion Middleware infrastructure domains on Kubernetes. For more information about the WebLogic Kubernetes Operator, see the public documentation at https://github.com/oracle/weblogic-kubernetes-operator.
- The Database Operator for Kubernetes add-on (OraOperator) helps developers, database administrators, DevOps, and GitOps teams reduce the time and complexity of deploying and managing Oracle databases. For more information, see the public documentation at https://github.com/oracle/oracle-database-operator/tree/main. For guidance on how to deploy and configure the add-on, we recommend this solution paper: Deploying Oracle Database Operator Cluster Add-on for Oracle Kubernetes Engine (OKE).
- The NVIDIA GPU Plugin add-on is a convenient way to manage the NVIDIA Device Plugin for Kubernetes. The NVIDIA Device Plugin for Kubernetes is the NVIDIA implementation of the Kubernetes device plug-in framework for exposing the number of NVIDIA GPUs on each worker node, and tracking the health of those GPUs. For more information about the NVIDIA Device Plugin for Kubernetes, see https://github.com/NVIDIA/k8s-device-plugin.
- The optional Certificate Manager add-on, also known as cert-manager, adds certificates and certificate issuers to Kubernetes clusters as resource types. The Certificate Manager also simplifies the process of obtaining, using, and renewing those certificates. For more information, see https://github.com/cert-manager/cert-manager.

Add-on Prerequisites

Review the following requirements before you install the add-ons.

Oracle Database Operator add-on

- Database Operator can only be enabled on an existing cluster.
- The Certificate Manager add-on must be installed, enabled, and in the ACTIVE state before you can use the Database Operator add-on. See Install the Add-on for an Existing Cluster on page 601.

Nvidia GPU Plugin add-on

- For both the cluster and node pool, Kubernetes version 1.29.14 or higher is required
- Enable the Nvidia GPU plugin *before* creating a GPU node pool.
- Once a node pool is created as either GPU or non-GPU, its type cannot be switched.
- Use the appropriate required base image for your use case.

Base Image	Use Case
`nvcr.io/nvidia/cuda:12.9.0-runtime-ubi9` or equal as a runtime environment	Deploying of a prebuilt AI application
`nvcr.io/nvidia/cuda:12.9.0-devel-ubi9` or equal as a development environment	Developing, compiling, or training AI with native CUDA/C++ code.

- All worker nodes come with default block volume size of 50 GB however, GPU applications when used with the required base image might run out of disk space. Ensure to use Persistent Volume Claim in the GPU applications.

- To avoid unintended scheduling, the GPU nodes are tainted by default. Without this taint any pod could be scheduled onto a GPU node, even if it doesn't need a GPU. This means only pods that explicitly request GPU resources and tolerate the taint are be scheduled on GPU nodes.
- To ensure that the Nvidia device-plugin pods are scheduled and run only on nodes that possess NVIDIA GPU hardware, GPU nodes are labeled by default. The OKE controller labels GPU nodes with the following node label `"nvidia.com/gpu": "true"`.

WebLogic Kubernetes Operator add-on

See Deploying the WebLogic Kubernetes Operator Add-on on page 604.

Install an Add-on When You Create a Cluster

To install an add-on when you create a cluster, you must use the Console.

Review the Add-on Prerequisites on page 600 before you begin.

Console

1. In the Compute Cloud@Customer Console navigation menu, select **Containers**, then select **Kubernetes Clusters (OKE)**.
2. On the clusters list page, select the **Create Cluster** button.
3. On the bottom of the first page of the **Create Cluster** dialog box, the **Add-ons** section shows the available cluster add-ons. In the **Create Cluster** dialog box, all add-ons are **Disabled**.

   > **Note:**
   >
   > Oracle Database Operator add-on can only be enabled on an existing cluster.

4. Select the add-on you want to deploy and configure.
   a. **Enable**: Select the checkbox for the add-on you want to deploy and enable for this cluster.
   b. **Add-on version updates**: Select how you want the version of the add-on to be updated as newer versions of the add-on become available and as newer versions of Kubernetes are supported for OKE. Select either **Automatic Updates** or select a **Version**. See descriptions of these options in Version Updates for Add-ons on page 609.

      If you select **Choose a Version**, then you must select a version from the list.
   c. **Configurations**: Select **Add configuration** to select a configuration option and specify a value.

      Select **Add configuration** to set another configuration parameter.

Install the Add-on for an Existing Cluster

On Compute Cloud@Customer, you can install an add-on for an existing cluster.

Outside Certificates

To install the WebLogic Kubernetes Operator add-on on an existing cluster that is using a certificate that's not the certificate that's specific to the Compute Cloud@Customer, perform the following steps on the cluster where you want to install the add-on:

1. Perform certificate rotation. See Certificate Authority Bundles on page 513.
2. Perform any updates to node pool configuration that are required, such as boot volume size change or shape changes, for example.
3. Cycle worker nodes as described in Node Cycling an OKE Node Pool on page 629.
4. Enable or install the WebLogic Kubernetes Operator add-on as described in this procedure.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Containers**, then select **Kubernetes Clusters (OKE)**.

2. In the clusters list, select the name of the cluster in which you want to install the add-on.
3. On the cluster details page, under **Resources**, select **Add-ons**.
4. In the add-ons list, for the **WLS Operator** add-on, select the Actions menu (⋮), and select **Edit**. On the **WLS Operator** dialog box, select the **Enable Add-on WLS Operator** checkbox to do one of the following:
   - Deploy and enable the WebLogic Kubernetes Operator add-on if the add-on hasn't been enabled on this cluster before.
   - Enable the WebLogic Kubernetes Operator add-on if the add-on was previously deployed for this cluster but is currently disabled.
5. Configure the add-on.
   a. **Add-on version updates**: Select the method you want to use to update the version of the add-on as newer versions of the add-on become available and as newer versions of Kubernetes are supported for OKE: either **Automatic Updates** or **Choose a Version**. See descriptions of these options in Version Updates for Add-ons on page 609.

      If you select **Choose a Version**, then you must select a version from the list.
   b. **Add-on configuration**: Select **Add configuration** to select a configuration option and specify a value. See the descriptions in Configuration Parameters for Add-ons on page 615.

      To set another configuration parameter, select **Add configuration**.
6. Select **Save Changes**.

Using the CLI

Use the oci ce cluster install-addon command and required parameters to install the add-on.

```
oci ce cluster install-addon --cluster-id cluster_OCID --addon-
name addon_name [OPTIONS]
```

Procedure

1. Get the OCID of the cluster for which you want to install an add-on: `oci ce cluster list`
2. (Optional) Construct an argument for the `--configurations` option.

   Use the `--configurations` option to specify one or more key/value pairs in JSON format to pass as arguments to the cluster add-on.

   For descriptions of the configuration parameters, see Configuration Parameters for Add-ons on page 615.

   The inline syntax is shown in the example in the next step of this procedure. You might find it easier to use a file:

   ```
   --configurations file://./weblogic-cfg.json
   ```

   The format and content of the configuration file is given by the following command:

   ```
   $ oci ce cluster install-addon --generate-param-json-input configurations
   [
     {
       "key": "string",
       "value": "string"
     },
     {
       "key": "string",
       "value": "string"
     }
   ]
   ```

   In the following example, both `requests` and `limits` are specified because the memory limit is lower than the default memory request. If a limit is less than the corresponding request, the deployment fails.

Double quotation marks within a value must be escaped with a single backslash.

```
[
    {
        "key": "weblogic-operator.ContainerResources",
        "value": "{
            \"requests\": {
                \"cpu\": \"250m\",
                \"memory\": \"150Mi\"
            },
            \"limits\": {
                \"cpu\": \"500m\",
                \"memory\": \"200Mi\"
            }
        }"
    },
    {
        "key": "weblogic-operator-webhook.ContainerResources",
        "value": "{
            \"limits\": {
                \"cpu\": \"150m\",
                \"memory\": \"200Mi\"
            }
        }"
    },
    {
        "key": "numOfReplicas",
        "value": "1"
    }
]
```

3. Run the install add-on command.

   Syntax:

   ```
   $ oci ce cluster install-addon --cluster-id cluster_OCID \
   --addon-name addon_name
   ```

   Example:

   If you specify a version, you're selecting the `Stay on the specific version` option for updating the add-on version, described in Version Updates for Add-ons on page 609. If you set the version to null, or you omit the `--version-parameterconflict` option, you're selecting the default behavior `Automatically update the add-on`.

   Note that the version string must begin with a `v`.

   Enclose the configurations argument in single quotation marks so that you don't need to escape every double quotation mark in the argument value.

   ```
   $ oci ce cluster install-addon --cluster-id ocid1.cluster.unique_ID \
   --addon-name WeblogicKubernetesOperator --version-parameterconflict
    "v4.2.13" \
   --configurations '[{"key": "weblogic-operator.ContainerResources",
    "value": "{\"limits\": {\"cpu\": \"500m\", \"memory\": \"512Mi\"}}"}, \
   {"key": "weblogic-operator-webhook.ContainerResources", "value":
    "{\"limits\": {\"cpu\": \"150m\", \"memory\": \"200Mi\"}}"}]'

   {
    "opc-work-request-id": "ocid1.cccworkrequest.unique_ID"
   }
   ```

Using the API

Use the InstallAddon operation to install an add-on.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deploying the WebLogic Kubernetes Operator Add-on

On Compute Cloud@Customer, you can enable the WebLogic Kubernetes Operator add-on when you create a cluster or for an existing cluster.

> **Note:**
>
> To bring the WebLogic Server to the running state, create additional rules in separate WebLogic Server security lists for the control plane and worker subnets, and for the pod subnet if you're using VCN-Native Pod Networking. See Specify Ports Required by WebLogic Server on page 605.

Add-on installation remains in Accepted state and waits until the cluster is in the Active state.

After the cluster is in the Active state, the WebLogic Kubernetes Operator is in Needs Attention state until a node pool is created for the cluster.

When a node-pool has been created for the cluster, the add-on is reconciled, and the add-on is in Ready state unless some other problem exists. See Add-on Reconciliation on page 609.

> **Note:**
>
> Enabling the WebLogic Kubernetes Operator add-on on a VCN-Native Pod Networking cluster requires an entry for 169.254.169.254 in `crio-noproxy` node metadata for the nodepools where the add-on pods might be scheduled. See "Proxy settings" in the CLI procedure in Creating an OKE Worker Node Pool on page 619.

> **Note:**
>
> The WebLogic Kubernetes Operator add-on is not supported on GPU-only node pools.

- Step 1: Check the Cluster's Certificate Authority on page 604
- Step 2: Install the Add-on on page 605
- Step 3: Specify Ports Required by WebLogic Server on page 605
- Step 4: Add-on Reconciliation on page 609

Before you install the WebLogic Kubernetes Operator add-on on an existing cluster, check the cluster's certificate authority.

Check the Cluster's Certificate Authority

To install the WebLogic Kubernetes Operator add-on on an existing cluster that is using a certificate that's not the certificate that's specific to the Compute Cloud@Customer, perform the following steps on the cluster where you want to install the add-on:

1. Perform certificate rotation. See Certificate Authority Bundles on page 513.
2. Perform any updates to node pool configuration that are required, such as boot volume size change or shape changes, for example.
3. Cycle worker nodes as described in Node Cycling an OKE Node Pool on page 629.
4. Enable or install the WebLogic Kubernetes Operator add-on as described in this procedure.

Install the Add-on

See Install an Add-on When You Create a Cluster on page 601 or Install the Add-on for an Existing Cluster on page 601.

Specify Ports Required by WebLogic Server

This section describes additional security rules needed to specify ports that are required to bring the WebLogic Server to the running state.

Create additional rules in separate WebLogic Server security lists for the control plane and worker subnets, and for the pod subnet if you are using VCN-Native Pod Networking.

The following rules are for the control plane subnet. These rules are used for both Flannel Overlay networking and VCN-Native Pod Networking.

WebLogic Server Security Rules for the Control Plane Subnet

Console property	CLI property

Ingress Rule 1
• Stateless: uncheck the box
• Ingress CIDR: ***kmi_cidr***
• IP Protocol: TCP
  • Destination Port Range: 8084
• Description: "This service port is the default for the WebLogic Server Console and is used to manage WebLogic Server domains."
	Ingress Rule 1
	• `isStateless`: `false`
	• `source`: ***kmi_cidr***
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `8084`
	  • `min`: `8084`
	• `description`: "This service port is the default for the WebLogic Server Console and is used to manage WebLogic Server domains."

Ingress Rule 2
• Stateless: uncheck the box
• Ingress CIDR: ***worker_cidr***
• IP Protocol: UDP
  • Destination Port Range: 8472
• Description: "WebLogic Server administration."
	Ingress Rule 2
	• `isStateless`: `false`
	• `source`: ***worker_cidr***
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `17`
	• `udpOptions`
	`destinationPortRange`
	  • `max`: `8472`
	  • `min`: `8472`
	• `description`: "WebLogic Server administration."

The following rules are for the worker subnet. These rules are used for both Flannel Overlay networking and VCN-Native Pod Networking.

WebLogic Server Security Rules for the Worker Subnet

Console property	CLI property

Ingress Rule 1
• Stateless: uncheck the box
• Ingress CIDR: ***kmi_cidr***
• IP Protocol: TCP
  • Destination Port Range: 8084
• Description: "This service port is the default for the WebLogic Server Console and is used to manage WebLogic Server domains."
	Ingress Rule 1
	• `isStateless`: `false`
	• `source`: ***kmi_cidr***
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `8084`
	  • `min`: `8084`
	• `description`: "This service port is the default for the WebLogic Server Console and is used to manage WebLogic Server domains."

Ingress Rule 2
• Stateless: uncheck the box
• Ingress CIDR: ***kmi_cidr***
• IP Protocol: UDP
  • Destination Port Range: 8472
• Description: "WebLogic Server administration."
	Ingress Rule 2
	• `isStateless`: `false`
	• `source`: ***kmi_cidr***
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `17`
	• `udpOptions`
	`destinationPortRange`
	  • `max`: `8472`
	  • `min`: `8472`
	• `description`: "WebLogic Server administration."

Ingress Rule 3
• Stateless: uncheck the box
• Ingress CIDR: ***worker_cidr***
• IP Protocol: UDP
  • Destination Port Range: 7001-9000
• Description: "These ports are used by WebLogic Server."
	Ingress Rule 3
	• `isStateless`: `false`
	• `source`: ***worker_cidr***
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `17`
	• `udpOptions`
	`destinationPortRange`
	  • `max`: `9000`
	  • `min`: `7001`
	• `description`: "These ports are used by WebLogic Server."

The following rules are for the pod subnet. These rules are used for VCN-Native Pod Networking.

WebLogic Server Security Rules for the Pod Subnet

Console property	CLI property

Ingress Rule 1
• Stateless: uncheck the box
• Ingress CIDR: ***kmi_cidr***
• IP Protocol: TCP
  • Destination Port Range: 8084
• Description: "This service port is the default for the WebLogic Server Console and is used to manage WebLogic Server domains."
	Ingress Rule 1
	• `isStateless`: `false`
	• `source`: ***kmi_cidr***
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `8084`
	  • `min`: `8084`
	• `description`: "This service port is the default for the WebLogic Server Console and is used to manage WebLogic Server domains."

Ingress Rule 2
• Stateless: uncheck the box
• Ingress CIDR: ***worker_cidr***
• IP Protocol: UDP
  • Destination Port Range: 8472
• Description: "WebLogic Server administration."
	Ingress Rule 2
	• `isStateless`: `false`
	• `source`: ***worker_cidr***
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `17`
	• `udpOptions`
	`destinationPortRange`
	  • `max`: `8472`
	  • `min`: `8472`
	• `description`: "WebLogic Server administration."

Viewing OKE Cluster Add-ons

On Compute Cloud@Customer, you can list all add-ons for a cluster and view more information about a specified cluster add-on.

To list versions of an add-on that are supported on a specific version of Kubernetes, use the CLI, or use the **Edit Add-on** option in the Compute Cloud@Customer Console.

Using the Console

1. On the dashboard, select **Containers / View Kubernetes Clusters (OKE)**.
2. In the clusters list, select the name of the cluster for which you want to view add-ons.
3. On the cluster details page, under **Resources**, select **Add-ons**.

   The list table shows the name of the cluster add-on, whether automatic updates are enabled, the lifecycle state of the add-on (for example, Active, Updating, Disabled), and the current version of the add-on.
4. To show more information, select the Actions menu (⋮), and select **Edit Add-on**.

   On the dialog box, you can view the current add-on configuration and change the configuration. You can enable the add-on, select either **Automatically Update** or **Choose a version**, and specify configuration parameters.

Using the CLI

Use the oci ce cluster list-addons command and required parameters to view add-ons.

```
oci ce cluster list-addons --cluster-id <cluster_OCID>
```

Use the oci ce cluster get-addon command and required parameters to view details for an add-on.

```
oci ce cluster get-addon --cluster-id <cluster_OCID> --addon-
name <addon_name>
```

Procedure

1. Get the OCID of the cluster for which you want to view add-ons: `oci ce cluster list`
2. List all add-ons in the specified cluster.

   ```
   $ oci ce cluster list-addons --cluster-id ocid1.cluster.unique_ID
   ```

   For each add-on that is available to the specified cluster, the output shows the name of the add-on, the current installed version, and the lifecycle state.
3. For more information about a specific add-on, run the get add-on command.

   The output is the same as for the `list-addons` command except that `get-addon` also shows the configuration values.

   If the add-on lifecycle state is NEEDS_ATTENTION, see Add-on Reconciliation on page 609.

   Example:

   ```
   $ oci ce cluster get-addon --cluster-id ocid1.cluster.unique_ID \
   --addon-name WeblogicKubernetesOperator
   {
     "data": {
       "addon-error": {
           "code": null,
           "message": null,
           "status": null
       },
       "configurations": [
         {
           "key": "numOfReplicas",
           "value": "0"
         },
         {
           "key": "weblogic-operator.ContainerResources",
           "value": "{'limits': {'cpu': '500m', 'memory': '200Mi'}}"
         },
         {
           "key": "weblogic-operator-webhook.ContainerResources",
           "value": "{'limits': {'cpu': '200m', 'memory': '300Mi'}}"
         }
       ],
       "current-installed-version": "v4.2.13",
       "lifecycle-state": "ACTIVE",
       "name": "WeblogicKubernetesOperator",
       "time-created": "2025-02-26T01:41:52.020696+00:00",
       "version": null
     },
     "etag": "5f3eef22-eb32-5c2c-774c-c7a98836a13a"
   }
   ```

Using the API

Use the ListAddons operation to list add-ons.

Use the GetAddon operation to get add-on details.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Version Updates for Add-ons

When you enable a cluster add-on On Compute Cloud@Customer, specify how you want to update the add-on version when new versions become available.

(Default) Automatic Updates

If possible, this option automatically updates the add-on when new versions become available. This is the default behavior.

The newest version of the add-on that supports the Kubernetes version that is specified for the cluster is deployed when you install the add-on. When a newer version of the add-on is released, the add-on is automatically updated if the new add-on version is compatible with the versions of Kubernetes that are supported by OKE at that time and the version of Kubernetes that the cluster is running.

Best practice is to keep your clusters upgraded so that they are always running versions of Kubernetes that are listed as currently supported by OKE. See *Supported Versions of Kuberrnetes* in Kubernetes Engine (OKE) on Compute Cloud@Customer on page 505 and Updating an OKE Cluster on page 598.

Choose a Version

This option keeps the add-on on the specific version that you select until you change it.

If you specify that you want to choose the version of the add-on to deploy, the version you choose is enabled. Ensure that the add-on version is compatible with the Kubernetes version that you have selected for the cluster or that is already running on the cluster.

When you use the Console, you select the version from a list. All versions on the list are compatible with the Kubernetes version that you have selected for the cluster or that is already running on the cluster.

When you use the CLI, use the following `addon-option list` commands to get the information you need before you run the `cluster install-addon` command.

List available versions of all cluster add-ons that are supported on the specified Kubernetes version.

```
$ oci ce addon-option list --kubernetes-version v1.29.9
```

List available versions of the specified cluster add-on that are supported on the specified Kubernetes version.

```
$ oci ce addon-option list --kubernetes-version v1.29.9 --addon-name
 WeblogicKubernetesOperator
```

Add-on Reconciliation

On Compute Cloud@Customer, the OKE service includes a reconciliation process that periodically evaluates the state of the add-on and updates the add-on if necessary.

> **Note:**
>
> You should not install, configure, update, or delete add-ons manually. Use the OKE service installation, configuration, update, and delete interfaces.

The reconciliation process behaves differently depending on the state of the add-on.

Active State

For add-ons in ACTIVE state, the reconciliation process runs every twelve hours. If resources have been manually deleted, the process detects the change and attempts recovery. Full recovery is not guaranteed.

If full recovery is successful, the add-on state returns to ACTIVE.

If recovery is partial or fails, the add-on state changes to NEEDS_ATTENTION.

Needs Attention State

For add-ons in NEEDS_ATTENTION state, the reconciliation process runs every few minutes, not every twelve hours, while the add-on remains in the NEEDS_ATTENTION state. The process checks whether all deployments associated with the add-on are ready, whether all pods are healthy. The interval between reconciliation process runs is a little longer each time.

Some issues, such as unschedulable nodes, might resolve during the reconciliation process. Other issues, such as configuration problems, require user intervention to fix. If the add-on remains in NEEDS_ATTENTION state after the reconciliation process has run, try to identify the issues. Check the K8s_app application in Grafana or check the state of the add-on operator manually.

Recovery actions that might be appropriate for a user to take include the following:

- Ensure that node pools in the cluster have at least one node available. If no nodes are available, the add-on pods can't be scheduled, the add-on can't be deployed.
- Ensure the configuration values and other settings are correct.
- Update the add-on as needed. Updating the add-on will trigger another reconciliation process.
- Disable and reinstall the add-on.

If the add-on recovers, the add-on is transitioned back to ACTIVE state.

If the add-on remains unhealthy, the system schedules the next check.

The reconciliation process continues to run for approximately 12.5 hours if the add-on remains in NEEDS_ATTENTION state. After the reconciliation process stops running, the add-on remains in NEEDS_ATTENTION state indefinitely. After approximately 30 minutes, the work request moves to FAILED state.

The following example shows an add-on in NEEDS_ATTENTION state.

```
$ oci ce cluster get-addon --cluster-id ocid1.cluster.unique_ID --addon-name
 WeblogicKubernetesOperator
{
  "data": {
    "addon-error": {
      "code": "409",
      "message": "Incorrect state for CR",
      "status": "IncorrectState"
    },
    "configurations": [
      {
        "key": "numOfReplicas",
        "value": "0"
      },
      {
        "key": "weblogic-operator.ContainerResources",
        "value": "{'limits': {'cpu': '500m', 'memory': '200Mi'}}"
      },
      {
        "key": "weblogic-operator-webhook.ContainerResources",
        "value": "{'limits': {'cpu': '200m', 'memory': '300Mi'}}"
      }
    ],
    "current-installed-version": "",
    "lifecycle-state": "NEEDS_ATTENTION",
```

```
    "name": "WeblogicKubernetesOperator",
    "time-created": "2025-02-26T01:41:52.020696+00:00",
    "version": null
  },
  "etag": "5f3eef22-eb32-5c2c-774c-c7a98836a13a"
}
```

Ports Required by WebLogic Server

This topic describes additional security rules needed to specify ports that are required to bring the WebLogic Server to the running state.

Create additional rules in separate WebLogic Server security lists for the control plane and worker subnets, and for the pod subnet if you are using VCN-Native Pod Networking.

The following rules are for the control plane subnet. These rules are used for both Flannel Overlay networking and VCN-Native Pod Networking.

WebLogic Server Security Rules for the Control Plane Subnet

Console property	CLI property

Ingress Rule 1
• Stateless: uncheck the box
• Ingress CIDR: ***kmi_cidr***
• IP Protocol: TCP
  • Destination Port Range: 8084
• Description: "This service port is the default for the WebLogic Server Console and is used to manage WebLogic Server domains."
	Ingress Rule 1
	• `isStateless`: `false`
	• `source`: ***kmi_cidr***
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `8084`
	  • `min`: `8084`
	• `description`: "This service port is the default for the WebLogic Server Console and is used to manage WebLogic Server domains."

Ingress Rule 2
• Stateless: uncheck the box
• Ingress CIDR: ***worker_cidr***
• IP Protocol: UDP
  • Destination Port Range: 8472
• Description: "WebLogic Server administration."
	Ingress Rule 2
	• `isStateless`: `false`
	• `source`: ***worker_cidr***
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `17`
	• `udpOptions`
	`destinationPortRange`
	  • `max`: `8472`
	  • `min`: `8472`
	• `description`: "WebLogic Server administration."

The following rules are for the worker subnet. These rules are used for both Flannel Overlay networking and VCN-Native Pod Networking.

WebLogic Server Security Rules for the Worker Subnet

Console property	CLI property

Ingress Rule 1
• Stateless: uncheck the box
• Ingress CIDR: ***kmi_cidr***
• IP Protocol: TCP
  • Destination Port Range: 8084
• Description: "This service port is the default for the WebLogic Server Console and is used to manage WebLogic Server domains."
	Ingress Rule 1
	• `isStateless`: `false`
	• `source`: ***kmi_cidr***
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `8084`
	  • `min`: `8084`
	• `description`: "This service port is the default for the WebLogic Server Console and is used to manage WebLogic Server domains."

Ingress Rule 2
• Stateless: uncheck the box
• Ingress CIDR: ***kmi_cidr***
• IP Protocol: UDP
  • Destination Port Range: 8472
• Description: "WebLogic Server administration."
	Ingress Rule 2
	• `isStateless`: `false`
	• `source`: ***kmi_cidr***
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `17`
	• `udpOptions`
	`destinationPortRange`
	  • `max`: `8472`
	  • `min`: `8472`
	• `description`: "WebLogic Server administration."

Ingress Rule 3
• Stateless: uncheck the box
• Ingress CIDR: ***worker_cidr***
• IP Protocol: UDP
  • Destination Port Range: 7001-9000
• Description: "These ports are used by WebLogic Server."
	Ingress Rule 3
	• `isStateless`: `false`
	• `source`: ***worker_cidr***
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `17`
	• `udpOptions`
	`destinationPortRange`
	  • `max`: `9000`
	  • `min`: `7001`
	• `description`: "These ports are used by WebLogic Server."

The following rules are for the pod subnet. These rules are used for VCN-Native Pod Networking.

WebLogic Server Security Rules for the Pod Subnet

Console property	CLI property

Ingress Rule 1
• Stateless: uncheck the box
• Ingress CIDR: ***kmi_cidr***
• IP Protocol: TCP
  • Destination Port Range: 8084
• Description: "This service port is the default for the WebLogic Server Console and is used to manage WebLogic Server domains."
	Ingress Rule 1
	• `isStateless`: `false`
	• `source`: ***kmi_cidr***
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `6`
	• `tcpOptions`
	`destinationPortRange`
	  • `max`: `8084`
	  • `min`: `8084`
	• `description`: "This service port is the default for the WebLogic Server Console and is used to manage WebLogic Server domains."

Ingress Rule 2
• Stateless: uncheck the box
• Ingress CIDR: ***worker_cidr***
• IP Protocol: UDP
  • Destination Port Range: 8472
• Description: "WebLogic Server administration."
	Ingress Rule 2
	• `isStateless`: `false`
	• `source`: ***worker_cidr***
	• `sourceType`: `CIDR_BLOCK`
	• `protocol`: `17`
	• `udpOptions`
	`destinationPortRange`
	  • `max`: `8472`
	  • `min`: `8472`
	• `description`: "WebLogic Server administration."

Updating Cluster Add-ons

On Compute Cloud@Customer, you can configure the method for updating the cluster add-on version, and you can update the number of replicas and the resource usage limits.

The add-on update remains in Accepted state and waits until the cluster is in the Active state.

Using the Console

1. On the dashboard, select **Containers / View Kubernetes Clusters (OKE)**.
2. In the clusters list, select the name of the cluster for which you want to update an add-on.
3. On the cluster details page, Under **Resources**, select **Add-ons**.
4. In the add-ons list, locate the add-on you want to modify, select the Actions menu (⋮), and select **Edit Add-on**.
5. In the dialog box that opens, configure the add-on:
   a. **Add-on version updates**: Select how you want the version of the add-on to be updated as newer versions of the add-on become available and as newer versions of Kubernetes are supported for OKE: either **Automatic Updates** or **Choose a Version**. See descriptions of these options in Version Updates for Add-ons on page 609.

      If you select **Choose a Version**, you must select a version from the list.
   b. **Add-on configuration**: Optionally select a configuration parameter and a value for that parameter.

      To set another configuration parameter, select the Add configuration button.
6. Select the Save Changes button in the dialog.

Using the CLI

Use the oci ce cluster update-addon command and required parameters to update addon details for a cluster.

```
oci ce cluster update-addon --addon-name <addon_name> --cluster-
id <cluster_OCID> [OPTIONS]
```

Procedure

1. Get the OCID of the cluster for which you want to update an add-on: `oci ce cluster list`
2. Run the update add-on command.

   Example:

   ```
   $ oci ce cluster update-addon --cluster-id ocid1.cluster.unique_ID \
   --addon-name WeblogicKubernetesOperator \
   --configurations file://./weblogic-cfg.json --force

   {
    "opc-work-request-id": "ocid1.cccworkrequest.unique_ID"
   }
   ```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateAddon operation to update addon details for a cluster.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Disabling and Removing OKE Cluster Add-ons

On Compute Cloud@Customer, when you disable an add-on, it's disabled and deleted from the cluster. If you later enable the add-on, the add-on is reinstalled.

The add-on must be in the Active state to be deleted.

Using the Console

1. On the dashboard, select **Containers / View Kubernetes Clusters (OKE)**.
2. In the clusters list, select the name of the cluster for which you want to delete the add-on.
3. On the cluster details page, under **Resources**, select **Add-ons**.
4. In the add-ons list, select the Actions menu (⋮), next to the appropriate add-on, then select **Edit**.
5. In the edit dialog, clear **Enable Add-on** to disable and remove the selected add-on for this cluster.
6. Select **Save Changes**.

Using the CLI

Use the oci ce cluster disable-addon command and required parameters to disable and remove an add-on.

```
oci ce cluster disable-addon --addon-name <addon_name> --cluster-
id <cluster_OCID> [OPTIONS]
```

Procedure

1. Get the OCID of the cluster that you want to delete: `oci ce cluster list`
2. Run the disable add-on command.

   Example:

   ```
   $ oci ce cluster disable-addon --cluster-id ocid1.cluster.unique_ID --
   addon-name WeblogicKubernetesOperator
   ```

The add-on is disabled and removed from the cluster:

Note that the `--is-remove-existing-add-on` option is ignored.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DisableAddon operation to disable and remove an add-on.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Configuration Parameters for Add-ons

This topic describes the configuration parameters that are available On Compute Cloud@Customer for cluster add-ons.

WebLogic Kubernetes Operator add-on parameters

When you enable the WebLogic Kubernetes Operator cluster add-on, you can pass the following key/value pairs as arguments.

> **Note:**
>
> For weblogic-operator container resources and weblogic-operator-webhook container resources, if you set a limit without specifying a request, and the limit is less than the default request, the deployment fails.

Use the values of the weblogic-operator container resources parameter and the weblogic-operator-webhook container resources parameter to determine the maximum number of replicas you can specify. The values of these parameters could be the default values shown in the table below or different values that you requested when you enabled the WebLogic Kubernetes Operator.

Parameter Name **Console** **CLI**	**Description**
numOfReplicas `numOfReplicas`	(Required) The integer number of replicas of the add-on deployment.
weblogic-operator container resources `weblogic-operator.ContainerResources`	(Optional) These are resource values for the main WebLogic Operator container. The resource quantities that the add-on containers request, and the resource usage limits that the add-on containers cannot exceed. See Resource Management for Pods and Containers in the Kubernetes documentation. If you do not specify a request, the default request values are: • cpu: 250m • memory: 512Mi If you set a usage limit, you must set a limit equal to or greater than these default resource request values or the request values that you specified.

Parameter Name **Console** **CLI**	Description
weblogic-operator-webhook container resources `weblogic-operator-webhook.ContainerResources`	(Optional) These are resource values for the webhook container used by the operator. The resource quantities that the add-on containers request, and the resource usage limits that the add-on containers cannot exceed. If you do not specify a request, the default request values are: • cpu: 100m • memory: 100Mi If you set a usage limit, you must set a limit equal to or greater than these default resource request values or the request values that you specified.

Example

A worker node with 6 Gb RAM and 2 OCPUs could accommodate 12 pods if you don't count other cluster-related or custom pods. Each WebLogic Server pod could require 250m/768Mi (cpu/memory), and Flannel Overlay pods could require 100m/50Mi. Best practice is not to exceed 70% of CPU/memory usage per node. Considering only weblogic-operator pods, 8 would be ideal to leave room for system daemons or fluctuating workloads.

Because you also must allocate weblogic-operator-webhook pods, a better configuration would be to schedule a maximum of 6 replicas per weblogic-operator pod per node, leaving room for a maximum of 8 weblogic-operator-webhook pods,

In this example (a worker node with 6 Gb RAM and 2 OCPUs), 6 replicas per node (12 for a 2 worker node cluster) is the best configuration.

This calculation of maximum number of replicas varies for each case, depending on the WebLogic requirements, the size of the node pools, and the shape configuration (CPU and memory) for each node pool.

Oracle Database Operator

When you enable the Oracle Database Operator for Kubernetes cluster add-on, you can pass the following key/value pairs as arguments.

Parameter Name **Console** **CLI**	Description
numOfReplicas `numOfReplicas`	(Required) The integer number of replicas of the add-on deployment.

Parameter Name **Console** **CLI**	Description
manager container resources `manager.ContainerResources`	(Optional) You can specify the resource quantities that the add-on containers request, and set resource usage limits that the add-on containers cannot exceed. Use JSON format in plain text or Base64 encoded. If you do not specify a request, the default request values are: **Limits** • cpu: 400m • memory: 400Mi **Requests** • cpu: 400m • memory: 400Mi Example: Create add-on containers that request 100 milllicores of CPU, and 100 mebibytes of memory. Limit add-on containers to 500 milllicores of CPU, and 200 mebibytes of memory. {"limits": {"cpu": "500m", "memory": "200Mi" }, "requests": {"cpu": "100m", "memory": "100Mi"}}

Nvidia GPU Plugin add-on parameters

When you enable the Nvidia GPU Plugin cluster add-on, you can pass the following key/value pairs as arguments.

Parameter Name **Console** **CLI**	Description

nvidia-gpu-device-plugin container resources
`nvidia-gpu-device-plugin container resources`
	You can specify the resource quantities that the add-on containers request, and set resource usage limits that the add-on containers cannot exceed.
	`{"limits": {"cpu": "500m", "memory": "200Mi" },`
	`"requests": {"cpu": "100m", "memory": "100Mi"}}`
	Create add-on containers that request 100 milllicores of CPU, and 100 mebibytes of memory. Limit add-on containers to 500 milllicores of CPU, and 200 mebibytes of memory.

Certificate Manager add-on parameters

When you enable the Certificate Manager cluster add-on, you can pass the following key/value pairs as arguments.

Parameter Name Console CLI	Description
numOfReplicas `numOfReplicas`	(Required) The integer number of replicas of the add-on deployment.

cert-manager-controller container resources
`cert-manager-controller.ContainerResources`
	(Optional) You can specify the resource quantities that the add-on containers request, and set resource usage limits that the add-on containers cannot exceed.
	`{"limits": {"cpu": "500m", "memory": "200Mi" },`
	`"requests": {"cpu": "100m", "memory": "100Mi"}}`
	Create add-on containers that request 100 milllicores of CPU, and 100 mebibytes of memory. Limit add-on containers to 500 milllicores of CPU, and 200 mebibytes of memory.

cert-manager-cainjector container resources
`cert-manager-cainjector.ContainerResources`
	(Optional) You can specify the resource quantities that the add-on containers request, and set resource usage limits that the add-on containers cannot exceed.
	`{"limits": {"cpu": "500m", "memory": "200Mi" },`
	`"requests": {"cpu": "100m", "memory": "100Mi"}}`
	Create add-on containers that request 100 milllicores of CPU, and 100 mebibytes of memory. Limit add-on containers to 500 milllicores of CPU, and 200 mebibytes of memory.

weblogic-operator-webhook container resources
`weblogic-operator-webhook.ContainerResources`
	(Optional) You can specify the resource quantities that the add-on containers request, and set resource usage limits that the add-on containers cannot exceed.
	`{"limits": {"cpu": "500m", "memory": "200Mi" },`
	`"requests": {"cpu": "100m", "memory": "100Mi"}}`
	Create add-on containers that request 100 milllicores of CPU, and 100 mebibytes of memory. Limit add-on containers to 500 milllicores of CPU, and 200 mebibytes of memory.

Creating and Managing OKE Worker Node Pools

Learn how to create, update, and delete node pools for an OKE cluster On Compute Cloud@Customer. Be sure to carefully read the descriptions of the node pool parameters before you create the node pool.

The following topics describe how to manage Kubernetes worker node pools in Compute Cloud@Customer:

- Creating an OKE Worker Node Pool on page 619
- Updating an OKE Node Pool on page 625
- Deleting an OKE Node Pool Node on page 630

- Deleting an OKE Node Pool on page 632

This topic also describes how to recognize node pool nodes in a list of all instances in a tenancy, and how to delete a single node from a node pool.

Creating an OKE Worker Node Pool

Learn how to create OKE worker node pools On Compute Cloud@Customer for a workload cluster.

Nodes are Compute Cloud@Customer compute instances. When you create a worker node pool, you specify the number of nodes to create and other parameters that define instances.

> **Note:**
>
> You can't customize the OKE `cloud-init` scripts.

To configure proxy settings, use the CLI or API to set the proxy in node metadata. If the cluster is using VCN-Native Pod Networking, add 169.254.169.254 to the noproxy setting.

Custom CA support is available only by using the CLI (`oci ce node-pool create` and `oci ce node-pool update`).

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Containers**, then select **Kubernetes Clusters (OKE)**.

   If the cluster to which you want to attach a node pool is not listed, select a different compartment from the compartment menu at the top of the page.
2. Select the name of the cluster to which you want to add a node pool.
3. On the cluster details page, scroll to the **Resources** sectin and select **Node Pools**.
4. On the **Node Pools** list, select the **Add Node Pool** button.
5. In the **Add Node Pool** dialog box, provide the following information:

   - **Name**: The name of the new node pool. Avoid entering confidential information.
   - **Compartment**: The compartment in which to create the new node pool.
   - **Node Pool Options**: In the **Node Count** field, enter the number of nodes you want in this node pool. The default is 0. The maximum number is 128 per cluster, which can be distributed across multiple node pools.
   - **Network Security Group**: If you check the box to enable network security groups, select the **Add Network Security Group** and select an NSG from the drop-down list. You might need to change the compartment to find the NSG you want. The primary VNIC from the worker subnet will be attached to this NSG.
   - **Placement configuration**
     - **Worker Node Subnet**: Select a subnet that has configuration such as the "worker" subnet described in Creating a Worker Subnet (Flannel Overlay) on page 536 or Creating a Worker Subnet (VCN-Native Pod) on page 571. For a public cluster, create the NAT private version of the "worker" subnet. For a private cluster, create the VCN-only private version of the "worker" subnet. Select only one subnet. The subnet must have rules set to communicate with the control plane endpoint. The subnet must use a private route table and must have a security list like the worker-seclist security list described in Creating a Worker Subnet (Flannel Overlay) on page 536 or Creating a Worker Subnet (VCN-Native Pod) on page 571.
     - **Fault Domain**: Select a fault domain, or select **Automatically select the best fault domain**, which is the default option.
   - **Source Image**: Select an image.

     a. Select the **Platform Image Source Type**.
     b. Select an image from the list.

        The image list has columns Operating System, OS Version, and Kubernetes Version. You can use the drop-down menu arrow to the right of the OS Version or Kubernetes Version to select a different version. If more than one image has the same Kubernetes version, select the newest image, according to the date in the image name.

If the image that you want to use is not listed, use the CLI procedure and specify the OCID of the image. To get the OCID of the image you want, use the `ce node-pool get` command for a node pool where you used this image before.

> **Note:**
>
> The image that you specify must not have a Kubernetes version that's newer than the Kubernetes version that you specified when you created the cluster. The Kubernetes Version for the cluster is in a column of the cluster list table.

- **Shape**: Select a shape for the worker nodes. If you're using the GPU Operator cluster add on, you can use the supported GPU shapes. For descriptions of shapes, see Compute Shapes on page 301.

  Specify the number of OCPUs you want. You can optionally specify the total amount of memory you want. The default value for gigabytes of memory is 16 times the number you specify for OCPUs. Select 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.

  Boot volume size (GB): The default boot volume size for the selected image is shown. To specify a larger size, enter a value from 50 to 16384 in gigabytes (50 GB to 16 TB) or use the increment and decrement arrows.

  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. Seeoci-growfs.
- **Pod Communication** (VCN-Native Pod Networking clusters only)

  Pod Communication Subnet: Select a subnet that has configuration like the "pod" subnet described in Creating a Pod Subnet (VCN-Native Pod) on page 568.

  Number of Pods per node: The maximum number of pods that you want to run on a single worker node in a node pool. The default value is 31. You can enter a number from 1 to 110. The number of VNICs allowed by the shape you specify (see "Shape" above) limits this maximum pods number. See Node Shapes and Number of Pods on page 549. To conserve the pod subnet's address space, reduce the maximum number of pods you want to run on a single worker node. This reduces the number of IP addresses that are pre-allocated in the pod subnet.

  If you check the box to Use Security Rules in Network Security Group (NSG), select the Add Network Security Group button and select an NSG from the drop-down list. You might need to change the compartment to find the NSG you want. Secondary VNICs from the pod subnet will be attached to this NSG.
- **Cordon and Drain**: (Optional) Use the arrows to decrease or increase the number of minutes of eviction grace duration. You cannot deselect "Force terminate after grace period." Nodes are deleted after their pods are evicted or at the end of the eviction grace duration, even if not all pods are evicted.

  For descriptions of cordon and drain and eviction grace duration, select the CLI tab on this page, and see *Node and node pool deletion settings*.
- **SSH Key**: The public SSH key for the worker nodes. Either upload the public key file or copy and paste the content of the file.
- **Kubernetes Labels**: Select the Add Kubernetes Label button and enter a key name and value. You can use these labels to target pods for scheduling on specific nodes or groups of nodes. See the description and example in the CLI procedure.
- **Node Pool Tags**: Defined or free-form tags for the node pool resource.

> **Note:**
>
> Don't specify values for the OraclePCA-OKE.cluster_id defined tag or for the ClusterResourceIdentifier free-form tag. These tag values are system-generated and only applied to nodes (instances), not to the node pool resource.

- **Node Tags**: Defined or free-form tags that are applied to every node in the node pool.

> **Important:**
>
> Don't specify values for the OraclePCA-OKE.cluster_id defined tag or for the ClusterResourceIdentifier free-form tag. These tag values are system-generated.

6. Select **Add Node Pool**.

   The details page for the node pool is displayed. Under **Resources**, select **Work Requests** to see the progress of the node pool creation and see nodes being added to the Nodes list. The work request status is Accepted until the cluster is in either Active state or Failed state.

   To identify these nodes in a list of instances, note that the names of these nodes are in the format `oke-`*`ID`*, where *`ID`* is the first 32 characters after the *`pca_name`* in the node pool OCID. Search for the instances in the list whose names contain the *`ID`* string from this node pool OCID.

   To identify these nodes in a list of instances, note that the names of these nodes contain the first 32 characters after the *`ccc_name`* of the node pool OCID, and they contain the cluster OCID in the OraclePCA-OKE.cluster_id tag.

What's Next:

1. Configure any registries or repositories that the worker nodes need. Ensure you have access to a self-managed public or intranet container registry to use with the OKE service and your application images.
2. Create a service to expose containerized applications outside the Compute Cloud@Customer. See Exposing Containerized Applications on page 633.
3. Create persistent storage for applications to use. See Adding Storage for Containerized Applications on page 634.

Using the CLI

Use the oci ce node-pool create command and required parameters to create a new node pool.

```
oci ce node-pool create --cluster-id <cluster_OCID> --compartment-
id <compartment_OCID> --name <pool_name> --node-shape <node_shape_name>
 [OPTIONS]
```

1. Get the information you need to run the command.

   - The OCID of the compartment where you want to create the node pool: `oci iam compartment list`
   - The OCID of the cluster for this node pool: `oci ce cluster list`
   - The name of the node pool. Avoid entering confidential information.
   - The placement configuration for the nodes, including the worker subnet OCID and fault domain. See the "Placement configuration" description in the Compute Web UI procedure. Use the following command to show the content and format of this option:

     ```
     $ oci ce node-pool create --generate-param-json-input placement-configs
     ```

     Use the following command to list fault domains: `oci iam fault-domain list`. Don't specify more than one fault domain or more than one subnet in the placement configuration. To allow the system to select the best fault domains, don't specify any fault domain.
   - (VCN-Native Pod Networking clusters only) The OCID of the pod subnet. See Creating a Pod Subnet (VCN-Native Pod) on page 568. See also the description in Pod Communication in the Compute Web UI procedure. Use the `--pod-subnet-ids` option. Although the `--pod-subnet-ids` option value is an array, you can specify only one pod subnet OCID.

     The maximum number of pods that you want to run on a single worker node in a node pool. Use the `--max-pods-per-node` option. The default value is 31. You can enter a number from 1 to 110. The number of VNICs allowed by the shape you specify (see "The name of the shape" below) limits this maximum pods number. See Node Shapes and Number of Pods on page 549. To conserve the pod subnet's address space,

reduce the maximum number of pods you want to run on a single worker node. This reduces the number of IP addresses that are pre-allocated in the pod subnet.

(Optional) The OCID of the Network Security Group to use for the pods in this node pool. Use the `--pod-nsg-ids` option. You can specify up to five NSGs.

- The OCID of the image to use for the nodes in this node pool.

  Use the following command to get the OCID of the image that you want to use:

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

  If the image that you want to use isn't listed, you can get the OCID of the image from the output of the `ce node-pool get` command for a node pool where you used this image before.

  > **Note:**
  >
  > The image that you specify must have `-OKE-` in its `display-name` and must not have a Kubernetes version that's newer than the Kubernetes version that you specified when you created the cluster.

  The Kubernetes version for the cluster is shown in `cluster list` output. The Kubernetes version for the image is shown in the `display-name` property in `image list` output. The Kubernetes version of the following image is 1.29.9.

  `"display-name": "uln-pca-Oracle-Linux8-OKE-1.29.9-20250325.oci"`

  Don't specify the `--kubernetes-version` option in the `node-pool create` command.

  You can specify a custom boot volume size in gigabytes. The default boot volume size is 50 GB. To specify a custom boot volume size, use the `--node-source-details` option to specify both the boot volume size and the image. You can't specify both `--node-image-id` and `--node-source-details`. Use the following command to show the content and format of the node source details option.

  ```
  $ oci ce node-pool create --generate-param-json-input node-source-details
  ```

  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. See oci-growfs.

- The name of the shape of the worker nodes in this node pool. If you're using the GPU Operator cluster add on, you can use the supported GPU shapes.

  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 gigabytes is 16 times the number of `ocpus`.

  ```
  --node-shape-config '{"ocpus": 32, "memoryInGBs": 512}'
  ```

  > **Note:**
  >
  > Allocate at least 2 OCPUs and 32 GB memory for every 10 running pods. You might need to allocate more resources, depending on the workloads that are planned. See Resource Management for Pods and Containers.

- (Optional) The OCID of the Network Security Group to use for the nodes in this node pool. Use the `--nsg-ids` option. Do not specify more than one NSG.
- (Optional) Labels. Setting labels on nodes enables you to target pods for scheduling on specific nodes or groups of nodes. Use this functionality to ensure that specific pods only run on nodes with certain isolation, security, or regulatory properties.

Use the `--initial-node-labels` option to add labels to the nodes. Labels are a list of key/value pairs to add to nodes after they join the Kubernetes cluster. See Metadata Key Restrictions on page 301 for more information about metadata limits.

The following is an example label to apply to the nodes in the node pool:

```
--initial-node-labels '[{"key":"disktype","value":"ssd"}]
```

An easy way to select nodes based on their labels is to use `nodeSelector` in the pod configuration. Kubernetes only schedules the pod onto nodes that have each of the labels that are specified in the `nodeSelector` section.

The following example excerpt from a pod configuration specifies that pods that use this configuration must be run on nodes that have the `ssd` disk type label:

```
nodeSelector:
  disktype: ssd
```

- (Optional) Node metadata. Use the `--node-metadata` option to attach custom user data to nodes. See the following proxy settings item for a specific example.

  See Metadata Key Restrictions on page 301. The maximum size of node metadata is 32,000 bytes.
- (Optional) Proxy settings. If your network requires proxy settings to enable worker nodes to reach outside registries or repositories, for example, create an argument for the `--node-metadata` option.

  In the `--node-metadata` option argument, provide values for `crio-proxy` and `crio-noproxy` as shown in the following example file argument:

  ```
  {
    "crio-proxy": "http://<your_proxy>.<your_domain_name>:<your_port>",
    "crio-noproxy":
   "localhost,127.0.0.1,<your_domain_name>,ocir.io,<Kubernetes_cidr>,<pods_cidr>"
  }
  ```

  If the cluster is using VCN-Native Pod Networking, add 169.254.169.254 to the noproxy setting, as in the following example:

  ```
  "crio-noproxy":
   "localhost,127.0.0.1,your_domain_name,ocir.io,Kubernetes_cidr,pods_cidr,169.254.16
  ```
- (Optional) Node and node pool deletion settings. You can specify how to handle node deletion when you delete a node pool, delete a specified node, decrement the size of the node pool, or change the node pool nodes placement configuration. These node deletion parameters can also be set or changed when you update the node pool, delete a specified node, or delete the node pool.

  To specify node pool deletion settings, create an argument for the `--node-eviction-node-pool-settings` option. You can specify the eviction grace duration (`evictionGraceDuration`) for nodes. Nodes are always deleted after their pods are evicted or at the end of the eviction grace duration.

  - Eviction grace duration. This value specifies the amount of time to allow to cordon and drain worker nodes.

    A node that's cordoned can't have new pods placed on it. Existing pods on that node aren't affected.

    When a node is drained, each pod's containers terminate gracefully and perform any necessary cleanup.

    The eviction grace duration value is expressed in ISO 8601 format. The default value and the maximum value are 60 minutes (PT60M). The minimum value is 20 seconds (PT20S). OKE always tries to drain nodes for at least 20 seconds.
  - Force delete. Nodes are always deleted after their pods are evicted or at the end of the eviction grace duration. After the default or specified eviction grace duration, the node is deleted, even if one or more pod containers aren't completely drained.

The following shows an example argument for the `--node-eviction-node-pool-settings` option. If you include the `isForceDeleteAfterGraceDuration` property, then its value must be `true`. Nodes are always deleted after their pods are evicted or at the end of the eviction grace duration.

```
--node-eviction-node-pool-settings '{"evictionGraceDuration": "PT30M",
 "isForceDeleteAfterGraceDuration": true}'
```

> **Note:**
>
> If you use Terraform and you specify `node_eviction_node_pool_settings`, then you must explicitly set `is_force_delete_after_grace_duration` to `true`, even though true is the default value. The `is_force_delete_after_grace_duration` property setting isn't optional if you're using Terraform.

- (Optional) Tags. Add defined or free-form tags for the node pool resource by using the `--defined-tags` or `--freeform-tags` options. Do not specify values for the OraclePCA-OKE.cluster_id defined tag or for the ClusterResourceIdentifier free-form tag. These tag values are system-generated and only applied to nodes (instances), not to the node pool resource.

  To add defined or free-form tags to all nodes in the node pool, use the `--node-defined-tags` and `--node-freeform-tags` options.

  > **Important:**
  >
  > Do not specify values for the OraclePCA-OKE.cluster_id defined tag or for the ClusterResourceIdentifier free-form tag. These tag values are system-generated.

2. Run the create node pool command.

   Example:

   See the Using the Console on page 619 procedure for information about the options shown in this example, and other options such as `--node-boot-volume-size-in-gbs` and `nsg-ids`. The `--pod-subnet-ids` option is only applicable if the cluster uses VCN-Native Pod Networking.

   ```
   $ oci ce node-pool create \
   --cluster-id ocid1.cluster.unique_ID --compartment-id
    ocid1.compartment.unique_ID \
   --name node_pool_name --node-shape shape_name --node-image-id
    ocid1.image.unique_ID \
   --placement-configs
    '[{"availabilityDomain":"AD-1","subnetId":"ocid1.subnet.unique_ID"}]' \
   --pod-subnet-ids '["ocid1.subnet.unique_ID"]' --size 10 --ssh-public-key
    "public_key_text"
   ```

   The output from this `node-pool create` command is the same as the output from the `node-pool get` command. The cluster OCID is shown, and a brief summary of each node is shown. For more information about a node, use the `compute instance get` command with the OCID of the node.

   Use the `work-request get` command to check the status of the node pool create operation. The work request OCID is in `created-by-work-request-id` in the `metadata` section of the `cluster get` output.

   ```
   $ oci ce work-request get --work-request-id workrequest_OCID
   ```

   The work request status will be `ACCEPTED` until the cluster is in either Active state or Failed state.

   To identify these nodes in a list of instances, note that the names of these nodes are in the format `oke-ID`, where *ID* is the first 32 characters after the name in the node pool OCID. Search for the instances in the list whose names contain the *ID* string from this node pool OCID.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

What's Next:

1. Configure any registries or repositories that the worker nodes need. Ensure you have access to a self-managed public or intranet container registry to use with the OKE service and your application images.
2. Create a service to expose containerized applications outside the Compute Cloud@Customer. See Exposing Containerized Applications on page 633.
3. Create persistent storage for applications to use. See Adding Storage for Containerized Applications on page 634.

Using the API

Use the CreateNodePool operation to create a new node pool.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Next Steps:

1. Configure any registries or repositories that the worker nodes need. Ensure you have access to a self-managed public or intranet container registry to use with the OKE service and your application images.
2. Create a service to expose containerized applications outside the Compute Cloud@Customer. See Exposing Containerized Applications on page 633.
3. Create persistent storage for applications to use. See Adding Storage for Containerized Applications on page 634.

Updating an OKE Node Pool

On Compute Cloud@Customer, you can update any configuration that you can set when you create a node pool except for the compartment where nodes are created.

When you update node properties, by default existing nodes aren't updated. The updated values only apply to new nodes that are created. New nodes are created when you increase the node count.

To replace existing nodes with new nodes that use these updated settings, see Node Cycling an OKE Node Pool on page 629.

If new worker nodes are added, perform the following steps:

1. Configure any registries or repositories that the worker nodes need. Ensure you have access to a self-managed public or intranet container registry to use with the OKE service and your application images.
2. Create a service to expose containerized applications outside the Compute Cloud@Customer. See Exposing Containerized Applications on page 633.
3. Create persistent storage for applications to use. See Adding Storage for Containerized Applications on page 634.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Containers**, then select **Kubernetes Clusters (OKE)**.
2. Select the name of the cluster that contains the node pool that you want to update.
3. On the cluster details page, under **Resources**, select **Node Pools**.
4. For the node pool that you want to update in the **Node Pools** list, select the Actions menu (⋮), then select **Edit**.

   The **Edit Node Pool** dialog box opens. You can change any configuration except the compartment where new nodes are created. See Creating an OKE Worker Node Pool on page 619 for property descriptions. The updated configuration only applies to new nodes that are created.

   Cordon and Drain: Enter the number of minutes of eviction grace duration, or use the arrows to decrease or increase the number of minutes of eviction grace duration. The maximum value and default value is 60 minutes.

You can't clear **Force terminate after grace period**. For descriptions of cordon and drain and eviction grace duration, go to Creating an OKE Worker Node Pool on page 619, select the CLI tab, and see *Node and node pool deletion settings*.

> **Note:**
>
> Don't specify values for the OraclePCA-OKE.cluster_id defined tag or for the ClusterResourceIdentifier free-form tag. These tag values are system-generated and only applied to nodes (instances), not to the node pool resource.

5. When you're finished making changes, select **Save Changes**.

   The details page for the node pool is displayed. In addition to **Node Pool Information** and **Tags** tabs, the node pool details page has a **Placement Configuration** tab.

   The updated configuration only applies to new nodes that are created by this procedure or in the future.

   To replace existing nodes with new nodes that use these updated settings, see Node Cycling an OKE Node Pool on page 629.

What's Next:

If you make changes that add new worker nodes, consider your next steps:

1. Configure any registries or repositories that the worker nodes need. Ensure you have access to a self-managed public or intranet container registry to use with the OKE service and your application images.
2. Create a service to expose containerized applications outside the Compute Cloud@Customer. See Exposing Containerized Applications on page 633.
3. Create persistent storage for applications to use. See Adding Storage for Containerized Applications on page 634.

Using the CLI

Use the oci ce node-pool update command and required parameters to update a node pool.

```
oci ce node-pool update --node-pool-id <node-pool_OCID> [OPTIONS]
```

1. Get the information you need to run the command.
   - The OCID of the node pool that you want to update: `oci ce node-pool list`
   - (Optional) Node and node pool deletion settings. Use the `--node-eviction-node-pool-settings` option or the `--override-eviction-grace-duration` option to set the eviction grace duration for nodes. Nodes are always deleted after their pods are evicted or at the end of the eviction grace duration. See the description in Creating an OKE Worker Node Pool on page 619.
   - (Optional) Labels. To add labels to new nodes, use the `--initial-node-labels` option. Labels on existing nodes cannot be changed by using the `--initial-node-labels` option. Labels on existing nodes can be modified using `kubectl`. For more information about node labels, see Creating an OKE Worker Node Pool on page 619.
   - (Optional) Tags. Add, change, or delete defined or free-form tags for the node pool resource by using the `--defined-tags` and `--freeform-tags` options. Do not specify values for the OraclePCA-OKE.cluster_id defined tag or for the ClusterResourceIdentifier free-form tag. These tag values are system-generated and only applied to nodes (instances), not to the node pool resource.

     To add tags to nodes that are newly added to the node pool, use the `--node-defined-tags` and `--node-freeform-tags` options.
   - (Optional) Custom CA Certificate Bundle. You can specify a custom CA certificate bundle to be used for verifying TLS connections to a private container registry. Use the `custom-ca-bundle-cert`, `registry-host`, and `registry-port` metadata parameters with the `--node-metadata` option.

In the `--node-metadata` option argument, you must provide values for `custom-ca-bundle-cert` and `registry-host`, and optionally you can provide a value for `registry-port` as shown in the following example file argument:

```
--node-metadata custom-ca-bundle-cert=<base64-encoded-cert-
content>,registry-host=<registry.example.com>,registry-port=5000
```

If you configured a private registry with a custom CA bundle, ensure nodes are newly provisioned or cycled (scaled down and back up or rolling restart) to apply the updated certificate to all worker nodes.

2. (Optional) Create an argument for the `--node-pool-cycling-details` option, and use that option to apply these updates to all of the nodes in the node pool.

   Without the `--node-pool-cycling-details` option, the updated configuration specified in this `node-pool update` command only applies to new nodes that are created by this command or in the future, as described at the beginning of this topic.

   To replace existing nodes with new nodes that use these updated settings, specify the `--node-pool-cycling-details` option as described in Node Cycling an OKE Node Pool on page 629.
3. Run the update node pool command.

```
$ oci ce node-pool update --node-pool-id ocid1.nodepool.unique_ID \
<new_configuration_settings>
```

What's Next:

If you make changes that add new worker nodes, consider these next steps:

1. Configure any registries or repositories that the worker nodes need. Ensure you have access to a self-managed public or intranet container registry to use with the OKE service and your application images.
2. Create a service to expose containerized applications outside the Compute Cloud@Customer. See Exposing Containerized Applications on page 633.
3. Create persistent storage for applications to use. See Adding Storage for Containerized Applications on page 634.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the UpdateNodePool operation to update a node pool.

If you make changes that add new worker nodes, consider these next steps:

1. Configure any registries or repositories that the worker nodes need. Ensure you have access to a self-managed public or intranet container registry to use with the OKE service and your application images.
2. Create a service to expose containerized applications outside the Compute Cloud@Customer. See Exposing Containerized Applications on page 633.
3. Create persistent storage for applications to use. See Adding Storage for Containerized Applications on page 634.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Using Node Doctor to Troubleshoot Worker Node Issues

On Compute Cloud@Customer, Node Doctor is a script that is included in the latest OKE images.

> **Note:**
>
> If Compute Cloud@Customer has worker nodes that were created before Node Doctor was included in OKE images, you can use node cycling to update older worker node images. See Node Cycling an OKE Node Pool on page 629.

If a cluster has a worker node that's in a state other than Active or Running, use the Node Doctor utility to troubleshoot the issues.

Node Doctor scans a worker node and reports the health status of the node. Node Doctor can do the following tasks:

- Identify potential problem areas and provide references to information to help you address those problem areas. See Print Troubleshooting Information on page 628.
- Collect node system information into a support bundle if you need help from Oracle Support. See Create a Support Bundle on page 629.

> **Important:**
>
> Use Node Doctor only on worker nodes. Because Node Doctor is installed on OKE images, Node Doctor is also available on cluster control plane nodes. Don't use Node Doctor on control plane nodes.

Connect to the Worker Node Using SSH

Perform the following steps to connect to the worker node that you want to troubleshoot.

1. Ensure that you have a private and public SSH key pair.

   You must have the private key that goes with the public key that was added to the node when the node was created.
2. Get the node username. OKE images have the initial username `opc` configured.
3. Get the IP address of the worker node that you need to troubleshoot.

   The IP address is on the Networking tab of the node details page in the Console.

   - If the node has a public IP address, use the public IP address.
   - If the node is on a private IP, then connect to the node through the bastion host.

     If a bastion host isn't available, see Creating a Bastion.
4. Enter the following command at a shell prompt on your local system (public IP address) or on the bastion host (private IP address):

   ```
   ssh -i private_key_file username@ip-address
   ```

   - *private_key_file*. The full path and name of the file that contains the private SSH key that goes with the public key that was added to the node when the node was created.
   - *username*. The default username for the node. This value probably is `opc`.
   - *ip-address*. The node IP address that you got in the previous step.
5. Ensure that you have execute permissions for the following script. You run the script later.

   ```
   ls -l /usr/local/bin/node-doctor.sh
   -rwxr-xr-x 1 user1 user1 6288 Dec  5  2024 usr/local/bin/node-doctor.sh
   ```

Print Troubleshooting Information

While logged in to the worker node as described in Connect to the Worker Node Using SSH on page 628, enter the following command to print information that identifies potential problem areas:

```
$ sudo /usr/local/bin/node-doctor.sh --check
```

Use the following command to see more options:

```
$ sudo /usr/local/bin/node-doctor.sh --help
```

Create a Support Bundle

If you can't resolve the issue, use the following command to create a support bundle with relevant information for Oracle Support:

```
$ sudo /usr/local/bin/node-doctor.sh --generate
```

The support bundle is in the `/tmp` directory as `oke-support-bundle-`*`date`*`T`*`time`*`.tar`.

> **Note:**
>
> Monitor the `/tmp` directory to ensure that it doesn't fill up. Remove old files using the `rm` command, for example.

See the following resources for information about submitting a Support Request and uploading a bundle:

- Creating a Support Request
- Quick User Guide to Upload Files to My Oracle Support - MOS (Doc ID 1588459.1)
- How to Upload Files to Oracle Support (Doc ID 1547088.2)

Node Cycling an OKE Node Pool

On Compute Cloud@Customer, when you update a node pool, only new nodes that are added during this update or that are added later receive the updates. To replace existing nodes with new nodes that use updated settings, enable the node cycling option.

Node cycling performs an in-place update of all existing nodes in the node pool to the latest specified configuration. New nodes are created, workloads moved onto them from existing nodes, current node pool updates applied, and the original nodes terminated.

You can set the maximum number of nodes that are starting or terminating at any particular time.

- **Maximum surge**. The maximum number of new nodes that can be starting at any time during this update operation. Set this value to avoid adding too many new nodes before existing nodes are terminated, which could incur excessive cost. The default value is 1. The maximum value is 5.
- **Maximum unavailable**. The maximum number of existing nodes that can be terminating at any time during this update operation. Set this value to ensure that enough nodes remain to handle the workload. The default value is 0. The maximum value is 7.

One of these values must be greater than 0.

Both of these values can be set to either a number (from 0 to the configured number of nodes in the node pool, but not greater than the maximum cited above) or a percentage (from 0% to 100%, but not a percentage that would result in a number greater than the maximum cited above). These values can be a maximum of four characters.

Both of these values must be a number of nodes, or both of these values must be a percentage. You cannot specify one of these values as a number of nodes and the other value as a percentage.

If you set either of these properties to a percent value that exceeds the maximum allowed number of nodes, the error message tells you the maximum allowed percent value for this node pool.

> **Note:**
>
> If the node cycling operation fails (for example, the operation times out), try re-running the operation. You might need to run the node cycling operation multiple times if the system is loaded and running at scale.

Using the Console

Follow the procedure in Updating an OKE Node Pool on page 625 to update the node pool configuration.

1. On the node pool details page, select the Cycle Nodes button.
2. In the Cycle Nodes dialog, enter values for the Maximum Surge and Maximum Unavailable properties.

   See the rules at the beginning of this topic.

3. Select the Cycle Nodes button in the dialog to start the node pool update operation.

   To monitor the progress of the update operation, view the status of the associated work request.

Using the CLI

Use the `--node-pool-cycling-details` option of the oci ce node-pool update command to update all existing nodes in a node pool.

```
oci ce node-pool update [OPTIONS]
```

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Procedure

1. Construct a command to update the node pool configuration as described in Updating an OKE Node Pool on page 625 under the CLI tab.
2. In that same command (not later) include the `--node-pool-cycling-details` option.

   In addition to setting `maximumUnavailable` and `maximumSurge`, enable node cycling by setting `isNodeCyclingEnabled` to `true`. By default, `isNodeCyclingEnabled` is `false`, and node cycling is not performed if you set only `maximumUnavailable` or `maximumSurge` and do not set `isNodeCyclingEnabled` to `true`.

   ```
   $ oci ce node-pool update --node-pool-id ocid1.nodepool.unique_ID \
   new_configuration_settings \
   --node-pool-cycling-details
    '{"isNodeCyclingEnabled":true,"maximumUnavailable":"value","maximumSurge":"value"}'
   ```

   See the beginning of this topic for the possible values.

   In the following example, the image is updated for all nodes in the node pool:

   ```
   $ oci ce node-pool update --node-pool-id ocid1.nodepool.unique_ID \
   --node-source-details
    '{"imageId":"ocid1.image.unique_ID","sourceType":"IMAGE"}' \
   --node-pool-cycling-details
    '{"isNodeCyclingEnabled":true,"maximumUnavailable":"5%","maximumSurge":"5%"}'
   ```

   To monitor the progress of the update operation, view the status of the associated work request.

   Find the work request OCID:

   ```
   oci ce work-request list --compartment-id ocid1.compartment.unique_ID \
   --resource-id ocid1.nodepool.unique_ID
   ```

   Show the current state of the work request:

   ```
   oci ce work-request get --work-request-id ocid1.workrequest.unique_ID
   ```

Using the API

Use the UpdateNodePool operation to update a node pool. To update all existing nodes in the node pool, use the nodePoolCyclingDetails attributes: isNodeCyclingEnabled, maximumSurge, and maximumUnavailable.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting an OKE Node Pool Node

Learn how to explicitly delete an OKE worker node On Compute Cloud@Customer.

Worker nodes are also deleted when you update a node pool to scale down the node pool. See Updating an OKE Node Pool on page 625.

> **Caution:**
>
> Deleting a worker node permanently deletes the node. You can't recover a deleted worker node.

When you delete a node, by default a new node is created to satisfy the node count set for the pool. To override this behavior, select the option to decrease node pool size.

> **Important:**
>
> Don't use the `kubectl delete node` command to terminate worker nodes in an OKE cluster. The `kubectl delete node` command removes the worker node from the cluster's etcd key-value store, but the command doesn't terminate the underlying compute instance.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Containers**, then select **Kubernetes Clusters**.
2. Select the name of the cluster that contains the node that you want to delete.
3. On the cluster details page, scroll to the **Resources** section and select **Node Pools**.
4. Select the name of the node pool that contains the node that you want to delete.
5. On the node pool details page, scroll to the **Resources** section and select **Nodes**.
6. For the node that you want to delete, select the Actions menu (⋮) and select **Delete**.
7. Confirm the deletion.
   a. If you don't want a new node to be automatically created to replace the deleted node, select **Decrease node pool size**.
   b. Check the box to override the eviction grace duration in the cordon and drain settings for the node. See the description of this field in Updating an OKE Node Pool on page 625.

      Use the arrows to decrease or increase the number of minutes of eviction grace duration.

      You can't clear "Force terminate after grace period." The node is deleted after its pods are evicted or at the end of the eviction grace duration, even if not all pods are evicted.

      For descriptions of cordon and drain and eviction grace duration, go to Creating an OKE Worker Node Pool on page 619, select the CLI tab, and see *Node and node pool deletion settings*.
   c. Select **Delete**.

Using the CLI

Use the oci ce node-pool delete-node command and required parameters to delete a node pool node.

```
oci ce node-pool delete-node --node-id <node_OCID> --node-pool-id <node-
pool_OCID> [OPTIONS]
```

1. Get the information you need to run the command.
   - OCID of the node pool: `oci ce node-pool list`
   - OCID of the node: `oci ce node-pool list`
2. Run the delete node pool node command.

   If you don't want a new node to be automatically created to replace the deleted node, specify the `--is-decrement-size` option.

   Example:

   ```
   $ oci ce node-pool delete-node --node-pool-id ocid1.nodepool.unique_ID \
   --node-id ocid1.instance.unique_ID --is-decrement-size true --force
   ```

You can use the `--override-eviction-grace-duration` option to set a new value for `evictionGraceDuration` for this node deletion. See the description of `--node-eviction-node-pool-settings` in Creating an OKE Worker Node Pool on page 619. For `node-pool delete-node`, this new eviction grace duration value only applies to the node being deleted.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteNode operation to delete a node pool node.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting an OKE Node Pool

Learn how to delete an OKE node pool On Compute Cloud@Customer.

> **Caution:**
>
> Deleting a node pool permanently deletes the node pool. You can't recover a deleted node pool.

Using the Console

1. In the Compute Cloud@Customer Console navigation menu, select **Containers**, then select **Kubernetes Clusters**.
2. Select the name of the cluster that contains the node pool that you want to delete.
3. On the cluster details page, scroll to the **Resources** section and select **Node Pools**.
4. For the node pool that you want to delete, select the Actions menu (⋮) and select **Delete**.
5. Confirm the deletion.
   a. Enter the name of the node pool to confirm that you want to delete the node pool.
   b. Check the box to override the eviction grace duration in the cordon and drain settings for the nodes in the pool.

      Use the arrows to decrease or increase the number of minutes of eviction grace duration. See the description of this field in Updating an OKE Node Pool on page 625.

      You can't clear **Force terminate after grace period**. Nodes are deleted after their pods are evicted or at the end of the eviction grace duration, even if not all pods are evicted.

      For descriptions of cordon and drain and eviction grace duration, go to Creating an OKE Worker Node Pool on page 619, select the CLI tab, and see *Node and node pool deletion settings*.
   c. Select **Delete**.

Using the CLI

Use the oci ce node-pool delete command and required parameters to delete a node pool.

```
oci ce node-pool delete --node-pool-id <node-pool_OCID> [OPTIONS]
```

1. Get the OCID of the node pool that you want to delete: `oci ce node-pool list`
2. Run the delete node pool command. You can use the optional `--force option` to delete the node pool without prompting for confirmation.

   Example:

   ```
   $ oci ce node-pool delete --node-pool-id ocid1.nodepool.unique_ID --force
   ```

   You can use the `--override-eviction-grace-duration` option to set a new value for `evictionGraceDuration` for this node pool deletion. See the description of `--node-eviction-node-pool-settings` in Creating an OKE Worker Node Pool on page 619.

For a complete list of CLI commands, flags, and options, see the Command Line Reference.

Using the API

Use the DeleteNodePool operation to delete a node pool.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Exposing Containerized Applications

On Compute Cloud@Customer, you can expose an application deployment so that worker node applications can be reached from outside the infrastructure.

Do the following to expose worker node applications outside the infrastructure:

- Create an external load balancer.
- Update ingress and egress rules as necessary to support the port requirements of the containerized applications. For example, if any application uses TCP port 3000, then an ingress rule needs to be added with port 3000.

Create an External Load Balancer

An external load balancer is a Service of type LoadBalancer. The service provides load balancing for an application that has multiple running instances.

If you use the `--service-lb-defined-tags` or `--service-lb-flexible-tags` options to specify tags to be applied to external load balancers, then ensure that the applicable dynamic group includes the following policy. See Create a Cluster Dynamic Group and Policies on page 508.

```
allow dynamic-group dynamic-group-name to use tag-namespaces in
 compartment compartment-name
```

Ensure that the load balancer shape parameter has one of the following values:

- `400Mbps`
- `flexible` – Requires that you also provide `flex-min` and `flex-max` annotations.

You might need to edit the application deployment file to modify the load balancer shape value. See Specifying Alternative Load Balancer Shapes and Specifying Flexible Load Balancer Shapes for more information and examples of how to set these values.

To create a service load balancer on a private cluster (a cluster with a private worker load balancer subnet), then use the following annotation in the external load balancer template:

```
service.beta.kubernetes.io/oci-load-balancer-internal: "true"
```

Use the following command to create the external load balancer:

```
# kubectl create -f expose_lb
```

The following is the content of the `expose_lb` file:

```
apiVersion: v1
kind: Service
metadata:
  name: my-nginx-svc
  labels:
    app: nginx
  annotations:
    oci.oraclecloud.com/load-balancer-type: "lb"
    service.beta.kubernetes.io/oci-load-balancer-shape: "400Mbps"
spec:
  type: LoadBalancer
  ports:
   - port: 80
```

```
  selector:
    app: nginx
```

The following command shows more information about this external load balancer. The LoadBalancer Ingress IP address is the IP address that's used to reach node applications from outside the Compute Cloud@Customer. In the Console, the LoadBalancer Ingress IP address is shown under the heading "IP Address" at the bottom of the first column of the load balancer details page, followed by the label "(Public)."

```
# kubectl describe svc my-nginx-svc
Name:                     my-nginx-svc
Namespace:                default
Labels:                   app=nginx
Annotations:              oci.oraclecloud.com/load-balancer-type: lb
                          service.beta.kubernetes.io/oci-load-balancer-
shape: 400Mbps
Selector:                 app=nginx
Type:                     LoadBalancer
IP Family Policy:         SingleStack
IP Families:              IPv4
IP:                       IP_address
IPs:                      IP_address
LoadBalancer Ingress:     Load_Balancer_IP_address
Port:                     <unset> 80/TCP
TargetPort:               80/TCP
NodePort:                 <unset> 32145/TCP
Endpoints:                IP_address:port, IP_address+1:port, IP_address
+2:port
Session Affinity:         None
External Traffic Policy:  Cluster
Events:
  Type    Reason                Age    From                Message
  ----
  Normal  EnsuringLoadBalancer  7m48s  service-controller  Ensuring load
 balancer
  Normal  EnsuredLoadBalancer   6m40s  service-controller  Ensured load
 balancer
```

For descriptions of traffic policies, see Preserving the client source IP.

Use the following command to list IP addresses and ports for the external load balancer:

```
# kubectl get svc
NAME          TYPE          CLUSTER-IP  EXTERNAL-IP               PORT(S)
    AGE
kubernetes    ClusterIP     IP_address  <none>                    443/TCP
    6h17m
my-nginx-svc  LoadBalancer  IP_address
         Load_Balancer_IP_address  80:32145/TCP  5h5m
```

What's Next:

Adding Storage for Containerized Applications on page 634

Adding Storage for Containerized Applications

On Compute Cloud@Customer, you can add persistent storage for use by applications on an OKE cluster node. Storage created in a container's root file system is deleted when you delete the container. For more durable storage for containerized applications, configure persistent volumes to store data outside of containers.

A persistent volume (PV) is storage that enables data to remain intact when the containers to which the storage is connected are terminated.

A PV is a resource in the cluster. A persistent volume claim (PVC) is a request for a PV resource. A PVC is a storage request that is met by binding the PVC to a PV. A PVC provides an abstraction layer to the underlying storage.

You can provision PVCs using the following methods:

- **Block volumes**. Attach volumes from the Compute Cloud@Customer Block Volume service. The volumes are connected to clusters created by OKE using a CSI (Container Storage Interface) volume plugin deployed on the clusters.
  - To provision a regular block volume, see Creating Persistent Block Volume Storage on page 635.
  - To provision a high performance block volume, see Creating Persistent High Performance Block Volume Storage on page 636.

  For information about block volumes On Compute Cloud@Customer, see Block Volume Storage on page 380.
- **File systems**. Mount file systems from the Compute Cloud@Customer File Storage service. The File Storage service file systems are mounted inside containers running on clusters created by OKE using a CSI volume plugin deployed on the clusters.
  - Provision a PVC on a new file system using the CSI volume plugin. Create a storage class and a PVC. The CSI volume plugin dynamically creates both a new File Storage service file system and a new persistent volume backed by the new file system. See Creating Persistent File System Storage Using the CSI Volume Plugin on page 638.
  - Provision a PVC on an existing file system. Create a file system, mount target, PV, and PVC. See Creating Persistent File System Storage Using an Existing File System on page 640.

  For information about file systems on the Compute Cloud@Customer, see File Storage on page 433.

Creating Persistent Block Volume Storage

The Compute Cloud@Customer Block Volume service provides persistent, durable, and high-performance block storage that you can use to store data outside of containers.

This procedure automatically creates the requested `oci-bv` storage class; you don't need to create it. This procedure starts with using the `kubectl` command to create the persistent volume claim.

1. Create a persistent volume claim, specifying the storage class name `oci-bv`.

   ```
   $ kubectl create -f csi-bvs-pvc.yaml
   ```

   The following is the content of the `csi-bvs-pvc.yaml` file:

   ```
   apiVersion: v1
   kind: PersistentVolumeClaim
   metadata:
     name: mynginxclaim
   spec:
     storageClassName: "oci-bv"
     accessModes:
       - ReadWriteOnce
     resources:
       requests:
         storage: 50Gi
   ```

   The persistent volume claim name in the `metadata` section is user-specified. You can have more than one persistent volume claim on a persistent volume.

   For the value of `accessModes`, specify `ReadWriteOnce`. Don't use `ReadWriteMany`.

   The value of the `storage` property must be at least 50 gigabytes.
2. Run the following command to verify that the PVC has been created:

   ```
   $ kubectl get pvc
   ```

```
NAME           STATUS    VOLUME    CAPACITY    ACCESSMODES    STORAGECLASS
 AGE
mynginxclaim   Pending                                       oci-bv
 4m
```

The PVC has a status of `Pending` because the `oci-bv` storage class definition includes the following:

```
volumeBindingMode: WaitForFirstConsumer
```

3. Use the PVC when creating other objects, such as pods.

   For example, you could create a new pod from the following pod definition, which instructs the system to use the `mynginxclaim` PVC as the `nginx` volume, which is mounted by the pod at `/data`:

```
apiVersion: v1
kind: Pod
metadata:
  name: nginx
spec:
  containers:
    - name: nginx
      image: nginx:latest
      ports:
        - name: http
          containerPort: 80
      volumeMounts:
        - name: data
          mountPath: /usr/share/nginx/html
  volumes:
    - name: data
      persistentVolumeClaim:
        claimName: mynginxclaim
```

   Run the following command to verify that the PVC has been bound to a new PV:

```
$ kubectl get pvc
NAME           STATUS   VOLUME          CAPACITY   ACCESSMODES
 STORAGECLASS   AGE
mynginxclaim   Bound    csi-unique_ID   50Gi       RWO           oci-bv
```

   Run the following command to verify that the pod is using the new PVC:

```
$ kubectl describe pod nginx
```

Creating Persistent High Performance Block Volume Storage

Learn to create a high performance block volume as persistent storage On Compute Cloud@Customer.

If you don't need a high performance block volume, use the instructions in Creating Persistent Block Volume Storage on page 635.

1. Create a high performance block volume using the CSI plugin specified by the `oci-bv-high` storage class definition (`provisioner: blockvolume.csi.oraclecloud.com`).

```
$ kubectl create -f csi-bvs-high.yaml
```

   The following is the content of the `csi-bvs-high.yaml` file:

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: oci-bv-high
```

```
provisioner: blockvolume.csi.oraclecloud.com
parameters:
  vpusPerGB: "20"
  attachment-type: "paravirtualized"
volumeBindingMode: WaitForFirstConsumer
allowVolumeExpansion: true
reclaimPolicy: Delete
```

2. Create a persistent volume claim, specifying the storage class name `oci-bv-high`.

```
$ kubectl create -f csi-bvs-high-pvc.yaml
```

The following is the content of the `csi-bvs-high-pvc.yaml` file:

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: mynginxclaim-high
spec:
  storageClassName: "oci-bv-high"
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 50Gi
```

The persistent volume claim name in the `metadata` section is user-specified. You can have more than one persistent volume claim on a persistent volume.

For the value of `accessModes`, specify `ReadWriteOnce`. Do not use `ReadWriteMany`.

The value of the `storage` property must be at least 50 gigabytes.

3. Run the following command to verify that the PVC has been created:

```
$ kubectl get pvc
NAME                STATUS   VOLUME   CAPACITY   ACCESSMODES
 STORAGECLASS   AGE
mynginxclaim-high   Pending                                      oci-bv-high
    4m
```

The PVC has a status of Pending because the `oci-bv-high` storage class definition includes the following:

```
volumeBindingMode: WaitForFirstConsumer
```

4. Use the PVC when creating other objects, such as pods.

For example, you could create a new pod from the following pod definition, which instructs the system to use the `mynginxclaim-high` PVC as the `nginx` volume, which is mounted by the pod at `/data`:

```
apiVersion: v1
kind: Pod
metadata:
  name: nginx-high
spec:
  containers:
    - name: nginx
      image: nginx:latest
      ports:
        - name: http
          containerPort: 80
      volumeMounts:
        - name: data
          mountPath: /usr/share/nginx/html
```

```
  volumes:
    - name: data
      persistentVolumeClaim:
        claimName: mynginxclaim-high
```

Run the following command to verify that the PVC has been bound to a new PV:

```
$ kubectl get pvc
NAME                STATUS   VOLUME          CAPACITY   ACCESSMODES
 STORAGECLASS   AGE
mynginxclaim-high   Bound    csi-unique_ID   50Gi       RWO           oci-
bv-high
```

Run the following command to verify that the pod is using the new PVC:

```
$ kubectl describe pod nginx-high
```

Creating Persistent File System Storage Using the CSI Volume Plugin

On Compute Cloud@Customer, you can provision a PVC on a new file system using the CSI volume plugin. Use the `kubectl` command to create the storage class and persistent volume claim. The CSI volume plugin provisions the PVC on a new file system.

You can have only one mount target and one file system per VCN. You can have multiple storage classes, persistent volumes, and persistent volume claims per cluster. All storage classes, persistent volumes, and persistent volume claims in a cluster share one NFS.

1. Create a new storage class that uses the `fss.csi.oraclecloud.com` provisioner.

   ```
   $ kubectl create -f sc.yaml
   ```

   The following is the content of the `sc.yaml` manifest file:

   ```
   kind: StorageClass
   apiVersion: storage.k8s.io/v1
   metadata:
     name: fss-dyn-storage
   provisioner: fss.csi.oraclecloud.com
   parameters:
     availabilityDomain: AD-1
     compartmentOcid: ocid1.compartment.unique_ID
     mountTargetSubnetOcid: ocid1.subnet.unique_ID
     exportPath: AUTOSELECT
     exportOptions: "[{\"source\":\"0.0.0.0/0\",\"requirePrivilegedSourcePort
   \":false,\"access\":\"READ_WRITE\",\"identitySquash\":\"NONE\"}]"
     encryptInTransit: "false"
   ```

   - The name for the new storage class is `fss-dyn-storage`.
   - Either `mountTargetSubnetOcid` or `mountTargetOcid` is required. The value of `mountTargetSubnetOcid` is the OCID of the subnet where you want the CSI plugin to create a mount target. The value of `mountTargetOcid` is the OCID of an existing mount target. If you specify both `mountTargetSubnetOcid` and `mountTargetOcid`, `mountTargetOcid` is used and `mountTargetSubnetOcid` is ignored.

     To ensure that the mount target can be reached from worker nodes, specify the subnet that has configuration like the "worker" subnet described in Creating a Worker Subnet (Flannel Overlay) on page 536 or Creating a Worker Subnet (VCN-Native Pod) on page 571 or create the mount target on the subnet that has configuration like the worker subnet. Ensure that TCP port 2049 to the NFS server is open on that subnet.
   - The `compartmentOcid` is optional. This value is the OCID of the compartment where the new file system (and the new mount target, if `mountTargetSubnetOcid` is specified) will be created. The default value is the same compartment as the cluster.

- You must specify `AUTOSELECT` as the value for `exportPath`.
- The `exportOptions` value is the NFS export options entry within the file system export that defines the access granted to NFS clients when they connect to a mount target. The `source` can be a single IP address or CIDR block range. This value is a set of parameters in JSON format.
- The value of `encryptInTransit` specifies whether to encrypt data in transit.

2. Create a PVC to be provisioned by the new file system in the File Storage service.

   ```
   $ kubectl create -f fss-dyn-claim.yaml
   ```

   The following is the content of the `fss-dyn-claim.yaml` manifest file:

   ```
   apiVersion: v1
   kind: PersistentVolumeClaim
   metadata:
     name: fss-dynamic-claim
   spec:
     accessModes:
     - ReadWriteMany
     storageClassName: "fss-dyn-storage"
     resources:
       requests:
         storage: 50Gi
   ```

3. Verify that the PVC has been bound to the new persistent volume.

   ```
   $ kubectl get pvc
   NAME              STATUS VOLUME
    CAPACITY ACCESS MODES STORAGECLASS    AGE
   fss-dynamic-claim Bound  csi-fss-f6823a66-8b6f-4c42-9d1f-d25723e69257 50Gi
        RWX          fss-dyn-storage 6m47s
   ```

4. Use the new PVC when you create objects such as pods.

   The following is an example object creation:

   ```
   $ kubectl create nginx.yaml
   ```

   The following is the content of the `nginx.yaml` file. See the `claimName` on the last line:

   ```
   apiVersion: apps/v1
   kind: Deployment
   metadata:
     name: nginx-deployment
   spec:
     replicas: 3
     selector:
       matchLabels:
         app: nginx
     template:
       metadata:
         labels:
           app: nginx
       spec:
         containers:
         - name: nginx
           image: nginx_image_url
           ports:
           - name: http
             containerPort: 80
           volumeMounts:
           - name: persistent-storage
             mountPath: /usr/share/nginx/html
         volumes:
   ```

```
      - name: persistent-storage
        persistentVolumeClaim:
          claimName: fss-dynamic-claim
```

Verify that the object is created and deployed:

```
$ kubectl get deploy
NAME               READY UP-TO-DATE AVAILABLE AGE
nginx-deployment 3/3   3          0         104s
```

Creating Persistent File System Storage Using an Existing File System

On Compute Cloud@Customer, you can provision a PVC on an existing file system.

This procedure creates a mount target, file system, and file system export. Then uses the `kubectl` command to create the storage class, persistent volume, and persistent volume claim.

1. Create a mount target.

   For instructions, see Creating a Mount Target on page 442.

   > **Important:**
   >
   > To ensure that the mount target can be reached from worker nodes, create the mount target on the subnet that has the worker subnet described in Creating OKE Network Resources on page 517. Ensure that TCP port 2049 to the NFS server is open on that subnet.

   Note the export set OCID and mount target OCID. The export set OCID is required to create the file system export, and the mount target OCID is required to create the storage class in later steps.

   You can have only one mount target per VCN.
2. Create a file system.

   For instructions, see Creating a File System on page 445.

   You can create only one file system per VCN. You can have multiple storage classes, persistent volumes, and persistent volume claims per cluster, and they all share one NFS.
3. Create a file system export to associate the mount target with the file system.

   For instructions, see Creating an Export for a File System on page 448.

   - Specify the export set OCID from the output from creating the mount target.
   - Specify the longest CIDR (smallest network) in the CIDR range that you specified when you created the "worker" subnet as described in Creating a Worker Subnet (Flannel Overlay) on page 536 or Creating a Worker Subnet (VCN-Native Pod) on page 571.

   Note the export path and the mount target IP address.
4. Create a storage class, specifying the mount target OCID from the output of the create mount target step.

```
$ kubectl create -f sc.yaml
```

The following is the content of the `sc.yaml` file:

```
kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
  name: pca-fss
provisioner: fss.csi.oraclecloud.com
parameters:
  mntTargetId: ocid1.mounttarget.unique_ID
```

The values of the `apiVersion` and `provisioner` properties are standard. The value of the storage class name in the metadata section is user-specified. You can create more than one storage class per mount target, and the storage class name is used in the following steps to create a persistent volume and persistent volume claim.

Use the `get sc` subcommand to view information about the new storage class:

```
$ kubectl get sc
```

5. Create a persistent volume, specifying the storage class name, the export path, and the mount target IP address.

   The storage class name is in the metadata in the `sc.yaml` file in the preceding step. The export path and the mount target IP address are output from the create file system export step. See Step 3 on page 640.

   ```
   $ kubectl create -f pv.yaml
   ```

   The following is the content of the `pv.yaml` file:

   ```
   apiVersion: v1
   kind: PersistentVolume
   metadata:
     name: fss-pv
   spec:
     storageClassName: pca-fss
     capacity:
       storage: 200Gi
     accessModes:
       - ReadWriteMany
     mountOptions:
       - nosuid
     nfs:
       server: mount_target_IP_address
       path: "/export/unique_ID"
       readOnly: false
   ```

   The persistent volume name in the `metadata` section is user-specified. You can have more than one persistent volume in a storage class.

   In the `nfs` section, the `server` value is the mount target IP address, and the `path` value is the export path.

   Use the `get pv` subcommand to view information about the new persistent volume:

   ```
   $ kubectl get pv
   NAME    CAPACITY  ACCESS MODES  RECLAIM POLICY  STATUS  CLAIM
    STORAGECLASS  REASON  AGE
   fss-pv  200Gi     RWX           Retain          Bound   default/fss-pvc
    pca-fss               20h
   ```

6. Create a persistent volume claim, specifying the persistent volume name and the storage class name.

   The persistent volume name and storage class name are in the output of the `get pv` command.

   Wait for the PVC status to be Bound before using this storage.

   ```
   kubectl create -f pvc.yaml
   ```

   The following is the content of the `pvc.yaml` file:

   ```
   apiVersion: v1
   kind: PersistentVolumeClaim
   metadata:
     name: fss-pvc
   spec:
     storageClassName: pca-fss
   ```

```
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 200Gi
  volumeName: fss-pv
```

The persistent volume claim name in the `metadata` section is user-specified. You can have more than one persistent volume claim on a persistent volume.

The value of the `accessModes` property must be `ReadWriteMany`.

The value of the `storage` property must be at least 50 gigabytes.

Run the following command to view information about the new persistent volume claim:

```
$ kubectl get pvc
NAME      STATUS   VOLUME   CAPACITY   ACCESSMODES   STORAGECLASS   AGE
fss-pvc   Bound    fss-pv   200Gi      RWX           pca-fss        2h
```

7. Use the PVC when creating other objects, such as pods.

   For example, you could create a new pod from the following pod definition, which instructs the system to use the `fss-pvc` PVC as the `nginx` volume, which is mounted by the pod at `/persistent-storage`:

```
apiVersion: v1
kind: Pod
metadata:
  name: fss-dynamic-app
spec:
  containers:
    - name: nginx
      image: nginx:latest
      ports:
        - name: http
          containerPort: 80
      volumeMounts:
        - name: persistent-storage
          mountPath: /usr/share/nginx/html
  volumes:
  - name: persistent-storage
    persistentVolumeClaim:
      claimName: fss-pvc
```

   Run the following command to verify that the pod is using the new PVC:

```
$ kubectl describe pod fss-dynamic-app
```

Using a Persistent Volume

On Compute Cloud@Customer, to use this persistent storage, create a Kubernetes deployment and assign a persistent volume claim as shown in the following sections.

- Using Block Volume Storage on page 642
- Using File System Storage on page 643
- Verify the New Storage Asset on page 644

Using Block Volume Storage

The following example uses block volume storage:

```
$ kubectl create -f nginx-deploy.yaml
```

The following output is the content of the `nginx-deploy.yaml` file.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-bv-deployment
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx-bv
  template:
    metadata:
      labels:
        app: nginx-bv
    spec:
      containers:
      - name: nginx
        image: available_internal_registry/nginx:latest
        volumeMounts:
        - mountPath: /usr/share/nginx/
          name: data
        ports:
        - containerPort: 80
          name: http
          protocol: TCP
      volumes:
      - name: data
        persistentVolumeClaim:
          claimName: mynginxclaim
```

Using File System Storage

The following example uses file system storage:

```
$ kubectl create -f nginx-deploy.yaml
```

The following output is the content of the `nginx-deploy.yaml` file.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-fss-deployment
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx-fss
  template:
    metadata:
      labels:
        app: nginx-fss
    spec:
      containers:
      - name: nginx
        image: nginx:latest
        volumeMounts:
        - mountPath: /usr/share/nginx/
          name: data
        ports:
        - containerPort: 80
          name: http
```

```
          protocol: TCP
      volumes:
      - name: data
        persistentVolumeClaim:
          claimName: fss-pvc
```

Verify the New Storage Asset

Use the `get pod` subcommand to show the names of the replicas in the pod:

```
$ kubectl get pod
nginx-deployment-55ff88b668-2k8rt 1/1 Running 0 4m54s
nginx-deployment-55ff88b668-79c2t 1/1 Running 0 4m54s
nginx-deployment-55ff88b668-qpdfd 1/1 Running 0 4m54s
```

Log in to the pod and use the Linux `df` command to show that the application replicas are using the `persistentVolumeClaim` storage. The `Filesystem` column in the `df` output shows the mount target IP address and the file system export path.

```
$ kubectl exec -it nginx-deployment-55ff88b668-2k8rt -- df -h /usr/share/
nginx/html
Filesystem
        Size Used Avail Use% Mounted on
xxx.xx.xxx.xxx:/
export/4fsderwh09ufyf84ei1lh3q2x8ou86pq5vcbx3aeeo060xxxxxxxxxxxxxxx 67T  0
  67T   0%   /usr/share/nginx/html
```

Deleting a Persistent Volume

Learn how to delete a persistent volume (PV), or retain a PV after all associated persistent volume claims (PVCs) are deleted on Compute Cloud@Customer.

To delete a PVC, see Deleting a Persistent Volume Claim on page 644.

For file system storage, the default behavior is to retain the PV when all associated PVCs are deleted.

For block volume storage, the default behavior is to delete the PV when all associated PVCs are deleted. You might prefer to retain the PV after all associated PVCs are deleted, for example, if the volume contains critical data. See Retaining a Persistent Volume on page 645.

If a PV is retained, you can optionally delete the PV later.

Deleting a Persistent Volume Claim

On Compute Cloud@Customer, to delete a PVC, first delete all pods that are using that PVC. If you attempt to delete the PVC while a pod is still using the PVC, the PVC will be stuck in Terminating state and will not be deleted. When all the pods that are using that PVC are deleted, the PVC is deleted.

1. List all pods that are using the PVC.

   Ensure that you have JQ command line utilities installed to query JSON objects.

   Use the following command to list pods across all the namespaces that are associated with the PVC that you want to delete.

   ```
   $ kubectl get pods --all-namespaces -o=json | jq -c '.items[]
    | {name: .metadata.name, namespace: .metadata.namespace,
    claimName: .spec | select(has("volumes")).volumes[] |
    select(has("persistentVolumeClaim")).persistentVolumeClaim.claimName} |
    select(.claimName != null)'

   {"name":"pod1_name","namespace":"namespace1_name","claimName":"claim1_name"}
   {"name":"pod2_name","namespace":"namespace1_name","claimName":"claim1_name"}
   {"name":"pod3_name","namespace":"namespace2_name","claimName":"claim2_name"}
   ```

To list pods only in the current namespace, use the same command as the preceding command except omit the `--all-namespaces` option.

2. Delete all pods that are using the PVC.

   Use the pod names reported by the `kubectl get pods` command that are associated with the `claimName` that you want to delete.

   ```
   $ kubectl delete pod pod1_name pod2_name
   ```

3. Delete the PVC.

   ```
   $ kubectl delete pvc claim1_name
   ```

4. (Optional) Delete the PV.

   If the Persistent Volume Reclaim Policy is Delete, the PV is automatically deleted when all PVCs that are associated with this PV are deleted.

   To list all PVCs, use the `kubectl get pvc` command.

   If the Persistent Volume Reclaim Policy is Retain, you can use the following command to delete the PV:

   ```
   $ kubectl delete pv pv_name
   ```

Retaining a Persistent Volume

Learn how to retain a persistent volume (PV) On Compute Cloud@Customer.

Rather than delete a PV, you might prefer to retain the PV after all associated PVCs are deleted, for example if the volume contains critical data. See Changing the Reclaim Policy of a Persistent Volume on page 645 for instructions to change the reclaim policy of the PV so that the PV will be retained after all associated PVCs are deleted.

If the Persistent Volume Reclaim Policy is Delete, the PV is automatically deleted when all PVCs that are associated with this PV are deleted. To prevent this behavior, specify the Retain policy. With the Retain policy, the PV isn't deleted but is released of its claim. See Recovering the Data from a Released Persistent Volume on page 646 for instructions to recover the data.

If you decide you want to delete the PV even though it was retained, or you want to delete the PV after you have recovered the data, use the following command:

```
$ kubectl delete pv pv_name
```

Changing the Reclaim Policy of a Persistent Volume

1. List the PVs in the cluster.

   ```
   $ kubectl get pv
   NAME    CAPACITY  ACCESS MODES  RECLAIM POLICY  STATUS  CLAIM
    STORAGECLASS  REASON  AGE
   fss-pv  200Gi     RWX           Delete          Bound   default/fss-pvc
    pca-fss              20h
   ```

2. Change the reclaim policy of the PV.

   ```
   $ kubectl patch pv fss-pv -p '{"spec":
   {"persistentVolumeReclaimPolicy":"Retain"}}'
   ```

3. Verify the reclaim policy change.

   The `RECLAIM POLICY` column should now say `Retain`.

   ```
   $ kubectl get pv
   ```

Recovering the Data from a Released Persistent Volume

The PV isn't available for another claim after the PV has been released of its previous claim because the previous claimant's data is still on the volume. Recover the data and then re-create the PV using the same storage to make a new claim on that storage.

1. Delete the PV.

```
$ kubectl delete pv pv_name
```

   The associated block volume or file system still exists after the PV is deleted.
2. Manually recover and clean up the data on the block volume or file system.
3. (Optional) Manually delete the block volume or file system.

   To reuse the same block volume or file system, create a new PV with the same storage asset definition.

Logging for Compute Cloud@Customer

Learn how to manage the Oracle Cloud Infrastructure (OCI) Logging service on Compute Cloud@Customer infrastructures.

The OCI Logging service provides a highly scalable and fully managed single interface for all the logs in your tenancy. Logs include diagnostic information that describes how resources are performing and being accessed. For general information about the Logging service, see Logging Overview.

You can use the Logging service with Compute Cloud@Customer to generate service logs. By default, the Logging service is disabled on Compute Cloud@Customer. If you enable logging, service logs are generated for user activity.

The logs are stored off-premises in an Object Storage bucket in your OCI tenancy so that they're accessible in the event of a disaster at your site. You can easily search and view the logs in the Oracle Cloud Console.

> **Note:**
>
> Enabling, disabling, and searching logs requires certain policies and permissions. See Logs and Log Groups.

The following sections describe how to manage the Logging service on Compute Cloud@Customer:

- Enabling Logging for Compute Cloud@Customer on page 646
- Disabling Logging for Compute Cloud@Customer on page 647

For information about Compute Cloud@Customer log entry details, see Details for Compute Cloud@Customer

More Logging Resources

- Logging service documentation
- Logging Overview
- Logging Search

Enabling Logging for Compute Cloud@Customer

Learn how to enable Compute Cloud@Customer logs in your OCI tenancy.

1. In the Oracle Cloud Console navigation menu, select **Observability & Management**.
2. Under **Logging**, select **Logs**.
3. Under **Compartment**, select the compartment where your Compute Cloud@Customer infrastructure is located.
4. Select **Enable service log**.

   The **Enable Resource Log** panel is displayed.
5. Enter the following information:

   a. **Service**: Select **Compute Cloud at Customer.**

b. **Resource**: Select the name of the infrastructure for which you want to enable logs.
c. **Log Category**: Select a predefined log category.
d. **Log Name**: Enter a log name. Avoid entering confidential information.

6. (Optional) Select **Show Advanced Options** to configure the following options:

a. **Log Location**: Create or change the log group that this log belongs to.
b. **Log Retention**: Change the default 30 day retention period.
c. **Add Tags**: Apply any tags that you use to organize resources.

For more information about these options, see Enabling Logging from a Service's Resource page.

7. Select **Enable Log**.

Compute Cloud@Customer logging is enabled.

After logging is enabled, you can select the link under **Log name** or **Log group** to view the log details or log group details pages.

Disabling Logging for Compute Cloud@Customer

Learn how to disable Compute Cloud@Customer logs in your OCI tenancy.

1. In the Oracle Cloud Console navigation menu, select **Observability & Management**.
2. Under **Logging**, select **Logs**.
3. Under **Compartment**, select the compartment where your Compute Cloud@Customer infrastructure is located.

A list of enabled logs is displayed.

If no logs are listed, the Logging service might not be enabled, or you might be in the wrong compartment, or you don't have permissions.

4. Select the log you want to disable.
5. Select **Disable Log**.
6. Confirm the action.

The log is disabled.

> **Note:**
>
> To reenable the log, follow Steps 1 on page 647 through 3 on page 647. Then select **Enable Log**, and confirm the action.

Troubleshooting Compute Cloud@Customer

Identify and address common issues that can occur while working with Compute Cloud@Customer.

Troubleshooting Versioning

Identify and address common issues that can occur while working with object versioning On Compute Cloud@Customer.

Unable to Enable Versioning

If enabling versioning fails, the most likely cause is missing or incomplete IAM permissions. Enabling versioning requires:

- User permissions that let you use the bucket and manage the objects in that bucket.
- BUCKET_UPDATE permissions.

Unable to Delete a Bucket

If deleting a bucket fails, the most likely cause is that the bucket is not empty.

You can permanently delete an empty bucket. You can't delete a bucket that contains any of the following:

- Any objects
- Previous versions of an object
- A multipart upload in progress
- A preauthenticated 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.

Unable to Delete a Previous Version

If deleting a previous object version fails, the most likely cause is missing or incomplete IAM permissions. Object version deletion requires:

- User permissions that let you use the bucket and manage the objects in that bucket.
- At minimum, OBJECT_VERSION_DELETE permissions.

Compute Cloud@Customer Known Issues

These known issues have been identified in Compute Cloud@Customer.

- Customers in certain OCI realms will not have access to usage/billing for resource consumption on page 648
- Take caution when creating or deleting compartments and tags against Compute Cloud@Customer resources on page 649
- System upgrades are not possible for specific OCI realms on page 649
- Upgrade might impact Load Balancer as a Service functionality on page 649
- GPU Drivers Not Included in Oracle Linux Platform Images on page 650
- The oci ccc get infrastructure and oci ccc infrastructure update CLI commands return null for the provisioning_pin value on page 653
- When using the ccc infrastructure list CLI command with the --compartment-id-in-subtree true option, no results are returned on page 653
- Output from oci iam user get doesn't list user capabilities on page 655

Customers in certain OCI realms will not have access to usage/billing for resource consumption

Details

Customers in realms outside of the following list currently will not have access to usage/billing for resource consumption on the Compute Cloud@Customer rack:

- OC1
- OC2
- OC3
- OC4
- OC5
- OC6
- OC7
- OC8
- OC9
- OC10
- OC11

- OC12
- OC14
- OC16
- OC19

Workaround: This resource consumption will “catch up” when onboarding of additional realms completes.

Take caution when creating or deleting compartments and tags against Compute Cloud@Customer resources

Details

Customers in realms outside of the following list need to be cautious when creating or deleting compartments and tags against Compute Cloud@Customer (CCC) resources.

Failure to follow the guidelines in the Workaround section could lead to out of sync environments and critically a loss of resource(s) that may require the system to be bootstrapped to factory configuration.

- OC1
- OC2
- OC3
- OC4
- OC5
- OC6
- OC7
- OC8
- OC9
- OC10
- OC11
- OC12
- OC14
- OC16
- OC19

Workaround: Ensure you do not:

- Delete compartments in OCI where you have created a resource on the Compute Cloud@Customer rack. For example, "Compartment C" on the rack and then deleting that compartment in OCI.
- Delete tags on any Compute Cloud@Customer resources if any are created.
- Delete the compartment that the CCC-Infrastructure resource is in. Failure to adhered to this could lead to a loss of the resource and require the system to be bootstrapped to its factory configuration.

System upgrades are not possible for specific OCI realms

Details

System upgrades are unavailable for Compute Cloud@Customer racks in any OCI realm outside of OC1.

Workaround: We're working on a resolution.

Upgrade might impact Load Balancer as a Service functionality

The first upgrade after May 2025, changes the Load Balancer as a Service (LBaaS) to a new background implementation. As a result, a few features are either different or no longer available. An existing configuration that's no longer supported in the new implementation, can have a negative impact on the software upgrade.

Details

Before the first upgrade after May 2025, if you use the Load Balancer service, review the following issues, and act if needed:

- **Response body regex parsing** (Bug: 37629014)

  If you have a Load Balancer configured with regular expression (regex) parsing of backend responses for health status information, that won't work after the upgrade. Health status reporting is limited to response codes.

  Workaround: Unconfigure the optional regex setting (`--response-body-regex`) for the response from the backend servers.
- **Cipher suites** (Bug: 37461876)

  In the new Load Balancer implementation, weaker cipher suites have been removed. Going forward, SSL/TLS connections can be secured with these cipher suites:

  ```
  AES128-GCM-SHA256, AES256-GCM-SHA384,
  ECDHE-ECDSA-AES128-GCM-SHA256, ECDHE-ECDSA-AES256-GCM-SHA384,
  ECDHE-RSA-AES128-GCM-SHA256, ECDHE-RSA-AES256-GCM-SHA384,
  AES128-SHA, AES256-SHA, DES-CBC3-SHA,
  ECDHE-ECDSA-AES128-SHA, ECDHE-ECDSA-AES256-SHA,
  ECDHE-RSA-AES128-SHA, ECDHE-RSA-AES256-SHA,
  PSK-AES128-CBC-SHA, PSK-AES256-CBC-SHA
  ```

  Workaround: Ensure that your Load Balancer configuration is using the latest cipher suites. If necessary, change the existing Load Balancer configurations.
- **Cookie-based session persistence** (Bug: 37473362)

  For existing load balancers, session persistence between clients and backend servers can be enabled using either application cookies or load balancer cookies. These are no longer supported after upgrade.

  Workaround: Unconfigure cookie-based session persistence. Or, load balancer cookies can be preserved on the condition that the load balancing policy is set to IP hash before the upgrade.
- **Server order preference**

  The SSL parameter to prioritize server ciphers over client ciphers isn't supported.

For more information, see Load Balancer as a Service on page 211.

GPU Drivers Not Included in Oracle Linux Platform Images

The Oracle Linux 8 or Oracle Linux 9 platform images provided with Compute Cloud@Customer don't include GPU drivers. If you create a GPU instance, you must manually install the GPU drivers.

Details

If a Compute Cloud@Customer installation includes compute nodes with GPUs, you can access them by selecting a dedicated shape. The GPU shapes can be selected for compute instances based on an Oracle Linux 8 or Oracle Linux 9 platform image. The current image versions don't include GPU drivers. The instance OS detects the allocated GPUs, but to use them, you need the CUDA Toolkit from the NVIDIA developer site to install the required drivers.

> **Note:**
>
> The large download and local repository installation need a large amount of disk space. The default 50GB boot volume is insufficient on Oracle Linux 9 and only just large enough on Oracle Linux 8. We highly recommend increasing the boot volume size to at least 60GB, and extend the file system accordingly.

Workaround

After creating the instance, log in to the instance and install the CUDA Toolkit. Follow the instructions for your version of Oracle Linux.

Installing GPU Drivers in an Oracle Linux 9 Instance

1. From the command line of the instance, download and install the CUDA Toolkit rpm for your OS.

```
$ wget https://developer.download.nvidia.com/compute/
cuda/12.8.0/local_installers/cuda-repo-rhel9-12-8-
local-12.8.0_570.86.10-1.x86_64.rpm
$ sudo rpm -i cuda-repo-rhel9-12-8-
local-12.8.0_570.86.10-1.x86_64.rpm
$ sudo dnf clean all
$ sudo dnf install cuda-toolkit-12-8
```

2. Enable the Oracle Linux 9 EPEL yum repository. Install the `dkms` package.

```
$ sudo yum-config-manager --enable ol9_developer_EPEL
$ sudo dnf install dkms
```

3. Install the GPU drivers.

```
$ sudo dnf install cuda-12-8
```

4. Verify the installation with the NVIDIA System Management Interface.

```
$ nvidia-smi
+-----------------------------------------------------------------------------------------
+
| NVIDIA-SMI 570.86.10              Driver Version: 570.86.10
 CUDA Version: 12.8     |
|-----------------------------------------+------------------------
+----------------------+
| GPU  Name                 Persistence-M | Bus-Id          Disp.A |
 Volatile Uncorr. ECC |
| Fan  Temp   Perf          Pwr:Usage/Cap |           Memory-Usage |
 GPU-Util  Compute M. |
|                                         |                        |
               MIG M. |
|=========================================+========================
+======================|
|   0  NVIDIA L40S                    Off |   00000000:00:05.0 Off |
                    0 |
| N/A   26C    P8             23W /  350W |       1MiB /  46068MiB |
      0%      Default |
|                                         |                        |
                   N/A |
+-----------------------------------------+------------------------
+----------------------+

+-----------------------------------------------------------------------------------------
+
| Processes:
                       |
|  GPU   GI   CI              PID   Type   Process name
          GPU Memory |
|        ID   ID
          Usage      |
|
=========================================================================================
|  No running processes found
                       |
+-----------------------------------------------------------------------------------------
+
```

Installing GPU Drivers in an Oracle Linux 8 Instance

1. From the command line of the instance, download and install the CUDA Toolkit rpm for your OS.

```
$ wget https://developer.download.nvidia.com/compute/
cuda/12.8.0/local_installers/cuda-repo-rhel8-12-8-
local-12.8.0_570.86.10-1.x86_64.rpm
$ sudo rpm -i cuda-repo-rhel8-12-8-
local-12.8.0_570.86.10-1.x86_64.rpm
$ sudo dnf clean all
$ sudo dnf install cuda-toolkit-12-8
```

2. Enable the Oracle Linux 8 EPEL yum repository. Install the `dkms` package.

```
$ sudo yum-config-manager --enable ol8_developer_EPEL
$ sudo dnf install dkms
```

3. Install the GPU drivers.

```
$ sudo dnf install cuda-12-8
```

4. Install the NVIDIA kernel module.

```
$ sudo scl enable gcc-toolset-13 bash
# dkms install nvidia-open -v 570.86.10
```

If this `make` error appears while the kernel module is built, you can safely ignore it.

```
Cleaning build area...(bad exit status: 2)
Failed command:
make -C /lib/modules/5.15.0-206.153.7.el8uek.x86_64/build M=/var/
lib/dkms/nvidia-open/570.86.10/build clean
```

5. Verify the installation with the NVIDIA System Management Interface.

```
# nvidia-smi
+-----------------------------------------------------------------------------------------
+
| NVIDIA-SMI 570.86.10              Driver Version: 570.86.10
 CUDA Version: 12.8     |
|-----------------------------------------+------------------------
+----------------------+
| GPU  Name                 Persistence-M | Bus-Id          Disp.A |
 Volatile Uncorr. ECC |
| Fan  Temp   Perf          Pwr:Usage/Cap |           Memory-Usage |
 GPU-Util  Compute M. |
|                                         |                        |
              MIG M. |
|=========================================+========================
+======================|
|   0  NVIDIA L40S                    Off |   00000000:00:05.0 Off |
                    0 |
| N/A   26C    P8             23W /  350W |       1MiB /  46068MiB |
      0%      Default |
|                                         |                        |
                 N/A |
+-----------------------------------------+------------------------
+----------------------+

+-----------------------------------------------------------------------------------------
+
| Processes:
                       |
|  GPU   GI   CI              PID   Type   Process name
          GPU Memory |
```

```
|        ID   ID
          Usage      |
|
==========================================================================================
|  No running processes found
                      |
+-------------------------------------------------------------------------------------
+
```

The `oci ccc get infrastructure` and `oci ccc infrastructure update` CLI commands return `null` for the `provisioning_pin` value

Details

When you create an infrastructure, a PIN is generated and displayed in the output.

However, if you use the `oci ccc get infrastructure` command right after creating or updating the infrastructure, the PIN might not be returned.

This happens because the PIN isn't available to the `get` command for up to 5 minutes after creation.

Example output:

```
{
  "compartment_id": "ocid1.compartment.oc1..uniqueID",
. . .
  },
  "display_name": "C3ResourcePrincipal_infra",
  "freeform_tags": {},
  "id": "ocid1.cccinfrastructure.uniqueID",
  "lifecycle_details": null,
  "lifecycle_state": "ACTIVE",
  "provisioning_fingerprint": null,
  "provisioning_pin": null,
  "rack_inventory": {
    "capacity_storage_tray_count": null,
    "compute_node_count": null,
    "management_node_count": null,
    "performance_storage_tray_count": null,
    "serial_number": null
. . .

}
```

Workaround

Obtain the PIN from the `create` command, or wait 5 minutes to retrieve the PIN using the `get` command.

For more information, see the ccc infrastructure CLI Reference page.

When using the ccc infrastructure list CLI command with the --compartment-id-in-subtree true option, no results are returned

Details

You get an empty list even though there are items in the subtree.

Example:

```
oci ccc infrastructure list --profile user1 --compartment-id-in-subtree
 true -c ocid1.tenancy.oc1..uniqueID
{
  "data": {
    "items": []
```

```
  }
}
```

Workaround

Instead of using the `--compartment-id-in-subtree` option, query each compartment directly using the `-compartment` option.

Example:

```
oci ccc infrastructure list --profile user1 -compartment
 ocid1.tenancy.oc1..uniqueID
{
  "data": {
    "items": [ list of compartment details ]
  }
}
```

For more information, see the ccc infrastructure CLI Reference page.

Output from `oci iam user get` doesn't list user capabilities

Details

The output from `oci iam user get` differs between Oracle Cloud Infrastructure (OCI) and Compute Cloud@Customer. The Compute Cloud@Customer output shows `null` for `capabilites` and omits the list of capabilities, as shown in the following table.

OCI Output	Compute Cloud@Customer Output

`oci iam user get --user-id ocid1.user.oc1..uniqueID`
`{`
`"data": {`
`"capabilities": {`
`"can-use-api-keys": true,`
`"can-use-auth-tokens": true,`
`"can-use-console-password": true,`
`"can-use-customer-secret-keys": true,`
`"can-use-o-auth2-client-credentials": true,`
`"can-use-smtp-credentials": true`
`},`
`"compartment-id": "ocid1.tenancy.oc1..uniqueID",`
`"defined-tags": {`
`"Oracle-Recommended-Tags": {`
`"ResourceType": "group",`
`"UtilExempt": "minrequired"`
`}`
`},`
`"description": "user-1",`
`"email": null,`
`"email-verified": false,`
`"external-identifier": null,`
`"freeform-tags": {},`
`"id": "ocid1.user.oc1..uniqueID",`
`"identity-provider-id": null,`
`"inactive-status": null,`
`"is-mfa-activated": false,`
`"last-successful-login-time": "2024-02-08T10:25:44.036000+00:00",`
`"lifecycle-state": "ACTIVE",`
`"name": "user-1",`
`"previous-successful-login-time": null,`
`"time-created": "2024-02-08T09:12:35.256000+00:00"`
`},`
`"etag": "60f0527b3bbd0f40f137d4149d131fbf77eb44ab"`
`}`
	`oci iam user get --user-id ocid1.user.oc1..uniqueID`
	`{`
	`"data": {`
	`"capabilities": null,`
	`"compartment-id": "ocid1.tenancy.oc1..uniqueID",`
	`"defined-tags": {`
	`"Oracle-Recommended-Tags": {`
	`"ResourceType": "group",`
	`"UtilExempt": "minrequired"`
	`}`
	`},`
	`"description": "user-1",`
	`"email": null,`
	`"email-verified": null,`
	`"external-identifier": null,`
	`"freeform-tags": {},`
	`"id": "ocid1.user.oc1..uniqueID",`
	`"identity-provider-id": null,`
	`"inactive-status": null,`
	`"is-mfa-activated": null,`
	`"last-successful-login-time": null,`
	`"lifecycle-state": "ACTIVE",`
	`"name": "user-1",`
	`"previous-successful-login-time": null,`
	`"time-created": "2023-02-08T09:12:35.256000+00:00"`
	`},`
	`"etag": "bee44237-6d70-4691-b7f9-a98fbb332b12"`

Workaround

To see the list of capabilities, run the `oci iam user get` command in your OCI tenancy.

Index

C

cooling requirements 48

F

floor load requirements 44

V

ventilation requirements 48